

UNIVERSITATEA "POLITEHNICA" TIMIȘOARA

FACULTATEA DE ELECTRONICĂ ȘI  
TELECOMUNICAȚII



---

**DAN FLORENTIN LASCU**

**TRANSFERUL CONTROLAT AL ENERGIEI  
PRIN CONVERTOARE PWM ȘI REZONANTE**

**Teză de doctorat**

**Conducător științific**

**Prof. dr. ing. MIRCEA CIUGUDEAN**

**Timișoara 1998**

BIBLIOTECA CENTRALĂ  
UNIVERSITATEA "POLITEHNICA"  
TIMIȘOARA

**Pentru Alice și Diana**  
**părinților mei**

## Notații, simboluri și abrevieri

- $A, B, E, F$  - matricile de stare ale unui sistem în reprezentarea clasică în spațiul stărilor
- CCM - “continuous conduction mode”, modul de funcționare în conducție continuă
- CF-QRC - “constant frequency quasirezonant converter”, convertor cvasirezonant cu funcționare la frecvență constantă
- $d$  - factorul de umplere continuu (mediata pe o perioadă de comutație a funcției de comutație)
- dc - “direct current”, curent continuu
- $d_k$  - factorul de umplere corespunzător celei de a  $k$ -a perioade
- $D$  - factorul de umplere în stare staționară
- DCVM - “discontinuous capacitor voltage mode”, modul de funcționare în conducție discontinuă din punct de vedere al tensiunii capacitive
- DICM - “discontinuous inductor current mode”, modul de funcționare în conducție discontinuă din punct de vedere al curentului inductiv
- DIVM - “discontinuous input voltage mode”, modul de funcționare în conducție discontinuă din punct de vedere al tensiunii de intrare în mediul de comutație
- $\delta$  - funcția de comutație a unui întrerupător. Are valoarea 1 pentru întrerupătorul în conducție și 0 pentru întrerupătorul blocat.
- EMI - “electromagnetic interference”, interferență electromagnetică
- FLC - “fuzzy logic control”, comandă în logică fuzzy
- $f_s, T_s$  - frecvența respectiv perioada de comutație
- FW (pentru convertoare cvasirezonante) - “full wave”, mod de conducție cu undă plină
- HW (pentru convertoare cvasirezonante) - “half wave”, mod de conducție în semiundă
- LTI (pentru sisteme sau model) - liniar și invariant în timp.
- $N_i$  - numărul de iterații
- $N_s$  - numărul stărilor topologice din funcționarea în stare staționară a unui convertor dc-dc.
- $N_{sim}$  - numărul de simulări
- PFC - “power factor correction”, corecția factorului de putere.
- PWM - “pulse width modulation”, modulația impulsurilor în durată
- $q(x)$  - funcția cvasirezonantă
- QRC - “quasirezonant converter”, convertor cvasirezonant
- QSW - “quasi square wave converter” convertor cu undă cvasidreptunghiulară
- $S$  - “switch”, întrerupător
- SRC - “series resonant converter”, convertor rezonant serie
- $T$  - operația de transpunere a matricilor.
- $V_g$  - tensiunea de alimentare
- $V_o$  - tensiunea de ieșire
- $x$  - vectorul de stare
- $Z$  - valoarea medie (în sens clasic) a variabilei  $z$
- $\bar{z}$  - mediata variabilei  $z$
- $\hat{z}$  - perturbația de semnal mic prezentă în variabila  $z$
- ZCS - “zero current switching”, comutare la curent zero
- ZVS - “zero voltage switching”, comutare la tensiune zero

## CUPRINS

<b>Introducere.</b>	1
<b>1. Sinteza convertoarelor dc-dc cu modulația impulsurilor în durată (PWM).</b>	7
1.1. Sinteza după natura sursei și sarcinii.	7
1.2. Sinteza prin cascada.	9
1.3. Metoda de sinteză având la bază întrerupătorul PWM.	12
1.4. Sinteza prin dualitate.	15
1.4.1. Tipuri de elemente de comutație și implementarea lor practică.	15
1.4.2. Aplicarea principiilor dualității la convertoarele în comutație.	20
1.5. Sinteza prin inversie bilaterală.	27
1.6. O nouă metodă de sinteză a convertoarelor PWM dc-dc în comutație: “metoda celulei de bază complete”.	30
1.6.1. Familia convertoarelor de ordinul 2. Celula BBB.	35
1.6.2. Familii de convertoare de ordinul 4.	40
1.6.3. Celule de bază complete care generează familia convertoarelor având curenți cu pulsații nule.	61
1.6.4. Familia convertoarelor pătratice.	63
1.7. Concluzii.	77
<b>2. Convertoare având curenți de intrare/ieșire cu pulsații nule</b>	79
2.1. Principiul obținerii curenților cu pulsații nule.	79
2.2. Obținerea convertoarelor dc-dc cu pulsații nule folosind generarea din celule de bază.	83
2.3. Noi aspecte și soluții privind magnetica integrată.	92
2.4. Concluzii.	97
<b>3. Sinteza convertoarelor cvasirezonante.</b>	99
3.1. Convertoare cvasirezonante. Definiții.	99
3.2. Sinteza unitară a convertoarelor cvasirezonante.	100
3.2.1. Sinteza convertoarelor cvasirezonante cu două elemente rezonante.	100
3.2.2. Variațiuni topologice.	104
3.2.3. Clase degenerat de convertoare QRC cu două elemente rezonante.	104
3.2.4. Sinteza convertoarelor cvasirezonante cu mai mult de două elemente rezonante.	105
3.2.5. Implementarea întrerupătoarelor.	105

3.2.6. Aplicații ale procedurii de sinteză.	106
3.3. Comanda cu frecvență constantă a convertoarelor cvasirezonante și rezonante.	108
3.4. Concluzii.	112
<b>4. Analiza unitară de curent continuu a convertoarelor dc-dc.</b>	113
4.1. Notații utilizate și principiul analizei unitare de curent continuu	113
4.2. Analiza unitară a convertoarelor PWM.	118
4.2.1. Regimul de conducție continuă (CCM).	118
4.2.2. Regimul de conducție discontinuă din punct de vedere al curentul inductiv (DICM).	119
4.2.2.1. Regimul DICM pentru convertoare cu două întrerupătoare.	119
4.2.2.2. Regimul DICM pentru convertoare pătratice.	122
4.2.3. Regimul de conducție discontinuă din punct de vedere al tensiunii capacitive	126
4.3. Analiza unitară de curent continuu a convertoarelor cvasirezonante.	134
4.3.1. Analiza unitară a convertoarelor ZCS și ZVS.	134
4.3.2. Analiza unitară a convertoarelor cvasirezonante cu funcționare la frecvență constantă.	140
4.4. Concluzii.	142
<b>5. Metode de comandă a convertoarelor dc-dc.</b>	145
5.1. Comanda în curent.	145
5.2. Tehnici de comandă integrative.	146
5.2.1. Comanda prin sarcină.	146
5.2.2. Comanda integrativă după o variabilă.	147
5.2.3. Comanda mixtă.	149
5.3. Comanda fuzzy.	152
5.4. Comanda cu rețele neuronale.	162
5.5. Concluzii.	165
<b>6. Modelarea convertoarelor dc-dc.</b>	169
6.1. O scurtă trecere în revistă a principalelor tehnici de modelare.	169
6.2. Modele canonice. Modele canonice generalizate. Problema filtrului de intrare.	173
6.3. Modele unificate pentru convertoarele rezonante serie 180	
6.3.1. O metodă generală de mediere.	180
6.3.2. Modele unificate pentru convertorul rezonant serie	181
6.4. Un model de semnal mic în spațiul stărilor pentru comanda prin sarcină.	191
6.5. Un model de semnal mic îmbunătățit pentru comanda integrativă după o variabilă.	195

6.6. Noi modele de semnal mare și de semnal mic pentru convertoare cvasirezonante. . . . .	199
6.7. Două modele discrete noi pentru convertoare în comutație . . . . .	203
6.8. Analiza de semnal mic exactă pentru convertoare dc-dc cu funcționare la frecvență constantă. . . . .	208
6.8.1. Algoritmul Tymerski de calcul exact al funcției de transfer control-ieșire pentru convertoare dc-dc cu funcționare la frecvență constantă. . . . .	208
6.8.2. Contribuții la calculul exact, accelerat și automat al stării staționare într-un convertor dc-dc. . . . .	211
6.8.2.1. Calculul stării staționare prin metoda Newton-Raphson vectorială. . . . .	211
6.8.2.2. Calculul și analiza accelerată a stării staționare prin legarea unui program de calcul matematic și un simulator de circuite de uz general. . . . .	215
6.8.3. Funcțiile de transfer control-ieșire exacte pentru principalele moduri de funcționare și tipuri de comandă în convertoare PWM. . . . .	230
6.8.3.1. Funcția de transfer control-ieșire exactă pentru convertoare PWM DCVM. . . . .	230
6.8.3.2. Funcția de transfer control-ieșire exactă pentru convertoare PWM DIVM. . . . .	232
6.8.3.3. Funcția de transfer control-ieșire exactă pentru convertoare PWM CCM cu feedforward. . . . .	233
6.8.3.4. Funcția de transfer control-ieșire exactă pentru convertoare PWM CCM cu comanda prin sarcină. . . . .	236
6.8.3.5. Funcția de transfer control-ieșire exactă pentru convertoare PWM CCM cu comandă integrativă . . . . .	239
6.8.3.6. Expresia generală exactă a funcției de transfer control-ieșire pentru orice tip de comandă integrativă în moduri discontinue. . . . .	244
6.8.3.7. Convertoare la care funcția de transfer control-ieșire mediată are valabilitate la orice frecvență. . . . .	245
6.9. Concluzii. . . . .	246
<b>7. Aplicații ale convertoarelor PWM și QRC în circuite de corecție a factorului de putere. . . . .</b>	<b>251</b>
7.1. Utilizarea convertoarelor cvasirezonante ca circuite pentru corecția factorului de putere. Analiză și proiectare. . . . .	251
7.2. Funcționarea convertoarelor DIVM ca circuite de corecție a factorului de putere. . . . .	259

7.3. Concluzii.	261
<b>Concluzii</b>	265
<b>Bibliografie</b>	273
<b>Anexa</b>	

## Introducere

Convertoarele dc-dc sunt, așa cum apreciază somitățile în materie, ca prof. Slobodan Čuk sau prof. Robert Erickson, poarta de intrare în electronica de putere. O justificare ar fi faptul că studiul lor se poate face mai direct decât în cazul convertoarelor care au la intrare sau ieșire energie alternativă. Argumentul decisiv este însă acela că aplicațiile cu convertoare dc-dc au penetrat și penetrează profund în categorii de convertoare care până nu demult formau clase “închise”, pentru că efectuau un cu totul alt tip de conversie decât cea dc-dc, studiindu-se de altfel separat. Ca și exemple se pot aminti invertoarele PWM, care au intrat deja în domeniul clasicului și mai nou redresoarele cu factor de putere unitar, sau circuitele de corecție a factorului de putere, domeniu în plină expansiune astăzi. De aceea studiul convertoarelor dc-dc continuă să suscite atenția cercetătorilor și să ridice noi provocări.

În studiul convertoarelor dc-dc se pot defini șase direcții principale de studiu: *sinteza, analiza, comanda, modelarea, proiectarea și aplicațiile în alte tipuri de conversie a energiei*. Desigur că aceste direcții se întrepătrund, un anumit model de exemplu, fiind util în măsura în care aduce informații noi sau ușurează analiza și sugerează noi relații de proiectare.

Această teză este structurată în șapte capitole de dezvoltare, o încheiere fiind destinată concluziilor. Fiecare capitol conține o succintă trecere în revistă fie a stadiului actual al problematicii discutate, fie a rezultatelor de care se uzează în paragrafele următoare, urmând ca în continuare să se prezinte numai problemele esențiale și contribuțiile proprii. În finalul fiecărui capitol autorul a sintetizat distinct contribuțiile pe care acesta le consideră că le-a adus domeniului abordat, cu precizarea paragrafelor în care ele pot fi găsite, permițându-și apoi să creioneze principalele direcții de cercetare în viitor, legate de tematica abordată în capitol.

*Capitolul 1* este destinat sintezei convertoarelor PWM dc-dc. Este cel mai vast capitol, autorul considerându-l important din cel puțin două puncte de vedere: realizează o grupare sistematică a convertoarelor în familii (toate convertoarele familiei având același ordin), furnizând totodată o metodă unitară de generare de noi convertoare. Sunt prezentate tehnici de sinteză combinaționale și tehnici de sinteză evoluționiste, ca instrumente în elaborarea tehnicii originale de sinteză denumite “metoda celulei de bază complete”, care îmbină caracterul evoluționist cu cel combinațional. În dezvoltarea acestei metode autorul enunță o serie de propoziții și teoreme din care cea mai importantă se consideră cea a inversiei bilaterale generalizate. Această teoremă folosește conceptul general de întrerupător introdus de Erickson, pentru care autorul extinde noțiunea de dualitate. De aceea dimensiunea capitolului se justifică prin numeroasele noțiuni și tehnici la care se apelează și a căror prezentare s-a considerat necesară. Rezultatele obținute se concretizează în noi tipuri de convertoare BUCK, BOOST, iar din familia denumită de autor ZSWJ se amintesc convertoarele WATKINS-JOHNSON și WATKINS-JOHNSON INVERS, fiecare cu foarte bune proprietăți de compatibilitate electromagnetică, toate contribuții ale autorului, o parte fiind verificate și experimental. Sunt prezentate de asemenea 12 convertoare pătrăce originale, ceea ce spune mult despre avantajele metodei.

*Capitolul 2* este destinat convertoarelor cu curenți de intrare/ieșire cu pulsații nule. Acest tip de convertoare este de actualitate prin proprietățile de interferență electromagnetică remarcabile pe care le oferă. Este expus principiul cuplării bobinelor cu tensiuni egale, ce stă la baza acestor structuri și este prezentată o tehnică prin care se poate renunța la cuplajul magnetic între bobine fără a se pierde proprietatea de pulsații nule ale curenților. Autorul extinde sinteza cu metoda celulei de bază la aceste convertoare, realizând astfel un studiu unitar și general. În acest fel, folosind analiza pe celula de bază, se argumentează de ce o serie de topologii prezentate în literatură sunt de fapt versiuni ale uneia și aceleiași structuri. Este demonstrată echivalența dintre



un convertor ZETA cu curent de ieșire fără pulsații, având bobinele cuplate, cu un convertor BUCK-BOOST cu filtru la ieșire, precum și imposibilitatea realizării condiției de curenți cu pulsații nule pentru convertoare cu tensiune de ieșire bipolară din familia ZSWJ. De asemenea este introdusă o variantă de convertor WATKINS-JOHNSON cu curent de ieșire fără pulsații.

În finalul capitolului este prezentat un nou model al transformatorului, propus recent de prof. Slobodan Ćuk. S-a insistat asupra acestui model deoarece autorul consideră că el va avea influența puternic evoluția din viitorul imediat a convertoarelor cu curenți de intrare/ieșire de pulsații nule și magnetica integrată în general. În lucrare, se exemplifică pentru argumentare cu primele soluții lansate pe piață și care au la bază acest nou model al transformatorului.

Sinteza convertoarelor cvasirezonaante este obiectul *capitolului 3*. Este dată o procedură de sinteză unitară a acestor convertoare, bazată pe o abordare cu elemente de teoria grafurilor, publicată în literatură. Sunt astfel obținute, pe lângă convertoarele ZCS și ZVS cunoscute și devenite clasice și convertoarele QSW și două clase noi denumite  $Q_n$ -PWM și  $Q_f$ -PWM. Autorul sistematizează algoritmul de sinteză aplicându-l nu pe convertoare PWM separate ci direct în celula de bază. În plus este demonstrată imposibilitatea existenței a patru cazuri privind inserția capacității și a bobinei rezonante. În final este examinat modul de obținere a convertoarelor QRC cu funcționare la frecvența constantă prin metoda celulei de bază, făcându-se considerații asupra modului de alegere și implementare practică a întrerupătoarelor.

*Capitolul 4* este dedicat analizei unitare de curent continuu a convertoarelor dc-dc. Plecând de la noțiunile de mediu de comutație și factor de umplere echivalent, introduse de Maksimović, autorul sistematizează, printr-o serie de teoreme și propoziții, strategia de analiză statică a convertoarelor PWM. Se arată că analiza de curent continuu se poate face pe celule de bază, autorul exemplificând acest lucru prin contribuția privind stabilirea condițiilor de funcționare CCM pentru toate convertoarele pătratică Sunt făcute considerații asupra celulei de bază, pentru un convertor din familie rezultatele obținându-se prin simplă particularizare. În legătură cu mediul de comutație și modul DCVM autorul introduce o nouă clasă de convertoare denumită DIVM ("discontinuous input voltage mode" - cu tensiune de intrare discontinuă) pentru care este efectuată și analiza de curent continuu și căreia i se subliniază o serie de proprietăți benefice. Rezultatele teoretice privind această clasă au fost confirmate de experimentele efectuate.

În privința convertoarelor QRC, autorul s-a bazat pe rezultatele a o serie de autori, publicate în literatură, elaborând programe de trasare automată a caracteristicilor control-ieșire și de ieșire. Aceste programe care sunt apreciate a fi instrumente utile în analiză și proiectare.

În *capitolul 5* sunt investigate diverse tehnici de comandă a convertoarelor dc-dc. În primele paragrafe sunt expuse principalele metode de comandă menționate în literatură: comanda în curent, comanda prin sarcină și comanda integrativă după o variabilă. Autorul introduce ca metodă nouă de comandă tot o tehnică integrativă, numită comandă mixtă, care realizează integrarea ponderată a mărimilor lent variabile din mediul de comutație. Sunt prezentate avantajele și dezavantajele acestei tehnici de comandă față de celelalte metode integrative.

În continuare sunt prezentate rezultatele teoretice și experimentale privind comanda fuzzy a convertoarelor dc-dc. Un astfel de sistem a fost construit în jurul modului EVM30 (Texas Instruments, cu procesor de semnal TMS320C30), făcându-se considerații privind alegerea frecvenței de eșantionare, a momentului eșantionării și efectele medierii eșantioanelor. Regulatorul fuzzy a fost de tip Mamdani folosind implicația min în metoda de inferență. Este prezentat comportamentul static și dinamic pentru un tabel de reguli cu singleton-uri și pentru un tabel de inferență cu termeni lingvistici. De asemenea este investigat efectul anumitor parametri în performanțele regulatorului. Atenția s-a focalizat pe comportarea dinamică în raport cu sarcina și sursa de alimentare, evidențiindu-se performanțe dinamice bune.

Tot ca o posibilă tehnică a viitorului s-a demonstrat fezabilitatea cu rețele neuronale a comenzii unui convertor rezonant serie. S-a plecat de la comanda pe traiectorie optimală pentru generarea

vectorilor de antrenament, citată în literatură și prin simulare s-a demonstrat că o rețea neuronală de complexitate medie poate conduce la un controller bun pentru un convertor SRC.

Modelării convertoarelor dc-dc îi este afectat *capitolul 6*. Este un capitol de mare extindere în raport cu altele, justificată și prin generozitatea domeniului, deși s-a încercat pe cât posibil concentrarea expunerii. Singurul paragraf fără contribuții originale din acest capitol este paragraful 6.1, în care sunt trecute în revistă principalele tehnici de modelare cunoscute.

În paragraful 6.2 este prezentată rezolvarea problemei clasice a filtrului de intrare în cazul convertoarelor QRC-FW, folosind modelul canonic generalizat al lui Witulski și Erickson.

În paragraful 6.3, folosind metoda generală de mediere a lui Sun și Grotstollen, sunt dezvoltate modele unificate pentru convertorul rezonant serie. Modelele sunt prezentate în formă închisă având valabilitate atât peste, cât și sub frecvența de rezonanță. Sunt prezentate atât modele statice cât și modele dinamice, de semnal mare și de semnal mic. Utilizând modelele statice, autorul a dezvoltat programe interactive de proiectare asistată a acestor convertoare. Modelele de semnal mare au fost verificate prin simulare, constatându-se o accelerare semnificativă a simulării, în timp ce modelul de semnal mic a fost verificat comparativ cu exemplele citate în literatură, unde există și rezultate experimentale.

În continuare este prezentat un nou model mediat pentru comanda prin sarcină, care, spre deosebire de modelele publicate până acum, are calitatea de a fi în spațiul stărilor. Modelul pleacă de la un model în forma unei scheme bloc, citat în literatură, căruia, după câteva transformări, folosind un observator canonic și introducând un vector de stare extins, i se poate deduce expresia în spațiul stărilor. Este prezentată o aplicație a noului model folosind un program scris de autor în studiul stabilității convertoarelor.

În paragraful 6.5 este elaborat tot un model de semnal mic îmbunătățit pentru comanda integrativă după o variabilă. Spre deosebire de modelul elaborat de Smedley și Čuk acesta ține cont de efectul de eșantionare-memorare prezent în perturbațiile din tensiunile capacitive. Utilitatea modelului a fost exemplificată pe un convertor ČUK cu comandă integrativă după o variabilă, instabilitatea prognozată de model fiind verificată prin simulare.

Sunt apoi prezentate noi modele de semnal mare și de semnal mic pentru convertoarele cvasirezonante. Aceste modele sunt dezvoltate plecând de la câteva observații privind forma tipică a unor tensiuni și curenți din convertor. Se pot astfel defini factori de umplere similari cu cei din convertoarele PWM cu singura deosebire că, spre deosebire de aceștia din urmă, ei depind de mărimile stare. Și aceste modele au fost verificate prin simulare sau prin comparație cu rezultate publicate în literatură.

În continuare sunt propuse două noi modele discrete bazate pe aproximarea backward-Euler respectiv trapezoidală, fiind evidențiată superioritatea lor față de modelul clasic bazat pe dezvoltare în serie Taylor (cu aproximare forward-Euler). În plus modelele au fost verificate și experimental cu rezultate excelente.

Penultimul paragraf al capitolului este consacrat calculului exact al funcției de transfer control-ieșire a convertoarelor dc-dc cu funcționare la frecvență constantă. Atributul "exact" trebuie înțeles în sensul că expresia obținută pentru funcția de transfer este valabilă pentru orice frecvență. Plecând de la algoritmul propus de Tymerski este prezentată o procedură sistematică de deducere a expresiei analitice a funcției de transfer control-ieșire exacte. Deoarece această expresie depinde puternic de punctul static de funcționare în jurul căruia se face liniarizarea se impune un calcul exact de stare staționară. În acest scop autorul propune întâi o metodă de calcul bazată pe un algoritm de tip Newton-Raphson vectorial, pe care de altfel îl și implementează în MATLAB. Se propune soluția furnizată de un model mediat ca punct de pornire în algoritmul iterativ.

Apoi, tot pentru calculul exact al stării staționare, se prezintă o tehnică bazată pe legarea unui program de calcul matematic de un simulator de circuite, Sunt comparate din punct de vedere al performanțelor (număr de iterații și număr de simulări necesare) metoda de tip Newton cu diferențe finite și metoda cu actualizare Broyden, ambele utilizate pentru calculul Jacobianului, cu ajutorul căruia se rezolvă starea staționară. Apoi se prezintă o serie de alte aplicații pe care legarea

celor două programe le oferă și anume: reprezentarea caracteristicilor statice reale și a familiilor de forme de undă în stare staționară și simularea convertoarelor dc-dc cu controller-e digitale sau controllere fuzzy.

Având starea staționară rezolvată, este dedusă expresia exactă a funcției de transfer control-ieșire și sunt reprezentate caracteristicile de amplitudine și de fază pentru următoarele tipuri de convertoare: convertoare PWM DCVM, PWM DIVM, PWM CCM cu feedforward, PWM CCM cu comandă prin sarcină, PWM CCM cu comandă integrativă după o variabilă. De remarcat că pentru convertoarele PWM CCM cu feedforward se dă o expresie unitară atât pentru feedforward realizat cu multiplicatoare cât și pentru feedforward realizat cu integroare, reprezentările arătând coincidența caracteristicilor până la aproximativ jumătate din frecvența de comutație. Este de asemenea prezentată o expresie generală exactă a funcției de transfer control-ieșire pentru orice tip de comandă integrativă în CCM și o expresie generală a aceleiași funcții pentru orice tip de comandă integrativă în moduri cu conducție discontinuă. În final este enunțată o teoremă care stabilește o condiție necesară pentru ca funcția de transfer dată de modelul mediat și funcția de transfer exactă să coincidă. Pentru o clasă de convertoare, pentru care Kassakian arată că reacția feedforward asigură eroare dinamică nulă, autorul tezei demonstrează că aceeași categorie de convertoare posedă proprietatea că funcția de transfer control-ieșire exactă și mediată coincid.

*Capitolul 7*, care poate fi privit ca unul introductiv în aplicații ale convertoarelor dc-dc în alte tipuri de conversie a energiei, punctează două soluții de circuite PFC bazate pe structuri QRC ZCS și PWM. Sunt determinate principalele relații ce stau la baza circuitelor PFC realizate cu convertoare ZCS, făcându-se considerații de proiectare și alegere a tipului de convertor și a modului de lucru. Este apoi prezentată o aplicație a convertoarelor DIVM ca circuite de corecție a factorului de putere pentru tensiuni de ieșiri mici.

Pe parcursul tezei au fost efectuate numeroase simulări. Ca și simulatoare de circuite au fost utilizate programul CASPOC (Simulation Research) și PETS (Boulder University Colorado) utilizat cu postprocesorul grafic PROBE din PSPICE. În ultimul capitol s-a fost folosit programul PSIM (Powersim Technologies Inc.). Pentru determinarea și trasarea diverselor caracteristici, precum și pentru programele de proiectare asistată a fost utilizat programul MATLAB, iar pentru simularea anumitor modele dinamice de semnal mare mediul SIMULINK. Antrenarea rețelei neuronale din capitolul 5 s-a realizat cu programul Neural Graphics. Anexa cuprinde toate programele dezvoltate de autor. S-a preferat, cu prețul creșterii dimensiunii Anexei, să se prezinte integral programele pentru fiecare aplicație în parte, chiar dacă multe din acestea aveau segmente comune. În acest fel ele se pot utiliza exact ca în Anexă, fără a se face trimiteri sau alte modificări. Calculatorul pe care s-au executat programele a fost un 486DX/40MHz rulând sub Windows95. Toate formele de undă au fost importate de pe un osciloscop digital Philips PM 3350A.

În ceea ce privește notațiile și abrevierile s-a optat pentru denumirile în limba engleză. Aceasta nu din cosmopolitism sau snobism, ci pentru că din experiența dobândită de autor a reieșit că ele au devenit unanim acceptate pe plan mondial, chiar și de țări destul de reticente la aceste aspecte, cum sunt Franța sau Germania.

În timpul anilor de pregătire a acestei teze foarte multe persoane mi-au fost de un real și deosebit ajutor. În particular aș dori să mulțumesc:

- prof. dr. ing. Mircea Ciugudean, care, în calitate de conducător științific al tezei, mi-a dat deplina libertate în explorarea domeniului, oferindu-mi un puternic sprijin în plan științific, dublat de exigență, dar și de o mare înțelegere, disponibilitate și echilibru în momentele de cumpănă. Domnia sa a fost principalul catalizator fără de care această teză ar fi avut puține șanse de reușită.
- membrilor comisiei de doctorat: prof. dr. ing. Dimitrie Alexa, prof. dr. ing. Șerban Lungu, și prof. dr. ing. Viorel Popescu, pentru bunăvoința de a fi acceptat să parcurgă lucrarea spre a o recenza, precum și pentru aprecierile și sugestiile pertinente făcute.

- prof. dr. ing. Alimpie Ignea, decanul Facultății de Electronică și Telecomunicații din Timișoara, pentru sprijinul logistic acordat permanent în perioada doctoratului.
- prof. dr. ing. Tiberiu Mureșan și prof. dr. ing. Viorel Popescu care m-au format, au crezut în mine oferindu-mi șansa de a mă perfecționa și mi-au influențat puternic modul de a gândi.
- ing. Ion Alexandru-Neag și ș.l. ing. Cristian Gavrilăscu pe care i-am simțit alături în perioadele de colaborare, presărate cu discuții fructuoase și pline de substanță.
- prietenului meu, prof. Peter Johann van Duijsen de la Technical University Delft, care mi-a oferit gratuit și complet dezinteresat simulatorul CASPOC, precum și o bogată bibliografie.
- prof. Dragan Maksimović de la Boulder University Colorado și prof. Predrag Pejović de la Technical University Belgrade care, fără ezitări, mi-au pus la dispoziție simulatorul PETS și o serie de informații valoroase legate de probleme de simulare a circuitelor de putere.
- colectivului catedrei de Electronică Aplicată care m-a înconjurat întotdeauna cu un climat de seriozitate și prietenie.
- familiei, pentru sprijinul și ajutorul permanent, dus uneori până la sacrificiu.

## CAPITOLUL 1

### Sinteza convertoarelor dc-dc cu modulația impulsurilor în durată (PWM)

Din punct de vedere istoric convertoarele dc-dc cu modulația impulsurilor în durată au apărut nu pe baza unei analize unitare ci spontan, la diferite intervale de timp și descoperite de persoane diferite. Excepție face convertorul ĆUK, el fiind rezultatul cercetărilor prof. Slobodan Ćuk (California Institute of Technology) [27] care și-a propus obținerea unui convertor cu cât mai multe calități posibile (număr minim de componente, pulsații de curent mici atât la intrare cât și la ieșire, atât ridicător cât și coborâtor de tensiune). Se menționează că toate metodele de sinteză prezentate în continuare presupun, dacă nu se precizează nimic, regimul de funcționare în conducție continuă (CCM).

#### 1.1. Sinteza după natura sursei și sarcinii.

Ferrieux și Forest [41] propun o generare sistematică a convertoarelor dc-dc PWM folosind noțiunile de surse de tensiune sau surse de curent continuu și sarcini de tip tensiune și de tip curent (capacitive respectiv inductive), conform schemei bloc din Fig. 1.1. Conectarea surselor și sarcinilor se face prin elemente de comutație, a căror comandă realizează modificarea (controlul) mărimii de ieșire.

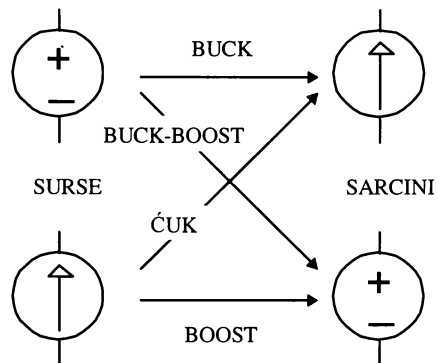


Fig. 1.1. Generarea convertoarelor PWM prin conectarea diferitelor tipuri de surse și sarcini.

Dacă sursele de tensiune și de curent sunt elemente de circuit cunoscute, sarcinile de tip tensiune respectiv curent sunt cele din Fig. 1.2. Se deduce ușor că denumirea de tip "tensiune" sau "curent" reiese din natura portului cu care sarcina se leagă spre sursă (cu capacitate paralel pentru sarcina de tip tensiune sau cu bobină serie pentru sarcina de tip curent). Se observă că în fond sarcina de tip tensiune este un filtru  $R-C$  de ordinul 1 în tensiune iar sarcina de tip curent un filtru  $L-R$  de ordinul 1 în curent. Totuși, în cele mai multe situații, la sarcina de tip curent se adaugă și o capacitate în paralel cu sarcina  $R$  care nu schimbă natura de curent a portului, dacă  $L$  are o valoare suficient de mare. Evident, în această abordare există 4 structuri de bază și anume: BUCK, BOOST, BUCK-BOOST și ĆUK deoarece există 4 posibilități de asocieri distincte între cele două tipuri de surse și cele două tipuri de sarcini.

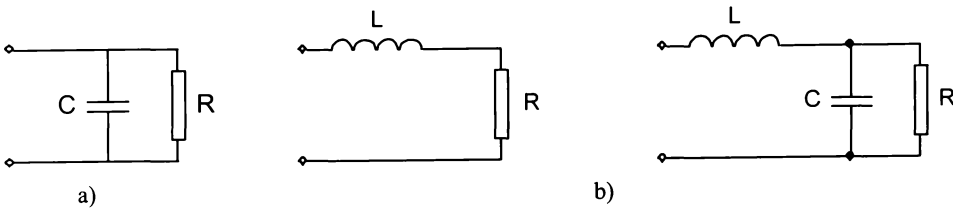


Fig. 1.2. a) sarcină de tip tensiune. b) sarcini de tip curent.

De exemplu convertorul BUCK se obține conectând prin intermediul unor elemente de comutație o sursă de tensiune cu o sarcină de tip inductiv, ca în Fig. 1.3. Deoarece o sursă de tensiune și una de curent se pot conecta atât în serie cât și în paralel fără a apare incompatibilitate, apare firească utilizarea unui comutator bipozițional care în poziția 1 aplică sursa de tensiune de la intrare pe sarcină, iar în poziția 2 deschide o cale de închidere a curentului sursei  $I_o$ , ca în Fig. 1.3a. Materializarea cu elemente de circuit concrete este prezentată în Fig. 1.3b, în care comutatorul bipozițional este înlocuit cu un întrerupător (de ex. tranzistor) la care aducerea în conducție și blocarea sunt ambele comandate cu un semnal extern și o diodă.

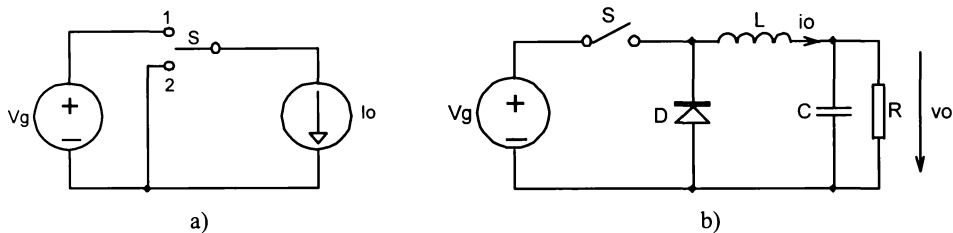


Fig. 1.3. Obținerea convertorului BUCK clasic. a) schema de principiu utilizând un întrerupător. b) materializare practică.

În cazul în care se asociază surse și sarcini de aceeași natură cu ajutorul unor elemente de comutație, care în anumite poziții realizează conectare directă, apare incompatibilitate (de ex. surse de tensiune conectate în paralel sau surse de curent conectate în serie). Pentru a evita conectarea directă sursa și sarcina se conectează pe rând la un element capabil să stocheze energie, deci un element reactiv. Într-una din pozițiile elementului de comutație acesta preia energie de la sursă, pentru ca în următoarea să cedeze această energie sarcinii. Natura elementului reactiv este dată de natura sursei și sarcinii: dacă acestea sunt de tip tensiune elementul reactiv va fi inductiv (natură de tip curent pentru a nu avea incompatibilitate la conectarea directă a sursei și sarcinii la el), iar dacă sursa și sarcina sunt de tip tensiune elementul stocant va fi capacitiv. Pentru exemplificare în Fig. 1.4a se prezintă obținerea convertorului BUCK-BOOST iar în Fig. 1.4b realizarea practică a sa.

Punctul slab al acestui procedeu îl reprezintă faptul că nu se spune ce elemente de comutație trebuie utilizate și cum anume se conectează în schemă atunci când se trece de la schema de principiu la cea practică, fapt ce poate crea unele dificultăți. Exemplul de la sinteza convertorului BUCK-BOOST este edificator în acest sens. În plus, surse de curent constant ideale nu există în practică, ceea ce se echivalează cu o sursă de curent constant fiind de regulă o sursă de tensiune constantă în serie cu o inductivitate suficient de mare prin care curentul să prezinte pulsații reduse. De aceea apare un element pasiv suplimentar (bobina serie cu sursa de tensiune) care nu era prezent

în procedura inițială de sinteză (vezi sinteza convertorului BOOST) și care față de schema de principiu mărește cu o unitate ordinul circuitului.

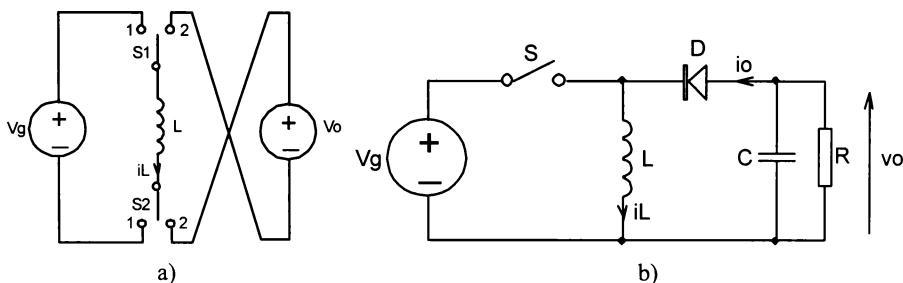


Fig. 1.4. Obținerea convertorului BUCK-BOOST clasic. a) schema de principiu utilizând un întrerupător. b) materializare practică.

Convertoarele obținute printr-o astfel de procedură de sinteză sunt date în Tab. 1.1.

Tab. 1.1. Sinteza convertoarelor PWM după natura sursei și sarcinii

Sursele și sarcinile conectate ( $V =$ tensiune, $I =$ curent, prima este specificată sursa)	Convertorul obținut
$V + I$	BUCK clasic
$V + V$	BUCK-BOOST clasic
$I + V$	BOOST cu alimentare de la o sursă de curent Dacă se înlocuiește sursa de curent cu o sursă de tensiune în serie cu o bobină se obține BOOST clasic
$I + I$	ĆUK cu alimentare de la o sursă de curent Dacă se înlocuiește sursa de curent cu o sursă de tensiune în serie cu o bobină se obține ĆUK clasic

## 1.2. Sinteza prin cascadare.

O altă abordare este prezentată de Severns și Bloom în [167]. Se consideră ca *structuri elementare* convertoarele BUCK și BOOST, convertoarele BUCK-BOOST și ĆUK, rezultând prin cascadarea structurilor elementare. Procedura are la bază următoarea teoremă:

**Teorema 1.** Orice convertor format prin cascadarea alternantă de convertoare BUCK și BOOST poate fi redus la o topologie având un singur întrerupător comandat și o singură diodă.

În Fig. 1.5 se prezintă sinteza convertorului BUCK-BOOST prin cascadarea unui convertor BUCK cu un convertor BOOST.

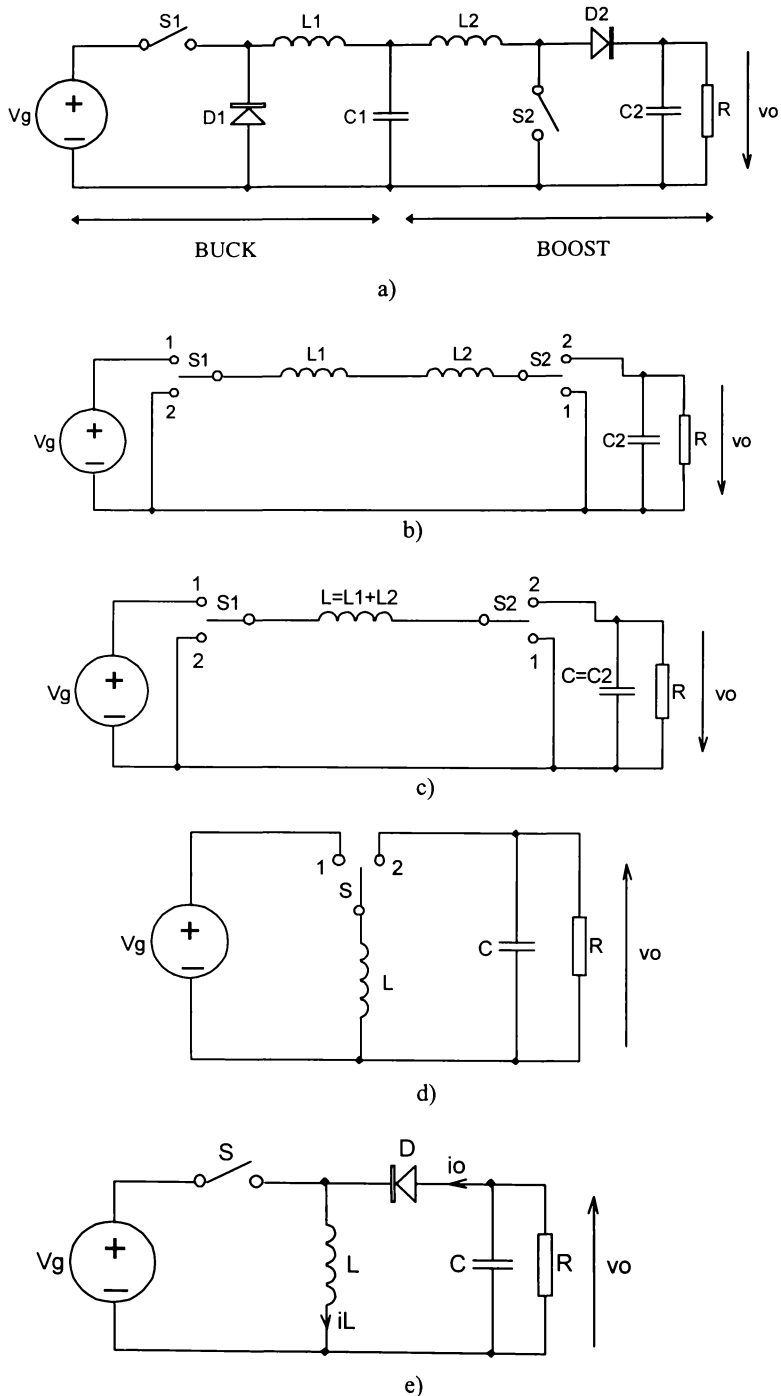


Fig. 1.5. Sinteza prin cascaderă a convertorului BUCK-BOOST.



În schema din Fig. 1.5b condensatorul  $C_1$  este înlăturat față de schema din Fig. 1.5a. Justificarea este aceea că în Fig. 1.5a grupul  $L_1 - C_1 - L_2$  formează un filtru trece-jos de ordinul 3 în T, dar raportul static de conversie nu depinde de ordinul filtrului trece jos utilizat, deci se poate renunța la  $C_1$ . Totodată perechile  $S_1 - D_1$  și  $S_2 - D_2$  din Fig. 5a se pot echivala cu comutatoarele cu 2 poziții  $S_1$  și  $S_2$  din Fig. 1.5b. În Fig. 5c  $L_1$  și  $L_2$  sunt combinate într-o singură bobină  $L$ . Dacă se acceptă inversarea polarității tensiunii de ieșire schema se poate simplifica în continuare ca în Fig. 1.5d, deoarece în această situație cele două comutatoare apar cu borna 2 comună și se pot înlocui cu un singur comutator. În sfârșit, în Fig. 1.5e se prezintă materializarea practică a convertorului înlocuind comutatorul cu un întrerupător și o diodă. Similar, convertorul ĆUK se obține prin cascada unui convertor BOOST cu un convertor BUCK - Fig. 1.6.

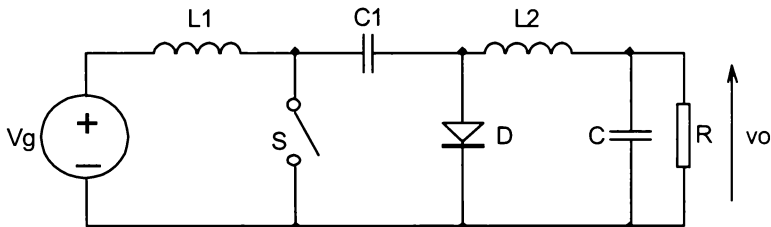


Fig. 1.6. Convertorul ĆUK.

Sunt cascadeate apoi convertoarele BUCK-BOOST și ĆUK cu convertoare elementare obținându-se topologiile ZETA - Fig. 1.7 și SEPIC - Fig. 1.8. Cele 4 structuri obținute prin cascada, împreună cu convertoarele elementare BUCK și BOOST au fost numite în literatură *topologii de bază*. Procedura de sinteză prin cascada ale celor 6 topologii de bază este rezumată în Tab. 1.2.

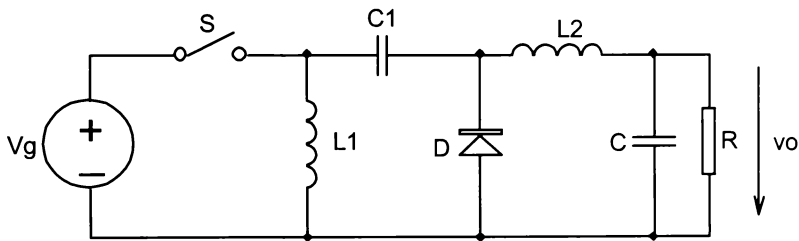


Fig. 1.7. Convertorul ZETA.

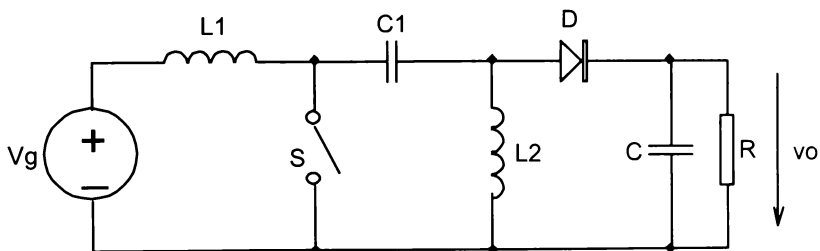


Fig. 1.8. Convertorul SEPIC.

Tab. 1.2. Sinteza prin cascada a celor 6 topologii de bază.

Convertorul	Convertoarele care intră la sinteza prin cascada
BUCK clasic	structură elementară
BOOST clasic	structură elementară
BUCK-BOOST clasic	BUCK+BOOST
ĆUK clasic	BOOST+BUCK
SEPIC clasic	BOOST+BUCK-BOOST ĆUK+BOOST BOOST+BUCK+BOOST
ZETA clasic	BUCK-BOOST+BUCK BUCK+ĆUK BUCK+BOOST+BUCK

Și în cazul sintezei prin cascada apare dificultatea simplificării convertorului inițial și reducerea acestuia la un convertor final cu aceeași caracteristică statică și cu număr minim de întrerupătoare active, pasive și elemente pasive de circuit. Aceasta pentru că nu sunt date în acest sens nici un fel de reguli, totul ținând de abilitatea celui ce efectuează sinteza, ceea ce la structuri mai complicate devine dificil.

### 1.3. Metoda de sinteză având la bază întrerupătorul PWM

Examinarea structurilor convertoarelor BUCK și BUCK-BOOST din Fig. 1.3b și respectiv Fig. 1.4b relevă prezența unui ansamblu de elemente comun ambelor convertoare: întrerupătorul  $S$  și dioda  $D$ . Dacă se redesenează convertorul BOOST ca în Fig. 1.9 regăsim din nou perechea  $S - D$  conectată în aceeași configurație.

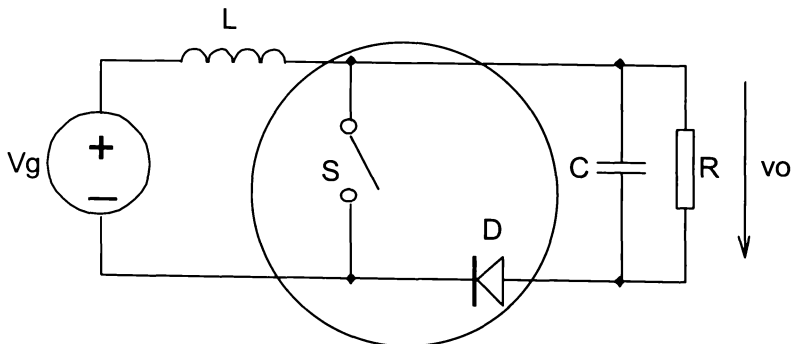


Fig. 1.9. Convertorul BOOST și identificarea întrerupătorului PWM.

Dispozitivul cu 3 terminale, prezentat în Fig. 1.10, a fost denumit *întrerupător (switch) PWM* [38], [39], [195].

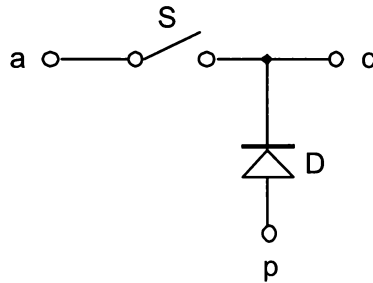


Fig. 1.10. Întrerupătorul PWM.

Terminalele au fost în mod firesc notate “activ”-  $a$ , “pasiv”-  $p$  și “comun”-  $c$  după cum ele sunt legate de întrerupătorul activ  $S$ , de întrerupătorul pasiv  $D$  sau de punctul lor comun.

Dacă întrerupătorului PWM i se adaugă o bobină rețeaua cu 3 terminale care se formează și prezentată în Fig. 1.11 poate fi privită ca o celulă de bază deoarece prin conectarea ei între sursă și sarcină se obțin convertoarele BUCK, BOOST și BUCK-BOOST clasice. Pentru prima oară s-a arătat în [195], [39] și apoi în [6] că există 3 posibilități distincte de conectare a acestei celule între sursă și sarcină. Deci convertoarele BUCK, BOOST și BUCK-BOOST pot fi privite ca fiind rezultate din aceeași celulă de bază conectată diferit între sursă și sarcină.

O altă celulă de bază cu trei terminale se obține prin alăturarea unei capacități la întrerupătorul PWM, ca în Fig. 1.12. Pentru a avea curenți de intrare și ieșire fără discontinuități (denumiți impropriu în literatură și nepulsatorii) se impune conectarea de inductivități în serie cu sursa și sarcina. Din nou există 3 posibilități distincte de conectare a acestei celule între sursă și sarcină. Se obțin astfel convertorul BUCK cu filtru  $L-C$  la intrare din Fig. 1.13, un convertor BOOST cu filtru  $L-C$  la ieșire, ca în Fig. 1.14 și convertorul ĆUK - Fig. 1.15. Rezultatele obținute prin sinteza plecând de la cele două celule sunt prezentate în Tab. 1.3.

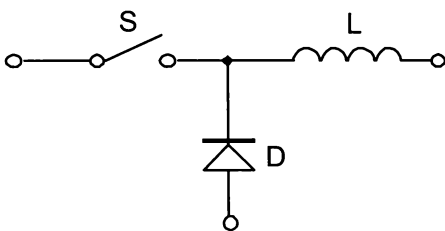


Fig. 1.11. Celulă de bază (întrerupător PWM și bobină).

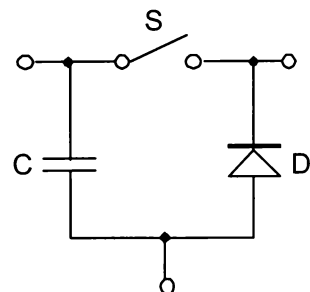


Fig. 1.12 Celulă de bază (întrerupător PWM și capacitate).

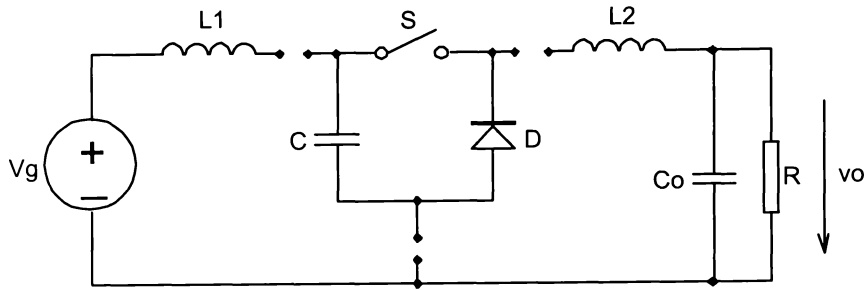


Fig. 1.13. Obținerea convertorului BUCK cu filtru  $L-C$  la intrare plecând de la celula de bază din Fig. 1.12.

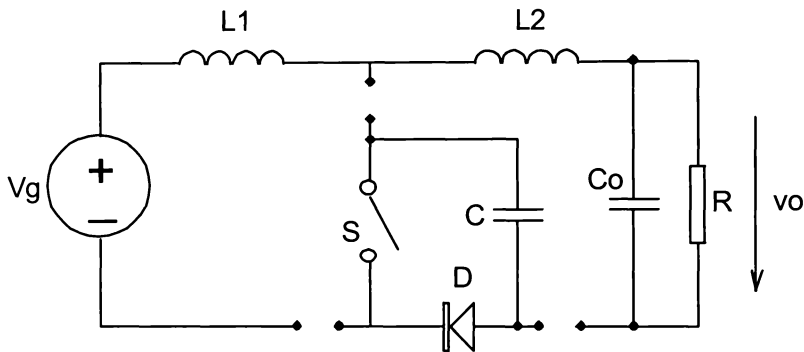


Fig. 1.14. Obținerea convertorului BOOST cu filtru  $L-C$  la ieșire plecând de la celula de bază din Fig. 1.12.

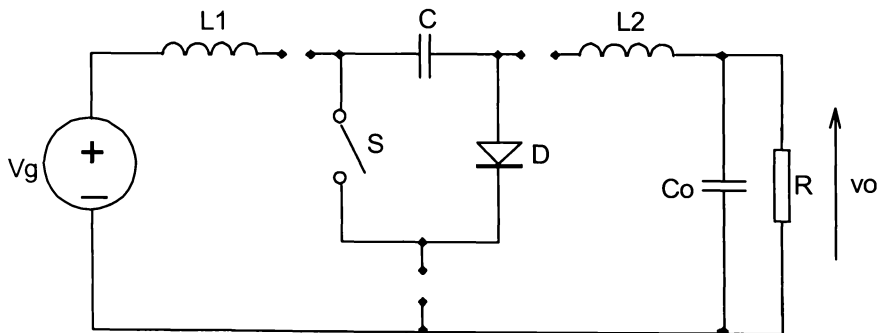


Fig. 1.15. Obținerea convertorului ĆUK clasic plecând de la celula de bază din Fig. 1.12.

Tab 1.3. Sinteza cu celule de bază construite pe întrerupătorul PWM.

Celula de bază	Convertoare obținute
Întrerupător PWM + $L$	BUCK clasic BOOST clasic BUCK-BOOST clasic
Întrerupător PWM + $C$	BUCK cu filtru $L-C$ la intrare BOOST cu filtru $L-C$ la ieșire ĆUK clasic

#### 1.4. Sinteza prin dualitate

Principiul dualității este un concept fundamental în circuitele electrice. Totuși cele mai multe aplicații ale sale au avut ca subiect circuitele liniare sau cele cu neliniarități simple. Până la rezultatele obținute de Ćuk [27] nu era clar dacă este posibil și cum să se aplice dualitatea la convertoarele în comutație, care sunt circuite profund neliniare. Abordarea propusă de Ćuk se bazează pe reducerea convertorului la o succesiune de circuite liniare echivalente pentru fiecare stare topologică și găsirea circuitului dual aferent fiecărei topologii. Apoi circuitele duale rezultate sunt recombinate într-un singur circuit, din nou neliniar. Înainte de a intra propriu-zis în studiul principiului dualității și a regulilor pentru generarea circuitelor duale se va consacra un scurt paragraf elementelor de comutație care conferă neliniaritate convertoarelor în comutație, deoarece sunt părți constitutive mai puțin investigate din componența acestora.

##### 1.4.1. Tipuri de elemente de comutație și implementarea lor practică.

Așa cum s-a observat în paragrafele precedente comutatorul bipozițional din Fig. 1.16 apare ca element constitutiv în generarea diferitelor structuri de convertoare (vezi Fig. 1.3a, Fig. 1.4a, Fig. 1.5b). Totuși această structură nu se poate materializa în practică cu un singur element de circuit. Din fericire, comutatorul bipozițional se poate echivala cu două întrerupătoare, pentru început complet comandate, funcționând în contratimp - Fig. 1.17. Deci studiul elementelor de comutație se va transfera pe modalități de implementare a întrerupătoarelor [Erickson, 39].

Fie întrerupătorul din Fig. 1.18. El conține terminalele de putere notate cu 0 și 1, cu sensurile tensiunii și curentului aferente lui aceleași, (definite și în figură) și fixate. Evident, în stare de conducție tensiunea  $v_S$  pe întrerupător este zero, curentul prin întrerupător putând lua orice valoare. În stare de blocare cel care este zero este curentul  $i_S$  prin întrerupător, tensiunea  $v_S$  putând lua orice valoare.

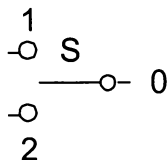


Fig. 1.16. Comutatorul bipozițional

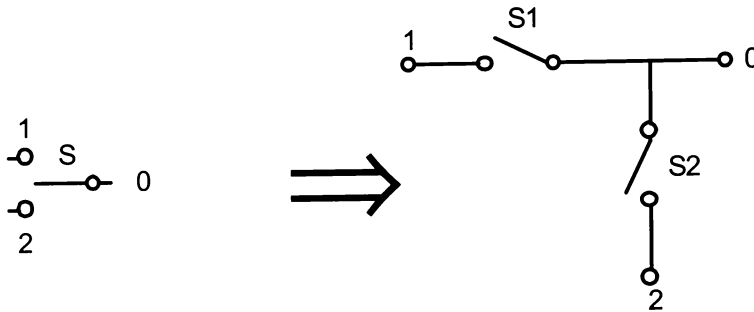


Fig. 1.17. Echivalarea comutatorului bipozițional cu două întrerupătoare.

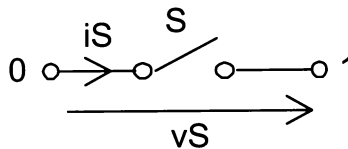


Fig. 1.18. Definirea sensurilor pentru tensiunea și curentul aferente unui întrerupător

Rezultă că în planul  $v_S - i_S$  caracteristica întrerupătorului coincide, în situația când este cea mai extinsă posibil, cu axele de coordonate. Întrerupătorul mai poate sau nu avea un terminal de comandă, ce nu va fi figurat în schemele analizate în continuare. Diferitele performanțe ale unui întrerupător se referă la metoda de comandă a sa (întrerupător activ sau pasiv) și la regiunea din planul  $v_S - i_S$  în care funcționează.

Un *întrerupător pasiv* nu are terminal de comandă. Starea întrerupătorului este determinată de mărimile  $i_S(t)$  și  $v_S(t)$  aplicate terminalelor 0 și 1. Cel mai întâlnit exemplu de întrerupător pasiv este dioda.

La un *întrerupător activ* starea de conducție este determinată de semnalul aplicat terminalului de comandă și abia apoi de formele de undă  $i_S(t)$  și  $v_S(t)$  aplicate terminalelor 0 și 1. Tranzistorul bipolar (BJT), tranzistorul MOS, tranzistorul bipolar cu poartă izolată (IGBT) și tiristorul cu comandă MOS (MCT) sunt exemplele cele mai des utilizate de întrerupătoare active.

Pentru a determina cum să se implementeze un întrerupător utilizând un tranzistor (BJT, MOS sau IGBT) și/sau o diodă se vor determina pozițiile punctelor de funcționare ale întrerupătorului în planul  $v_S - i_S$ , poziții date de *valorile instantanee* de curent și tensiune din circuitul concret. Atunci când este plasat într-un anumit circuit, punctul de funcționare se poate situa și într-un subdomeniu al axelor de coordonate. Evident, deoarece suntem interesați în primul rând de polaritatea mărimilor mai sus menționate, *semiaxele* vor fi *domeniile elementare* de interes. Sunt posibile următoarele situații (se vor menționa și câteva exemple des întâlnite de implementări practice, fără a avea pretenția că acestea sunt unicele):

1. Funcționarea pe semiaxele unui singur cadran (ambele mărimi au o singură polaritate)
  - a) funcționarea este în cadranul I - Fig. 1.19. În cele mai frecvente cazuri materializarea practică se face cu un tranzistor bipolar, un IGBT sau un MOS de putere.

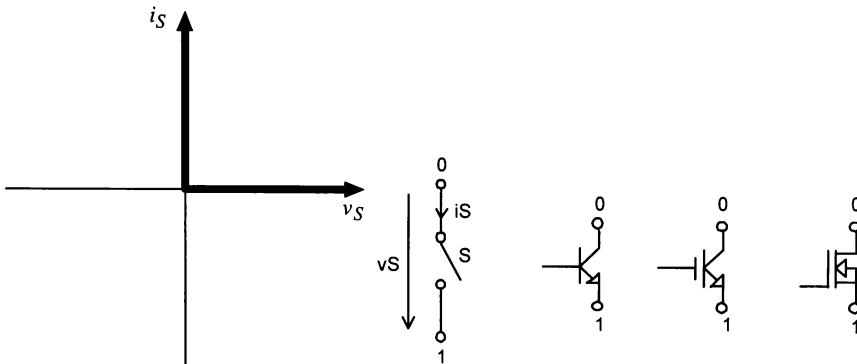


Fig. 1.19. Întreprător în cadranul I și materializarea sa practică.

b) funcționarea este în cadranul II - Fig. 1.20. Materializarea se face cu o diodă.

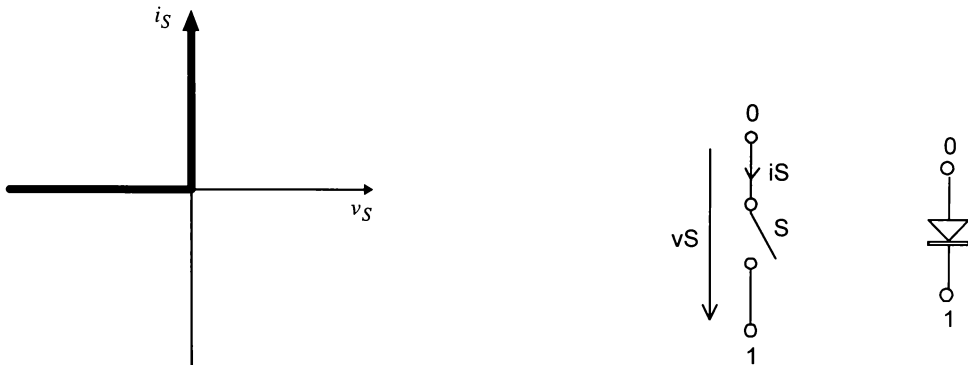


Fig. 1.20. Întreprător în cadranul II și materializarea sa practică.

c) funcționarea este în cadranul III - Fig. 1.21. Materializarea se face cu elementele specificate în cazul a) conectate invers față de situația de la a).

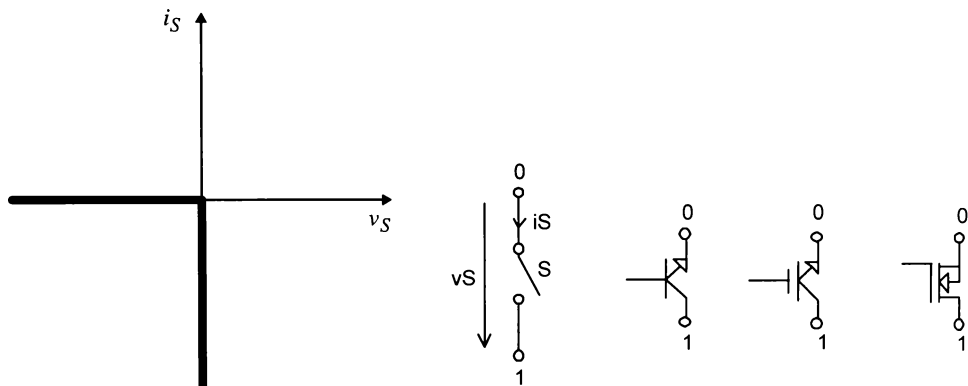


Fig. 1.21. Întreprător în cadranul III și materializarea sa practică.

Prof. Dr. Ing. G. Bănuș

- d) funcționarea este în cadranul IV - Fig. 1.22. Materializarea se face cu o diodă conectată invers față de situația din cazul b).

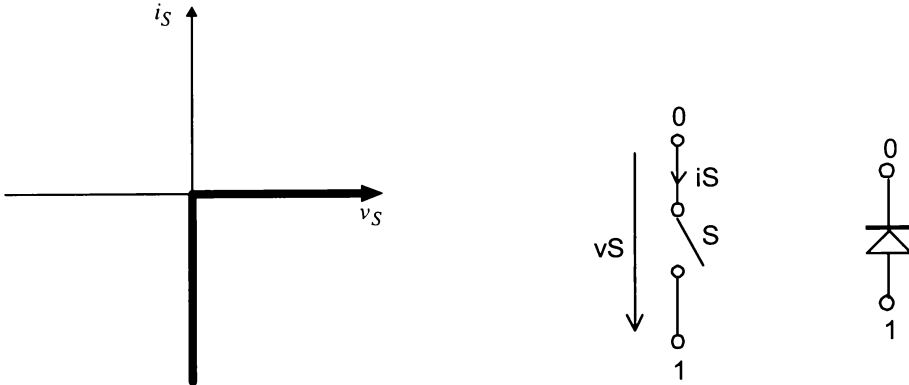


Fig. 1.22. Întrerupător în cadranul IV și materializarea sa practică.

2. Funcționarea pe o semiaxă și o axă (o mărime nu-și schimbă semnul, cealaltă fiind de ambele polarități) - funcționarea în 2 cadrane adiacente.

- a) funcționarea este în cadranele I și II - Fig. 1.23. Materializarea se face cu un tranzistor bipolar sau MOS în serie cu o diodă.

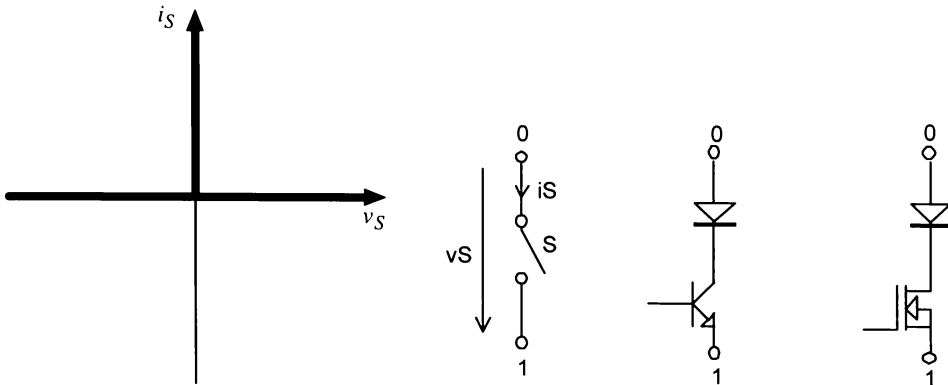


Fig. 1.23. Întrerupător în cadranele I-II și materializarea sa practică.

- b) funcționarea este în cadranele III și IV - Fig. 1.24. Materializarea se face ca în cazul a) cu elementele conectate invers față de cazul a).



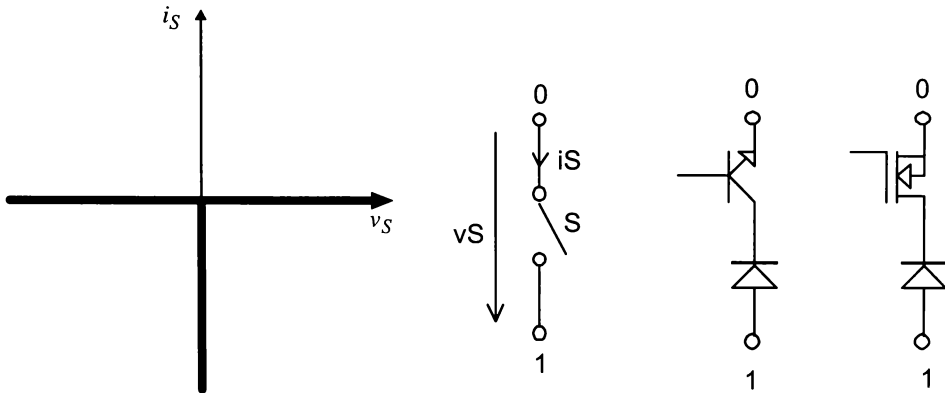


Fig. 1.24. Întrerupător în cadranele III-IV și materializarea sa practică.

- c) funcționarea este în cadranele I și IV - Fig. 1.25. Se poate materializa fie cu un tranzistor bipolar având o diodă în antiparalel, fie cu un MOS cu diodă internă FRED, fie, dacă MOS-ul nu are diodă internă performantă, cu un MOS având inseriată o diodă și cu o diodă în antiparalel pe ansamblu.

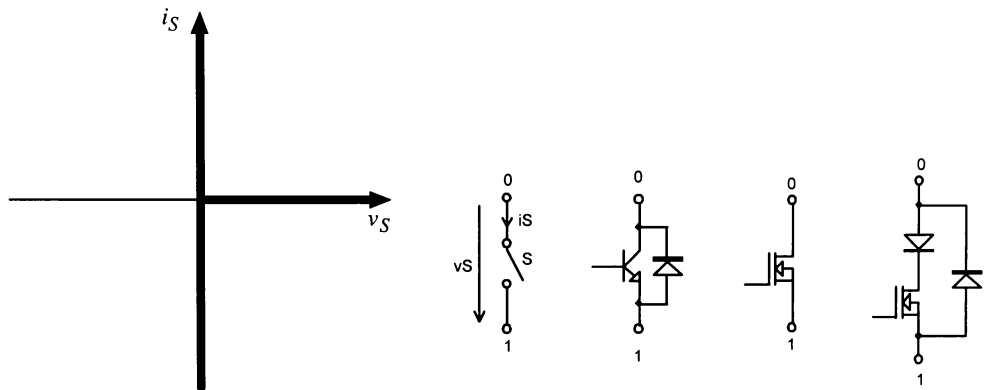


Fig. 1.25. Întrerupător în cadranele I-IV și materializarea sa practică.

- d) funcționarea este în cadranele II și III - Fig. 1.26. Materializarea se face ca în cazul c) cu elementele conectate invers față de situația precedentă.

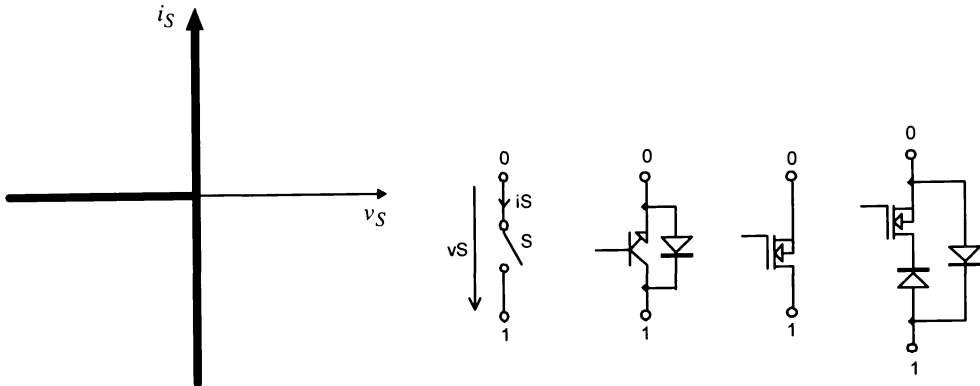


Fig. 1.26. Întrerupător în cadranele II-III și materializarea sa practică.

3. Funcționarea este pe axele întregului plan - Fig. 1.27. Trei posibile variante de implementare folosind tranzistoare bipolare sunt date alături în figură (implementările cu tranzistoare MOS sunt asemănătoare).

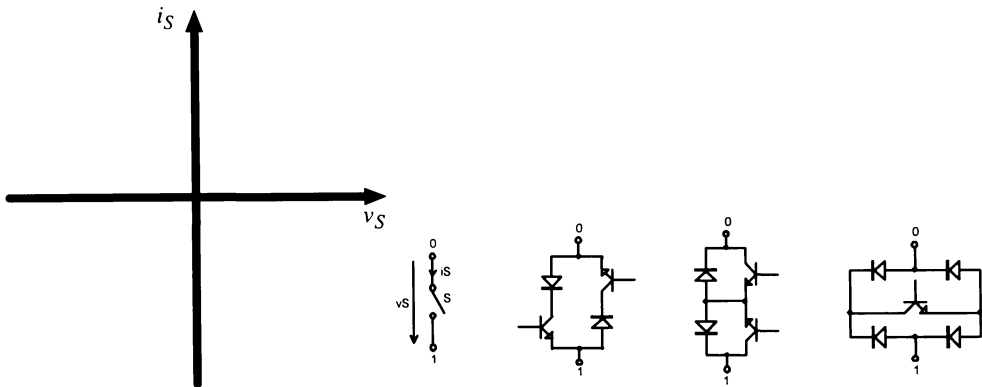


Fig. 1.27. Întrerupător în întregul plan și materializarea sa practică.

Observație. Desigur că orice implementare adevărată într-un anumit domeniu rămâne adevărată și într-un subdomeniu al acestuia. De aceea implementările găsite pentru cadranele I și IV sunt adevărate spre exemplu și în cadrantul I, la fel cum implementările găsite pentru cadranele I și II sunt adevărate în cadrantul I. Acestea nu au mai fost menționate la implementările prezentate pentru cadrantul I deoarece s-au căutat corespondențele practice cele mai simple.

#### 1.4.2. Aplicarea principiilor dualității la convertoarele în comutație.

În circuitele investigate dualitatea este de cele mai multe ori limitată la acele circuite liniare ale căror grafuri asociate sunt planare.

Definiția 1. [167] Un graf  $G$  se numește *planar* dacă poate fi desenat într-un plan astfel încât să nu existe două ramuri care să se intersecteze într-un punct care nu este nod.

De exemplu graful din Fig. 1.28a este planar în timp ce cel din Fig. 1.28b nu este planar.

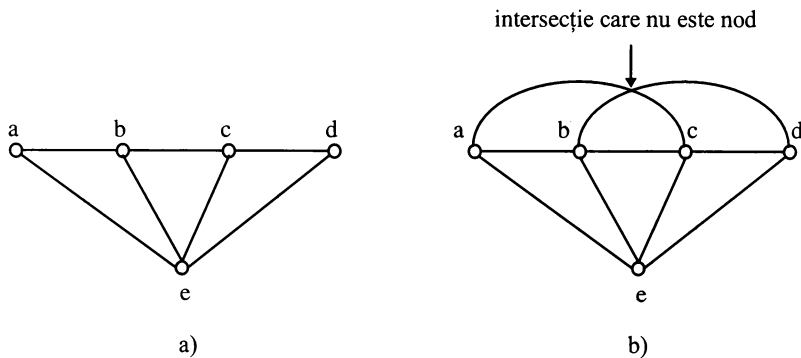


Fig. 1.28. a) graf planar. b) graf neplanar.

**Definiția 2.** Într-un graf planar orice buclă închisă care nu conține nici o ramură în interior se numește *ochi*.

În Fig. 1.28 buclele  $a-b-e-a$ ,  $c-d-e-c$  și  $b-c-e-b$  sunt ochiuri.

**Definiția 3.** Un ochi care nu conține nici o ramură în exterior se numește *ochi extern*.

**Definiția 4.** Dându-se un graf  $G$ , un graf  $G^*$  se spune că este graful *dual* lui  $G$  dacă sunt satisfăcute următoarele două condiții:

**Condiția 1.** Există o corespondență biunivocă (de unu la unu) între toate ochiurile lui  $G$  și toate nodurile lui  $G^*$ .

**Condiția 2.** Există o corespondență biunivocă între ramurile fiecărui graf astfel încât ori de câte ori două ochiuri ale unuia dintre grafuri au o ramură în comun, nodurile corespunzătoare din graful dual sunt conectate de ramura corespunzătoare ramurii în comun din graful inițial.

Și în cazul convertoarelor în comutație se presupune un anumit sens în parcurgerea unei ramuri din graful asociat, sens care coincide cu sensul curentului din latura circuitului. Acest sens este indicat de obicei pe ramură. Sensul adoptat este cel uzual, al sensului curentului prin elementele de circuit cu excepția faptului că pentru o ramură ce conține o sursă de tensiune ideală sensul curentului prin ramură se adoptă dinspre borna + spre borna - a sursei, vezi Fig. 1.29. Aceasta pentru că regulile de dualitate presupun aceleași sensuri pentru tensiunile și curenții la borne. Pentru o ramură ce conține o sursă de curent ideală sensul curentului rămâne același cu sensul generatorului.

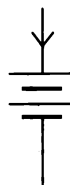


Fig. 1.29. Sensul adoptat în transformările prin dualitate pentru curentul ramurii ce conține o sursă de tensiune ideală la care tensiunea are specificată polaritatea.

**Definiția 5.** Graful asociat unui circuit la care fiecare ramură are un sens de parcurgere se numește *graf orientat*.

Există următoarea leamnă [167]:

**Lema 1.** Pentru un graf orientat  $G$  orientarea grafului dual  $G^*$  se obține rotind sensul de referință al fiecărei ramuri din  $G$  în sens *invers trigonometric* (sensul acelor de ceasornic) cu  $90^\circ$ .

Deși cele de mai sus au un caracter formal și complex, obținerea practică a grafurilor duale este destul de simplă și decurge astfel:

- se plasează câte un nod în fiecare ochi din grafurile originale și unul în afara ochiului extern. Acestea vor fi nodurile viitorului graf dual.
- se conectează ramuri între noile noduri, astfel încât fiecare nouă ramură conectată să intersecteze fiecare ramură din grafurile originale.
- din fiecare intersecție a unei ramuri din grafurile originale cu o ramură din grafurile duale se determină orientarea (sensul) ramurii din grafurile duale aplicând Lema 1.
- deoarece este o practică frecventă ca noile circuite să fie prezentate în scheme cât mai asemănătoare se pot efectua transformări de rotație, translație sau transformări în oglindă pe grafurile duale obținute. De reținut că la transformarea în oglindă sensul fiecărei ramuri a grafurilor oglindit trebuie inversat.

Fie de exemplu că dorim obținerea grafurilor duale celui din Fig. 1.30a. În Fig. 1.30b. se arată plasarea noilor noduri ale grafurilor duale (notate cu  $1^*$ ,  $2^*$ ,  $3^*$ ,  $4^*$ ), desenarea ramurilor grafurilor duale și sensurilor corespunzătoare. Grafurile dual rezultat este desenat separat în Fig. 1.30c.

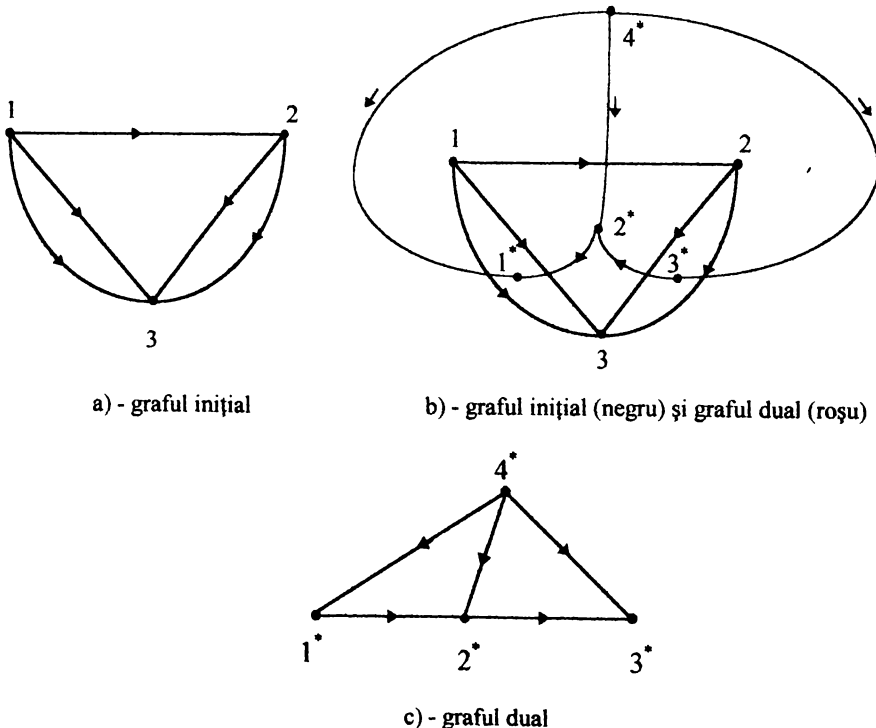


Fig. 1.30. Procedura de obținere a unui graf dual

Într-un circuit electric fiecare ramură conține componente cu diferite caracteristici electrice (rezistoare, bobine surse de tensiune sau de curent). Dualitatea presupune următoarele corespondențe:

$$v \rightarrow i^*; \quad i \rightarrow v^*; \quad q \rightarrow \phi^* \quad \phi^* \rightarrow q \quad (1)$$

unde  $v$  = tensiunea pe ramură,  $i$  = curentul prin ramură,  $q$  = sarcina ramurii,  $\phi$  = fluxul ramurii și se reamintește că simbolul \* se referă la mărimi din circuitul sau graful dual.

Relațiile dintre tensiunea și curentul la bornele principalelor elemente de circuit sunt:

$$\left\{ \begin{array}{ll} v = iR & v^* = i^* R^* \\ v = L \frac{di}{dt} & v^* = L^* \frac{di^*}{dt} \\ i = C \frac{dv}{dt} & i^* = C^* \frac{dv^*}{dt} \end{array} \right. \quad (2)$$

Din (2) se deduc ușor corespondențele prin dualitate din Tab. 1.4. De exemplu elementul dual unei bobine se deduce plecând de la relația că pentru o bobină avem

$$v = L \frac{di}{dt} \quad (3)$$

Știind că pentru elementul dual, conform (1) trebuie făcut  $v \rightarrow i^*$  și  $i \rightarrow v^*$ , rezultă că elementul dual, fie el  $X$ , trebuie să satisfacă o relație de forma:

$$i^* = X \frac{dv^*}{dt} \quad (4)$$

Comparând (4) cu (2) se observă că singura din relațiile cu variabile stelate din (2) similară cu (4) este cea corespunzătoare capacității, deci  $X$  trebuie să fie egal cu  $C$ .

Tab. 1.4. Elemente și mărimi de circuit duale.

Element	Element dual și valoarea sa
$v$	$i^*$
$i$	$v^*$
$q$	$\phi^*$
$\phi$	$q$
$R$	$G^* = R \Rightarrow R^* = \frac{1}{R}$
$G$	$R^* = G \Rightarrow G^* = \frac{1}{G}$
$L$	$C^*$
$C$	$L^*$
Generator de tensiune	Generator de curent
Generator de curent	Generator de tensiune
Circuit întrerupt (deschis)	Scurtcircuit
Scurtcircuit	Circuit întrerupt (deschis)

În cazul elementelor de comutație neliniare găsirea elementelor duale are la bază următoarea leamnă [41], consecință directă a relațiilor (1):

**Lema 2.** Două elemente (liniare sau neliniare) sunt duale dacă au caracteristicile curent-tensiune  $i = f(v)$  respectiv  $i^* = f(v^*)$  simetrice față de prima bisectoare a planului tensiune-curent. Evident că această leamnă justifică ușor echivalențele și relațiile din Tab. 1.4.

Consecințele acestei leme rezultă în următoarele propoziții, ce se pot argumenta imediat examinând Fig. 1.31, 1.32, 1.33 și respectiv 1.34.

**Propoziția 1.** Un întrerupător funcționând în cadranele I și II este dual cu un întrerupător funcționând în cadranele I și IV și reciproc.

**Propoziția 2.** Un întrerupător funcționând în cadranele II și III este dual cu un întrerupător funcționând în cadranele III și IV și reciproc.

**Propoziția 3.** Un întrerupător funcționând în cadranul I și/sau în cadranul III sau în toate cele 4 cadrane este dual cu el însuși. Deci un BJT sau IGBT sunt duale cu ele însele (cu conectare corectă din punct de vedere al sensurilor tensiunii și curentului în circuitul dual.)

**Propoziția 4.** Un întrerupător funcționând în cadranul II este dual cu un întrerupător funcționând în cadranul IV și reciproc. Deci o diodă este duală cu ea însăși (cu conectare inversă).

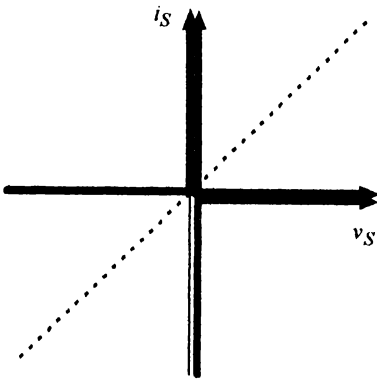


Fig. 1.31. Explicativă la propoziția 1

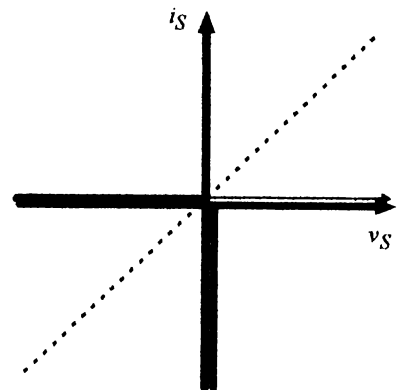


Fig. 1.32. Explicativă la propoziția 2

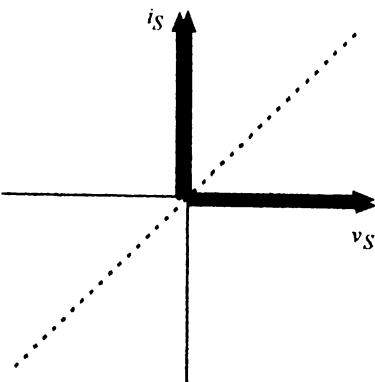


Fig. 1.33. Explicativă la propoziția 3

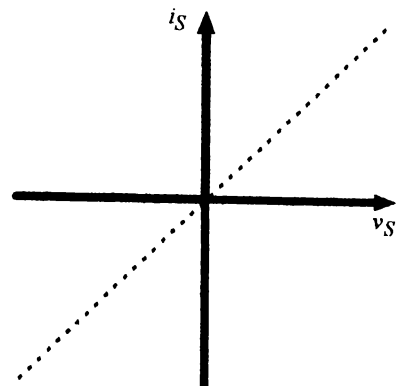


Fig. 1.34. Explicativă la propoziția 4

Ținând cont că dualul unui întrerupător este tot un întrerupător (la care eventual doar funcționarea în cadrane diferă) și luând în considerație transformările prin dualitate date de ultimele două linii din Tab. 1.4 rezultă imediat următoarea propoziție:

**Propoziția 5.** Un întrerupător funcționând cu factorul de umplere  $D$  într-un circuit va avea ca și corespondent în circuitul dual un întrerupător (cu funcționare adecvată ca și cadrane) și comutând cu factorul de umplere  $1-D$ .

Fie

$$M = \frac{V_o}{V_g} \quad (5)$$

raportul static de conversie al unui convertor în comutație presupus a funcționa cu randament unitar. În CCM se cunoaște că acest raport depinde numai de factorul de umplere  $D$ , adică  $M = M(D)$ . Notăm cu

$$M^* = \frac{V_o^*}{V_g^*} \quad (6)$$

raportul static de conversie al convertorului în comutație dual, unde  $V_g^*$  și  $V_o^*$  sunt tensiunea de intrare respectiv de ieșire din convertorul dual. Avem atunci următoarea propoziție:

**Propoziția 6.** Pentru convertorul dual, în CCM, raportul static de conversie este dat de relația:

$$M^* = \frac{1}{M(1-D)} \quad (7)$$

**Demonstrație.** Deoarece prin dualitate  $V_g \rightarrow I_g^*$ ,  $V_o \rightarrow I_o^*$ ,  $D^* \rightarrow 1-D$  din (5) rezultă că

$$M(D^*) = M(1-D) = \frac{I_o^*}{I_g^*} \quad (8)$$

Deoarece funcționarea convertorului original a fost presupusă cu randament unitar și funcționarea convertorului dual este tot cu randament unitar pentru că elementele disipative se transformă prin dualitate tot în elemente disipative. Are deci loc conservarea puterii și deci:

$$V_g^* I_g^* = V_o^* I_o^* \quad (9)$$

de unde, folosind (6) se obține:

$$M^* = \frac{V_o^*}{V_g^*} = \frac{I_g^*}{I_o^*} \quad (10)$$

Din (8) și (10) rezultă că:

$$M^* = \frac{V_o^*}{V_g^*} = \frac{I_g^*}{I_o^*} = \frac{1}{M(1-D)} \quad (11)$$

adică q.e.d.

Cele discutate până acum sugerează că dualitatea are un caracter mai larg, principiul general al dualității enunțându-se, fără a-l demonstra aici, astfel:

**Principiul general al dualității.** Fie o rețea (circuit) planară  $N$  și  $N^*$  rețeaua duală a sa. Fie  $S$  o propoziție adevărată privind comportarea rețelei  $N$ . Fie  $S^*$  propoziția obținută din  $S$  înlocuind fiecare cuvânt sau frază desemnând elemente legate de grafuri (nod, ochi, ramură etc.) prin dualul său și fiecare mărime sau element de natură electrică (tensiune, curent, impedanță, întrerupător) cu duala sa. Atunci  $S^*$  este o propoziție adevărată legată de comportarea rețelei  $N^*$ .

Există evident pentru orice convertor un convertor dual. Deoarece toate convertoarele cunoscute sunt alimentate de la o sursă de tensiune ne vom aștepta ca, plecând de la un astfel de convertor, să nu obținem prin dualitate un convertor cunoscut ci unul cu unele modificări, deoarece va fi alimentat de la o sursă de curent. Fie de exemplu convertorul BOOST clasic și ne propunem găsirea convertorului dual. Procedura de obținere este prezentată în Fig. 1.35.

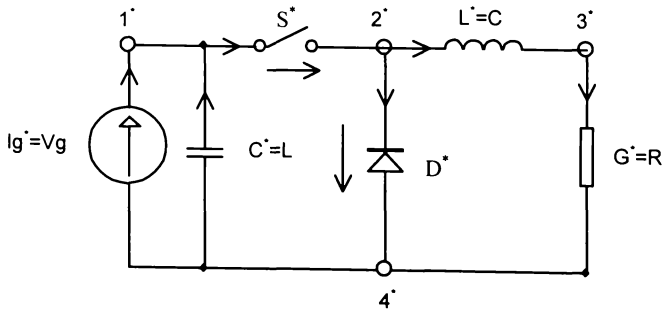
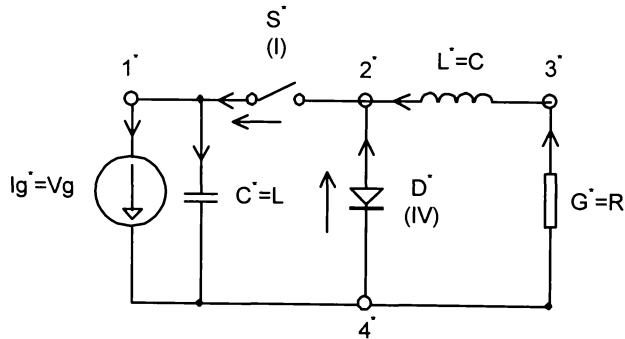
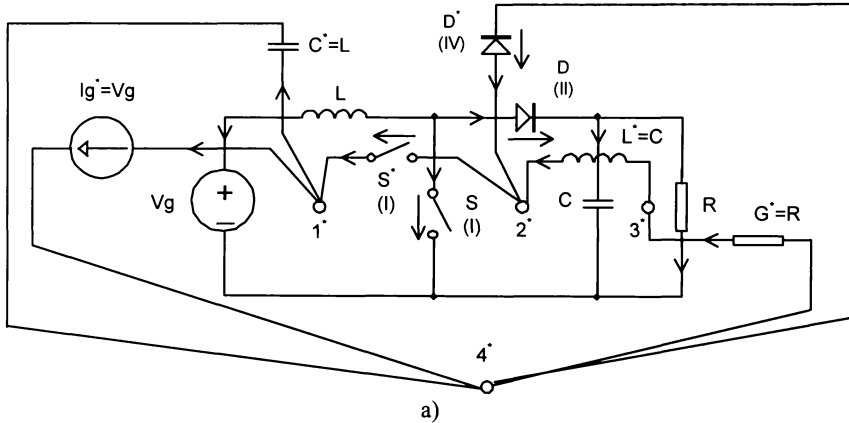


Fig. 1.35. Găsirea convertorului dual unui convertor BOOST clasic.



În Fig. 1.35a au fost plasate nodurile convertorului dual și au fost schițate elementele duale convertorului original. Pentru elementele de comutație, atât în convertorul original cât și în cel dual au fost menționate sensurile curentului și tensiunii și cadranul de funcționare. De asemenea au fost trecute și sensurile curenților aferenți ramurilor. În Fig. 1.35b a fost desenat convertorul dual așa cum rezultă direct din Fig. 1.35b. În final, în Fig. 1.35c a fost redesenat convertorul dual printr-o transformare în oglindă.

Deci dualul convertorului BOOST clasic este convertorul din Fig. 1.35 care este un convertor BUCK alimentat de la o sursă de curent constant. Rolul condensatorului  $C^*$  este de a preveni creșterea la infinit a tensiunii între nodurile 1\* și 4\* atunci când întrerupătorul  $S^*$  este deschis. Convertoarele duale celor 6 topologii de bază sunt prezentate în Tab. 1.5.

Tab. 1.5. Convertoarele duale celor 6 topologii de bază.

Convertorul	Convertorul dual
BUCK clasic	BOOST alimentat în curent și cu un filtru $L_O - G$ de ordinul 1 suplimentar la ieșire, dacă în convertorul BUCK original la ieșire filtrul era $L - C_O$ de ordinul 2
BOOST clasic	BUCK alimentat în curent
BUCK-BOOST clasic	ĆUK alimentat în curent cu un filtru $L - G$ de ordinul 1 la ieșire
ĆUK clasic	BUCK-BOOST alimentat în curent și cu filtru $L_O - G$ de ordinul 1 suplimentar la ieșire, dacă în convertorul ĆUK original la ieșire filtrul era $L - C_O$ de ordinul 2
SEPIC clasic	ZETA alimentat în curent
ZETA clasic	SEPIC alimentat în curent, fără bobina $L_1$ și cu filtru $L_O - G$ de ordinul 1 suplimentar la ieșire, dacă în convertorul ZETA original la ieșire filtrul era $L_2 - C_O$ de ordinul 2

Și în cazul dualității transformarea conduce la convertoare alimentate în curent care se materializează mai anevoios în practică și atunci când acest lucru se realizează nu se menține ordinul convertorului (similar cu sinteza după natura sursei și sarcinii).

### 1.5. Sinteza prin inversie bilaterală.

Sinteza prin inversie bilaterală a apărut ca o consecință a introducerii convertoarelor dc-dc bidirecționale. În [167] s-a arătat că un convertor oarecare, indiferent de complexitate, poate fi făcut bidirecțional din punct de vedere al transferului de putere dacă în antiparalel cu tranzistoarele cu rol de întrerupătoare active se conectează diode și în paralel cu diodele se plasează tranzistoare. Noile tranzistoare adăugate se vor comanda în contratimp (deci cu un factor de umplere complementar celor din convertorul original). Un exemplu de convertor bidirecțional plecând de la convertorul BUCK este prezentat în Fig. 1.36. Indicele "o" se referă la elementele convertorului BUCK original, pe când indicele "b" este asociat elementelor de circuit asociate ulterior. Sunt figurate și comenzile tranzistoarelor. De remarcat că pentru vehicularea energiei de la dreapta la stânga este necesar ca să existe o sursă de energie conectată la portul din dreapta al convertorului. În desen ea este reprezentată de sursa  $V_o$ , dar poate fi un simplu element reactiv dacă vehicularea

puterii în sens invers se face numai în regimuri tranzitorii. Direcția de vehiculare a energiei, pentru convertorul studiat depinde de  $V_g$ ,  $V_o$  și  $D$  astfel:

$$D \geq \frac{V_o}{V_g} \quad \text{energia se vehiculează de la stânga la dreapta} \quad (12)$$

$$D \leq \frac{V_o}{V_g} \quad \text{energia se vehiculează de la dreapta la stânga} \quad (13)$$

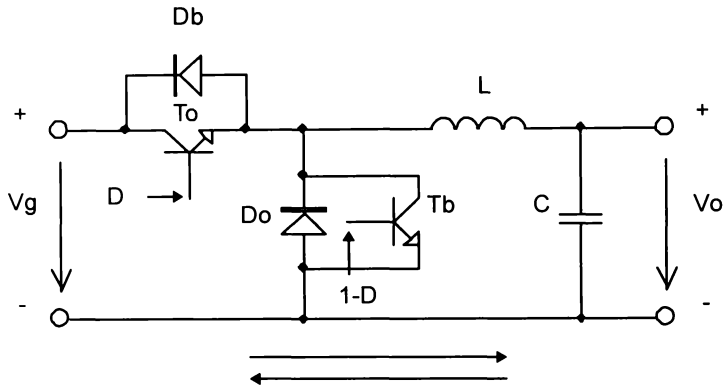


Fig. 1.36. Convertor bidirecțional plecând de la convertorul BUCK.

În [167] se arată că, plecând de la un convertor oarecare, se poate obține convertorul corespunzător prin *inversie bilaterală* parcurgând următorii pași:

Pasul 1. Fiecare tranzistor din convertorul original este înlocuit cu o diodă care să conducă curentul în sens invers decât tranzistorul.

Pasul 2. Fiecare diodă se înlocuiește printr-un tranzistor care este conectat astfel încât să conducă curentul în sens invers decât conduce dioda.

Pasul 3. Factorul de umplere  $D_b$  pentru noul tranzistor este  $D_b = 1 - D$  unde  $D$  este factorul de umplere al tranzistorului din convertorul original.

Pasul 4. Dacă  $V_g$  este tensiunea de intrare din convertorul original și  $V_o$  tensiunea sa de ieșire pentru o anumită valoare a lui  $D$ , sursa de alimentare va fi înlocuită în convertorul obținut prin inversie bilaterală cu o sarcină de potențial  $V_g$ , iar sarcina originală se înlocuiește cu o sursă de tensiune de valoare  $V_o$ .

Pasul 5. Pentru aceeași valoare a lui  $D$  în pasul 4, sarcina și capacitatea de filtraj de la ieșire au valorile:

$$R_b = R \left( \frac{V_o}{V_g} \right)^2 \quad C_b = \left( \frac{V_g}{V_o} \right)^2 \quad (14)$$

Pasul 6. La valori fixate pentru  $V_g$ ,  $V_o$ ,  $D$  și putere de ieșire fixată, convertorul obținut prin inversie bilaterală păstrează modul de funcționare al convertorului original (conducție continuă sau discontinuă din punct de vedere al curentului, etc.).

În Fig. 1.37 se exemplifică modul de obținere prin inversie bilaterală a convertorului corespunzător unui convertor BOOST clasic unidirecțional și anume: convertorul original - Fig. 1.37a, convertorul

bidirecțional obținut într-o etapă intermediară - Fig.1.37b (unde se observă factorii de umplere în comanda tranzistoarelor) și convertorul final - Fig. 1.37c (care, parcurs de la dreapta la stânga, este un convertor BUCK unidirecțional clasic).

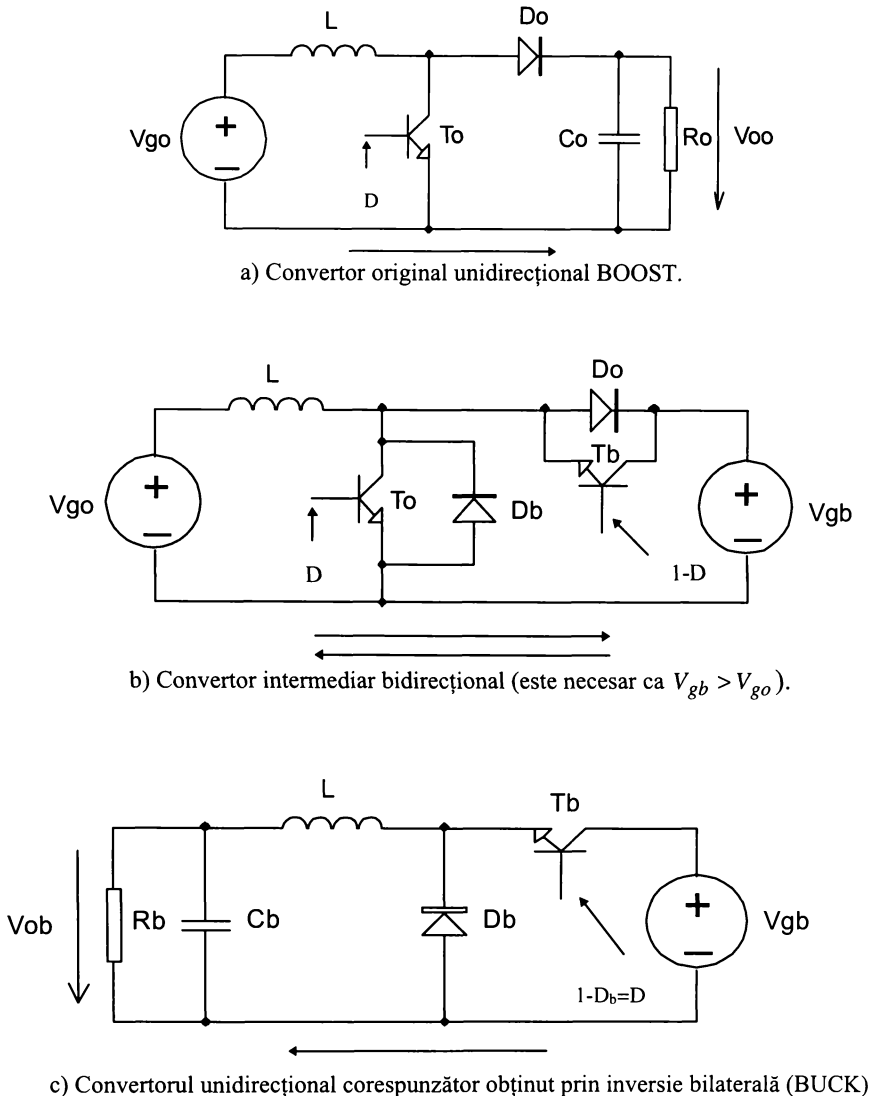


Fig. 1.37. Obținerea convertorului BUCK prin inversie bilaterală din convertorul BOOST.

În mod similar se pot demonstra corespondențele prin inversie bilaterală sintetizate în Tab. 1.6 [167]. Se observă faptul că inversia bilaterală transformă convertoarele cu stocare de energie în ele însele.

Se subliniem că deși unii autori consideră inversia bilaterală un caz particular al dualității, totuși acest lucru este eronat atâta timp cât cele două procedee, plecând de la același convertor,

conduc la convertoare diferite. Inversia bilaterală însă nu modifică ordinul convertorului, fapt ce este un avantaj față de transformarea prin dualitate.

Tab. 1.6. Transformatele prin inversie bilaterală ale celor 6 topologii de bază.

Convertor original	Convertor corespunzător prin inversie bilaterală
BUCK clasic	BOOST clasic
BOOST clasic	BUCK clasic
BUCK-BOOST clasic	BUCK-BOOST clasic
ĆUK clasic	ĆUK clasic
SEPIC clasic	ZETA clasic
ZETA clasic	SEPIC clasic

### 1.6. O nouă metodă de sinteză a convertoarelor PWM dc-dc în comutație: “metoda celulei de bază complete”

În cele ce urmează se propune o nouă metodă de sinteză a convertoarelor în comutație. Ideea de la care s-a pornit a fost a fost sugerată de următoarele observații:

1. În practică dispunem în majoritate covârșitoare de surse de energie sub formă de surse de tensiune, deci ne vom axa pe convertoare alimentate de la astfel de surse. Nu dorim deci generarea de convertoare alimentate în curent.
2. Pe cât posibil metoda de sinteză care utilizează aceeași celulă să conducă la convertoare de același ordin. Deci legarea celulei cu sursa și sarcina să se facă “direct”, fără a se mai adăuga elemente suplimentare (cum s-a văzut de exemplu în paragraful 1.3, unde sinteza convertorului ĆUK cerea două bobine suplimentare pentru a se realiza filtrajul corespunzător).
3. Pe cât posibil convertoarele să prezinte curenți de intrare și/sau de ieșire cât mai netezi.

Elementul central al sintezei este așa numita *celulă de bază completă* (pe care uneori o vom numi în acest paragraf și cu termenii de “celulă” sau “celulă de bază”, înțelegând însă că este vorba de celula de bază completă), care conține toate elementele de comutație și elementele pasive, cu excepția sursei de tensiune de alimentare și a sarcinii admise rezistive. Observațiile de mai jos se dovedesc utile în alegerea celulei.

- Obs. 1. Celula nu trebuie să conțină elemente disipative de energie pentru ca randamentul convertorului să fie, teoretic, unitar.
- Obs. 2. Deoarece dorim o conversie dc-dc dinspre sursă și dinspre sarcină spre celulă trebuie să se “vadă” porturi de curent continuu.
- Obs. 3. Deoarece presupunem sarcini cu caracter pur rezistiv natura portului “văzut” dinspre sarcină spre celulă nu are importanță, acesta putând fi fie un port de tensiune (capacitate paralel) fie un port de curent (bobină serie).
- Obs. 4. Dacă dorim ca toate convertoarele provenind din aceeași celulă să aibă același ordin atunci *toate elementele reactive trebuie să fie înglobate în celulă*. Chiar și așa, atunci când portul cu care se leagă sursa de tensiune continuă este de tensiune, ordinul convertorului obținut va fi cu o unitate mai mic deoarece capacitatea apare în paralel pe sursă. Din fericire, în

practică ea poate rămâne în circuit, sursele nefiind ideale și prezentând cel puțin o rezistență de ieșire nenulă.

Obs. 5. Ne propunem să obținem numai convertoare la care *sursa de alimentare și sarcina au o bornă comună*. În consecință rezultă că *celula de bază completă va avea trei terminale*.

În cele ce urmează se vor utiliza următoarele două teoreme și următoarea propoziție:

**Teorema 2.** În cazul unei celule de bază cu 3 terminale convertoarele rezultate din conectări ca cele din Fig. 1.38 și Fig. 1.39 conduc la una și aceeași structură.

Demonstrația acestei teoreme se găsește în mare parte în [195] și a fost completată în [39].

O consecință a acestei teoreme rezidă în aceea că este suficient a se analiza toate configurațiile posibile prin conectarea celulei de bază ca în Fig. 1.38, adică atunci când borna negativă a sursei este legată cu o bornă a sarcinii rezistive. Nu mai trebuie deci studiat cazul când borna pozitivă a sursei se leagă la sarcină. Acest lucru care se va practica în analizele ce vor urma.

**Definiția 6.** Fie un convertor dc-dc oarecare funcționând în conducție continuă și  $M$  raportul său static de conversie. Notăm  $m = M(D) \Big|_{D \in [0,1]}$ . Un convertor va fi de tip BUCK dacă  $m = [0, 1]$ , de tip BOOST dacă  $m = [1, \infty)$  și de tip BUCK-BOOST dacă  $m = (-\infty, 0]$ .

**Teorema 3.** În cazul conectării unei celule cu 3 terminale între o sursă și o sarcină ce au în comun o bornă, conform cu Fig. 1.38, condiția necesară și suficientă ca prin conectare să se obțină două convertoare de tip BUCK, două convertoare de tip BOOST și două convertoare de tip BUCK-BOOST este ca raportul  $\mu = \frac{V_{23}}{V_{13}}$  să parcurgă unul din intervalele  $[0, 1]$ ,  $[1, \infty)$  sau  $(-\infty, 0]$  atunci când factorul de umplere  $D$  parcurge intervalul  $[0, 1]$ .

**Demonstrație.** Se pleacă de la observația că raportul  $M = \frac{V_o}{V_g}$  se poate întotdeauna exprima în

funcție de  $\mu = \frac{V_{23}}{V_{13}}$ . În plus, pentru terminalul comun fixat la o bornă a celulei, cele două rapoarte de conversie care se obțin schimbând între ele terminalele sursei și sarcinii care nu sunt comune sunt inverse unul altuia. Analizând cele 6 posibilități de conectare rezultă imediat că  $M$  nu poate lua decât una din valorile:  $\mu$ ,  $\frac{1}{\mu}$ ,  $\frac{1}{1-\mu}$ ,  $1-\mu$ ,  $-\frac{\mu}{1-\mu}$ ,  $-\frac{1-\mu}{\mu}$ . Necesitatea se demonstrează arătând prin simple rezolvări de inegalități algebrice că:

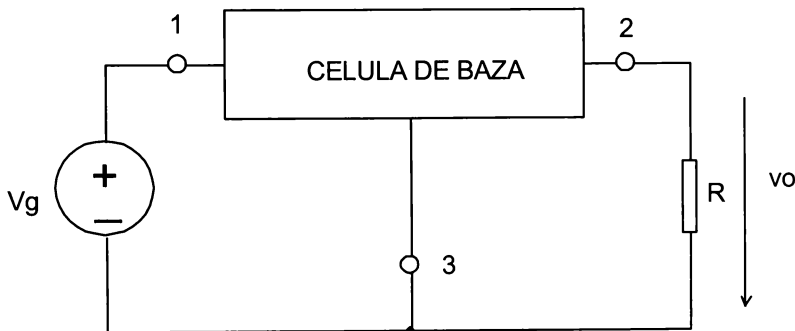


Fig. 1.38. Celula de bază cu 3 terminale ce face obiectul teoremei 2. Sursa are borna negativă comună cu sarcina.

- dacă  $\mu$  parcurge intervalul  $[0, 1]$  atunci și  $1-\mu$  parcurge intervalul  $[0, 1]$ ,  $\frac{1}{1-\mu}$  și  $\frac{1}{\mu}$  parcurg intervalul  $[1, \infty)$ , iar  $-\frac{\mu}{1-\mu}$  și  $-\frac{1-\mu}{\mu}$  parcurg intervalul  $(-\infty, 0]$ .

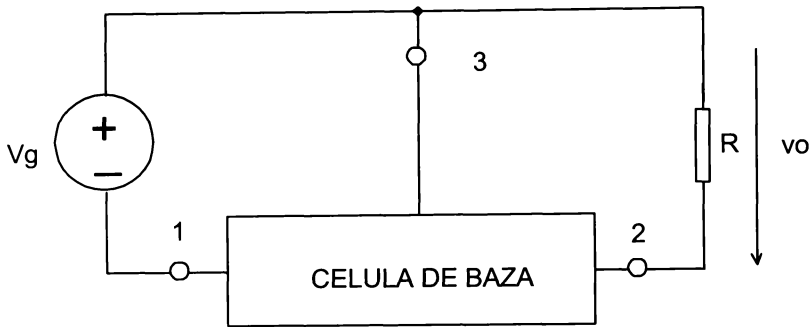


Fig. 1.39. Fig. 1.38. Celula de bază cu 3 terminale ce face obiectul teoremei 2. Sursa are borna negativă comună cu sarcina.

- dacă  $\mu$  parcurge intervalul  $[1, \infty)$  atunci și  $-\frac{\mu}{1-\mu}$  parcurge intervalul  $[1, \infty)$ ,  $-\frac{1-\mu}{\mu}$  și  $\frac{1}{\mu}$  parcurg intervalul  $[0, 1]$ , iar  $\frac{1}{1-\mu}$  și  $1-\mu$  parcurg intervalul  $(-\infty, 0]$ .
- dacă  $\mu$  parcurge intervalul  $(-\infty, 0]$  atunci și  $\frac{1}{\mu}$  parcurge intervalul  $[0, 1]$ ,  $-\frac{1-\mu}{\mu}$  și  $1-\mu$  parcurg intervalul  $[1, \infty)$ , iar  $-\frac{\mu}{1-\mu}$  și  $\frac{1}{1-\mu}$  parcurg intervalul  $(-\infty, 0]$ .

Pentru a demonstra suficiența observăm că printre cele 6 convertore obținute este și cel care are raportul de conversie  $\mu$ . Conform definiției 6 dacă acest convertor este de tip BUCK atunci  $\mu = [0, 1]$ , dacă este de tip BOOST atunci  $\mu = [1, \infty)$ , iar dacă este de tip BUCK-BOOST atunci  $\mu = (-\infty, 0]$ , ceea ce încheie demonstrația. Această teoremă va fi folosită în studiul convertoarelor pătratice.

**Teorema 4.** Dacă la o celulă de bază cu 3 terminale valoarea curentului mediu injectat într-un terminal este nulă, atunci valoarea absolută a tensiunii medii de ieșire este fie zero, fie egală cu  $V_g$ , deci tensiunea medie de ieșire nu poate fi controlată, o astfel de celulă neavând aplicabilitate practică.

**Demonstrație.** Fie celula din Fig. 1.40 și, fără a reduce generalitatea, considerăm că este nul curentul mediu injectat în terminalul 3. Apar trei posibilități:

- în terminalul 3 se conectează terminalul liber al sarcinii. Atunci evident tot curentul din terminalul 3 trece și prin sarcină și cum valoarea medie a acestuia este zero și tensiunea medie pe sarcină va fi zero.
- în terminalul 3 se conectează terminalul liber al sursei. În acest caz tot curentul din terminalul 3 trecând prin sursă și având valoare medie nulă puterea activă absorbită de la sursă va fi și ea nulă. Cum randamentul convertorului este unitar (Obs. 1) rezultă valoare nulă a tensiunii pe sarcină.

- în terminalul 3 se conectează borna comună a sursei și sarcinii. Deoarece

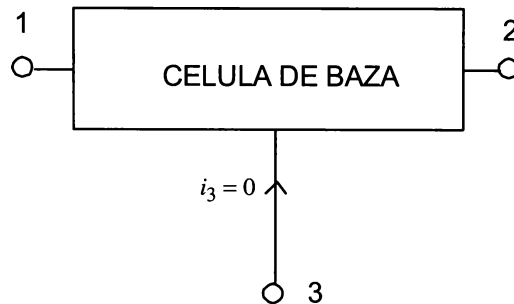


Fig. 1.40. Explicativă la teorema 4.

$$i_1 + i_2 + i_3 = 0 \quad (15)$$

și valoarea medie a curentului în borna 3 este nulă, deci

$$I_3 = 0 \quad (16)$$

rezultă că în modul valorile medii ale curenților injectați în bornele 1 și 2 sunt egale:

$$|I_1| = |I_2| \quad (17)$$

Din conservarea puterii în convertorul presupus cu randament unitar avem:

$$|V_g| |I_1| = |V_o| |I_2| \quad (18)$$

unde, fără a reduce generalitatea, am presupus că  $V_g$  este conectată între bornele 1 și 3 iar  $R$  între bornele 1 și 3. Din (17) și (18) rezultă imediat că:

$$|V_o| = |V_g| \quad (19)$$

În acest fel, prin cele trei variante posibile studiate, propoziția este demonstrată.

**Propoziția 7.** Dacă celula cu trei terminale are structura din Fig. 1.41 atunci curenții de intrare și de ieșire ai convertorului nu prezintă fronturi nule (discontinuități).

**Demonstrație.** Din examinarea Fig. 1.41 se deduce ușor că:

$$i_o = \frac{V_g - v_C}{R} \quad (20)$$

$$i_I = i_L + i_o \quad (21)$$

Din (19) rezultă imediat că  $i_o$  nu are fronturi nule, pentru că  $V_g$  și  $v_C$  nu au la rândul lor fronturi nule, fiind o mărime constantă respectiv tensiunea pe o capacitate ideală. Din (20) rezultă că nici  $i_I$  nu are fronturi nule, fiind sumă de mărimi fără fronturi nule, anume  $i_o$ , cum s-a demonstrat anterior și  $i_L$ , care este curentul printr-o inductivitate ideală. Această ultimă aserțiune finalizează demonstrația.

Propoziția 7 este deosebit de importantă din punct de vedere al perturbațiilor electromagnetice (EMI) pe care le generează convertorul. Din acest punct de vedere se cunoaște că spectrul curentului de intrare, dar și al celui de ieșire, ridică probleme de *compatibilitate electromagnetică* dacă sunt largi. Convertoarele cu o structură de genul celei din Fig. 1.41 sunt net superioare din acest punct de vedere. De remarcat faptul interesant că într-o astfel de structură cu elemente ideale

nu este necesară o capacitate de filtraj în paralel pe  $R$ . Rezistența echivalentă serie a condensatorului  $C$  și pulsațiile tensiunii  $V_g$  care alimentează celula sunt însă decisive în calitatea tensiunii de ieșire deoarece

$$v_o = V_g - v_C \quad (22)$$

Această dependență puternică a tensiunii de ieșire de tensiunea de intrare conduce la o audiosusceptibilitate mare, ceea ce este un dezavantaj. De aceea, pentru îmbunătățirea audiosusceptibilității, se practică frecvent introducerea unei capacități  $C_o$  în paralel cu sarcina aleasă astfel încât [6]:

$$C_o < \frac{1}{20} C \quad (23)$$

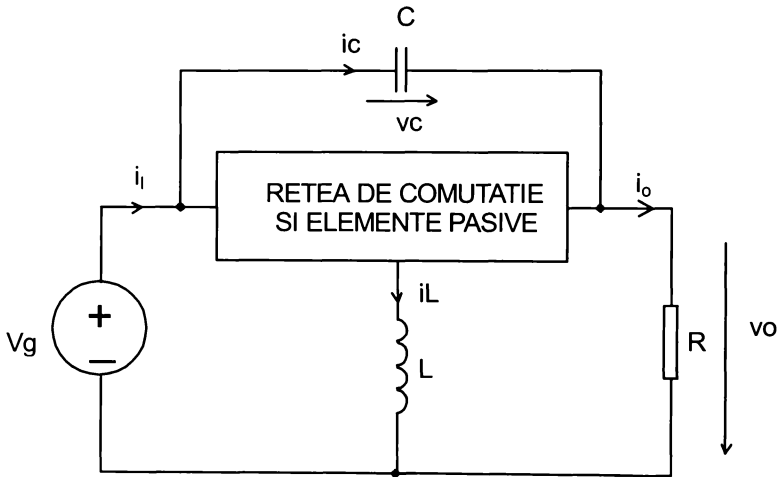


Fig. 1.41. Explicativă la propoziția 7.

Prezența acestuia nu modifică alura fără fronturi nule a curentului de intrare și de ieșire întrucât elementele de circuit sunt neideale.

Fie "g", "1" și "com" terminalele ansamblului sursă-sarcină ce trebuie conectate la terminalele "1", "2" și "3" ale celulei de bază - Fig. 1.42.

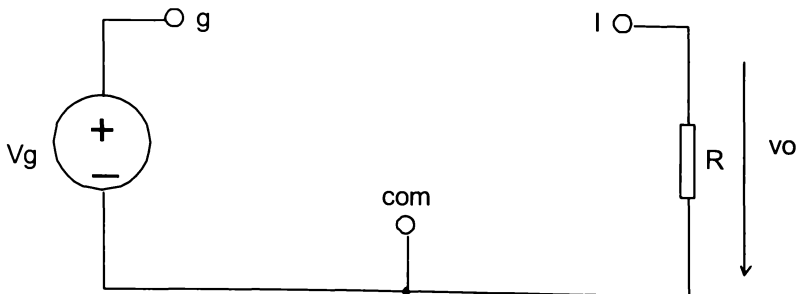


Fig. 1.42. Terminalele ansamblului sursă-sarcină care se conectează la celula de bază completă



Deoarece 3 terminale distincte trebuie conectate la 3 alte terminale distincte fără a lega două terminale ale celulei sau ale ansamblului sarcină-sursă împreună rezultă că acest lucru se poate realiza în câte bijecții se pot construi între două mulțimi cu 3 elemente fiecare, adică în  $3! = 6$  moduri distincte. Se pot obține deci maxim 6 convertoare distincte prin conectarea unei celule de bază complete la ansamblul sursă-sarcină (s-a ținut aici cont și de teorema 2).

Se investighează în continuare generarea diverselor familii de convertoare plecând de la diferite celule de bază. Elementele de comutație vor fi considerate în cazul cel mai general ca fiind pentru început comutatoare bipoziționale echivalate apoi cu perechi de *întrerupătoare în 4 cadrane*, urmând ca pentru fiecare structură concretă de convertor ele să se materializeze, alegând în funcție de pozițiile punctului de funcționare în planul tensiune-curent, elementele de circuit adecvate pentru el.

### 1.6.1. Familia convertoarelor de ordinul 2. Celula BBB.

Analizând convertorul BOOST apare firească alegerea celulei de bază ca în Fig. 1.43. Vom numi această celulă *celula BBB* pentru că toate convertoarele generate de ea au ca prefix litera B. Aceasta seamănă atât cu celula din Fig. 1.11, cât și cu cea din Fig. 1.12, fiindu-i caracteristic faptul că atât bobina  $L$  cât și condensatorul  $C$  sunt parte integrantă a celulei. Pentru a mări gradul de generalitate, dioda a fost înlocuită (așa cum s-a specificat anterior) tot cu un întrerupător în 4 cadrane. Cele două întrerupătoare se comandă în contratimp, așa cum este marcat și pe schemă. Se disting cele două porturi de curent-continuu, unul de tensiune și altul de curent.

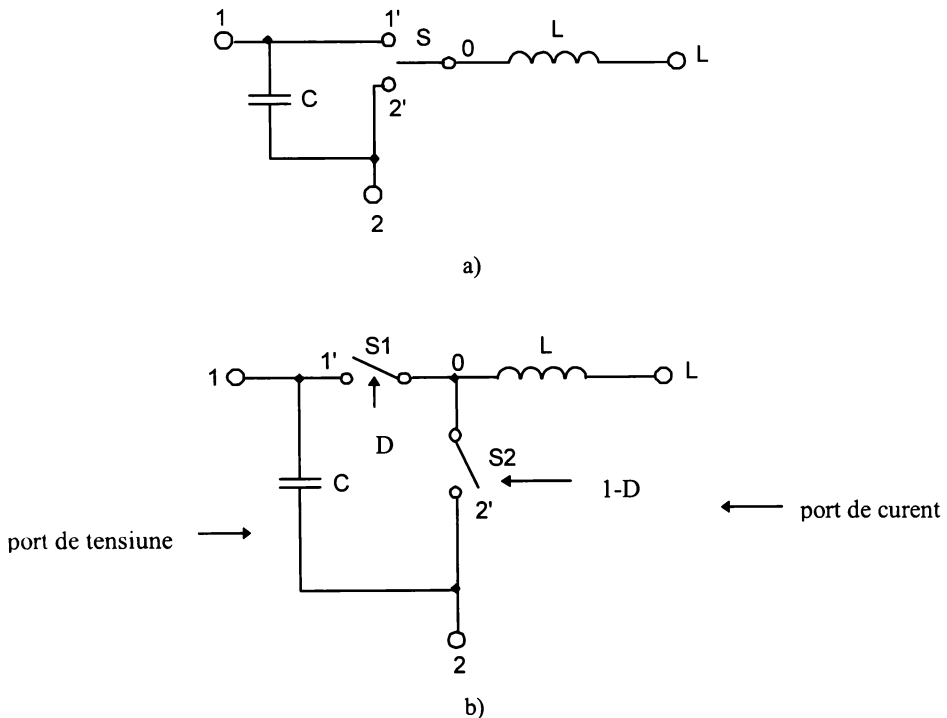


Fig. 1.43. Celula BBB. a) celula de bază cu comutator bipozițional. b) celula de bază cu 2 întrerupătoare în 4 cadrane.

Examinând celula din Fig. 1.43b se observă că aceasta prezintă simetrie privită dinspre bornele 1 și 2. În consecință tipul de convertor este dat de locul de conectare al bornei  $L$ . Pentru o poziție stabilită a bornelor  $L$ , 1 și 2, convertorul care se obține schimbând între ele bornele 1 și 2 este practic același ca și topologie, dar având comandă complementară. Există deci pentru această celulă de bază doar 3 convertoare distincte și anume BUCK, BOOST și BUCK-BOOST, prezentate în Fig. 1.44, 1.45 și respectiv 1.46. În fiecare schemă, pentru fiecare întrerupător, a fost trecut în paranteză domeniul de funcționare (numărul cadranelui) în cazul funcționării CCM. S-a specificat de asemenea și cum au fost legate terminalele “g”, “com”, “l” ale sursei și sarcinii la terminalele “1”, “2” și “L” ale celulei de bază. A doua schemă din cadrul figurilor conține aceeași topologie cu materializări practice ale întrerupătoarelor.

Modul în care se implementează practic întrerupătoarele, admitând regimul CCM, reclamă parcurgerea următoarelor etape:

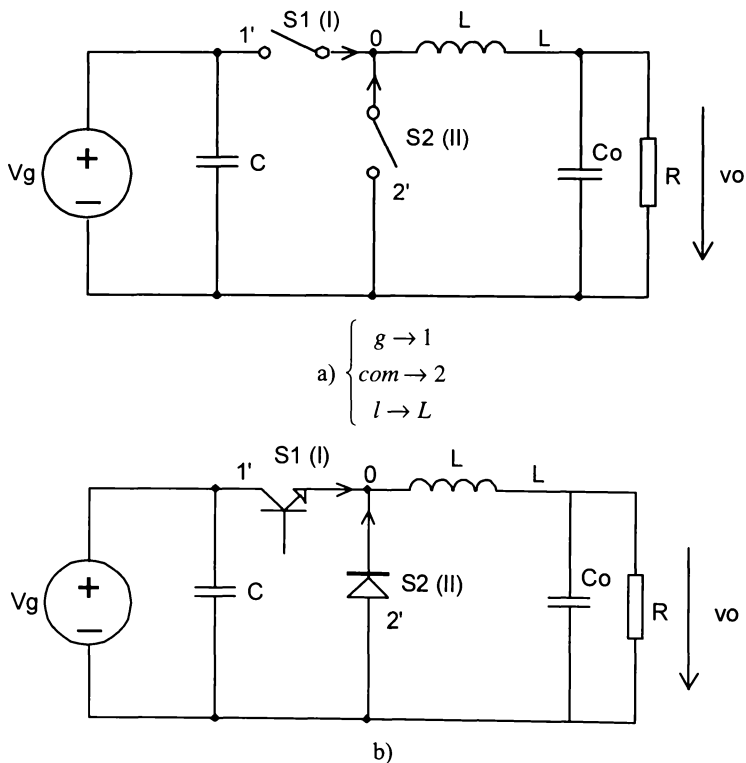


Fig. 1.44. Obținerea convertorului BUCK din celula BBB.

- se deduc valorile medii ale tensiunilor pe capacitățile care intervin în schemă ca și combinații liniare dintre tensiunea de alimentare,  $V_g$  și cea de ieșire,  $V_o$ . În acest scop se ține cont că din punct de vedere al valorilor medii bobinele se pot înlocui cu scurtcircuite.
- se determină valorile tensiunii pe bobine în cele două stări topologice caracteristice CCM tot ca și combinații liniare  $V_g$  și  $V_o$ , folosind rezultatele găsite la punctul anterior.

- se egalează valoarea medie a tensiunii pe oricare din bobinele circuitului cu zero, obținându-se expresia caracteristicii statice de transfer.

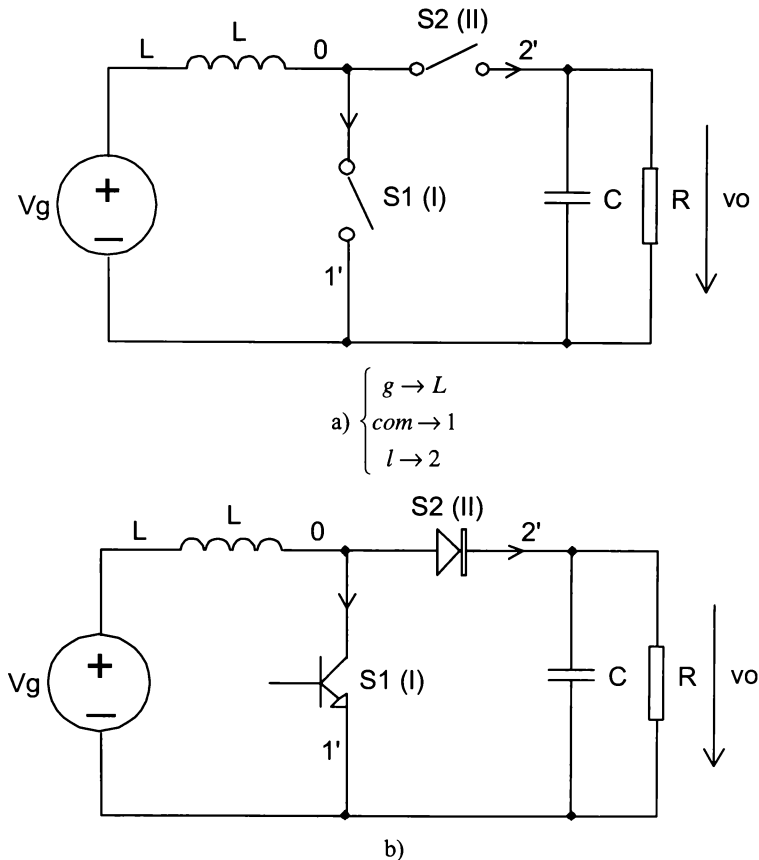


Fig. 1.45. Obținerea convertorului BOOST din celula BBB.

- folosind rezultatele punctului precedent, pentru fiecare întrerupător se determină:
  - polaritatea (semnul) tensiunii în stare blocată, exprimând în acest scop această tensiune în funcție de  $V_g$  și  $D$ . Se vor considera tensiunile de pe capacități aproximativ constante, astfel că semnul tensiunii pe întrerupătoare este dat de semnul componentei continue a combinației liniare de tensiuni pe capacități și  $V_g$  prin care se exprimă tensiunea pe întrerupător.
  - polaritatea (semnul) curentului în stare de conducție, exprimând acest curent ca o combinație liniară numai de curenți prin bobinele circuitului. Se vor determina în prealabil valorile medii ale curenților prin aceste bobine în funcție de  $V_g$ ,  $R$  și  $D$  și se va presupune că pulsațiile combinației liniare de curenți inductivi prin bobine (care dă valoarea curentului prin întrerupător) sunt neglijabile în raport cu componenta continuă. Astfel semnul curenților prin întrerupătoarele închise coincide cu semnul valorii medii a combinației liniare de curenți prin bobine.

În ambele situații se va avea alege un sens arbitrar pentru una din mărimi (curent sau tensiune), având însă grijă ca a doua mărime să aibă același sens cu cea aleasă arbitrar.

- pe baza punctului precedent se evaluează pentru fiecare întrerupător, poziția punctului de funcționare în planul  $v_S - i_S$ . În funcție de cadranele în care acesta se plasează se face materializarea cu elemente concrete, așa cum s-a descris în paragraful 1.4.1.

Se exemplifică cele de mai sus pentru convertorul BUCK-BOOST. Acest convertor conține două capacități pentru care, examinând Fig. 1.46a, găsim:

$$v_C = V_g - V_o \quad (24)$$

$$v_{C_o} = V_o \quad (25)$$

Circuitul conține o singură bobină la care:

$$v_{LON} = V_g \quad (26)$$

$$v_{LOFF} = -V_o \quad (27)$$

Egalând valoarea medie a tensiunii pe bobină cu zero se obține:

$$\frac{1}{T_S} [DT_S V_g + (1-D)T_S (-V_o)] = 0 \quad (28)$$

de unde se rezultă:

$$V_o = -\frac{D}{1-D} V_g \quad (29)$$

Când  $S_1$  este în conducție și  $S_2$  este blocat curentul care trece prin  $S_1$  este:

$$i_{S_1} = i_L \equiv I_L = I_I - I_o = -\frac{D}{1-D} I_o - I_o = -\frac{1}{1-D} I_o = \frac{D}{(1-D)^2} \frac{V_g}{R} \quad (30)$$

Când  $S_1$  este blocat și  $S_2$  conduce, tensiunea care apare pe  $S_1$  este

$$v_{S_1} = v_C \equiv V_C = V_g - V_o = V_g \frac{1}{1-D} \quad (31)$$

În mod asemănător se găsește:

$$i_{S_2} = i_L \equiv I_L = \frac{D}{(1-D)^2} \frac{V_g}{R} \quad (32)$$

$$v_{S_2} = -v_C \equiv -V_C = -V_g \frac{1}{1-D} \quad (33)$$

Examinând relațiile (30) și (31) respectiv (32) și (33), deoarece  $V_g > 0$  și  $0 \leq D \leq 1$  rezultă că  $i_{S_1} > 0$ ,  $v_{S_1} > 0$  respectiv  $i_{S_2} > 0$ ,  $v_{S_2} < 0$ . Deci  $S_1$  funcționează în cadrantul I, iar  $S_2$  în cadrantul II - Fig. 1.47. Se poate alege deci pentru  $S_1$  un tranzistor bipolar, iar pentru  $S_2$  o diodă conectate ca în Fig. 1.46b.

Dacă în ceea ce privește convertorul BOOST structura este cea clasică, convertoarele BUCK și BUCK-BOOST nu mai au conectată capacitatea în paralel cu sarcina. Aceasta pentru că portul de curent al celei de bază asigură un filtraj printr-un filtru de ordinul 1. Desigur, gradul filtrului de la ieșire se poate mări la 2 dacă se conectează ulterior  $C_o$ , cum se sugerează cu linie punctată în Fig. 1.46. De altfel aceasta este o practică frecventă atunci când portul celei care se leagă spre sarcină este portul de curent. De acum înainte notația  $C_o$ , fără alte explicații, va desemna tocmă această capacitate care nu apare prin generarea pe baza celei, dar care practic se leagă în paralel cu sarcina la portul de curent. De asemenea, teoretic,  $C$  nu mai are nici un rol dacă sursa de alimentare este ideală. În montajele practice însă este chiar indicat ca el să se utilizeze, pentru a asigura o tensiune de alimentare de calitate. De aceea el va continua să apară în schemele care urmează deși va fi conectat în paralel pe sursă.

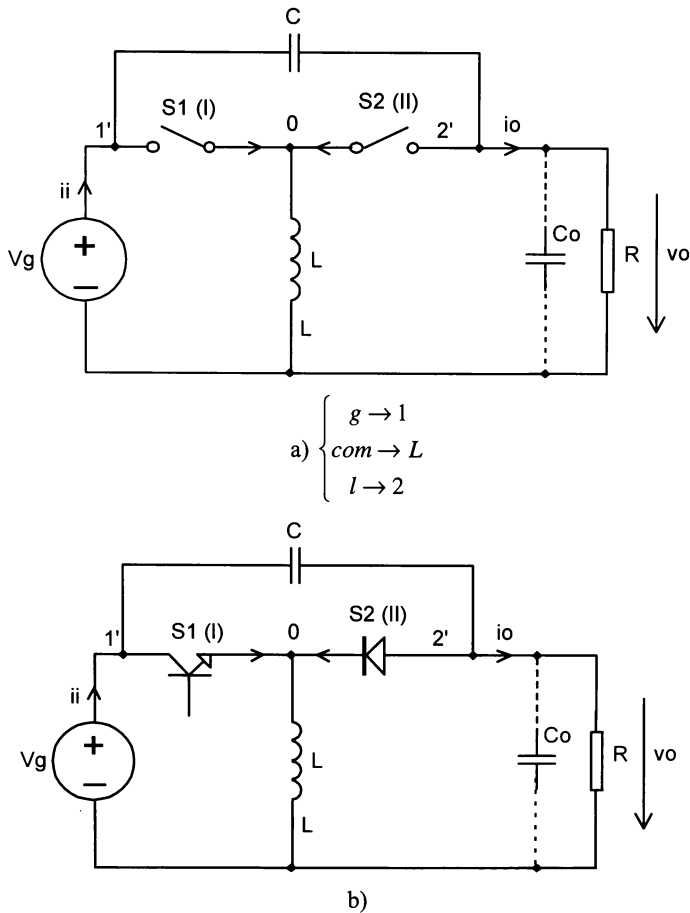


Fig. 1.46. Obținerea convertorului BUCK-BOOST din celula BBB.

O situație interesantă, care constituie de fapt un prim avantaj al abordării propuse, se regăsește la convertorul BUCK-BOOST. Conform propoziției 7 curenții de intrare și ieșire din convertor nu au fronturi nule (se pot bine aproxima cu o formă triunghiulară), deci din punct de vedere al interferenței electromagnetice o astfel de structură BUCK-BOOST are performanțe superioare celei clasice, la care atât curentul de intrare cât și cel de ieșire sunt cu fronturi nule, fapt ce-i limita aplicabilitatea.

Această topologie de convertor BUCK-BOOST a fost descoperită de o manieră deductivă de Bârcă-Gălățeanu [6], care s-a concentrat numai pe topologia BUCK-BOOST. Metoda de sinteză prezentată reprezintă însă o abordare generală ale cărei rezultate spectaculoase vor continua să se vadă în cele ce urmează.

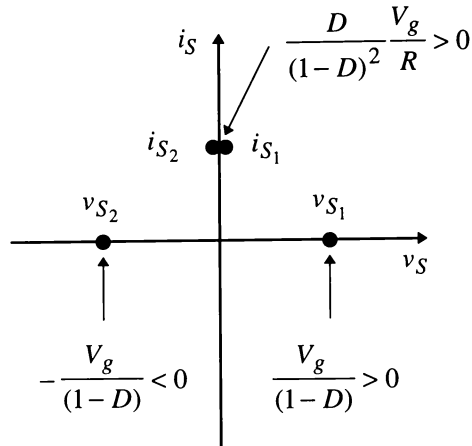


Fig. 1.47. Plasarea punctelor de funcționare ale întrerupătoarelor la convertorul BUCK-BOOST

#### 1.6.2. Familii de convertoare de ordinul 4.

O modalitate simplă și elegantă de a obține familii de convertoare de ordinul 4 se bazează pe observația că, plecând de la celula de bază de ordinul 2 din Fig. 1.43b, se poate construi o celulă de bază de ordinul 4 astfel: se taie legătura din celula de bază de ordinul 2 în punctul marcat în figura 1.48a și se introduce condensatorul  $C_1$ , de valoare suficient de mare încât din punct de vedere dinamic el să se comporte ca un scurtcircuit. Deci în convertoarele generate pe baza noii celule tensiunea pe  $C_1$  va putea fi admisă constantă. Celula nu poate rămâne însă în această formă pentru că,  $C_1$  apărând în serie cu  $S_2$ , valoarea medie a curentului injectat în borna 2 este nulă și conform teoremei 3 o astfel de celulă nu are aplicabilitate practică. Din acest motiv, pentru a deschide o cale de curent se mai adaugă celulei un element care nu poate fi decât o bobină (în caz că ar fi condensator valoarea medie a curentului injectat în borna 2 rămâne tot zero), rezultând structura din Fig. 1.48c.

Structura din Fig. 1.48c are însă 4 terminale, în timp ce sursa și sarcina, având un terminal comun, au numai 3 terminale. Prin urmare două dintre terminalele celulei din Fig. 1.48c trebuie unite. Vom nota întotdeauna cu  $X$  terminalul care rezultă din conectarea a două terminale ale celulei. Având 4 terminale și 2 din ele trebuind să fie unite rezultă că avem  $C_4^2=6$  variante posibile de celule de bază. Totuși nu toate acestea prezintă interes.

Astfel celula de bază din Fig. 1.49. reprezintă în fond tot celula de bază din Fig. 1.43b, dar cu bobina  $L$  secționată în 2 bobine,  $L_1$  și  $L_2$ , având un punct comun  $X$  și conectate din punct de vedere dinamic în paralel prin  $C_1$ . O astfel de celulă de bază, deși generează convertoare funcționale, va conduce tot la structurile BUCK, BOOST și BUCK-BOOST modificate în sensul menționat mai sus, deci cu aceleași caracteristici statice, curenți de intrare și de ieșire, etc.

Pentru celulele de bază din Fig. 1.50 și Fig. 1.51 se poate demonstra ușor că valoarea medie a curentului prin borna  $X$ ,  $I_X$ , este zero deoarece  $i_X$  este o sumă de curenți capacitivi. Conform teoremei 3 aceste celule nu vor conduce la structuri cu tensiune de ieșire controlabilă, deci nu intră în discuție. Rămân încă trei variante de celule de bază care se vor analiza mai jos. Ca și până acum, fără a se mai nota pe desene, se va presupune că  $S_1$  este comandat cu factorul de umplere  $D$  iar  $S_2$  cu factorul de umplere  $1-D$ , complementar față de  $S_1$ .

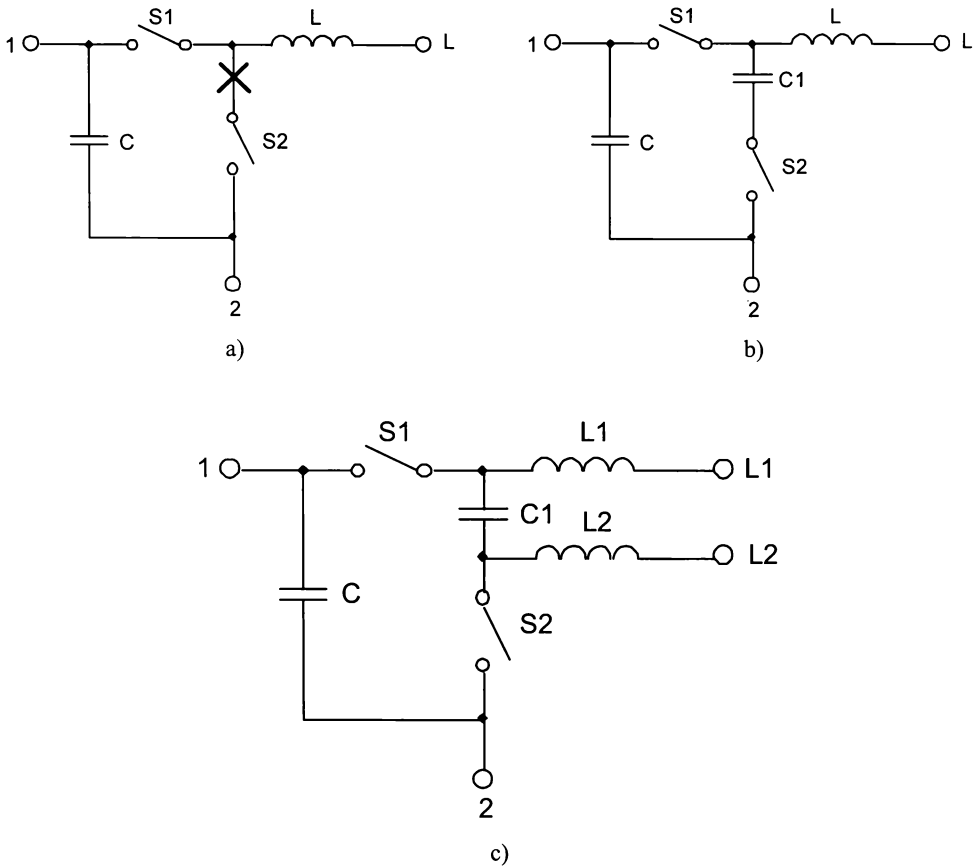


Fig. 1.48. Obținerea unei celule de bază complete de ordinul 4 cu 4 terminale din celula de bază completă BBB.

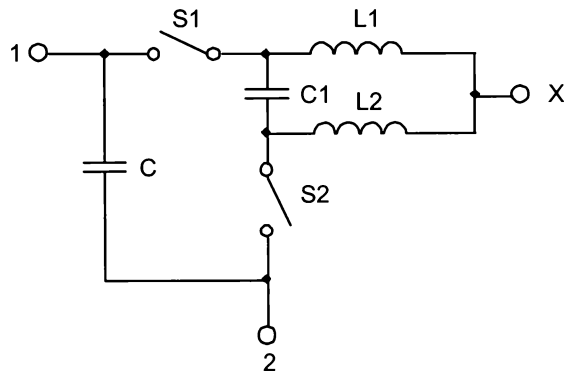


Fig. 1.49. Celula obținută din celula de bază din Fig. 1.48 unind bornele  $L_1$  și  $L_2$ .

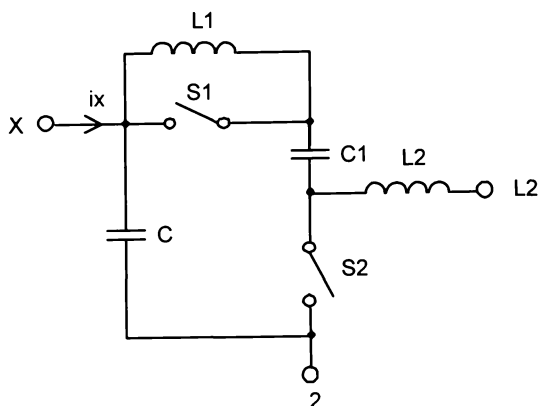


Fig. 1.50. Celula obținută din celula de bază din Fig. 1.48 unind bornele 1 și  $L_1$ .

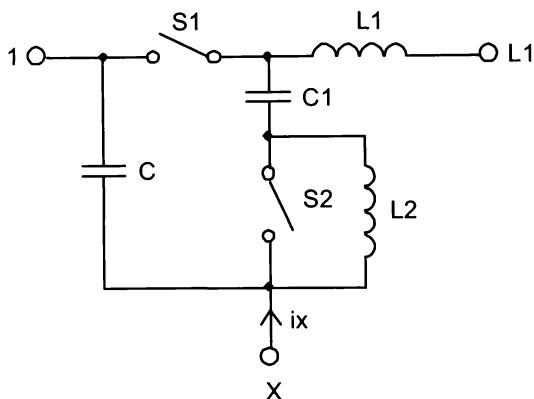


Fig. 1.51. Celula obținută din celula de bază din Fig. 1.48 unind bornele 2 și  $L_2$ .

#### A) Celula ĆUK (convertoare din familia ĆUK)

Celula de bază provine din structura din Fig. 1.48c unind bornele "1" și "2" într-o bornă notată cu  $X$  - Fig. 1.52. Desigur acest lucru face să dispară capacitatea  $C$ , care astfel este scurtcircuitată. Deci această familie de convertoare va fi de ordinul 3. O a doua observație este aceea că celula prezintă simetrie față de o axă verticală, deci practic din cele 6 convertoare posibile distincte sunt numai 3, celelalte fiind aceleași dar cu comandă complementară. Pentru fiecare convertor se dau schema cu întrerupătoare împreună cu materializarea lor practică. În paranteze au fost trecute cadranele de funcționare pentru sensurile alese. Ambele porturi ale celulei fiind de curent se poate ulterior îmbunătăți filtrajul pe sarcină conectând în paralel capacitatea notată pe scheme cu  $C_o$ , convertoarele redevenind de ordinul 4 și având pe sarcină o tensiune cu pulsații tipice, aproximativ parabolice.



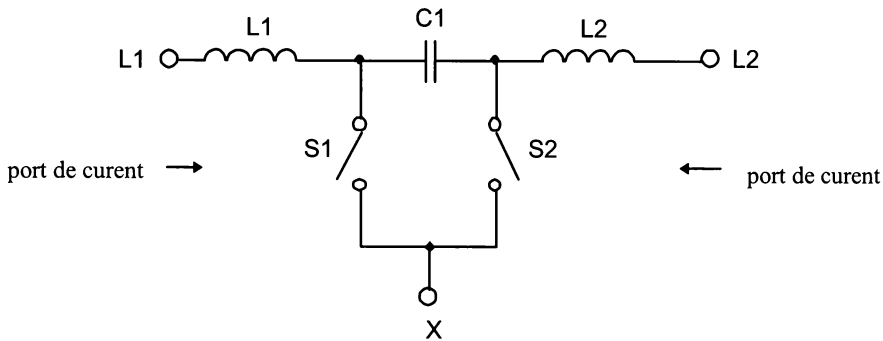


Fig. 1.52. Celula ĆUK.

Dată fiind natura de curent a porturilor, toți curenții din terminalele celulei de bază vor fi fără fronturi nule, de formă aproximativ triunghiulară, deoarece sunt combinații liniare de curenți inductivi. Aceasta înseamnă că și curenții de intrare și de ieșire ai convertorului au aceeași caracteristică, deci au proprietăți bune privind interferența electromagnetică.

Se obțin astfel următoarele structuri:

A1. Convertorul ĆUK - Fig. 1.53.

A2. O nouă variantă de convertor BUCK - Fig. 1.54.

A3. O nouă variantă de convertor BOOST - Fig. 1.55.

Cele două noi variante de convertoare BUCK și BOOST [Lascu, 75, 76] reprezintă două structuri propuse de autor. Cu numai două componente suplimentare (o bobină și un condensator) se obțin aceleași caracteristici statice și aceleași solicitări ale componentelor ca și în corespondentele clasice. Superioritatea acestor noi topologii constă în curenții de intrare și de ieșire cu spectrul mult restrâns față de structurile clasice, deci *proprietăți net superioare privind compatibilitatea electromagnetică*. În privința noii variante de convertor BUCK se observă că bucla  $V_g - L_1 - C_1 - L_2$  nu are amortizare, ceea ce determină un regim tranzitoriu oscilant, care teoretic nu se atenuează. În practică însă există totdeauna rezistențe parazite care realizează amortizarea.

Se cuvine subliniat un aspect important. Și un convertor BUCK clasic la care se adaugă la intrare un filtru  $L-C$  cu bobina suficient de mare, sau un convertor BOOST cu filtru similar la ieșire realizează curenți de intrare și ieșire fără fronturi nule, în condițiile menținerii numărului de componente. Totuși în acest caz elementele reactive rezultă de valori mult mai mari. Aceasta se observă și din studiul trecerii din CCM în DICM. În cazul noului convertor dioda este străbătută de curenții ambelor bobine, deci în condiția de limită va apare  $L_1 \parallel L_2$ . În schimb în varianta clasică cu filtru  $L-C$  prin diodă trece același curent din convertorul clasic și anume cel al bobinei convertorului.

Un alt fapt interesant este acela că în noile convertoare tensiunile pe cele două bobine sunt egale. Aceasta permite cuplarea acestora și obținerea de curenți cu pulsații nule la intrare - pentru convertorul BUCK și la ieșire - pentru convertorul BOOST, după tehnica prezentată în capitolul 2. Cele două convertoare au fost simulate și apoi realizate experimental.

Pentru noua variantă de convertor BUCK valorile parametrilor de circuit au fost:

$$V_g = 30 \text{ V}; \quad L_1 = 828 \text{ } \mu\text{H}; \quad R_{L_1} = 0,51 \text{ } \Omega; \quad L_2 = 521 \text{ } \mu\text{H}; \quad R_{L_2} = 0,19 \text{ } \Omega; \quad C_1 = 102 \text{ } \mu\text{F};$$

$$C_o = 12,8 \text{ } \mu\text{F}; \quad R = 10,07 \text{ } \Omega; \quad S_1 = BD 137; \quad S_2 = SB 540; \quad f_S = 40 \text{ kHz}; \quad D = 0,4.$$

Rezultatele simulărilor (pentru fișierul de simulare vezi anexa A1) sunt prezentate în Fig. 1.56, iar cele experimentale în Fig. 1.57, Fig. 1.58 și Fig. 1.59.

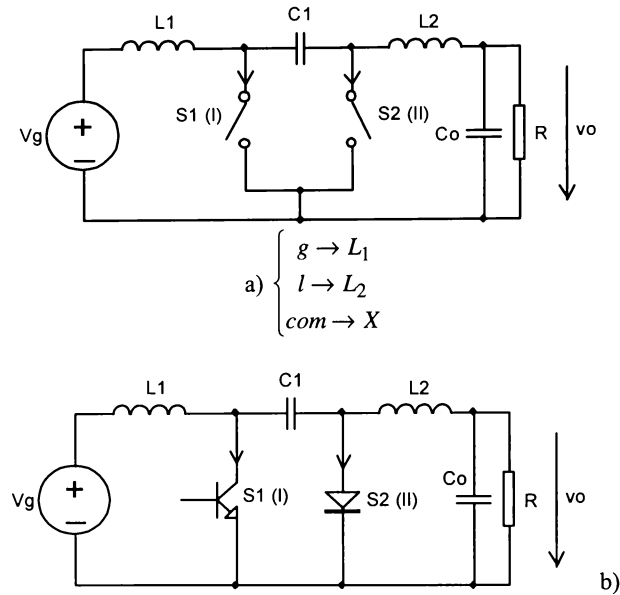


Fig. 1.53. Convertorul ĆUK obținut din celula de bază completă din Fig. 1.52.

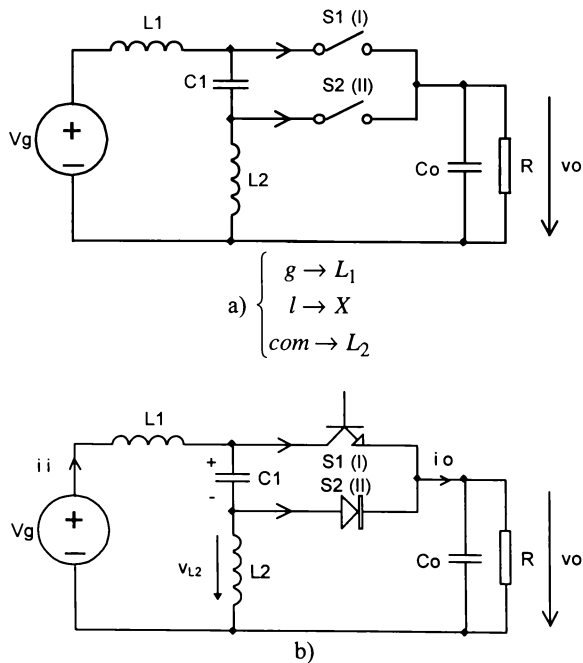


Fig. 1.54. O nouă variantă de convertor BUCK obținut din celula de bază completă din Fig. 1.52.

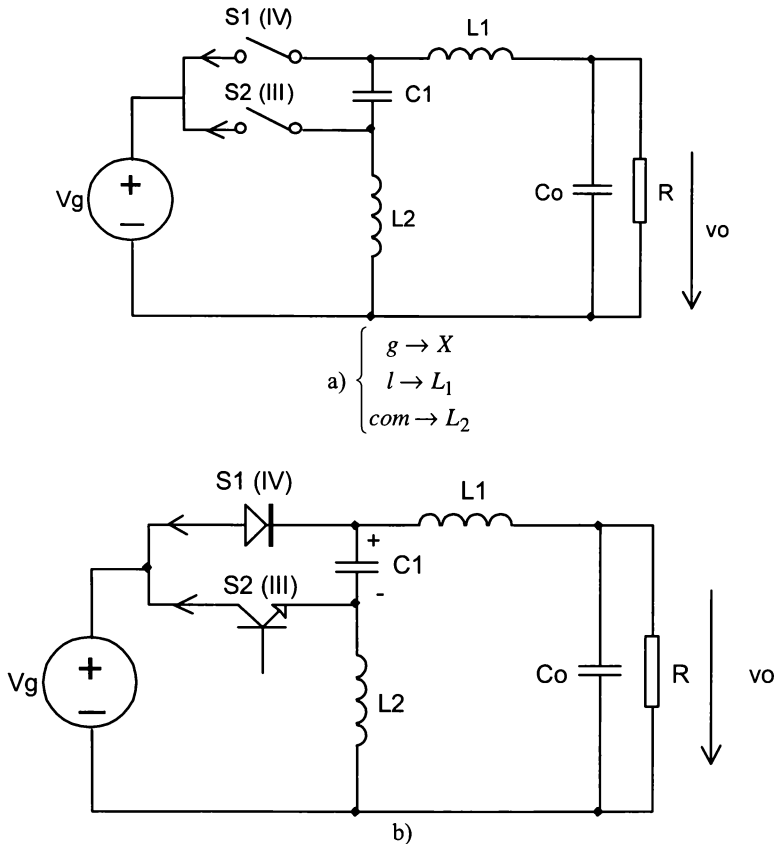


Fig. 1.55. O nouă variantă de convertor BOOST obținut din celula de bază completă din Fig. 1.52.

Pentru noua variantă de convertor BOOST valorile parametrilor de circuit au fost:

$$V_g = 10 \text{ V}; \quad L_1 = 625 \mu\text{H}; \quad R_{L_1} = 0,35 \Omega; \quad L_2 = 600 \mu\text{H}; \quad R_{L_2} = 0,68 \Omega; \quad C_1 = 199,7 \mu\text{F};$$

$$C_o = 21,04 \mu\text{F}; \quad R = 68,02 \Omega; \quad S_1 = BD 139; \quad S_2 = SB 540; \quad f_S = 40 \text{ kHz}; \quad D = 0,6.$$

Rezultatele simulărilor (pentru fișier vezi anexa A2) sunt prezentate în Fig. 1.60, iar cele experimentale în Fig. 1.61, Fig. 1.62 și Fig. 1.63.

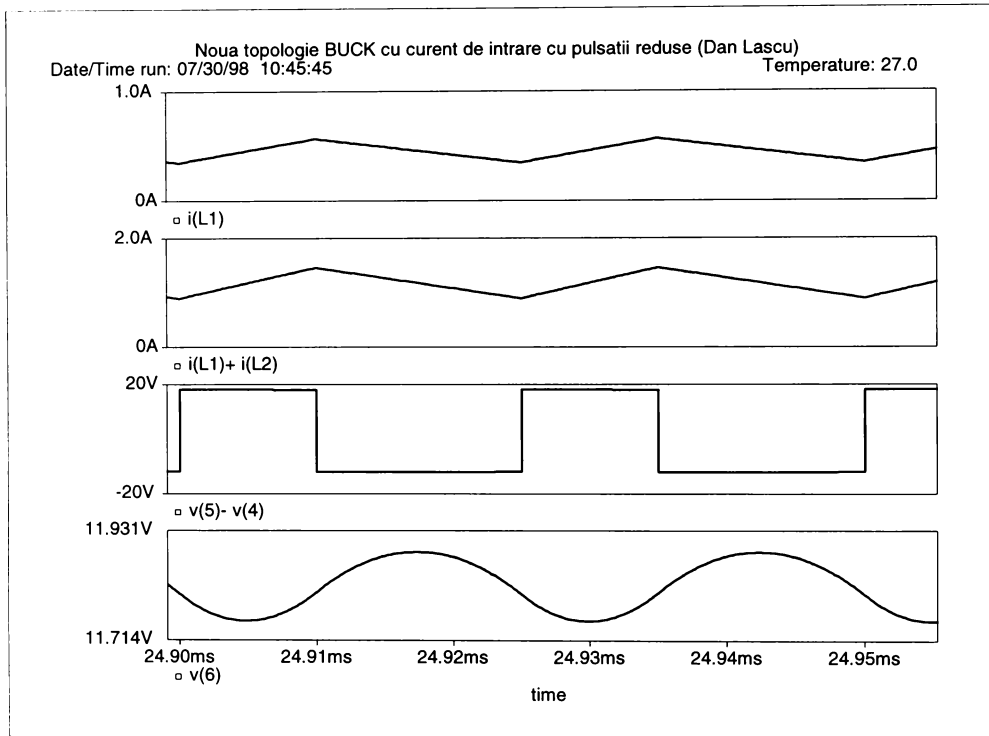


Fig. 1.56. Noua variantă de convertor BUCK - rezultatele simulărilor.  
 Simulator PETS (Boulder University Colorado) cu postprocesor grafic PROBE (Microsim Corp.)  
 De sus în jos:  $i_i$ ,  $i_o$ ,  $v_{L2}$ ,  $v_o$ .

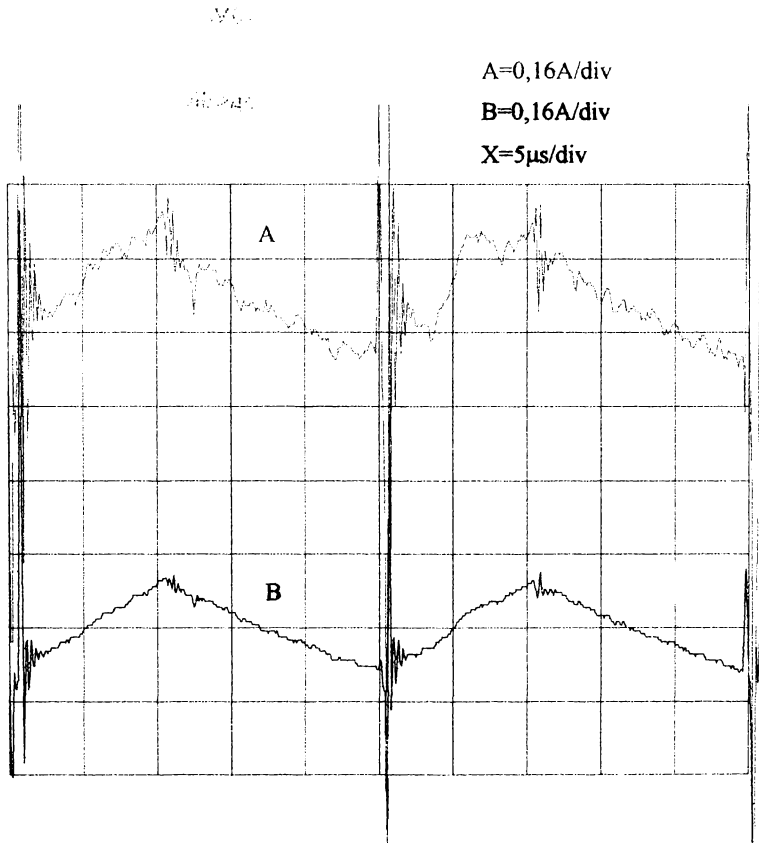


Fig. 1.57, Noua variantă de convertor BUCK - rezultate experimentale.

$i_i = i_{L_1}$  (sus) - nivelul de zero la jumătatea ecranului.

$i_o = i_{L_1} + i_{L_2}$  (jos) - nivelul de zero în partea inferioară a ecranului.

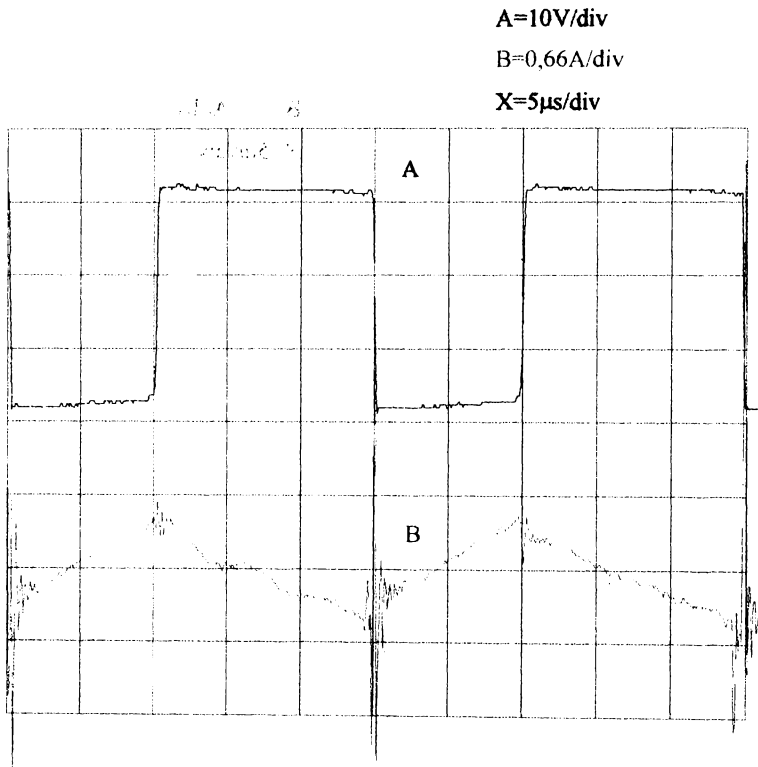


Fig. 1.58. Noua variantă de convertor BUCK - rezultate experimentale.

$v_{L_2}$  (sus) - nivelul de zero la a doua diviziune de sus în jos.

$i_{o.} = i_{L_1} + i_{L_2}$  (jos) - nivelul de zero în partea inferioară a ecranului.

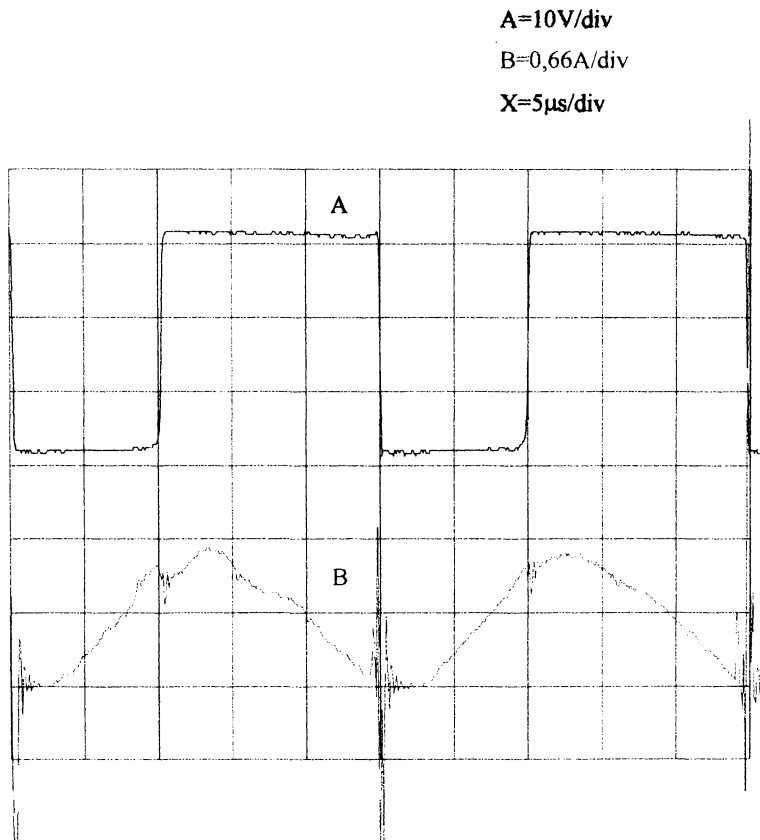


Fig. 1.59. Noua variantă de convertor BUCK - rezultate experimentale.

$v_{L_2}$  (sus) - nivelul de zero la a doua diviziune de sus în jos.

$v_o$  (jos) - fără componentă continuă, nivelul de zero la a doua diviziune de jos în sus.

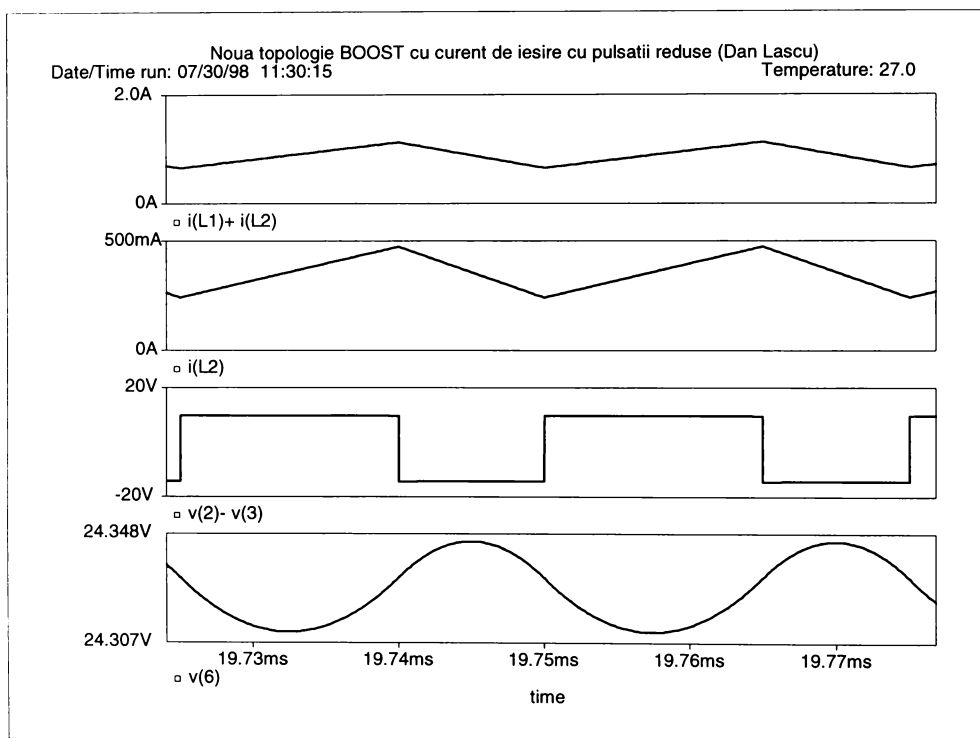


Fig. 1.60. Noua variantă de convertor BOOST - rezultatele simulărilor.  
 Simulator PETS (Boulder University Colorado) cu postprocesor grafic PROBE (Microsim Corp.)  
 De sus în jos:  $i_i$ ,  $i_o$ ,  $v_{L1}$ ,  $v_o$ .



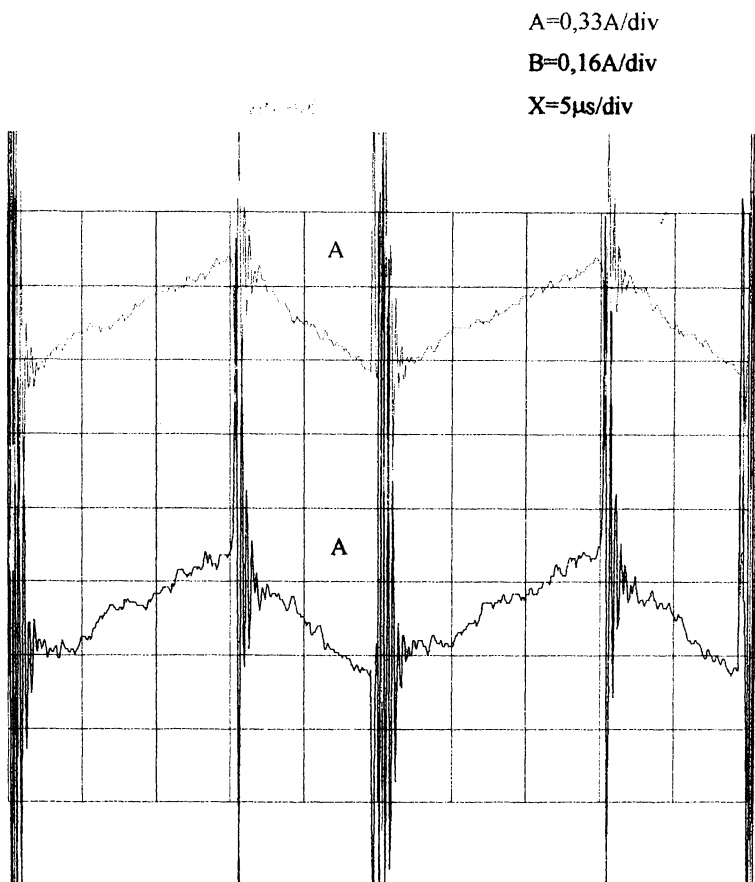


Fig. 1. 61. Noua variantă de convertor BOOST - rezultate experimentale.

$i_i = i_{L_1} + i_{L_2}$  (sus) - nivelul de zero la a jumătatea ecranului.

$i_o = i_{L_2}$  (jos) - nivelul de zero în partea inferioară a ecranului.

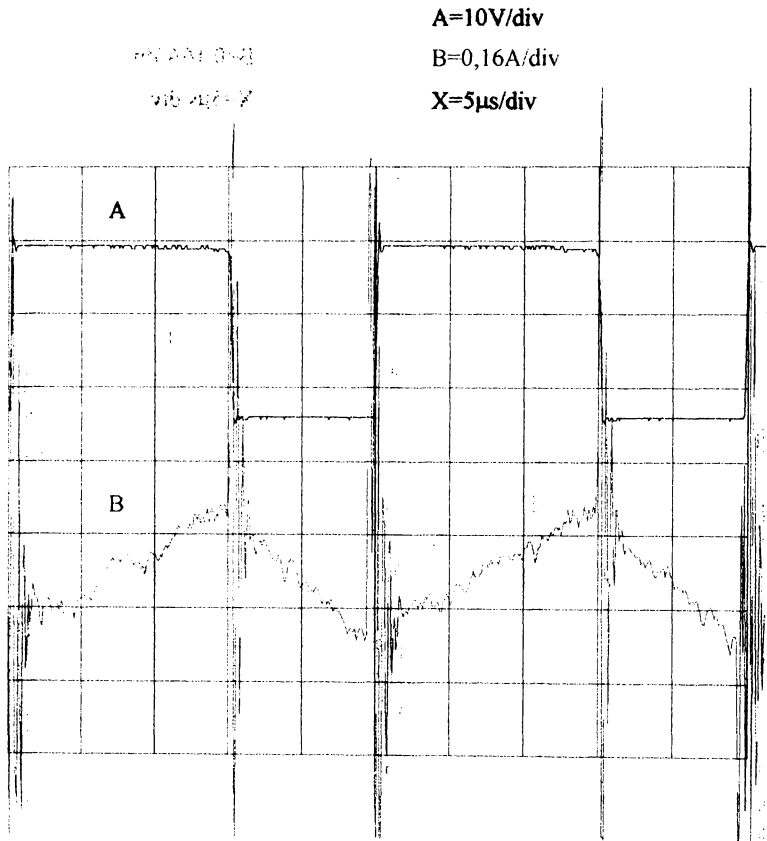


Fig. 1.62. Noua variantă de convertor BOOST - rezultate experimentale.

$V_{L_1}$  (sus) - nivelul de zero la a doua diviziune de sus în jos.

$V_{L_2}$  (jos) - nivelul de zero în partea inferioară a ecranului.

Urmare: Aproximativ 1000 de cuvinte pe pagina.

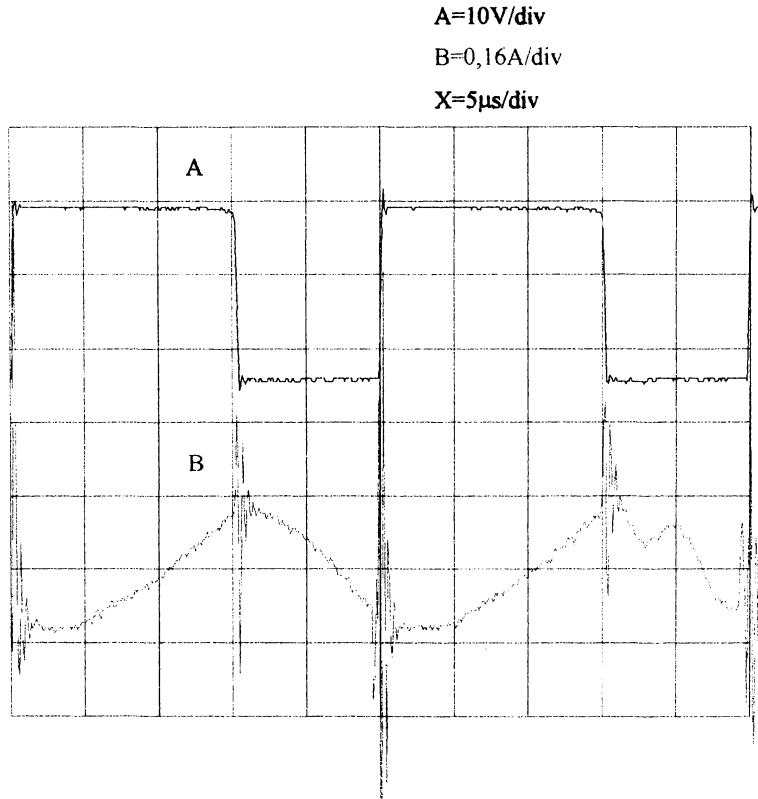


Fig. 1.63. Noua variantă de convertor BOOST - rezultate experimentale.

$v_{L_1}$  (sus) - nivelul de zero la a doua diviziune de sus în jos.

$v_o$  (jos) - fără componentă continuă, nivelul de zero la a doua diviziune de jos în sus.

## B) Celula ZSWJ (ZETA-SEPIC-WATKINS-JOHNSON)

Celula de bază, al cărui nume este propus de autor după tipurile de convertoare pe care le generează, provine din structura din Fig. 1.48c unind bornele "2" și "L<sub>1</sub>" într-o bornă notată cu X, ca în Fig. 1.64a, sau din aceeași celulă unind bornele "1" și "L<sub>2</sub>" - Fig. 1.64b. Se remarcă faptul că a doua celulă de bază provine din prima făcând:

$$S_1 \rightarrow S_2; S_2 \rightarrow S_1; L_1 \rightarrow L_2; L_2 \rightarrow L_1$$

De aceea celula din Fig. 1.64b va genera aceleași convertoare ca cea din Fig. 1.64a, dar cu comandă complementară. Se mai face observația că această familie de convertoare va fi de ordinul 4.

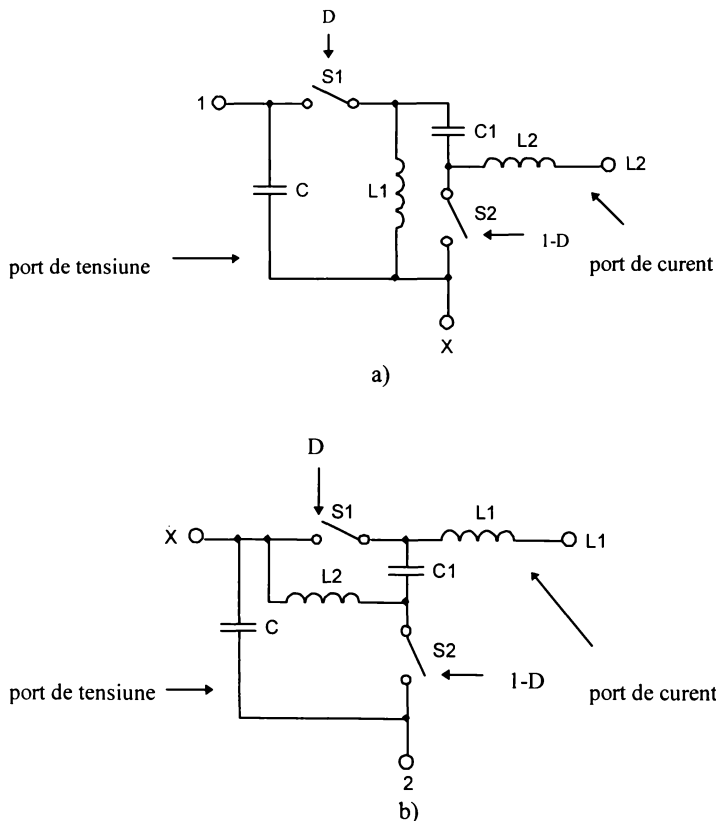


Fig. 1.64. Două celule ZSWJ.

Dacă facem generarea cu celula din Fig. 1.64a se observă că, față de celulele de bază analizate până acum nu mai avem simetrie, deci se vor obține 6 convertoare distincte și anume:

B1. ZETA

B2. SEPIC

B3. WATKINS-JOHNSON cu curenți de intrare și ieșire netezi

B4. O nouă VARIANTĂ de convertor WATKINS-JOHNSON

B5. WATKINS-JOHNSON INVERS cu curenți de intrare și de ieșire netezi

B6. O nouă VARIANTĂ de convertor WATKINS-JOHNSON INVERS

Procesul de obținere a acestora și schemele corespunzătoare sunt prezentate în Fig. 1.65, Fig. 1.66, Fig. 1.67, Fig. 1.68, Fig. 1.69 și respectiv Fig. 1.70.

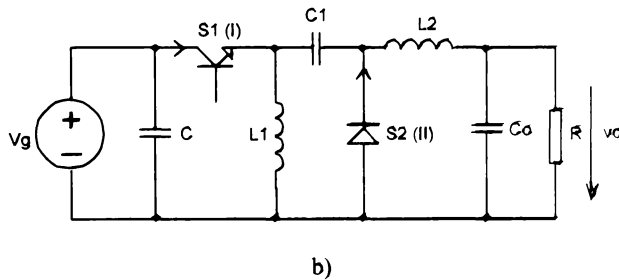
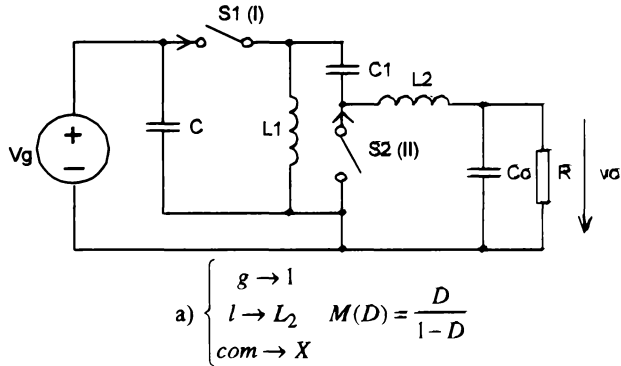


Fig.1.65. Obținerea convertorului ZETA.

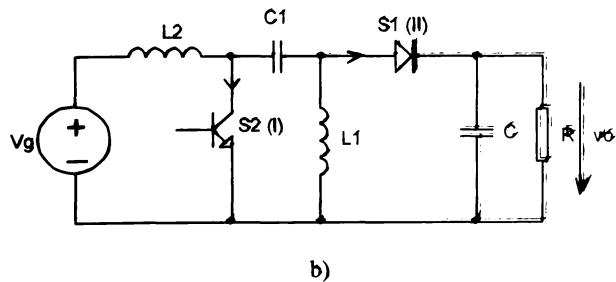
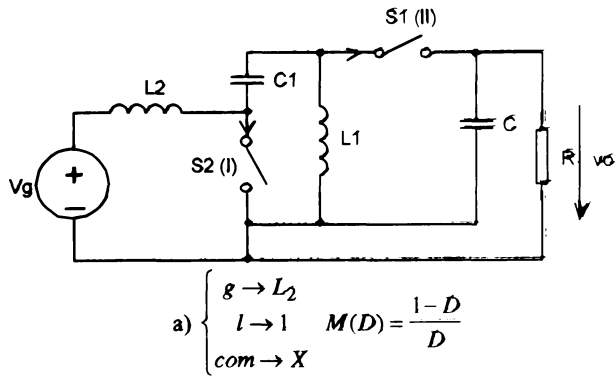


Fig. 1.66. Obținerea convertorului SEPIC.

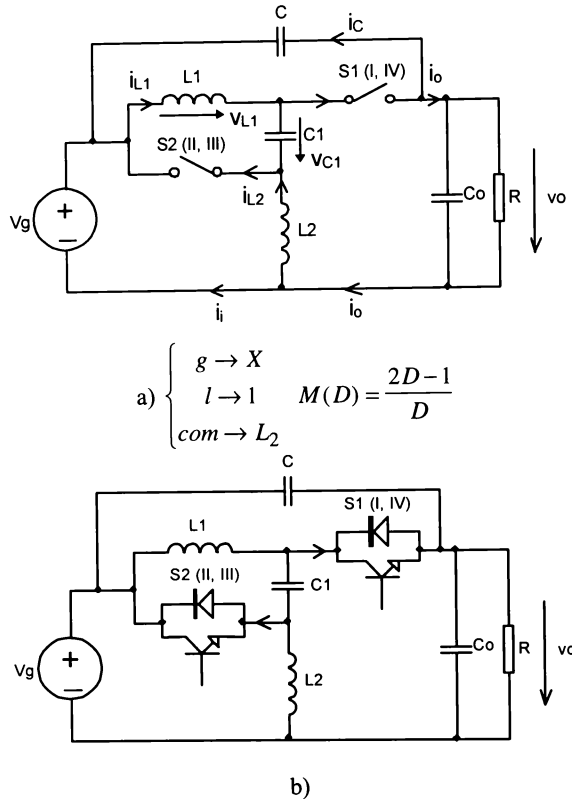


Fig. 1.67. Obținerea convertorului Watkins-Johnson cu curenți de intrare și de ieșire fără discontinuități.

Pe varianta de convertor cu întrerupătoarele încă nematerializate a fost trecută expresia raportului static de conversie. Pentru convertorul SEPIC se știe însă că expresia ei este aceeași cu a convertorului ZETA. Rezultatul diferit pe care îl produce procedura de sinteză se datorează faptului că, în implementarea practică, numai în convertorul SEPIC,  $S_1$  are materializarea într-o diodă. Pe de altă parte, în structurile clasice expresia caracteristicii statice de conversie este dată funcție de factorul de umplere al tranzistorului din convertor. Este însă clar că dacă în expresia lui  $M$  din Fig. 1.66a se face  $D \rightarrow 1-D$  se obține rezultatul clasic. Din aceleași considerente s-a preferat denumirea de "variantă" pentru convertoarele din Fig. 1.68 și 1.70, pentru că dacă în expresiile caracteristicilor statice de conversie corespunzătoare dacă se face  $D \rightarrow 1-D$  se obțin expresiile caracteristicilor convertoarelor Watkins-Johnson și Watkins-Johnson invers. Cu toate acestea topologiile acestor convertoare sunt complet diferite de convertoarele clasice cu același nume propriu. În fine, denumirea de Watkins-Johnson invers vine de la faptul că expresia caracteristicii sale statice de conversie este inversa (în sens matematic) celei a convertorului Watkins-Johnson.

Ultimele 4 convertoare au caracteristici statice de conversie bipolare ( tensiuni de ieșire atât pozitive, cât și negative, în funcție de valorile concrete ale lui  $D$ ). Alegerea întrerupătoarelor pentru ele conduce la întrerupătoare cu funcționare în două cadrane. De exemplu pentru convertorul Watkins-Johnson din Fig. 1.67, tensiunea la care se încarcă  $C_1$  este dată de bucla  $V_g - L_1 - C_1 - L_2$  și are valoarea:

$$v_{C_1} = V_g \quad (34)$$

Tensiunile pe bobina  $L_1$  aferente stărilor închis și deschis pentru  $S_1$  sunt:

$$v_{L_1 ON} = V_g - V_o \quad (35)$$

$$v_{L_1 OFF} = -v_{C_1} = -V_g \quad (36)$$

Egalând cu zero valoarea medie a tensiunii pe  $L_1$  se obține raportul static de conversie:

$$\frac{V_o}{V_g} = \frac{2D-1}{D} \quad (37)$$

Valorile medii ale curenților prin cele două bobine sunt:

$$I_{L_1} = I_{S_1} = I_o = \frac{2D-1}{D} \frac{V_g}{R} \quad (38)$$

$$I_{L_2} = I_o - I_l = \left(1 - \frac{2D-1}{D}\right) I_o = \frac{(1-D)(2D-1)}{D^2} \frac{V_g}{R} \quad (39)$$

Tensiunea pe  $S_1$ , în conformitate cu sensul înscris în desen, este (atunci când  $S_2$  este închis):

$$v_{S_1} = v_{C_1} + V_g - v_o \equiv V_{C_1} + V_g - V_o = 2V_g - V_o = \frac{1}{D} V_g > 0 \quad (40)$$

Curentul prin  $S_1$  când acesta este închis este:

$$i_{S_1} = i_{L_1} + i_{L_2} \equiv I_{L_1} + I_{L_2} = \frac{2D-1}{D^2} \frac{V_g}{R} \quad (41)$$

Este clar că pentru  $D > 0,5$   $I_{S_1}$  este pozitiv iar pentru  $D < 0,5$   $I_{S_1}$  este negativ. Deci  $S_1$  trebuie să suporte o tensiune pozitivă (unidirecțional în tensiune) dar să fie bidirecțional în curent. Funcționarea sa este deci în cadranele I și IV. Îl materializăm aici cu un tranzistor bipolar în antiparalel cu o diodă, dar se poate utiliza și în tranzistor MOS. Similar, pentru  $S_2$  găsim:

$$V_{S_2} = -v_{C_1} + V_o - V_g = V_o - 2V_g = -\frac{1}{D} V_g < 0 \quad (42)$$

$$i_{S_2} = i_{L_1} + i_{L_2} \equiv I_{L_1} + I_{L_2} = \frac{2D-1}{D^2} \frac{V_g}{R} \quad (43)$$

Deci  $S_2$  are punctul de funcționare în cadranul II (pentru  $D > 0,5$ ) și III (pentru  $D < 0,5$ ) deci are o materializare analogă cu  $S_1$ .

Ceea ce procedura de sinteză aduce nou este că pentru convertoarele *Watkins-Johnson și convertorul Watkins-Johnson invers* se obțin, spre deosebire de structurile clasice, convertoare *cu curenți de intrare fără fronturi nule* (formă aproximativ triunghiulară), deci cu proprietăți de interferență și compatibilitate electromagnetică superioare.

În al doilea rând, procedura de sinteză conduce la *două structuri noi de convertoare* care au fost numite variantă de convertor Watkins-Johnson - Fig. 1.68 - și variantă de convertor Watkins-Johnson invers - Fig. 1.70. Aceasta pentru că aceste convertoare au topologii diferite de corespondentele clasice, dar păstrează caracteristica statică de conversie a acestora dacă se comandă întrerupătoarele invers ( $S_1$  cu factor de umplere  $D$  și  $S_2$  cu  $1-D$ ). În plus, aceste structuri noi de convertoare au avantajul că se pot utiliza atunci când se doresc curenți de intrare sau de ieșire fără pulsații, ceea ce se obține prin tehnica de cuplare a bobinelor (vezi capitolul 2).

Anticipând puțin, dacă ne referim la convertorul Watkins-Johnson din Fig. 1.67 se observă că prin cuplarea bobinelor nu se poate obține curent cu pulsații nule pentru curentul de ieșire sau de intrare pentru că în serie cu intrarea sau ieșirea nu apare nici o bobină. Cu varianta de convertor Watkins-Johnson din Fig. 1.68 acest lucru este însă posibil pentru că  $L_2$  apare în serie la ieșire. Similar, într-un convertor Watkins-Johnson invers clasic - Fig. 1.69 - nu se poate obține curent de

ieșire sau intrare fără pulsații. În noua variantă de convertor Watkins-Johnson invers - Fig. 1.70 - se poate însă obține curent de intrare cvasicontinuu datorită prezenței bobinei  $L_2$  în serie la intrare.

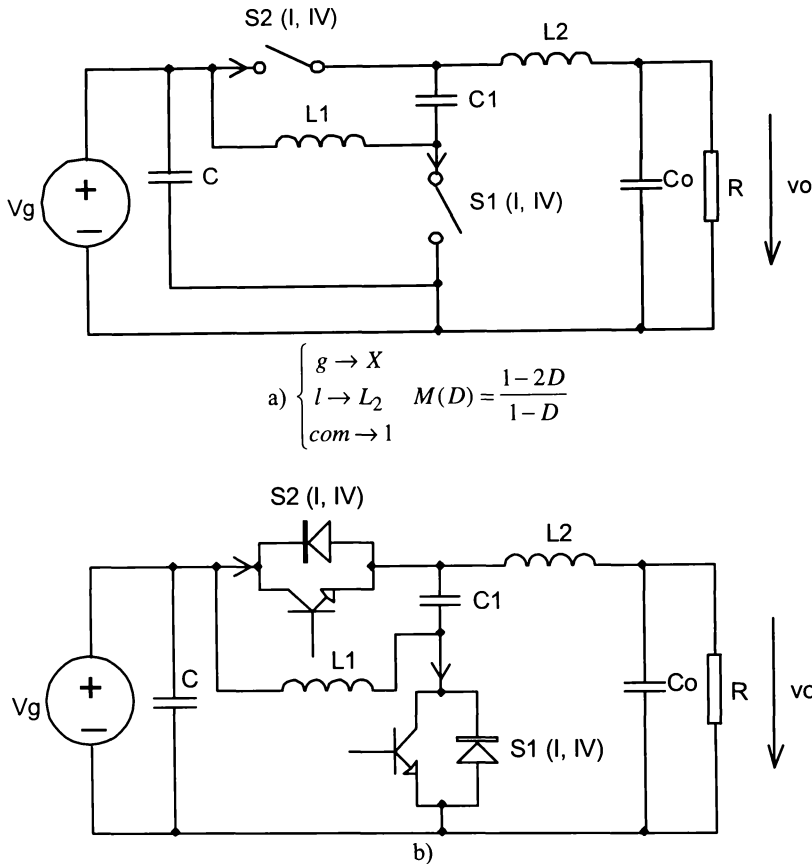


Fig. 1.61. Obținerea noii variante de convertor Watkins-Johnson.

În acest punct trebuie făcută o remarcă importantă. O dată cu existența convertoarelor cu tensiune de ieșire bipolară conceptul de inversie bilaterală intră într-un impas deoarece regulile inversiei bilaterale clasice presupun implicit convertoare cu întrerupătoare într-un singur cadran, pe când de exemplu convertorul Watkins-Johnson are întrerupătoare în două cadrane. Se poate însă găsi și în acest caz o soluție prin metoda inversei bilaterale generalizate propusă de autor. Ideea de la care se pornește este aceeași cu cea de la care pleacă și inversia bilaterală clasică, adică găsirea convertorului corespunzător prin schimbarea între ele a sursei și sarcinii. De aici însă nu ne vom axa pe înlocuirea componentelor concrete din convertorul original ci vom abstractiza topologia manipulând numai întrerupătoare în sensul general, ținând seama că inversia bilaterală vizează o nouă topologie ca rețea de comutație. Aceste întrerupătoare vor fi materializate practic abia ulterior.

Teorema inversei bilaterale generalizate

Fie un convertor dc-dc original. Pentru găsirea convertorului corespondent prin inversie bilaterală generalizată se parcurg următorii pași:



**Pasul 1.** Se identifică elementele sau grupele de elemente prin care se materializează întrerupătoare și se înlocuiesc cu întrerupătoare generale.

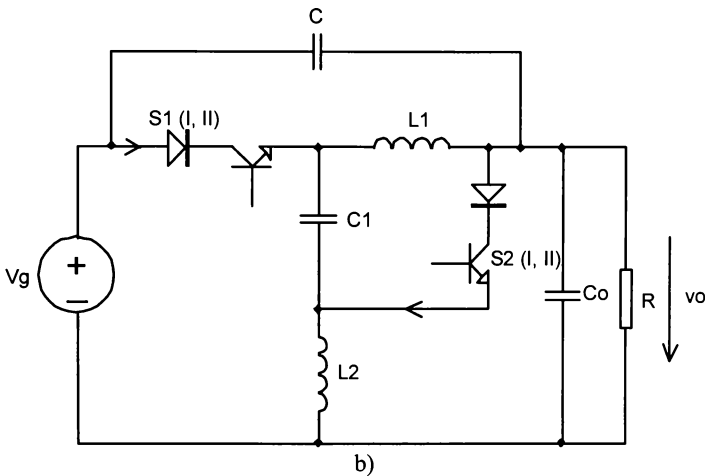
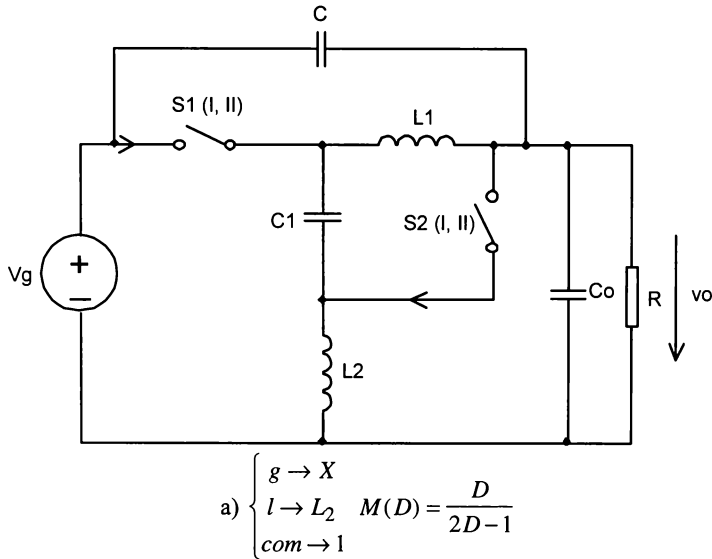
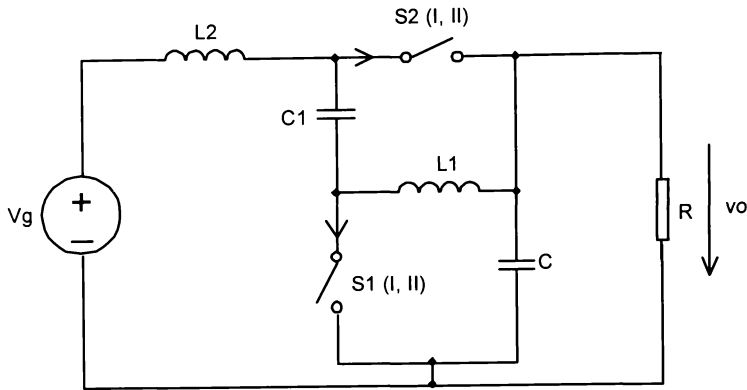


Fig. 1.69. Obținerea convertorului Watkins-Johnson invers.

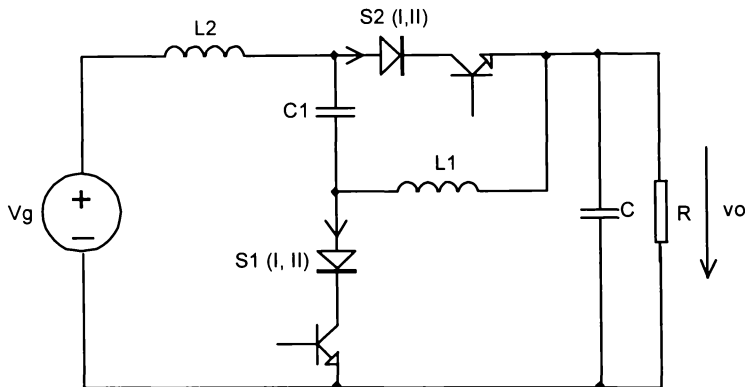
**Pasul 2.** Se înlocuiește sursa de tensiune de alimentare cu o sarcină și sarcina cu o sursă de tensiune, astfel încât polaritatea sursei de alimentare introduse este aceeași cu polaritatea definită pentru tensiunea de ieșire a convertorului original, iar polaritatea tensiunii de ieșire pe noua sarcină este aceeași cu polaritatea tensiunii de alimentare a convertorului original.

**Pasul 3.** În noul convertor rezultat se menține modul de comandă al întrerupătoarelor, apoi se găsește materializarea practică a acestora în urma analizei funcționării noului convertor în CICM.

**Pasul 4.** Pentru comodate se exprimă raportul static de conversie funcție de factorul de umplere al întrerupătorului (întrerupătoarelor) active ale noului convertor.



$$a) \begin{cases} g \rightarrow X \\ l \rightarrow L_2 \\ com \rightarrow 1 \end{cases} M(D) = \frac{1-D}{1-2D}$$



b)

Fig. 1.70. Obținerea unei noi variante de convertor Watkins-Johnson invers.

În acest mod, dacă se merge până la pasul 4 și se notează cu  $M_o(D)$  și  $M_b(D)$  rapoartele statice de conversie ale convertorului original respectiv al celui obținut prin inversie bilaterală generalizată avem:

$$M_b(D) = \frac{1}{M_o(D)} \quad (44)$$

Se poate constata ușor că teorema inversiei bilaterale generalizate are ca și caz particular inversia bilaterală clasică. Cu această teoremă rezultă că următoarele perechi de convertoare se obțin unul din altul prin inversie bilaterală generalizată:

ZETA  $\leftrightarrow$  SEPICWATKINS-JOHNSON  $\leftrightarrow$  WATKINS-JOHNSON INVERS

(ambele cu curenți de intrare și de ieșire netezi)

VARIANTĂ DE WATKINS-JOHNSON  $\leftrightarrow$  VARIANTĂ DE WATKINS-JOHNSON INVERS

### 1.6.3. Celule de bază complete care generează familia convertoarelor având curenți cu pulsații nule

Deoarece aceste convertoare fac obiectul unui capitol separat se prezintă numai celulele de bază și principalele structuri. Dacă se pleacă de la celula de bază completă BBB din Fig. 1.43b se remarcă faptul că în orice convertor tensiunile  $V_{12}$  și  $V_{L2}$  (și deci și  $V_{1L}$ , ca diferență) sunt tensiuni bine filtrate, fiind în fond combinații liniare dintre tensiunea de intrare și cea de ieșire. Pentru obținerea de convertoare cu pulsații de curent nule ideea de bază este cuplarea unor bobine ce au proprietatea că au tensiuni egale în funcționare. Prin urmare în celula din Fig. 1.43 mai trebuie plasată o bobină  $L_1$ , care să aibă tensiunea egală cu a bobinei originale  $L$ . Acest lucru nu este posibil decât dacă se mai inserează o capacitate  $C_1$ , cu rol de memorare a unei combinații liniare dintre tensiunile  $V_{12}$ ,  $V_{L2}$  și  $V_{1L}$ . Rezultă imediat variantele de celule de bază din Fig. 1.71. De menționat că celula de bază din Fig. 1.71a a fost introdusă în [6] de Bîrcă-Gălățeanu.

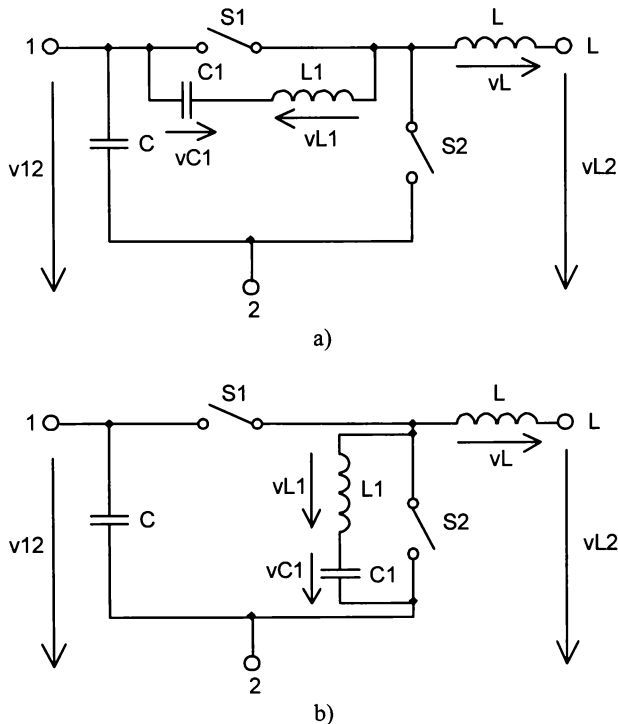


Fig. 1.71. Celule de bază care complete deduse din celula BBB și care generează convertoare cu curenți potențialii cu pulsații nule.

În cazul celulei de bază din Fig. 1.71a condensatorul  $C_1$ , ales suficient de mare, se încarcă la tensiunea cvasiconstantă

$$V_{C_1} = V_{12} - V_{L2} \quad (45)$$

Pe durata cât  $S_1$  conduce și  $S_2$  este blocat tensiunile pe bobine sunt:

$$v_{LON} = V_{12} - V_{L2} \quad (46)$$

$$v_{L_1ON} = v_{C_1} = V_{12} - V_{L2} \quad (47)$$

În starea topologică cu  $S_1$  blocat și  $S_2$  în conducție tensiunile pe bobine au valorile:

$$v_{LOFF} = -V_{L2} \quad (48)$$

$$v_{L1OFF} = -V_{12} + V_{C1} = -V_{L2} \quad (49)$$

unde s-a folosit și (45) pentru a deduce (49). Din (4), (47), (48) și (49) rezultă că  $v_L = v_{L1}$ , q.e.d.

În cazul celei din Fig. 1.71b tensiunile pe capacitatea  $C_1$  și pe bobine se găsesc similar ca fiind:

$$v_{C1} = V_{L2} \quad (50)$$

$$v_{LON} = V_{12} - V_{L2} \quad (51)$$

$$v_{L1ON} = V_{12} - v_{C1} = V_{12} - V_{L2} \quad (52)$$

$$v_{LOFF} = -V_{L2} \quad (53)$$

$$v_{L1OFF} = -v_{C1} = -V_{L2} \quad (54)$$

Este interesant de observat că cele două celule de bază din Fig. 1.71 se obțin una din alta dacă se permută între ele  $S_1$  și  $S_2$ . Deci ele vor genera din punct de vedere topologic aceeași familie de convertoare, numai că în cazul celei de a doua comanda va fi complementară. Familia de convertoare obținută plecând de la celula din Fig. 1.71a este prezentată în Fig. 1.72, 1.73 și 1.74. De reținut că prin locul de conectare a bornei "L" se stabilește tipul de convertor.

Alte convertoare cu pulsații reduse au fost raportate în [14]. Celula de bază este prezentată în Fig. 1.75. Din motive de spațiu nu se mai dau și convertoarele rezultante, modul lor de obținere urmărind întocmai procedura prezentată.

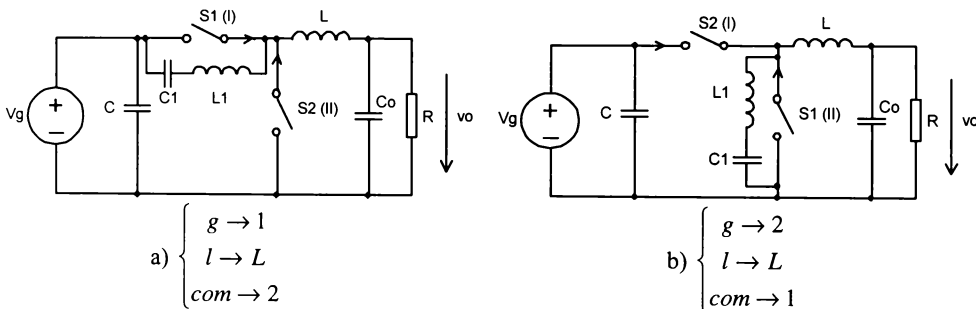


Fig. 1.72. Convertoare BUCK cu curent de ieșire potențial fără pulsații.

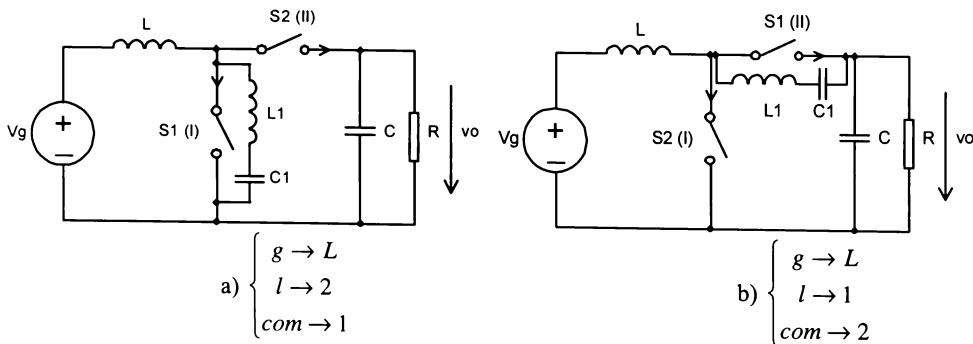


Fig. 1.73. Convertoare BOOST cu curent de intrare potențial fără pulsații.

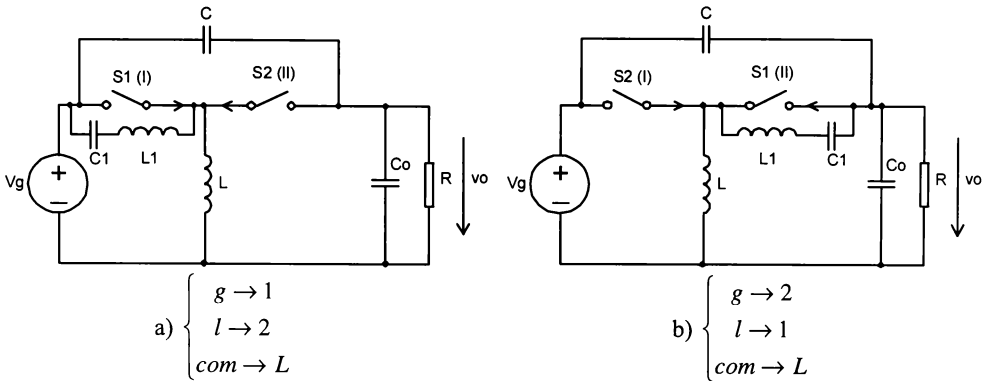


Fig. 1.74. Convertoare BUCK-BOOST cu curent de intrare potențial fără pulsații.

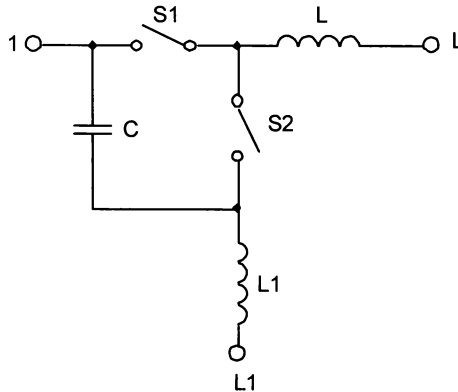


Fig. 1.75. O celulă de bază completă capabilă să genereze convertoare cu curenți potențiali fără pulsații.

#### 1.6.4. Familia convertoarelor pătratice

Se cunoaște că în cazul convertoarelor PWM convenționale raportul de conversie  $M = \frac{V_o}{V_g}$

este limitat atât superior cât și inferior astfel:

- valoarea minimă  $M_{\min}$  este limitată de timpul minim de conducție al tranzistorului (consecință a timpilor de comutație finiți). Evident,  $M_{\min}$  devine cu atât mai restrictiv cu cât frecvența de comutație este mai mare.
- valoarea maximă,  $M_{\max}$ , nu poate depăși 1 în cazul convertorului BUCK, în timp ce pentru celelalte convertoare  $M_{\max}$  este limitat de scăderea randamentului convertorului pe măsură ce factorul de umplere  $D$  se apropie de 1, datorită componentelor neideale, dar și de solicitările de tensiune și curent ale dispozitivelor semiconductoare.

Există o serie de aplicații de conversie dc-dc care necesită un raport de conversie foarte mare sau foarte mic. Se pot astfel enumera:

- stabilizatoare de tensiune foarte mică (3,3V sau 5V) alimentate de la tensiuni medii (50V) la care nu este obligatorie izolarea galvanică (de exemplu în aparatura cosmică).
- surse de tensiune continuă de laborator, în care tensiunea de ieșire are o gamă largă.
- surse de tensiune pentru alimentarea instalațiilor de electroliză.
- surse de tensiune “universale”, alimentate de la rețele 110V/220V.
- surse de tensiune pentru aplicații cu ultrasunete.

În toate aceste cazuri topologiile de convertoare PWM convenționale vor trebui să funcționeze cu factori de umplere fie foarte mici fie foarte mari, ceea ce nu se poate realiza decât la frecvențe nepermis de joase.

Maksimović și Čuk [118] au introdus o serie de convertoare numite “pătratică”. Desigur că astfel de convertoare se pot realiza și cascând convertoare convenționale, așa cum se propune în [124]. Totuși astfel de structuri necesită cel puțin două tranzistoare și au o complexitate relativ ridicată atât în partea de putere cât și în partea de comandă, fapt ce compromite potențialele avantaje ale raportului de conversie extins. Tot Maksimović, împreună cu Čuk, arată însă în [118] că se pot realiza convertoare cu dependență pătratică de factorul de umplere cu cel mult 2 capacități, 2 bobine și 4 întrerupătoare dintre care numai unul este tranzistor, restul fiind diode. Autorii dau, pentru un raport static de conversie, un număr de elemente reactive și un număr de întrerupătoare fixate, un procedeu sistematic de sinteză a tuturor topologiilor de convertoare PWM posibile care furnizează raportul respectiv de conversie. Procedeu de sinteză este implementat într-un program care caută toate configurațiile posibile și le reține pe acelea care satisfac specificațiile de intrare. Deci nu se poate preciza apriori, pentru un set de condiții de intrare, câte convertoare soluție se vor obține. Maksimović și Čuk au aplicat acest procedeu pentru a găsi toate convertoarele PWM de ordinul 4 care furnizează următoarele rapoarte de conversie:

$$M(D) = D^2, \quad -\frac{D^2}{1-D}, \quad \frac{D^2}{(1-D)^2} \quad (55)$$

Nu vom relua rezultatele celor doi autori, ci vom încerca să introducem noua procedură de sinteză arătând că structurile de convertoare obținute de ei fac parte din clase mai largi de convertoare cu rapoarte de conversie pătratică, ce au ca rădăcină o unică celulă de bază. Vom obține astfel noi structuri de convertoare cu caracteristică pătratică. În plus, se va sintetiza o nouă clasă de convertoare cu caracteristică pătratică pe care o vom denumi  $\delta_1$ , ce are ca reprezentant convertorul cu caracteristica de conversie de forma  $M(D) = \frac{1}{(1-D)^2}$ . Se vor utiliza denumirile propuse de autori pentru convertoare, acestea fiind prin extensie atribuite și claselor (de exemplu  $\alpha_1$ ,  $\beta_3$ , etc.).

Celula de bază pentru convertoare de clasă  $\alpha_1$  este prezentată în Fig. 1.76. Întrerupătoarele  $S_1$  și  $S_1'$  se comandă cu același semnal având factorul de umplere  $D$ , iar  $S_2$  și  $S_2'$  cu semnal complementar. Acest mod de comandă se va păstra și pentru viitoarele celule de bază analizate. Datorită structurii asimetrice a celulei vor rezulta 6 convertoare distincte. Aplicând teorema 3 rezultă că vom obține 2 convertoare de tip BUCK, 2 de tip BOOST și 2 de tip BUCK-BOOST. Același lucru rămâne valabil și pentru celelalte celule. După implementarea practică a întrerupătoarelor se obțin 3 convertoare având fiecare câte 3 diode și un întrerupător funcționând în cadranul I și 3 convertoare având o singură diodă și 3 întrerupătoare funcționând în cadranul I. Aceste din urmă 3 structuri nu prezintă interes din punct de vedere practic din cauza comenzii dificile, motiv pentru care nu le mai reproducem mai jos și la fel vom face și în cazul celorlalte celule. Cele 3 convertoare conținând 3 diode și un întrerupător de primul cadran sunt exemplificate în Fig. 1.77-1.79 împreună cu rapoartele de conversie corespunzătoare.

În continuare, în figurile 1.80-1.103 sunt reprezentate celelalte celule de bază complete și convertoarele cu un tranzistor și trei diode rezultate. Convertorul  $\beta_1$ -BUCK-BOOST, a fost descoperit independent de firma Lambda Electronics. Convertoarele  $\alpha_1$ -BUCK,  $\alpha_2$ -BUCK,  $\beta_1$ -BUCK-BOOST,  $\beta_2$ -BUCK-BOOST,  $\beta_3$ -BUCK-BOOST și  $\gamma_1$ -BUCK, au fost raportate de Maksimović și Ćuk în [118]. Este interesant de menționat faptul că unele topologii pătrățice au o oarecare asemănare cu o cascadă de convertoare.

De exemplu în Fig. 1.77  $S_1'$ ,  $S_2'$ ,  $L_1$  și  $C_1$  au o funcționare de genul convertorului BUCK, numit convertor BUCK pasiv, iar  $S_1$ ,  $S_2$ ,  $L$  și opțional  $C_o$  formează un nou convertor BUCK numit convertorul BUCK activ. Comutarea diodelor din convertorul pasiv este rezultatul curentului de intrare pulsatoriu (discontinuu) al convertorului activ. Rezultatul poate fi generalizat [119] în sensul că orice convertor PWM cu curent de intrare pulsatoriu precedat de un convertor BUCK pasiv își va multiplica raportul static de conversie propriu cu  $D$ . Acest lucru se regăsește de exemplu în convertorul  $\beta_2$ -BUCK-BOOST - Fig. 1.91 - unde se recunoaște o cascadă dintre un convertor activ BUCK-BOOST precedat de un convertor BUCK pasiv. Este important de reținut că însuși convertorul BUCK are curent de intrare pulsatoriu. În consecință se pot cascada  $n-1$  etaje BUCK pasive cu un singur convertor BUCK activ, obținându-se astfel un convertor cu un singur tranzistor ce are raportul static de conversie egal cu  $D^n$ . Structura unui astfel de convertor cu  $M(D) = D^n$  este prezentată în Fig. 1.104. Tranzistorul și diodele legate cu anodul la bara de minus a sursei conduc simultan, în timp ce diodele ce au catodii legați în bobine au conducție complementară. Cu toate acestea, la unele topologii nu se poate stabili o cascadă sau o altă combinație de convertoare de bază. Este, de exemplu, cazul topologiilor  $\alpha_2$  și  $\beta_3$ .

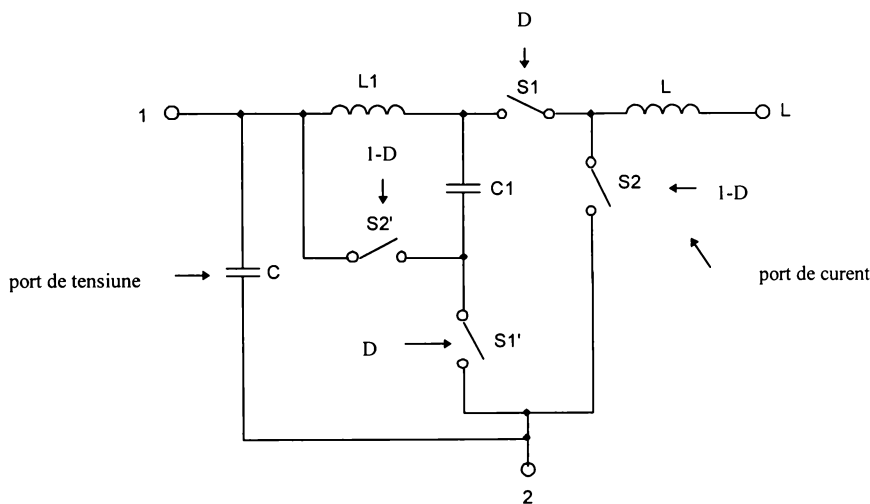
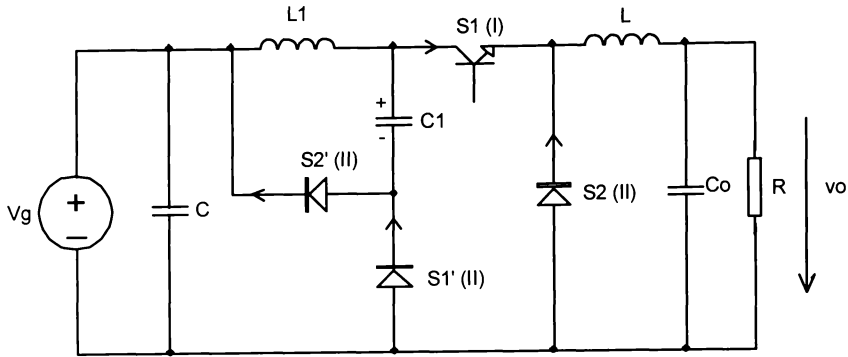
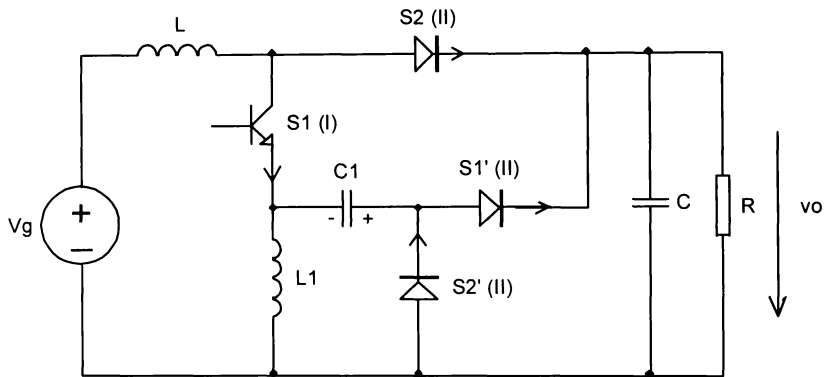
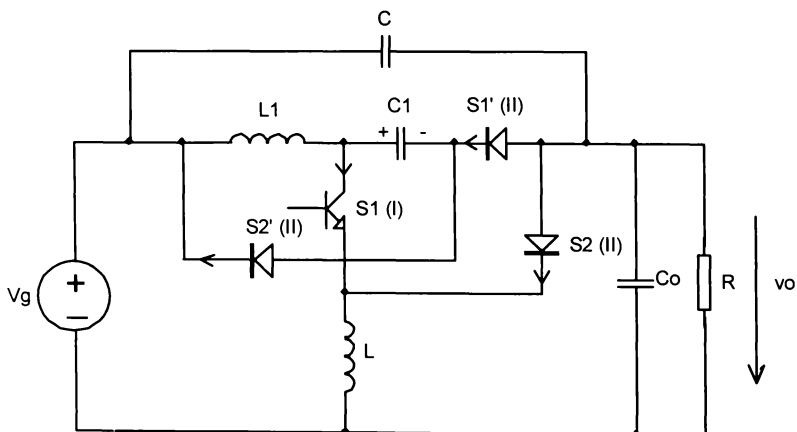


Fig. 1.69. Celula de bază  $\alpha_1$ .

Fig. 1.77. Convertorul  $\alpha_1$ -BUCK.  $M(D) = D^2$ .Fig. 1.78. Convertorul  $\alpha_1$ -BOOST.  $M(D) = \frac{1}{1-D^2}$ .Fig. 1.79. Convertorul  $\alpha_1$ -BUCK-BOOST.  $M(D) = -\frac{D^2}{1-D^2}$ .



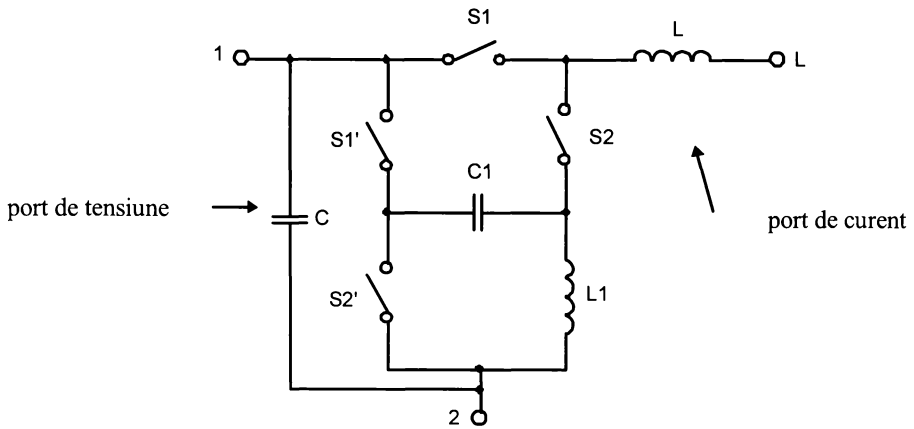


Fig. 1.80. Celula de bază  $\alpha_2$ .

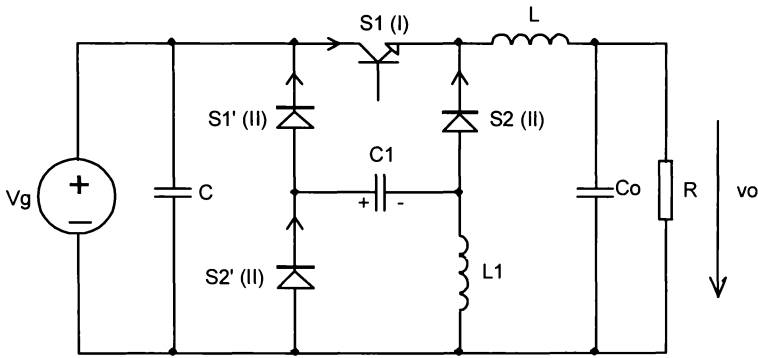


Fig. 1.81. Convertorul  $\alpha_2$ -BUCK.  $M(D) = D^2$ .

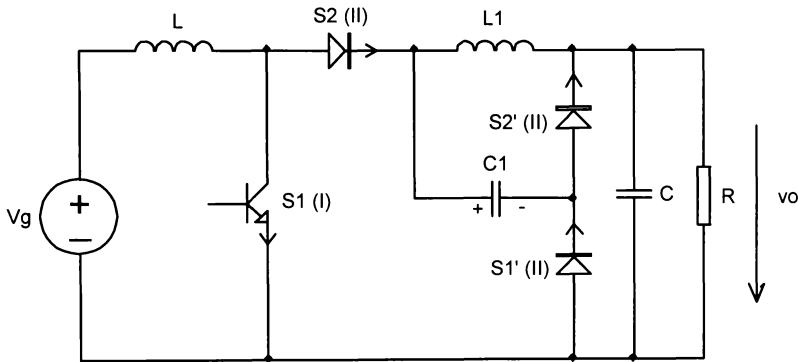


Fig. 1.82. Convertorul  $\alpha_2$ -BOOST.  $M(D) = \frac{1}{1-D^2}$ .

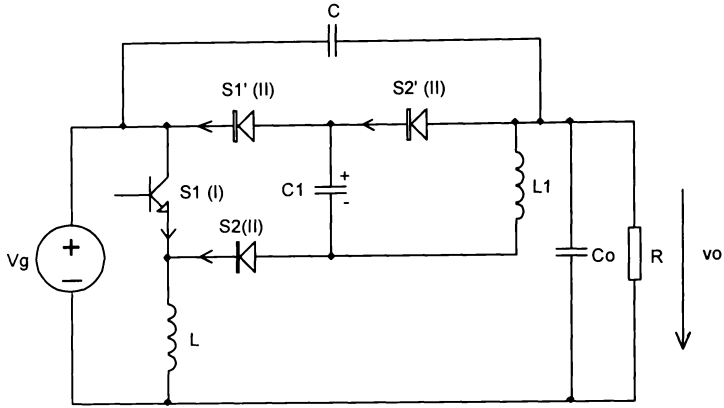


Fig. 1.83. Convertorul  $\alpha_2$ -BUCK-BOOST.  $M(D) = -\frac{D^2}{1-D^2}$ .

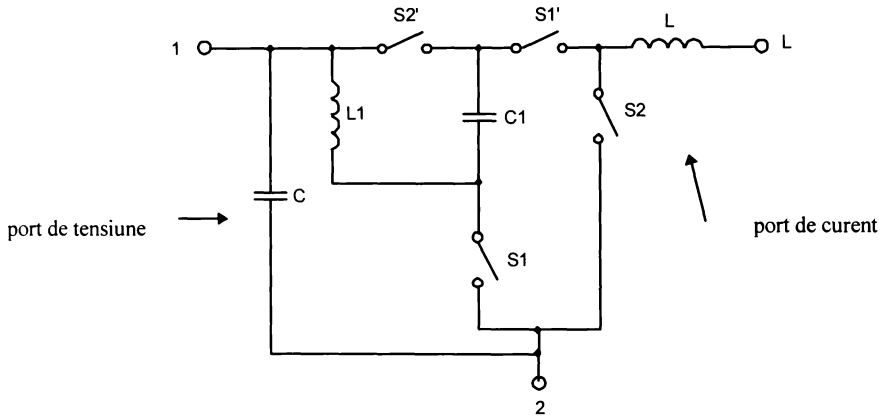


Fig. 1.84. Celula de bază  $\beta_1$ .

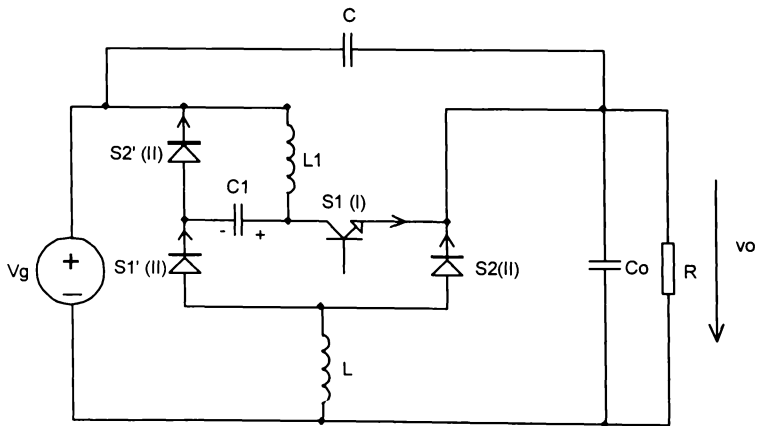


Fig. 1.85. Convertorul  $\beta_1$ -BUCK.  $M(D) = \frac{D^2}{D^2 - D + 1}$ .

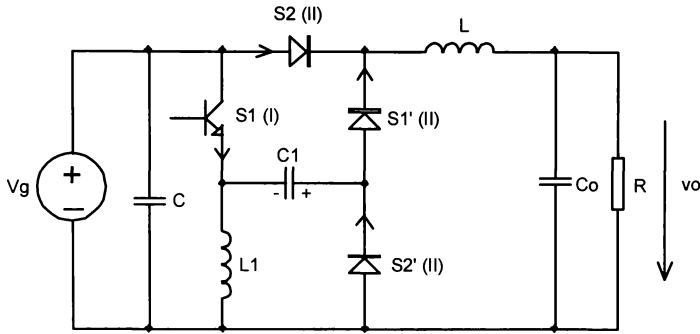


Fig. 1.86. Convertorul  $\beta_1$ -BOOST.  $M(D) = \frac{D^2 - D + 1}{1 - D}$ .

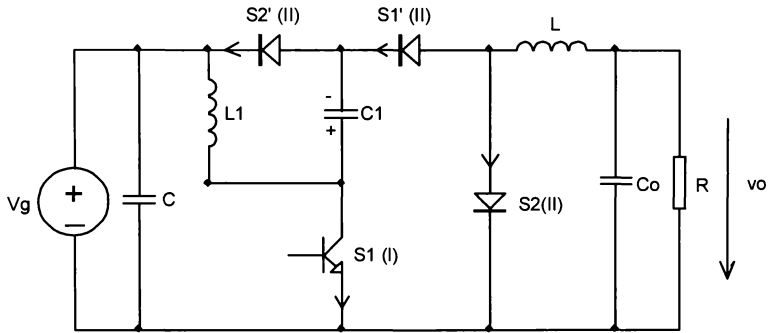


Fig. 1.87. Convertorul  $\beta_1$ -BUCK-BOOST.  $M(D) = -\frac{D^2}{1 - D}$ .

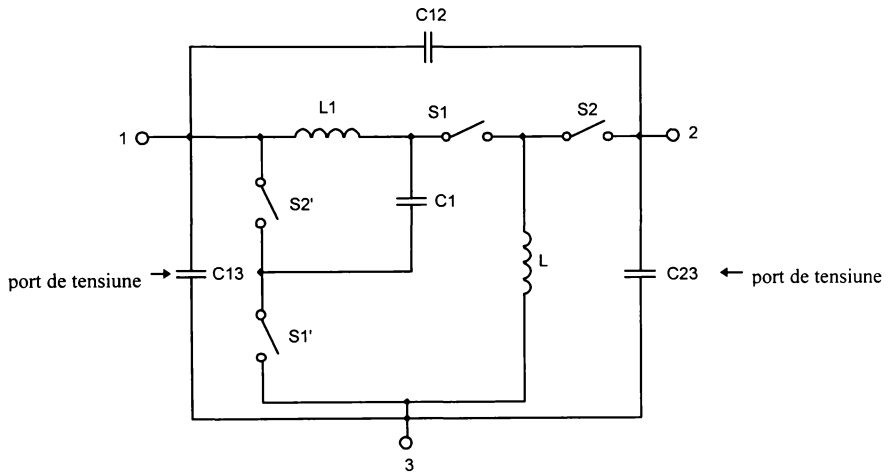


Fig. 1.88. Celula de bază  $\beta_2$ .

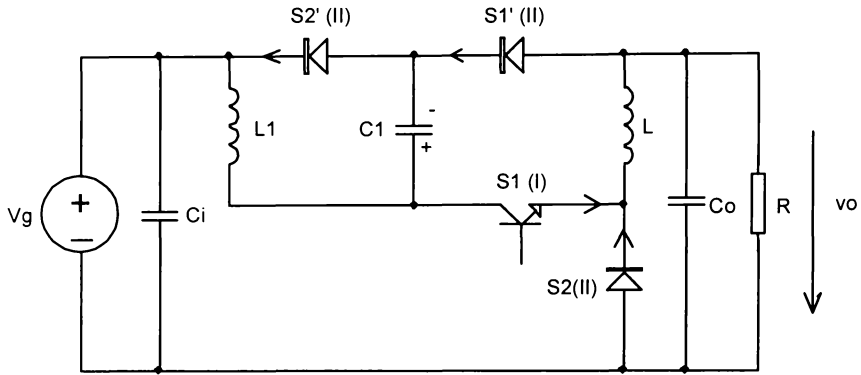


Fig. 1.89. Convertorul  $\beta_2$ -BUCK.  $M(D) = \frac{D^2}{D^2 - D + 1}$ .

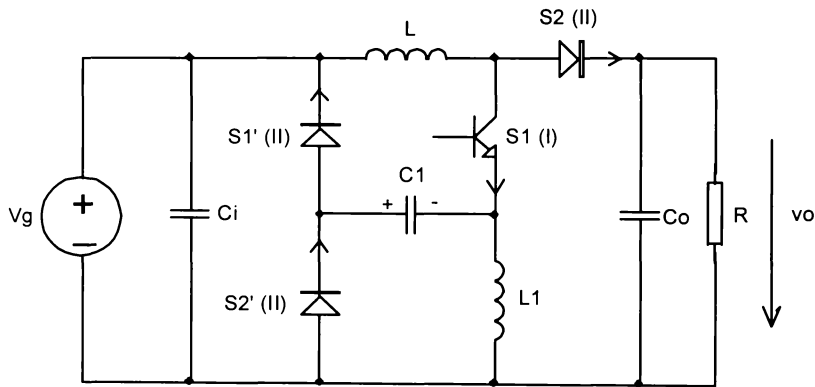


Fig. 1.90. Convertorul  $\beta_2$ -BOOST.  $M(D) = \frac{D^2 - D + 1}{1 - D}$ .

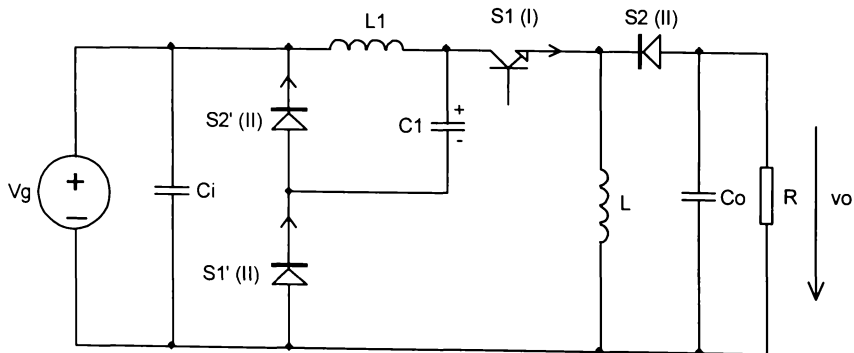


Fig. 1.91. Convertorul  $\beta_2$ -BUCK-BOOST.  $M(D) = -\frac{D^2}{1 - D}$ .

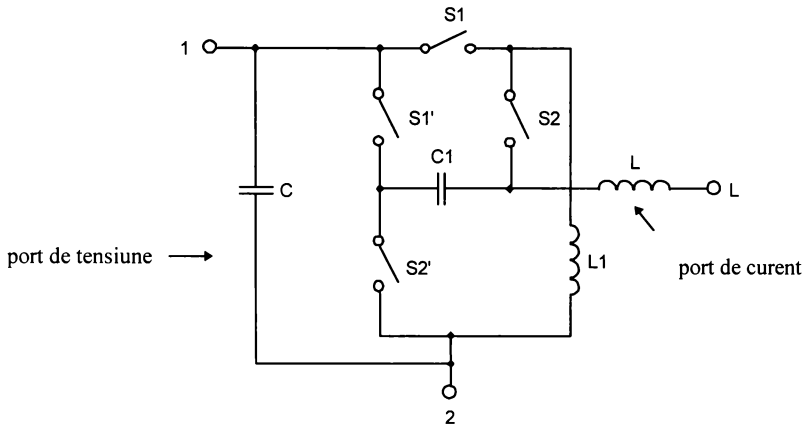


Fig. 1.92. Celula de bază  $\beta_3$ .

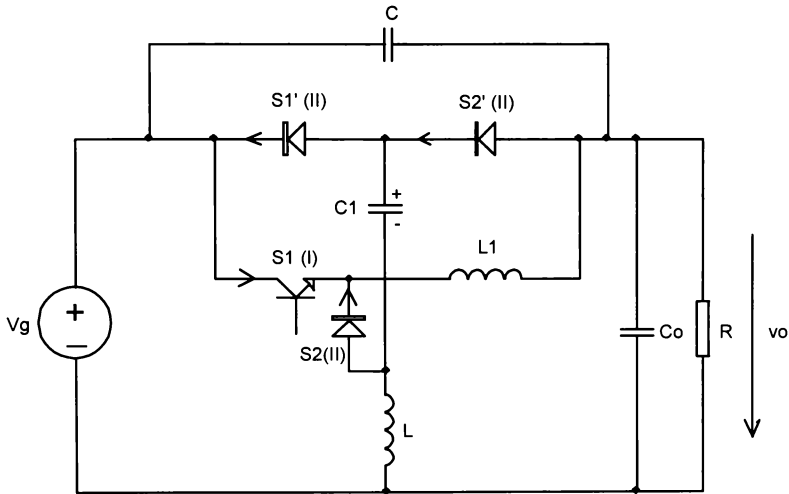


Fig. 1.93. Convertorul  $\beta_3$ -BUCK.  $M(D) = \frac{D^2}{D^2 - D + 1}$ .

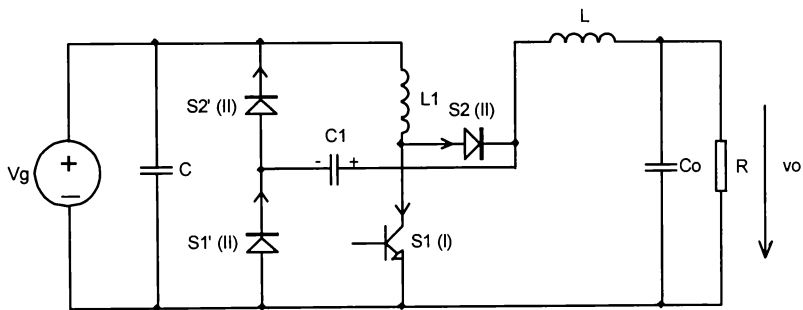
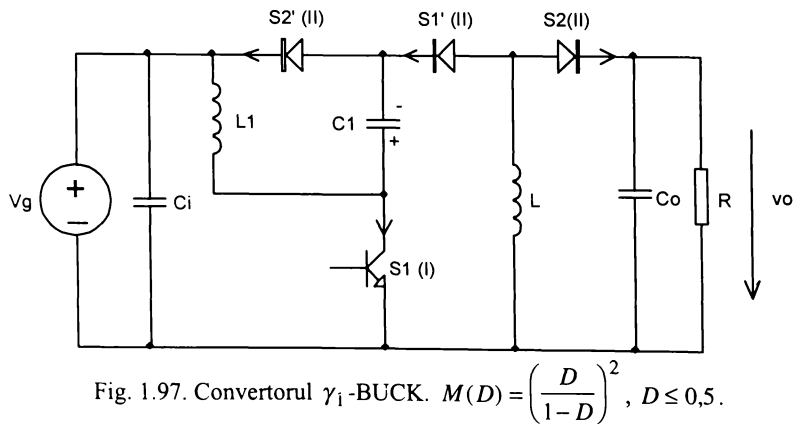
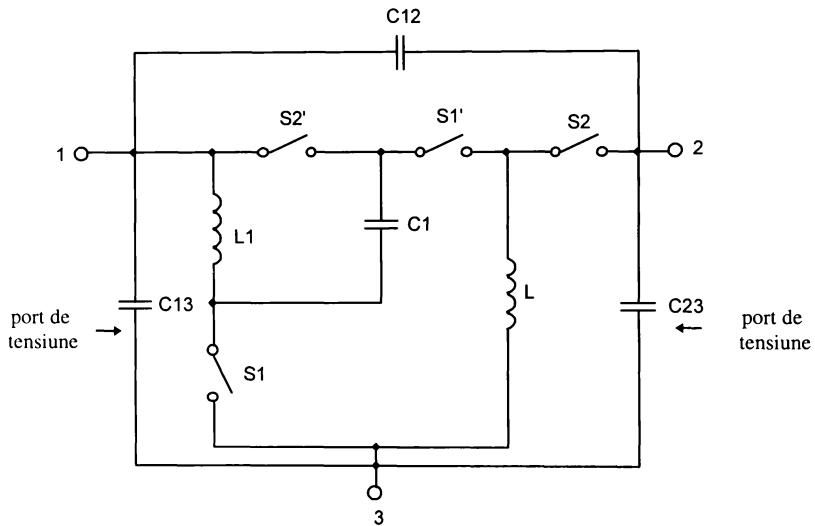
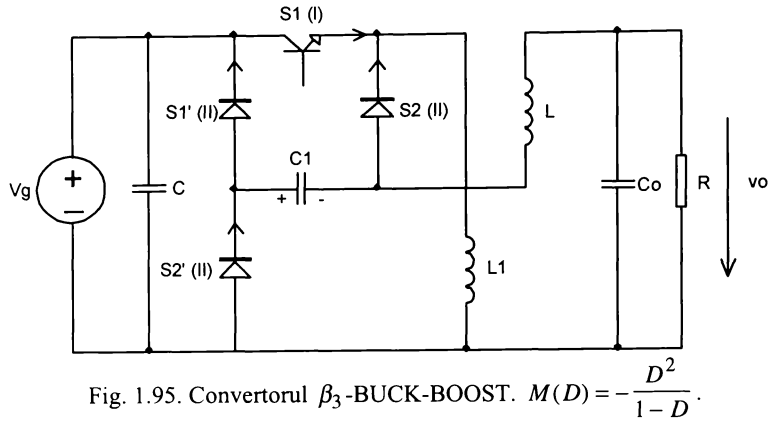


Fig. 1.94. Convertorul  $\beta_3$ -BOOST.  $M(D) = \frac{D^2 - D + 1}{1 - D}$ .



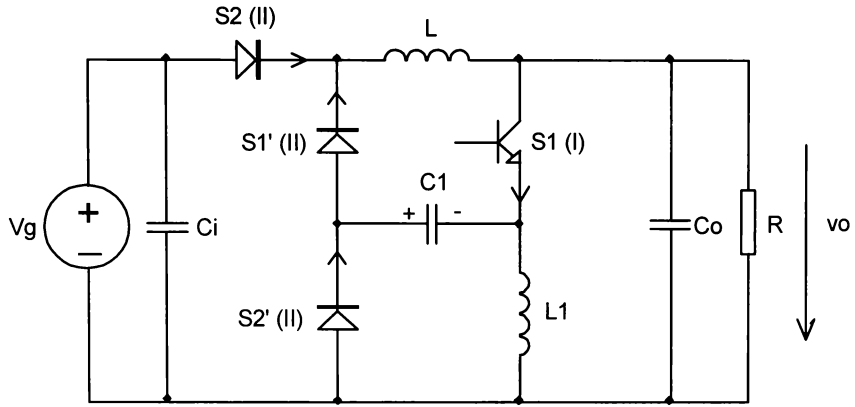


Fig. 1.98. Convertorul  $\gamma_1$ -BOOST.  $M(D) = \frac{(1-D)^2}{1-2D}$ ,  $D \leq 0,5$ .

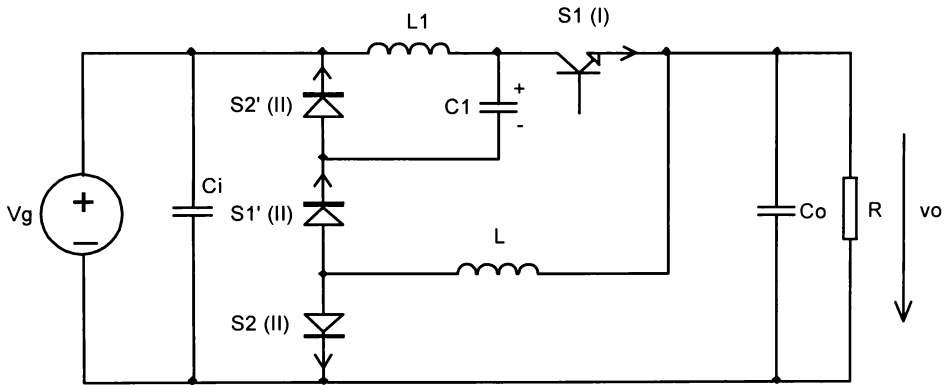


Fig. 1.99. Convertorul  $\gamma_1$ -BUCK-BOOST.  $M(D) = -\frac{D^2}{1-2D}$ ,  $D \leq 0,5$ .

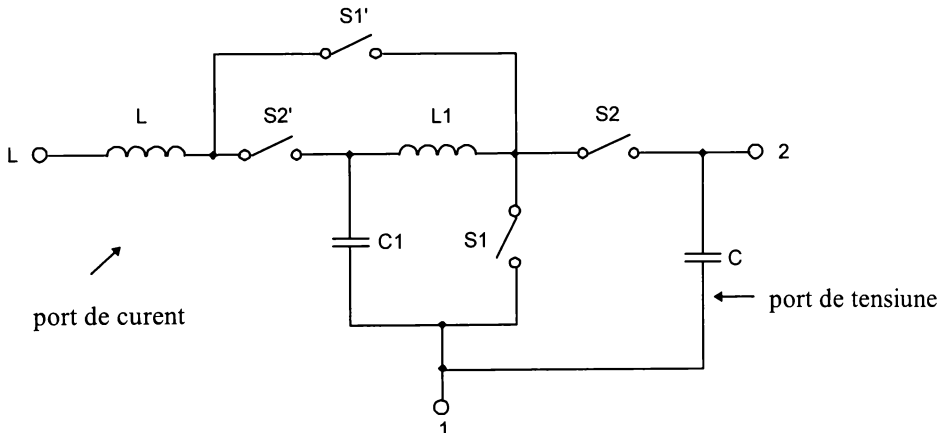
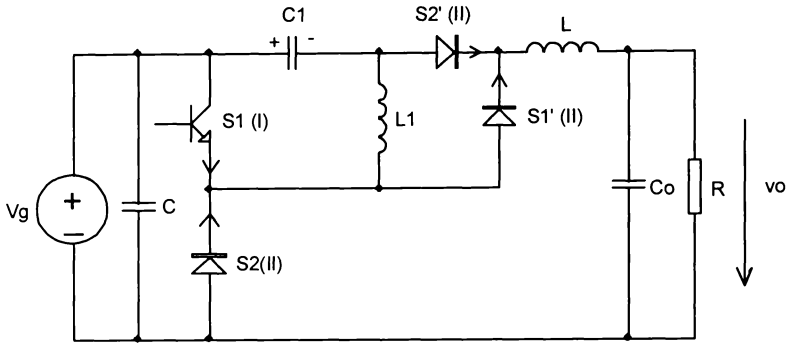
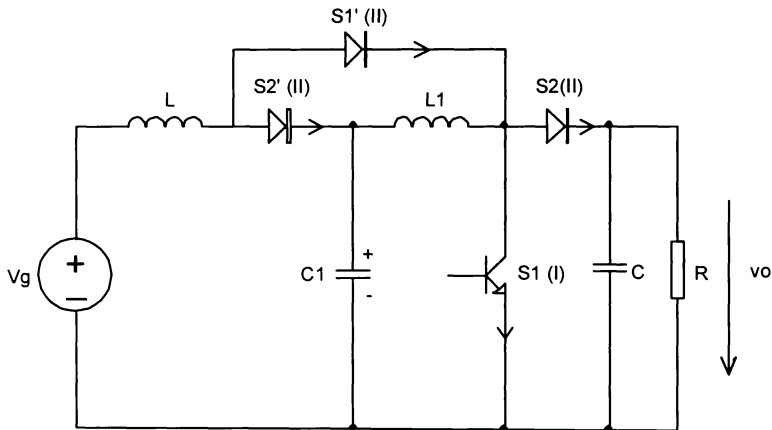
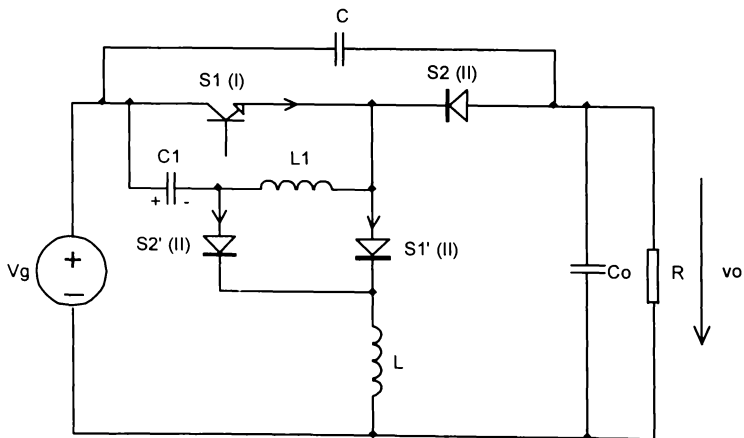


Fig. 1.100. Celula de bază  $\delta_1$ .

Fig. 1.101. Convertorul  $\delta_1$ -BUCK.  $M(D) = 1 - (1 - D)^2$ .Fig. 1.102. Convertorul  $\delta_1$ -BOOST.  $M(D) = \frac{1}{(1 - D)^2}$ .Fig. 1.103. Convertorul  $\delta_1$ -BUCK-BOOST.  $M(D) = -\frac{1 - (1 - D)^2}{(1 - D)^2}$ .



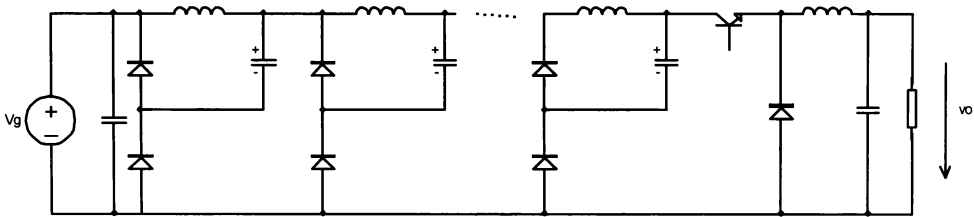


Fig. 1.104. Convertor cu  $M(D) = D^n$  format prin cascada a  $n-1$  convertore BUCK pasive și un convertor BUCK activ.

De menționat că în cazul celulelor  $\beta_2$  și  $\gamma_1$ , ambele porturi ale celulei sunt de tensiune. De aceea dintre capacitățile  $C_{13}$ ,  $C_{23}$  și  $C_{12}$  desenate în Fig. 1.88 în convertore se vor utiliza evident numai două, unul la intrarea convertorului și altul la ieșirea sa, notate  $C_i$  respectiv  $C_o$ . De asemenea, în celula  $\gamma_1$ , pentru ca trei dintre întrerupătoare să fie diode este necesar ca  $D \leq 0,5$ .

În ceea ce privește alegerea unuia sau altuia dintre convertorele ce au același raport static de conversie, aceasta se face în urma unor simple analize de curent continuu, coroborate cu dispozitivele pe care proiectantul le are la dispoziție. În cazul convertoarelor  $\alpha_1$ -BUCK și  $\alpha_2$ -BUCK cel care diferă este numai curentul mediu  $I_1$  prin bobina  $L_1$  și alegerea se face funcție de valoarea factorului de umplere  $D$ . Dintre convertorele  $\beta_1$ -BUCK-BOOST,  $\beta_2$ -BUCK-BOOST și  $\beta_3$ -BUCK-BOOST, primul are cei mai mici curenți medii prin bobine și prin dispozitivele de comutație. Deci dacă se utilizează aceleași componente în realizarea convertorului  $\beta_1$ -BUCK-BOOST va avea cele mai mici pierderi de conducție și energie reactivă stocată pentru aceeași putere procesată.

Considerente similare se pot face pentru alte categorii de convertore având același raport static de conversie.

Cella de bază  $\delta_1$  și convertorele  $\delta_1$ -BUCK,  $\delta_1$ -BOOST,  $\delta_1$ -BUCK-BOOST au fost raportate pentru prima oară de autor

Faptul esențial este însă acela că metoda de sinteză propusă conduce la noi tipuri de convertore pătratic. Astfel convertorele  $\alpha_1$ -BOOST,  $\alpha_1$ -BUCK-BOOST,  $\alpha_2$ -BOOST,  $\alpha_2$ -BUCK-BOOST,  $\beta_1$ -BUCK,  $\beta_1$ -BOOST,  $\beta_2$ -BUCK,  $\beta_2$ -BOOST,  $\beta_3$ -BUCK,  $\beta_3$ -BOOST,  $\gamma_1$ -BOOST,  $\gamma_1$ -BUCK-BOOST sunt originale, fiind convertore descoperite de autor. În acest fel, împreună cu convertorele propuse de Maksimović și Čuk, ele sunt cuprinse într-o clasificare sistematică a convertoarelor pătratic.

În mod similar cu analiza făcută pentru convertorul  $\alpha_1$ -BUCK și în convertorul  $\delta_1$ -BOOST se poate identifica un convertor BOOST pasiv format din  $L-S_1-S_2-C_1$  și un convertor BOOST pasiv alcătuit din  $L_1-S_1-S_2-C$ . Se pot conecta  $n-1$  celule BOOST pasive și o celulă BOOST activă ca în Fig. 1.105, rezultând un convertor de tip BOOST cu  $M(D) = \frac{1}{(1-D)^n}$ .

Convertorele de tip  $\delta_1$  au fost verificate prin simulare remarcându-se coincidența predicțiilor teoretice cu rezultatele simulărilor. În Fig. 1.106 sunt prezentate rezultatele simulării pentru convertorul  $\delta_1$ -BOOST din Fig. 1.102 (vezi anexa A3 pentru fișierul de simulare), având următorii parametri:

$$V_g = 10V; \quad C_1 = 4,7 \mu F; \quad L_1 = 2 mH; \quad C = 2,2 \mu F; \quad R = 2 k\Omega; \quad f_s = 50 kHz; \quad D = 0,75$$

Elementele semiconductoare s-au modelat prin rezistențe de  $10 m\Omega$  în conducție și  $10 M\Omega$  în blocare.

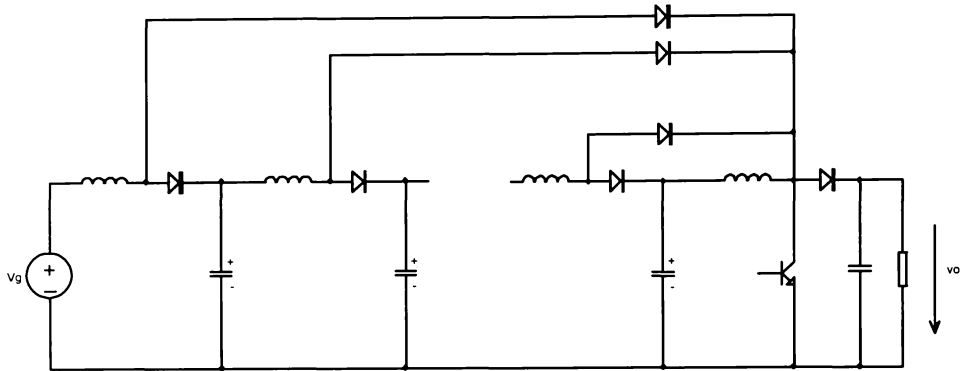


Fig. 1.105. Convertor cu  $M(D) = \frac{1}{(1-D)^n}$  format prin cascada a  $n-1$  convertoare BOOST pasive și un convertor BOOST activ.

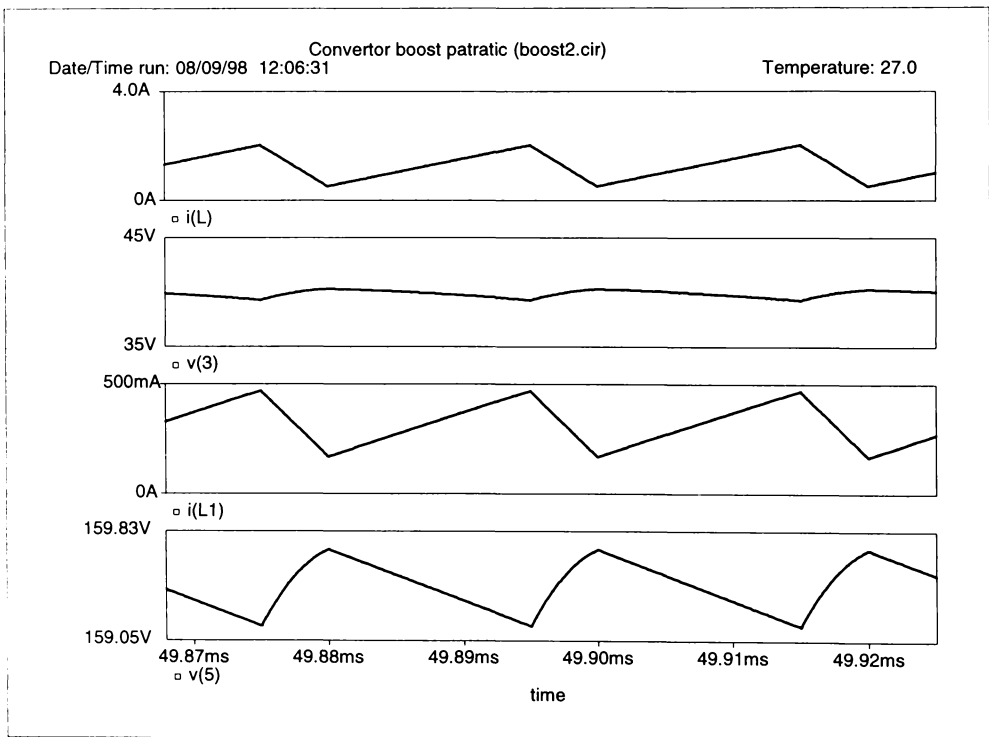


Fig. 1.106. Noua variantă de convertor propusă de autor și denumită  $\delta_1$ -BOOST - rezultatele simulărilor. Simulator PETS (Boulder University Colorado) cu postprocesor grafic PROBE (Microsim Corp.). De sus în jos:  $i_L$ ,  $v_{C1}$ ,  $i_{L1}$ ,  $v_o$ .

## 1.7. Concluzii

Tehnicile de sinteză după natura sursei și sarcinii și prin cascaderă sunt de natură pur *combi-națională*. Sinteza prin inversie bilaterală și prin dualitate reprezintă tehnici de sinteză *evoluționiste*. Contrar așteptărilor, deși tehnicile evoluționiste sunt categoric superioare celor combi-naționale, totuși ele singure nu realizează o sinteză completă, care să și grupeze convertoarele în familii. Capitolul parcurs duce la concluzia că tehnicile *mixte* sunt cele mai performante. Astfel de tehnici pleacă de la celule de bază (natură evoluționistă) care apoi sunt conectate în toate variantele posibile (aspectul combi-național) între sursă și sarcină. Mai mult, celulele de bază se pot obține din rețele de comutație făcând simple considerații de funcționare dinamică și modificându-le în consecință.

Contribuțiile autorului în privința sintezei convertoarelor dc-dc sunt date de:

1. propozițiile 1, 2, 3 și 4 (paragraful 1.4.2) care extind noțiunea de dualitate în cazul conceptului general și abstract de întrerupător introdus de Erickson.
2. teorema 3 (paragraful 1.6), care stabilește o condiție necesară și suficientă legată de generarea convertoarelor de tip BUCK, BOOST și BUCK-BOOST plecând de la o celulă de bază cu 3 terminale. Aplicabilitatea ei se regăsește spectaculos în generarea convertoarelor pătratic.
3. teorema 4 (paragraful 1.6), ce stabilește o condiție necesară relativă la aplicabilitatea practică a unor celule de bază.
4. propoziția 7 (paragraful 1.6), referitoare la o celulă de bază completă care generează convertoare cu curenți de intrare și ieșire fără fronturi nule (discontinuități) și deci cu foarte bune proprietăți de compatibilitate electromagnetică.
5. o nouă celulă de bază - Fig. 1.43 (paragraful 1.6.1) - pentru generarea convertoarelor PWM de bază (BUCK, BOOST, BUCK-BOOST)
6. o rețea generală de comutație - Fig. 1.48c (paragraful 1.6.2) - din care rezultă două celule de bază: una din care se generează familia ĆUK - Fig. 1.52 - și alta generatoare a familiei numite de autor ZSWJ - Fig. 1.64. Cu noua metodă de sinteză propusă se obțin astfel următoarele noi structuri de convertoare, contribuții originale ale autorului:
  - un convertor BUCK și un convertor BOOST, ambele convertoare având atât curenți de intrare cât și curenți de ieșire fără fronturi nule (discontinuități), deci cu foarte bune proprietăți de compatibilitate electromagnetică comparativ cu structurile clasice corespunzătoare. Convertoarele au fost testate și experimental confirmând analizele teoretice.
  - un convertor Watkins-Johnson și un convertor Watkins-Johnson invers. Fiecare din aceste convertoare are curenți de intrare și ieșire fără fronturi nule, spre deosebire de convertoarele clasice cu același nume în care acești curenți au un spectru larg datorat fronturilor nule.
  - o variantă de convertor Watkins-Johnson și o variantă de convertor Watkins-Johnson invers (termenul de "variantă" trebuie înțeles în sensul că topologiile sunt complet diferite deși au același raport static de conversie) având posibilitatea de a obține, prin cuplarea bobinelor, curenți de ieșire respectiv de intrare eminate constanți, lucru imposibil în convertoarele clasice cu același nume.
7. enunțarea teoremei inversiei bilaterale generalizate (paragraful 1.6.2), teoremă originală ce cuprinde ca și caz particular inversia bilaterală clasică, cu aplicații imediate în sinteza convertoarelor cu tensiune de ieșire bipolară și a oricărei structuri de convertor în general.
8. găsirea (paragraful 1.6.3), pe lângă convertoarele cu pulsații nule deja raportate, a noi variante de astfel de topologii.
9. aplicarea (paragraful 1.6.4) noii metode de sinteză în obținerea și clasificarea pe familii, în funcție de celula generatoare, a convertoarelor pătratică cu rezultate după cum urmează:

- obținerea a 12 convertoare pătratice noi, convertoare originale și anume:  $\alpha_1$ -BOOST,  $\alpha_1$ -BUCK-BOOST,  $\alpha_2$ -BOOST,  $\alpha_2$ -BUCK-BOOST,  $\beta_1$ -BUCK,  $\beta_1$ -BOOST,  $\beta_2$ -BUCK,  $\beta_2$ -BOOST,  $\beta_3$ -BUCK,  $\beta_3$ -BOOST,  $\gamma_1$ -BOOST și  $\gamma_1$ -BUCK-BOOST.
- propunerea unei noi celule de bază denumite  $\delta_1$  și generarea pe baza ei a unei noi familii de convertoare:  $\delta_1$ -BUCK,  $\delta_1$ -BOOST,  $\delta_1$ -BUCK-BOOST la care convertorul  $\delta_1$ -BOOST are o importanță deosebită ca și convertor puternic ridicător de tensiune dar cu o funcționare a tranzistorului la factori de umplere moderați.

Autorul estimează și propune ca direcții viitoare de cercetare sinteza unitară a convertoarelor având sarcina flotantă în raport cu sursa de alimentare și a convertoarelor cu surse multiple de alimentare.

## CAPITOLUL 2

### Convertoare având curenți de intrare/ieșire cu pulsații nule

Convertoarele PWM dc-dc având curenți de intrare și/sau de ieșire cu pulsații nule au constituit o preocupare majoră, îndeosebi după introducerea reglementărilor severe de compatibilitate electromagnetică și poluare armonică din ultimii ani, dar și cu scopul creșterii gradului de integrare, reducerii gabaritului, a densității de putere ( $W/in^3$ ) și, bineînțeles, a costurilor.

Problema a fost prima dată ridicată de Slobodan Ćuk [26], [29], [30], [31] care introduce și conceptul de magnetică integrată (“integrated magnetics”) legat de convertorul ĆUK. Apoi Wang, Dunford și Mauch [207] propun topologii cu curenți de ieșire respectiv de intrare cu pulsații nule pentru convertoarele BUCK și respectiv BOOST, deși astfel de convertoare au fost raportate mult mai devreme de Severns [167]. Bîrcă-Gălățeanu [6] propune apoi un convertor BUCK-BOOST cu curent de intrare cu pulsații reduse și generalizează modul de obținere al convertoarelor BUCK, BOOST și BUCK-BOOST cu pulsații nule prin rotirea unei rețele cu trei terminale între sursă și sarcină.

A urmat apoi analiza structurilor de convertoare având curenți cu pulsații nule în conducție discontinuă. Wang, Dunford și Mauch [206] fac acest lucru pentru convertoare BUCK și BOOST iar Lascu [85] pentru convertorul BUCK-BOOST. În sfârșit, Kolar Sree și Zach [70], [71] propun topologii noi de convertoare cu curenți de intrare sau de ieșire fără pulsații la care cuplarea bobinelor nu mai este necesară, plecând de la convertoare cu bobine cuplate.

Prezentul capitol își propune să reia problematica, fără a efectua analiza în mod individual pe convertoare ci utilizând celulele de bază introduse în capitolul 1, regăsindu-se astfel topologiile propuse de autorii sus-menționați dar și alte noi topologii. Este apoi făcută o scurtă analiză comparativă a acestora, iar în final sunt expuse câteva probleme de ultimă oră privind magnetica integrată.

#### 2.1 Principiul obținerii curenților cu pulsații nule

Obținerea unui curent cu pulsații nule presupune o impedanță dinamică infinită sau, dacă se dorește ca pulsațiile să fie nenule dar foarte mici, o impedanță foarte mare pe calea serie prin care acesta se închide, această proprietate trebuind să se mențină la toate frecvențele nenule. În curent continuu desigur că impedanța va avea o valoare finită nenulă.

*Tehnica pe care se bazează obținerea de curenți cu pulsații nule este cea a bobinelor cuplate la bornele cărora se imprimă tensiuni egale ca valoare instantanee.* Se subliniază faptul că, spre deosebire de un transformator clasic cu tensiune imprimată în primar și cu secundar terminat de exemplu pe o sarcină rezistivă, atât tensiunea din primar cât și cea din secundar sunt imprimate. Fie cele două bobine cuplate din Fig. 2.1 pentru care notăm cu  $k \in [0, 1]$  coeficientul de cuplaj magnetic și cu  $L_M$  inductivitatea mutuală.

Se pot scrie următoarele relații:

$$\begin{cases} v_{L_1} = L_1 \frac{di_{L_1}}{dt} + L_M \frac{di_{L_2}}{dt} \\ v_{L_2} = v_{L_1} = L_M \frac{di_{L_1}}{dt} + L_2 \frac{di_{L_2}}{dt} \end{cases} \quad (1)$$

de unde prin rezolvarea sistemului cu necunoscutele  $v_{L_1}$  și  $v_{L_2}$  se obține:

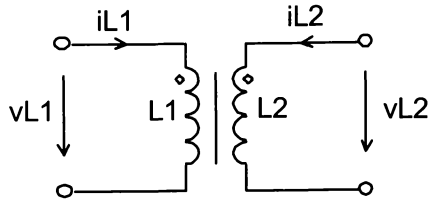


Fig. 2.1. Sensurile curenților și tensiunilor celor două bobine cuplate magnetic examinate.

$$\frac{di_{L_1}}{dt} = \begin{vmatrix} v_{L_1} & L_M \\ v_{L_2} & L_2 \end{vmatrix} \begin{vmatrix} L_1 & L_M \\ L_M & L_2 \end{vmatrix}^{-1} = v_{L_1} \frac{L_2 - L_M}{L_1 L_2 - L_M^2} \quad (2)$$

$$\frac{di_{L_2}}{dt} = \begin{vmatrix} L_1 & v_{L_1} \\ L_M & v_{L_2} \end{vmatrix} \begin{vmatrix} L_1 & L_M \\ L_M & L_2 \end{vmatrix}^{-1} = v_{L_1} \frac{L_1 - L_M}{L_1 L_2 - L_M^2} \quad (3)$$

Examinând relațiile (2) și (3) reiese ușor că pentru a avea curenții  $i_{L_1}$  constant este necesar:

$$L_M = L_2 \quad (4)$$

și simultan să avem

$$L_1 L_2 - L_M^2 \neq 0 \quad (5)$$

Similar, pentru a avea curenții  $i_{L_2}$  constant condițiile sunt:

$$L_M = L_1 \quad (6)$$

simultan cu (5). Deoarece

$$k = \frac{L_M}{\sqrt{L_1 L_2}} \quad 0 \leq k \leq 1 \quad (7)$$

$k = 0$  însemnând bobine necuplate, iar  $k = 1$  bobine cu cuplaj perfect, condițiile (4) și (5) ale unui curent cu pulsații nule prin  $L_1$  au și forma echivalentă;

$$k = \sqrt{\frac{L_2}{L_1}} \quad (8)$$

$$k \neq 1 \quad (9)$$

Deoarece  $k$  este subunitar (8) și (9) au loc simultan numai dacă  $L_1 > L_2$ . Analog se arată că o formă echivalentă a relațiilor (6) și (7) necesare pentru a avea curent cu pulsații nule prin  $L_2$  este dată de relațiile:

$$k = \sqrt{\frac{L_1}{L_2}} \quad (10)$$

$$k \neq 1 \quad (11)$$

și este necesar ca  $L_2 > L_1$ .

Se cunoaște însă că valoarea inductivității unei bobine  $L$  având  $n$  spire este dată de relația

$$L = A_L n^2 \quad (12)$$

unde  $A_L$  este o constantă ce depinde numai de geometria miezului și caracteristicile sale magnetice. Folosind (12), (8) și (10) devin:

$$k = \frac{n_2}{n_1} \quad (13)$$

respectiv

$$k = \frac{n_1}{n_2} \quad (14)$$

ceea ce în cuvinte se enunță că pentru a avea curent cu pulsații nule printr-o bobină coeficientul de cuplaj magnetic trebuie să fie egal cu raportul de transformare al înfășurărilor cuplate, având la numitor înfășurarea prin care se dorește curent constant. Se constată de asemenea că nu se poate obține curent constant decât prin înfășurarea cu inductivitatea cea mai mare.

O problemă practică deosebit de stringentă care apare este *cum să se realizeze practic* relațiile (8), (10), (13) sau (14). O primă soluție este dată în [167] în care se propune utilizarea unei *inductivități suplimentare*, externe, ajustabile, în serie cu cealaltă înfășurare decât cea prin care se dorește curent fără pulsații. De exemplu fie inductivitatea externă,  $L_{ext}$ , plasată așa cum se arată în Fig. 2.2.

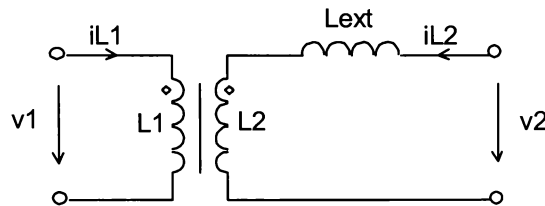


Fig. 2.2. Inserarea unei inductivități externe pentru realizarea curenților cu pulsații nule.

În aceeași ipoteză a egalității tensiunilor aplicate la cele două porturi,  $v_1 = v_2$ , scriind expresiile acestor tensiuni și rezolvând sistemul ce rezultă având ca necunoscute pe  $\frac{di_{L_1}}{dt}$  și  $\frac{di_{L_2}}{dt}$  se obține:

$$\frac{di_{L_1}}{dt} = v_1 \frac{L_2 - L_M + L_{ext}}{L_1 L_2 - L_M^2 + L_1 L_{ext}} \quad (15)$$

$$\frac{di_{L_2}}{dt} = v_1 \frac{L_1 - L_M}{L_1 L_2 - L_M^2 + L_1 L_{ext}} \quad (16)$$

Examinând relațiile (15) și (16) este evident că modificând pe  $L_{ext}$  se poate anula numai derivata primului curent. Acest lucru se realizează dacă

$$L_{ext} = L_M - L_2 \quad (17)$$

Pentru ca  $L_{ext}$  să fie realizabilă fizic este desigur necesar ca  $L_{ext} > 0$ , ceea ce atrage cu sine

$$L_M > L_2 \quad (18)$$

Se poate arăta fără dificultate că aceasta implică obligatoriu

$$L_1 > L_2 \quad (19)$$

Deci inserarea unei inductivități suplimentare are efect numai dacă bobina prin care se dorește curent fără pulsații este cea mai mare, adică aceeași condiție pe care am dedus-o anterior.

Pentru a obține curent constant prin a doua înfășurare  $L_{ext}$  se plasează în serie cu  $L_1$ , condițiile obținându-se permutând indicii "1" și "2" între ei, adică:

$$L_{ext} = L_M - L_1 \quad (20)$$

$$L_2 > L_1 \quad (21)$$

Procedul care se practică frecvent este cuplarea cât mai strânsă a bobinelor  $L_1$  și  $L_2$  pe miez. În acest fel

$$k \cong 1 \quad (22)$$

$$L_M \cong \sqrt{L_1 L_2} \quad (23)$$

și condiția pentru curent constant prin  $L_1$  de exemplu, devine:

$$L_{ext} = L_M - L_2 \cong \sqrt{L_1 L_2} - L_2 = L_2 \left( \sqrt{\frac{L_1}{L_2}} - 1 \right) = L_2 \left( \sqrt{\frac{n_1}{n_2}} - 1 \right) \quad (24)$$

Această metodă se preferă pentru că valoarea necesară pentru  $L_{ext}$  se poate determina rapid, prin măsurarea fiecărei înfășurări având cealaltă înfășurare în gol (deci două măsurători) sau printr-o singură măsurătoare (a lui  $L_2$ ) dacă se cunoaște câte spire s-au bobinat pentru fiecare înfășurare. Bineînțeles că se poate determina și valoarea necesară pentru  $L_{ext}$  atunci când cuplajul nu este perfect dar acest lucru reclamă mai multe măsurători asupra celor două bobine cuplate pentru a determina inductivitatea mutuală.

Dezavantajul metodei utilizării unei inductivități externe este existența unui dispozitiv magnetic suplimentar.

Altă posibilitate de a realiza practic condiția de pulsații nule este ajustarea coeficientului de cuplaj mutual  $k$ , prin soluții constructive mai sofisticate ce vizează modificarea geometriei miezului. Soluția are avantajul că păstrează un singur dispozitiv magnetic, dar complicațiile de fabricație sunt apreciable.

Fie acum aceleași bobine cuplate având însă o pereche de capete de același fel legate într-un punct comun, ca în Fig. 2.3a. Relațiile (1) se mai pot scrie:

$$\begin{cases} v_{L_1} = (L_1 - L_M) \frac{di_{L_1}}{dt} + L_M \left( \frac{di_{L_1}}{dt} + \frac{di_{L_2}}{dt} \right) \\ v_{L_2} = (L_2 - L_M) \frac{di_{L_2}}{dt} + L_M \left( \frac{di_{L_1}}{dt} + \frac{di_{L_2}}{dt} \right) \end{cases} \quad (25)$$

relații ce sugerează schema echivalentă în T din Fig. 2.3b.

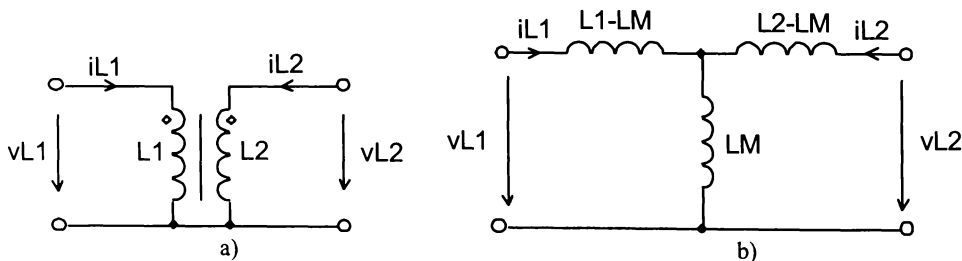


Fig. 2.3. Echivalența a două bobine cuplate cu punct comun cu un circuit în T cu bobine necuplate.

Se accentuează faptul că circuitul echivalent din Fig. 2.3b înlocuiește 2 bobine cuplate cu 3 bobine necuplate. Ordinul sistemului (dat de numărul elementelor capabile să înmagazineze energie) rămâne însă nemodificat (ordinul 2) pentru că prin  $L_M$  curge suma variabilelor de stare  $i_{L_1} + i_{L_2}$ , deci  $L_M$  nu reprezintă un element independent de stocare a energiei.

**Observație.** Evident relațiile (25) sunt formal identice cu relațiile (1) și fără ca bobinele să aibă vreun punct comun și fără ca tensiunile pe ele să fie egale. Aceasta în ceea ce privește relațiile. În ceea ce privește configurația circuitelor echivalente este clar că atunci când se materializează



relațiile (25) prin circuitul din Fig. 2.3b se pierde izolarea galvanică. Pierderea izolării este inevitabilă în această materializare. De aceea, pentru ca pe lângă echivalența relațiilor să avem și echivalența circuitelor a fost necesară precizarea că cele două înfășurări au un punct comun.

Dacă în ipoteza că tensiunile  $v_{L1}$  și  $v_{L2}$  sunt egale se aplică relațiile (4) și (6) pentru a avea curenții  $i_{L1}$  respectiv  $i_{L2}$  constanți circuitele devin cele din Fig. 2.4.

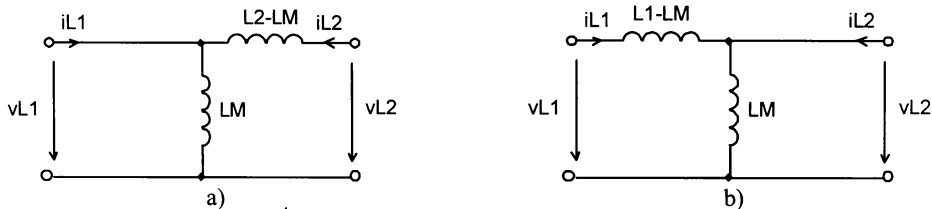


Fig. 2.4. Configurațiile circuitului în T pentru curenții  $i_{L1}$  fără pulsații (a) și curenții  $i_{L2}$  fără pulsații (b).

Aceste circuite se pot intui și fizic, deoarece pentru ca  $i_{L1}$  să aibă pulsații nule tensiunea aplicată pe inductivitatea  $L_1 - L_M$ , finită, trebuie nulă. Cum la intrare tensiunea este  $v_{L1}$  trebuie ca la borna din dreapta a inductivității  $L_1 - L_M$  să regăsim tot tensiunea  $v_{L1}$ . Cum  $v_{L1} = v_{L2}$  acest lucru este posibil numai dacă inductivitatea  $L_2 - L_M$  este nulă. Un raționament similar și pentru a avea curenții  $i_{L2}$  constant.

**Observație.** Se pune în mod firesc întrebarea: se pot obține *simultan* curenții  $i_{L1}$  și  $i_{L2}$  fără pulsații? Răspunsul este *negativ* și argumentarea este simplă. Dacă atât  $i_{L1}$  cât și  $i_{L2}$  au pulsații nule atunci relațiile

$$\begin{cases} L_2 = L_M \\ L_1 = L_M \\ L_1 L_2 - L_M^2 \neq 0 \end{cases} \quad (26)$$

trebuie să aibă loc simultan, ceea ce evident este imposibil deoarece înmulțind primele două relații se obține a treia relație negată. Obținerea simultană a celor doi curenți fără pulsații este deci posibilă numai dacă circuitul este modificat. Acest lucru este realizabil, așa cum a fost publicat în [29], [31], prin inserarea a încă unui transformator.

**Remarcă importantă.** Obținerea unuia sau altuia dintre curenții  $i_{L1}$  și  $i_{L2}$  fără pulsații nu este legată de o anumită formă de undă pentru  $v_{L1}$  și/sau  $v_{L2}$  ci singura condiție este egalitatea acestor tensiuni. Acest lucru constituie argumentul pentru care convertoarele vor funcționa cu curenți fără pulsații și în conducție discontinuă.

## 2.2 Obținerea convertoarelor dc-dc cu pulsații nule folosind generarea din celule de bază

Ideea pe care o prezentăm în continuare este aceea de a nu examina individual fiecare convertor și de a căuta modificarea structurii pentru a obține curenții doriți cu pulsații nule, ci de a analiza și modifica convenabil celulele de bază care generează familii de convertoare, în acest fel

convertoarele cu pulsații nule obținându-se prin simpla metodă de sinteză prezentată în capitolul 1. Se acoperă în acest fel toate clasele posibile de convertoare și dacă se poate utiliza echivalența cu structura în T pentru bobine prezentată în paragraful 2.1, se pot obține structuri interesante de noi convertoare.

#### *A. Celula generatoare a convertoarelor BUCK, BOOST și BUCK-BOOST.*

Evident această celulă conține o singură bobină, deci principiul de obținere a curenților cu pulsații nule enunțat în paragraful 2.1 nu se poate aplica decât dacă în celulă se mai introduce o bobină. În paragraful 1.6.3. s-a arătat cum se poate introduce această bobină și a fost demonstrată egalitatea tensiunilor pe bobine în ipoteza unor capacități suficient de mari pentru ca pulsațiile tensiunii pe ele să fie neglijabile față de componenta continuă. Câte 2 variante pentru fiecare tip de convertor au fost prezentate în Fig. 1.72, 1.73 și 1.74. Deci la toate aceste 6 convertoare cuplarea bobinelor și alegerea coeficientului de cuplaj conform relațiilor (13) sau (14) asigură curentul de ieșire cu pulsații nule pentru convertoarele BUCK, curent de intrare cu pulsații nule pentru convertoarele BOOST și BUCK-BOOST.

În acest punct trebuie făcută o precizare importantă. În paragraful 1.6.3 egalitatea tensiunilor pe bobine s-a demonstrat în ipoteza unor capacități suficient de mari pentru ca pulsațiile tensiunii pe ele să fie neglijabile față de componenta continuă. De fapt numai egalitatea componentelor continue este adevărată, iar în practică nici măcar aceasta pentru că bobinele prezintă rezistențe serie nenule pe care apar căderi suplimentare de tensiune. Chiar și în ipoteza unor bobine ideale egalitatea tensiunilor pe ele ar cere capacități infinite. Deci prezența pulsațiilor nenule pe capacități și a rezistențelor parazite din circuit face ca egalitatea tensiunilor instantanee pe bobine să fie numai aproximativă. În consecință și curenții nu vor fi eminentamente constanți ci vor avea pulsații mici, dar nenule. Aceeași situație se va regăsi la toate convertoarele cu curenți cu pulsații nule. De aceea unii autori denumesc aceste convertoare drept convertoare cu pulsații nule punând cuvântul “nule” între ghilimele (“ ‘zero’ current ripple ”).

O parte de topologii au fost raportate în articolele amintite în introducere [6], [206], [207], [85] altele sunt topologii noi diferind numai prin punctul în care se conectează a doua bornă a bobinei  $L_1$ , de aceea nu le vom atribui o valoare și o importanță deosebită.

Vom insista însă asupra modului în care se modifică aceste convertoare dacă se aplică transformarea în T tratată la pct. 2.1. Pentru aceasta primul pas este modificarea celulei de bază astfel încât cele două bobine componente să aibă o pereche de începuturi legate într-un punct comun. Luând ca exemplu celula de bază din Fig. 1.71b, reprodusă în Fig. 2.5a cu începuturile marcate, aplicarea transformării în T este acum posibilă, celula evoluând ca în Fig. 2.5b. Cum terminalul  $L$  al celei este cel ce se leagă la sursă sau la sarcină desigur dorim ca prin acest terminal curentul să fie fără pulsații. În consecință este necesar ca  $M = L_1$ , celula devenind cea din fig. 2.6. Comparând această celulă cu celula de bază generatoare a convertoarelor BUCK, BOOST, BUCK-BOOST clasice din Fig. 1.43b se observă că ea conține suplimentar filtrul  $(L - L_1) - C_1$ . Este clar acum că structurile care se obțin din această celulă vor fi variante ale convertoarelor clasice având la intrare (pentru BOOST), la ieșire (pentru BUCK) sau intern (BUCK-BOOST) un filtru suplimentar.

Se pune în mod normal problema care dintre topologii sunt mai performante: cele obținute cu celula din Fig. 2.5a sau cele obținute cu celula din Fig. 2.6, în condițiile în care ambele asigură curenți cu pulsații nule prin terminalul  $L$  ? Răspunsul nu este unul tranșant în favoarea uneia sau alteia dintre celule, fiecare având avantaje și dezavantaje după cum urmează.

Avantajele celulei din Fig. 2.5a și implicit ale convertoarelor pe care le generează sunt:

- utilizarea unui singur miez magnetic pe care se realizează ambele bobine.

- bobina  $L_1$  fiind în serie cu capacitatea  $C_1$  este parcursă de un curent fără componentă continuă, spre deosebire de aceeași bobină din Fig. 2.6 care va prelua și componente continue. Deci miezul va fi de dimensiuni mai mici și pierderile ohmice în înfășurare mai mici. La pierderi egale se va utiliza o sârmă de secțiune mai mică. [207].

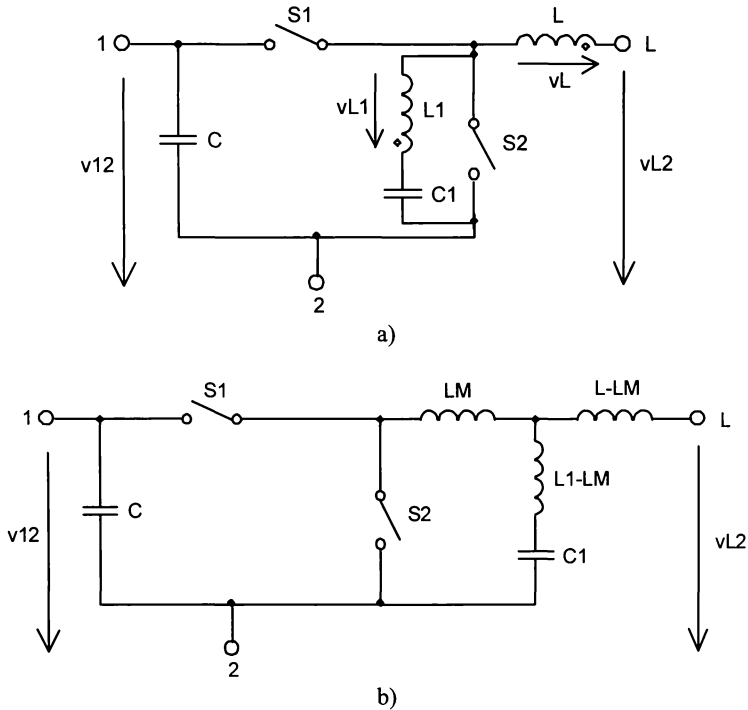


Fig. 2.5. Transformarea celulei având bobinele cuplate prin echivalarea acestora cu circuitul în T.

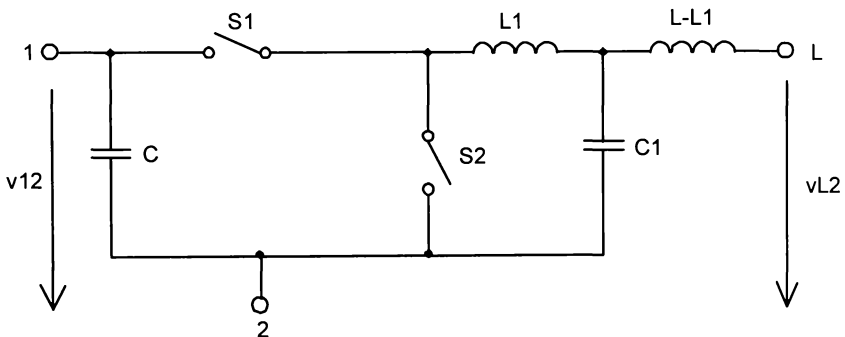


Fig. 2.6. În condiții de curent cu pulsații nule prin borma  $L$  celula este echivalentă cu celula originală cu bobine necuplate la care se adaugă un filtru de ordinul 2 în borma  $L$ .

Dezavantajele celulei din Fig. 2.5a comparativ cu cea din Fig. 2.6 sunt:

- este necesară asigurarea unei reproductibilități a cuplajului dintre bobine, lucru ce se poate realiza, așa cum s-a văzut, prin utilizarea unei bobine suplimentare pentru ajustare în serie cu  $L_1$ . Dar în acest caz se pierde avantajul utilizării unui singur miez magnetic. O altă posibilitate este ajustarea coeficientului de cuplaj  $k$  prin procedee mecanice și de fabricație a miezului, unde apare “problema numărului întreg” [165] care se poate evita numai cu o geometrie specială a miezului.
- cuplarea bobinelor pe același miez poate duce și la capacități parazite și deci o supresie mai puțin eficientă a fenomenelor la înaltă frecvență. Acest lucru cere deci un bobinaj special și îngrijit.

În concluzie, decizia alegerii uneia sau alteia din variante trebuie să se facă din punctul de vedere al fabricației, corelată cu cerințele de gabarit și compatibilitate electromagnetică.

### B. Celula generatoare a convertoarelor ĆUK, și a noilor variante de convertoare BUCK și BOOST

Din capitolul 1, paragraful 1.6.2, Fig. 1.52 celula de bază pentru aceste convertoare este cea din Fig. 2.7. Față de celula analizată anterior aceasta conține din start două inductanțe. Se poate arăta că dacă se aleg sensurile tensiunilor pe bobine ca în figură atunci tensiunea pe capacitatea  $C$  (presupusă suficient de mare) este  $V_{L_1X} + V_{XL_2}$  unde  $V_{L_1X}$  și  $V_{XL_2}$  sunt tensiuni cvasiconstante fiind combinații liniare dintre tensiunea de intrare și de ieșire. Ținând cont de acest fapt se demonstrează imediat că tensiunile pe bobine pe duratele celor două stări topologice sunt  $V_{L_1X}$  respectiv  $V_{L_2X}$ , deci egale. Rezultă deci că bobinele se pot cupla, alegând începuturile așa cum s-au marcat în figură. Pentru a putea utiliza configurația în T pentru bobine celula trebuie modificată astfel încât bobinele să aibă o pereche de capete de același fel comune. Două moduri posibile de realizare se dau în Fig. 2.8a și în Fig. 2.8b. Se remarcă imediat că cele două celule sunt de fapt una și aceeași structură, una fiind derivată din cealaltă prin oglindire. De aceea se va studia numai una dintre ele și anume cea din Fig. 2.8b. Efectuând transformarea în T a bobinelor pe această celulă se obține structura din Fig. 2.9. Impunând curenți cu pulsații nule prin bornele  $L_1$  și  $L_2$  schemele celulelor corespunzătoare devin cele din Fig. 2.10a respectiv 2.10b

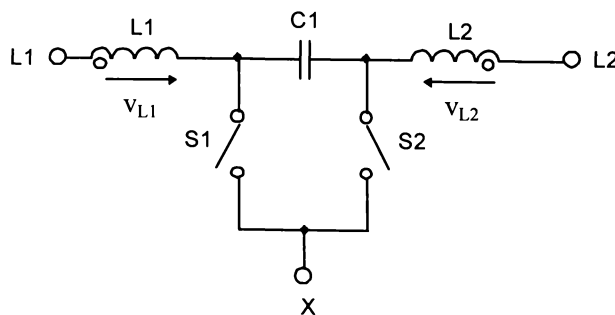


Fig. 2.7. Celula de bază pentru familia ĆUK și modul de cuplare a bobinelor pentru curenți fără pulsații prin  $L_1$  sau  $L_2$ .

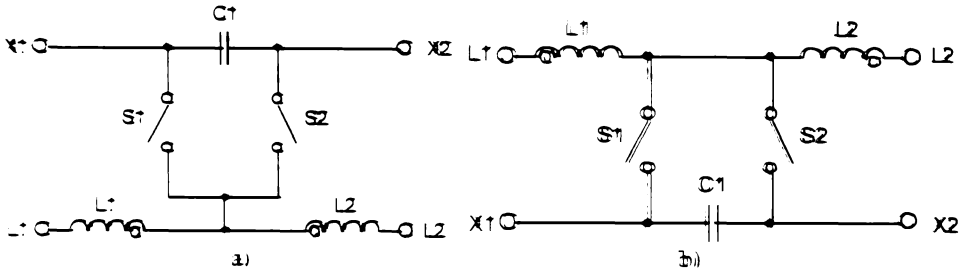


Fig. 2.3. Două modificări ale celulei CUK pentru a obține bobinele conectate între ele.

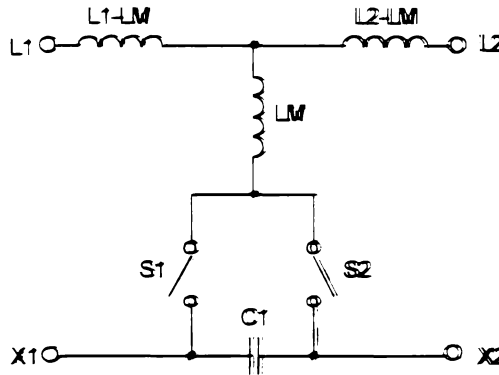


Fig. 2.9. Celula de bază cu echivalarea în T a bobinelor cuplate.

Din nou se distinge faptul că cele două scheme din Fig. 2.10 sunt echivalente structural, fiind simetrice ca și topologie după o dreaptă verticală ce trece prin  $C_1$ . De aceea este suficient să alegem pentru studiu una dintre ele, fie ea cea din Fig. 2.10a. Această structură se poate la rândul său transforma în celula din Fig. 2.11, prin mutarea bobinei  $L_1 - L_2$  pe ramura serie externă bornelor  $L_1$  și  $X_1$  (în exteriorul celulei între  $L_1$  și  $X_1$  nu se conectează decât un element) și apoi prin oglindire.

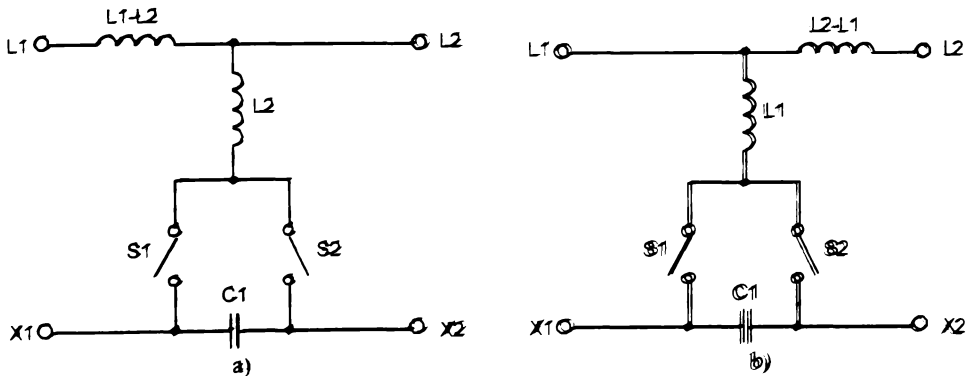


Fig. 2.10. Structura celulei cu bobine necuplate care asigură curent fără pulsații prin borna  $L_1$  (a) și  $L_2$  (b).

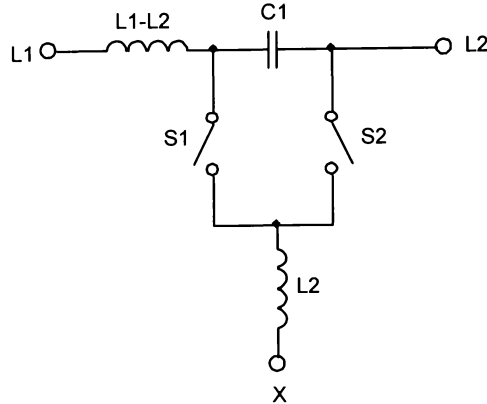


Fig. 2.11. Rearanjarea celulei din Fig. 2.10a pentru a obține o celulă cu 3 terminale.

Transformarea este deosebit de utilă pentru că furnizează o celulă cu 3 terminale. Să remarcăm că celula din Fig. 2.11 este identică cu cea prezentă în convertoarele propuse de Capel în [14] în mod independent și prezentată în capitolul 1, Fig. 1.75, unde este desenată rotit cu  $90^{\circ}$ . Plecând de la această celulă și folosind procedura de sinteză expusă în capitolul 1 se pot genera toate structurile de convertoare posibile, acestea moștenind proprietățile celei, în particular aceea că prin inductivitatea  $L_1 - L_2$  curentul este fără pulsații. Pentru că celula nu prezintă simetrie vor rezulta prin generare 6 topologii pentru borna minus a sursei comune cu sarcina. Din punct de vedere practic, al ușurinței filtrajului, gabariturii și perturbațiilor electromagnetice dorim fie curent de intrare în convertor fie curent de ieșire din convertor fără pulsații. Cum acest curent este cel care intră în borna  $L_1$  a celulei rezultă că numai topologiile la care borna  $L_1$  este legată la sursă sau la sarcină sunt de interes. Rămân astfel de analizat din cele 6 topologii numai 4. Cele 4 structuri de convertoare rezultate sunt cele din Fig. 2.12.

Să subliniem faptul că se obțin două noi topologii de convertoare ĆUK cu bobine necuplate care asigură curent fără pulsații fie la intrare - Fig. 2.12a - fie la ieșire - Fig. 2.12b. Aceste două topologii ĆUK prezintă o serie de avantaje, cum se va vedea mai jos. În celelalte două structuri se regăsesc convertoarele BUCK și BOOST cu filtre suplimentare  $L - C$  la intrare respectiv ieșire. Dacă se analizează individual noile topologii BUCK și BOOST obținute prin sinteză în capitolul 1 din Fig. 1.54 și Fig. 1.55 și se aplică transformarea în T a bobinelor (după mutarea convenabilă în circuite a acestora) se obțin exact aceleași circuite din Fig. 2.12c și 2.12d. Evident ele prezintă un interes scăzut din punct de vedere practic.

De precizat că în [70] și [71] se efectuează o analiză individualizată pe convertoare obținându-se o serie de topologii denumite de autori ca noi. Ele însă reprezintă funcțional câte una din structurile din Fig. 2.12. Nu este greu de văzut că oricare din ele se poate obține dintr-unul din circuitele din Fig. 2.12 efectuând ușoare modificări (deplasând bobine în alte poziții pe cale serie, etc.). Acest fapt subliniază o dată în plus superioritatea, prin gradul de generalitate pe care îl are, a studiului pe celule de bază generatoare de familii de convertoare, comparativ cu studiul individual pe convertor.

Cele două noi topologii de convertoare ĆUK cu curenți de intrare sau ieșire fără pulsații și fără bobine cuplate au o serie de avantaje și dezavantaje după cum urmează:

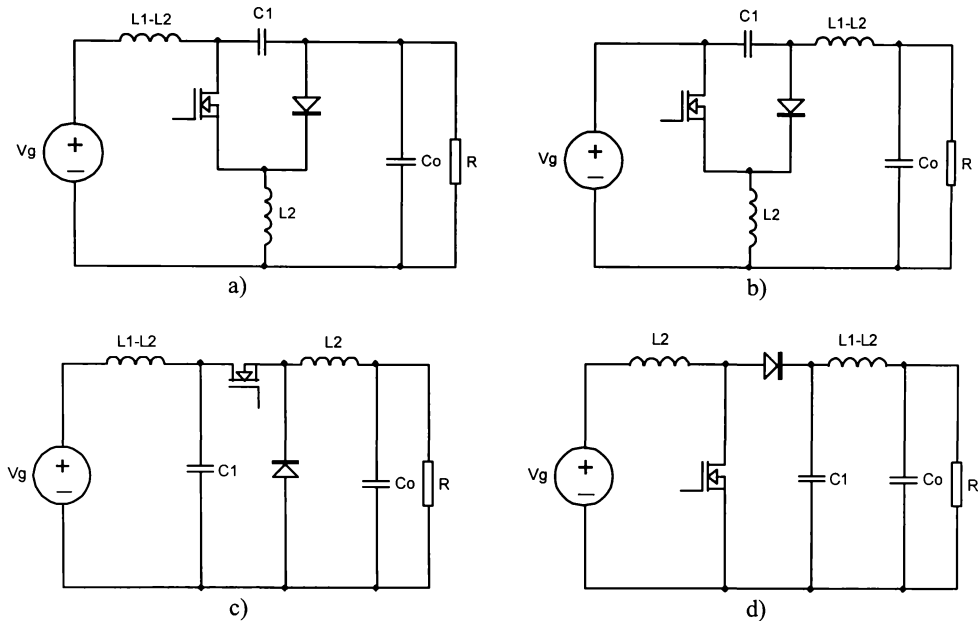


Fig. 2.12. Converter  $\dot{C}UK$  cu curent de intrare fără pulsații (a), converter  $\dot{C}UK$  cu curent de ieșire fără pulsații (b), converter BUCK cu curent de ieșire fără pulsații (c), converter BOOST cu curent de intrare fără pulsații (d).

Avantaje:

- comparativ cu convertorul  $\dot{C}UK$  convențional corespunzător, ce utilizează tot bobine necuplate, circuitele au pulsații mult mai reduse, la complexități (ca număr de elemente) de circuit egale. Deci efortul de filtrare va fi mai mic. Mai mult, inductivitățile acestor convertoare sunt mai mici decât cele din convertoarele convenționale ( $L_1 - L_2$  față de  $L_1$  pentru curent de intrare fără pulsații și  $L_2 - L_1$  față de  $L_2$  pentru curent de ieșire fără pulsații) deci dispozitivele magnetice vor fi mai reduse.
- în comparație cu convertorul  $\dot{C}UK$  convențional cu bobine cuplate noile structuri au realizabilitate mai mare, întrucât nu se pune problema ajustării cuplajului sau al vreunui element.
- în primă aproximație circuitele au densități de putere egale cu convertorul  $\dot{C}UK$  convențional cu bobine cuplate.

Dezavantaje:

- tranzistoarele sunt flotante față de punctul comun al sursei și sarcinii, ceea ce pune unele probleme pentru circuitul de comandă.
- nu există posibilitatea ca prin modificări simple ale circuitelor să se obțină curenți cu pulsații nule simultan la intrare și la ieșire, așa cum este posibil în cazul convențional [29].

C. Celula generatoare a convertoarelor ZETA, SEPIC, WATKINS-JOHNSON, WATKINS-JOHNSON INVERS, și a noilor variante de convertoare WATKINS-JOHNSON și WATKINS-JOHNSON INVERS.

Celula, prescurtată ca și nume ZSWJ, este cea din Fig. 1.64. Efectuând transformări ale celulei se poate arăta că există posibilitatea de a obține convertoare ZETA și SEPIC cu pulsații nule la ieșire respectiv intrare folosind două bobine necuplate. Cele două convertoare sunt cele din Fig. 2.13.

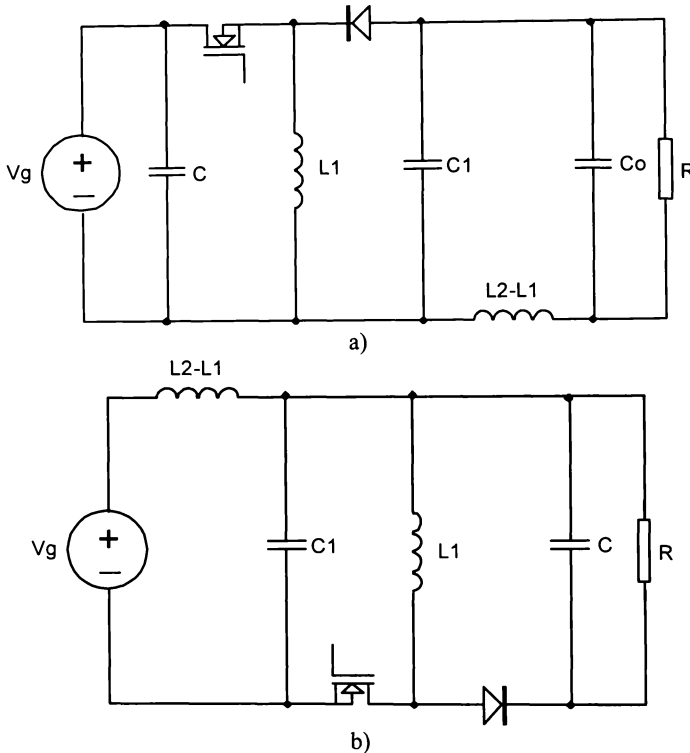


Fig. 2.13. Converter ZETA (a) și converter SEPIC (b) cu bobine necuplate.

Este interesant faptul că ambele convertoare sunt echivalente cu un convertor BUCK-BOOST cu un filtru suplimentar la ieșire pentru ZETA, respectiv un filtru suplimentar la intrare pentru SEPIC. Deci practic ele nu reprezintă topologii noi și în această formă prezintă mai puțin interes. Totuși, dacă bobinele sunt cuplate, avantajul față de topologiile clasice fără bobine cuplate este mare, obținându-se pe lângă curenți cu pulsații nule și utilizarea unui singur dispozitiv magnetic.

În privința celor 4 convertoare Watkins-Johnson, Watkins-Johnson invers și a variantelor lor - Fig. 1.67, Fig. 1.68, Fig. 1.69 respectiv Fig. 1.70 - se arată ușor că tensiunile pe cele două bobine sunt egale. Este însă imposibilă rearanjarea topologiei celulei astfel încât cele două bobine să aibă o pereche de capete comună. Deci fără modificări de substanță ale celulei (adăugarea de noi elemente) este imposibil de obținut curenți cu pulsații nule cu bobine necuplate. Totuși cuplarea bobinelor asigură pulsații nule pentru curentul de intrare la varianta de convertor WATKINS-JOHNSON INVERS și pentru curentul de ieșire la varianta de convertor WATKINS-JOHNSON.



D. Celulele generatoare ale convertoarelor pătratice

O analiză a celulelor relevă imediat faptul că deși ele conțin câte două bobine tensiunile pe acestea nu sunt egale. Vor trebui deci introduse în celule o bobină suplimentară și o capacitate în serie, conectate adecvat, în funcție de ce curent se vizează (dacă este posibil, bineînțeles) să fie fără pulsații.

Din fericire modul de inserare a acestor elemente este analog cu cel practicat în cazul celulei generatoare a convertoarelor BUCK, BOOST și BUCK-BOOST, pentru că se pot identifica astfel de subcelule în celulele convertoarelor pătratice. În Fig. 2.14 se arată pentru exemplificare cum se poate modifica în acest sens celula  $\delta_1$  inserând grupul  $L_c - C_c$  și, ca aplicație, după efectuarea tuturor transformărilor în Fig. 2.15 se prezintă un convertor  $\delta_1$ -BOOST cu curent de intrare fără pulsații și bobine necuplate. Se observă din nou echivalența cu un convertor  $\delta_1$ -BOOST clasic cu filtru la intrare.

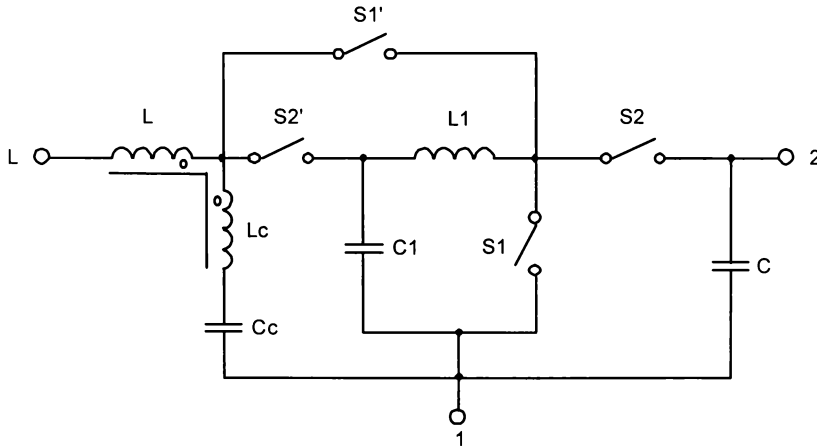


Fig. 2.14. Celula  $\delta_1$  pentru curent cu pulsații nule prin bobina  $L$

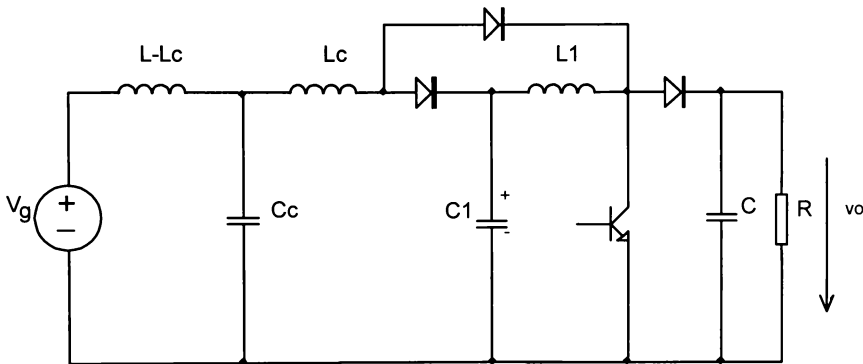


Fig. 2.15. Convertor  $\delta_1$ -BOOST cu curent de intrare fără pulsații și bobine necuplate.

O ultimă remarcă se referă la dinamica acestor convertoare. Cuplarea bobinelor și prezența capacităților din schemă duc la formarea unor circuite oscilante slab amortizate. Acest lucru se observă de exemplu clar în Fig. 2.13b unde filtrul de la intrare are amortizare redusă. De aceea timpul de răspuns al acestor sisteme va fi lung, iar regimul tranzitoriu se va caracteriza prin curenți și tensiuni oscilatorii cu amortizare mică.

### 2.3 Noi aspecte și soluții privind magnetica integrată

Tendința actuală în sistemele moderne de telecomunicații este înlocuirea surselor de alimentare centralizate, cu gabarite mari, cu mai multe module de putere distribuite. Modelele sunt utilizate ca surse de alimentare pe punct utilizator, prescurtat PUPS (Point of Use Power Supply), termen introdus de firma Nortel. Acest mod de alimentare are o serie de avantaje. În loc să se distribuie putere de 5V și 3,3V de la 100A până la 200A, magistrala va distribui aceeași putere la 48V și 10A până la 20A spre modulele individuale. Modulele de convertoare plasate direct pe modulele funcționale vor procesa puterea de la tensiunea nominală nestabilizată a magistralei de 48V (tipic între 36V și 72V) coborând-o la nivelele dorite de 5V sau 3,3V și curenți de sarcină de 10A până la 20A.

Modulele funcționale sunt plasate în rack-uri distanțate însă la 0,6". Aceasta impune deci ca profilele modulelor (și în particular ale celor de alimentare) să fie sub 0,5 inch. Se restricționează deci în primul rând proiectarea dispozitivelor magnetice și a radiatoarelor. Soluția este utilizarea de miezuri plate și de convertoare de randament foarte ridicat, această din urmă cerință fiind impusă de suprafața disponibilă mică pentru evacuarea căldurii. Convertoare de randament ridicat se obțin folosind diverse tehnici de comutare nedisipativă (soft-switching). Problema de bază rămâne aceea a proiectării dispozitivelor magnetice.

De departe cel mai utilizat convertor în aceste module este convertorul ĆUK [30] [31], datorită multiplelor proprietăți pe care le are și care au devenit deja clasice în literatură. Pentru a avea tensiuni de ieșire de calitate se utilizează varianta de convertor cu bobine cuplate și curent fără pulsații la ieșire. Se reamintește condiția pentru pulsații nule în curentul de ieșire:

$$k = \frac{n_1}{n_2} \quad (27)$$

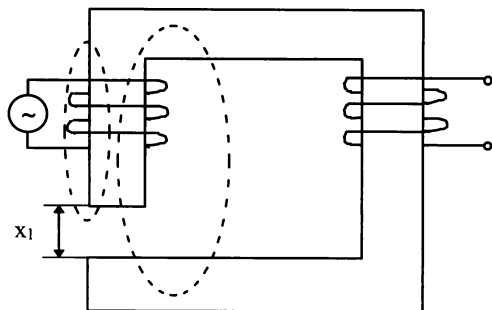
$$L_2 \geq L_1 \Leftrightarrow n_2 \geq n_1 \quad (28)$$

Așa cum s-a văzut, convertorul trebuie să fie unul coborâtor de tensiune, de la 48V nominal la 5V și 3,3V, însă cu curent de ieșire mare, 10A-20A. Acest curent de ieșire mare la o tensiune de ieșire mică cere ca rezistența înfășurării bobinei de la ieșire să fie cât mai mică. Deci această bobină va avea un număr  $n_2$  mic de spire, practic 1, 2, maxim 3 spire. Conform (28) rezultă că și  $n_1$  va fi mic, fiind mai mic decât  $n_2$ . De exemplu dacă  $n_2 = 3$   $n_1$  nu poate fi decât 1, 2 sau 3, pentru că este un număr natural. În consecință  $k$  va putea lua un număr finit de valori și anume:  $\frac{1}{3}, \frac{2}{3}, 1$ .

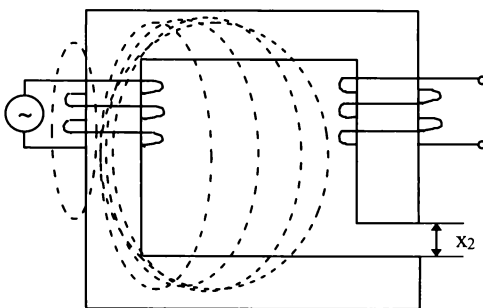
Condiția (27) se poate realiza fie modificând raportul  $\frac{n_1}{n_2}$ , fie modificând  $k$ . Evident prima soluție este mult mai comodă, însă în situația concretă prezentată acest lucru este imposibil, întrucât raportul  $\frac{n_1}{n_2}$  nu ia decât câteva valori distincte. Deci va trebui ajustat coeficientul de cuplaj mutual

la raport de transformare fix. Acest lucru se obține ajustând întrefierul miezului.

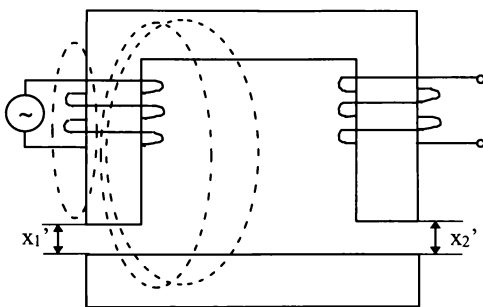
Slobodan Ćuk [28] a prezentat în premieră în 1997 o metodă prin care se garantează curent cu pulsații nule la ieșire *fără a se mai ajusta întrefierul*. Pentru aceasta el a propus *un nou model pentru două bobine cuplate pe același miez*, studiind circuitele din Fig. 2.16. Experimentul a arătat că proprietățile structurii magnetice cu bobine cuplate depind decisiv de *poziția întrefierului*.



$$a) \begin{cases} k = 0,98 \\ L_1 = 6,42 \text{ mH} \\ z_{gomot} = 0,12 \text{ V} \end{cases}$$



$$b) \begin{cases} k = 0,57 \\ L_1 = 11,03 \text{ mH} \\ z_{gomot} = 1,3 \text{ V} \end{cases} \quad x_2 = x_1$$



$$c) \begin{cases} k = 0,72 \\ L_1 = 7,13 \text{ mH} \\ z_{gomot} = 0,85 \text{ V} \end{cases} \quad x_1' + x_2' = x_1$$

Fig. 2.16. Cele trei experimente ale prof. Ćuk pentru elaborarea noului model magnetic a două bobine cuplate.

În toate experimentele numărul de spire ale celor două înfășurări este egal. Înfășurarea 1 este comandată cu o tensiune sinusoidală, în timp ce înfășurarea 2 este lăsată în gol. S-au măsurat în cele 3 cazuri coeficientul de cuplaj magnetic  $k$ , inductivitatea  $L_1$  văzută dinspre prima înfășurare și zgomotul radiat la distanța fixă de 1 cm de miez (acesta din urmă important atunci când sursa de alimentare este în apropierea unui echipament sensibil la zgomot, de exemplu un dispozitiv de stocare magnetică). În toate experimentele întrefierul total este același.

Până acum se considera că, întrefierul total fiind același, reluctanța magnetică  $R_x$  și circuitul magnetic echivalent sunt aceleași pentru toate trei circuitele din Fig. 2.16. Este vorba de circuitul din Fig. 2.17a, care, transformat prin dualitate, furnizează cunoscutul model al transformatorului din Fig. 2.17b.

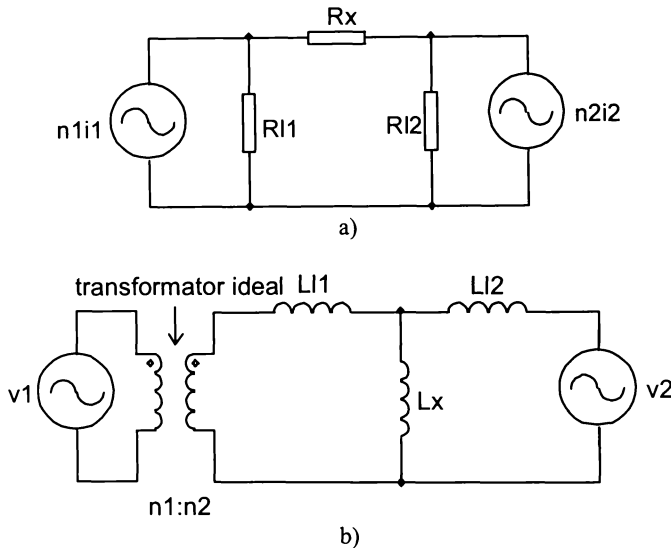


Fig. 2.17. Modelul clasic al circuitului magnetic cu două bobine și întrefier (a) și modelul clasic al transformatorului (b).

Măsurătorile au arătat însă că circuitele nu au aceeași comportare. De exemplu coeficientul de cuplaj nu este același, așa cum prezice schema din Fig. 2.17a, ci se modifică de la 0,98 în Fig. 2.16a la 0,57 în Fig. 2.16b și la fel și celelalte mărimi anexate figurilor. Explicația dată de prof. Cuk este aceea că în cazul Fig. 2.16a întrefierul este aproape de înfășurarea 1 care produce excitația. Partea superioară a miezului U și miezul I sunt la același potențial magnetic pentru că fluxul de scăpări (de pierderi) în aer este mic și cea mai mare parte a fluxului produs se închide și prin a doua înfășurare. În situația din Fig. 2.16b întrefierul este departe de înfășurarea de excitație. Potențialele magnetice ale miezurilor U și I vor fi substanțial diferite existând un flux de scăpări în aer semnificativ și deci cuplajul este mai slab. Fig. 2.16c este o situație intermediară existând întrefier și aproape și departe de înfășurare. Valorile diferite pentru inductivitatea  $L_1$  au o explicație asemănătoare. Așa cum am văzut, fluxul de scăpări este mai mare în cazul Fig. 2.16b. Acest flux contribuie la valoarea totală a inductivității  $L_1$ , mărind-o. Zgomotul radiat este dat de fluxul de scăpări, fiind desigur mai mare în cazul Fig. 2.16b.

Apare deci necesitatea unei scheme echivalente. Se propune pentru aceasta [28] separarea fluxului dintr-o înfășurare în două componente: fluxul de scăpări care se închide în aer înainte de a intersecta întrefierul și fluxul de scăpări care se închide în aer după ce se intersectează întrefierul, așa cum este arătat în Fig. 2.18a. Corespunzător acestor fluxuri apar permeanțe (reluctanțe) în noul

model al circuitului magnetic, așa încât acesta va arăta ca în Fig. 2.18b. Permeanțele  $P_{x_1}$  și  $P_{x_2}$  sunt permeanțele întrefierurilor  $x_1$  și  $x_2$ .  $P_{l_1}$  și  $P_{l_2}$  sunt permeanțele fluxurilor din aer care nu intersectează întrefierurile. Măsurători pe miezuri de diferite forme au arătat că *permeanțele  $P_{l_1}$ ,  $P_{l_2}$  și  $P_l$  sunt constante*, indiferent de mărimea întrefierului și de poziția sa. Numai  $P_{x_1}$  și  $P_{x_2}$  depind de întrefieruri.

Permeanțele de scăpări  $P_{l_1}$ ,  $P_{l_2}$  și  $P_l$  se pot determina experimental din 3 măsurători independente, care întotdeauna se pot obține la un diport linear și reciproc. De reținut că celelalte două permeanțe din model,  $P_{x_1}$  și  $P_{x_2}$ , sunt deja cunoscute, ele calculându-se cu formula uzuală a permeanței, cunoscând lungimea întrefierului, secțiunea miezului și numărul de spire. Principalul avantaj al modelului este că, pentru un miez dat, cu orice valori de întrefier, permeanțele de pierderi se pot determina dintr-un singur set de măsurători. Acestea caracterizează în mod unic miezul iar modelul este adevărat pentru orice alte valori ale întrefierurilor. Astfel noul model cu reluctanțe este cel din Fig. 2.18b.

Măsurătorile au arătat că permeanțele  $P_{l_1}$  și  $P_{l_2}$  sunt cu un ordin de mărime (8-20 de ori) mai mici decât  $P_l$ . Reluctanțele  $R_{l_1}$  și  $R_{l_2}$  devenind foarte mari în raport cu  $R_l$  pot fi omise din schema echivalentă și modelul apare în versiunea simplificată din Fig. 2.18c. Prin dualitate acest model se transformă în noul model al transformatorului cu elementele raportate la secundar din Fig. 2.18d. Comparându-l cu modelul clasic din Fig. 2.17b, se constată că, pe lângă diferența aranjării inductanțelor în  $\Pi$  față de structura în  $T$ , mai apare o *diferență importantă* și anume: în noul model permeanța  $P_l$  (reluctanța  $R_l$ ) este constantă și independentă de întrefieruri, în timp ce în vechiul model permeanțele  $P_{l_1}$  și  $P_{l_2}$  (reluctanțele  $R_{l_1}$  și  $R_{l_2}$ ) sunt puternic dependente de întrefieruri.

Aceasta se poate demonstra și analitic, transformând modelul în  $T$  în model  $\Pi$ .

Pentru tipurile de module de alimentare de telecomunicații aplicația practică imediată a acestui model se regăsește în utilizarea aranjamentului din Fig. 2.16a. În această configurație nu există întrefier pe coloana aferentă înfășurării secundare, deci în model  $R_{x_2} = 0$ , iar în modelul nou al transformatorului  $L_{x_2} = 0$ . Coeficientul de cuplaj este  $k = 1$  *indiferent de întrefierul  $x_1$  și condiția de pulsații nule devine:*

$$1 = \frac{n_1}{n_2} \quad (29)$$

deci numere egale de spire,  $n_1 = n_2$ . Acesta este un mare avantaj deoarece coeficientul de cuplaj fiind 1 indiferent de întrefier singura cerință este egalitatea numărului de spire. Un alt avantaj este că aceasta este structura cu cel mai mic zgomot radiat.

**Observație.** Măsurătorile, așa cum s-a amintit, au arătat un coeficient de cuplaj de 0,98 și nu de 1.

Acest lucru era de așteptat întrucât modelul simplificat care a dat valoarea unitară a neglijat fluxul de scăpări din primar înainte de întrefier, care e răspunzător de această mică diminuare.

Realizările practice cu această configurație magnetică aparțin firmei TESLaco și sunt din seria 90<sup>+</sup>ČUKonverter® [28]. Ele s-au făcut cu un convertor ČUK la care dioda de ieșire a fost înlocuită cu un MOSFET (gen redresor sincron) iar comutarea dispozitivelor se face nedisipativ prin aducerea în conducție la tensiune zero (ZVS). Modulul de 5V/100W are o frecvență de comutație de 140kHz, cu dispozitive magnetice integrate plate, radiatoarele fiind chiar substratul de aluminiu, deci practic fără radiatoare suplimentare. Randamentul ridicat de peste 90% la sarcină maximă face să nu fie necesară răcirea forțată cu aer, răcirea prin convecție naturală asigurând o temperatură de sub 100 °C la 55 °C temperatură ambientă. Dimensiunile modulului sunt de 2,4” x 4,5” x 0,5” iar raportul putere/suprafață de 1W/in<sup>2</sup>. Modulul respectă toate standardele de compatibilitate electromagnetă.

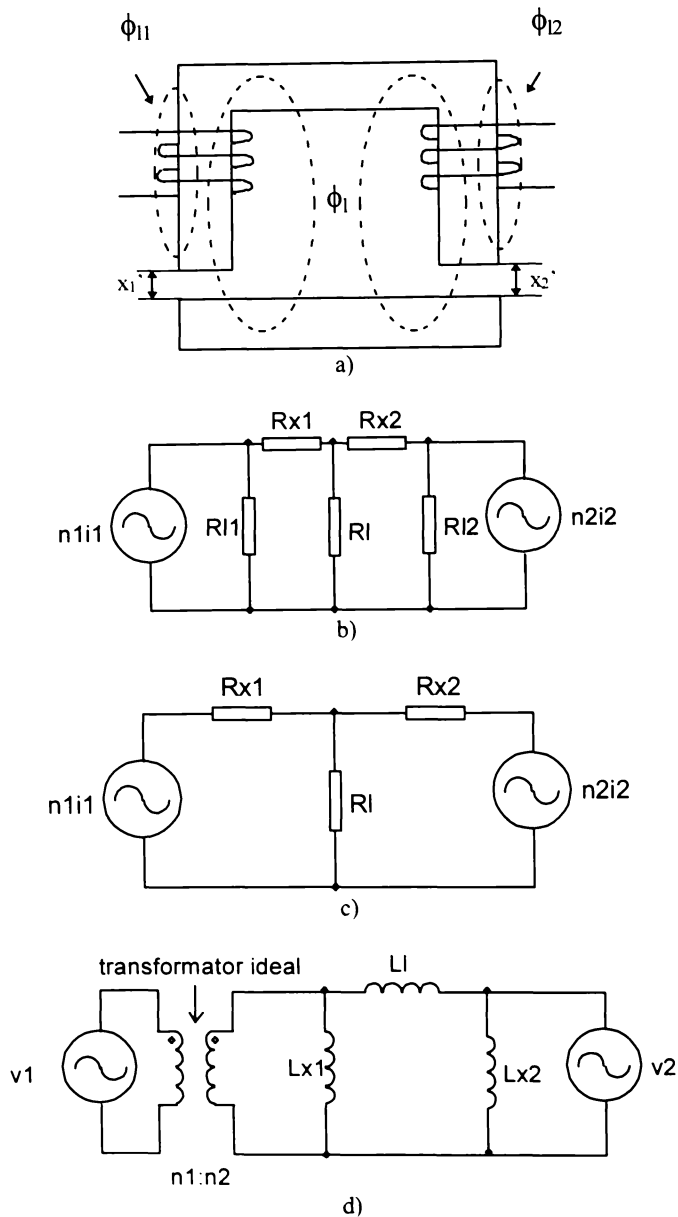


Fig. 2.18. Noul model al circuitului magnetic cu două bobine și întrefier.  
 (a) evidențierea fluxurilor; (b) modelul circuitului magnetic cu reluctanțe; (c) modelul simplificat al circuitului magnetic cu reluctanțe; (d) noul model al transformatorului.

## 2.4 Concluzii

Sinteza sistematică a convertoarelor cu pulsații nule de curent este o problemă deosebit de importantă în proiectare. Contribuțiile autorului tezei în acest domeniu, toate în paragraful 2.2, sunt:

- principiul generării convertoarelor având curenți cu pulsații nule studiind nu convertoare individuale ci celula de bază completă generatoare a familiei, realizându-se astfel un studiu general și unitar.
- demonstrarea echivalenței convertorului ZETA cu curent de ieșire fără pulsații și bobine necuplate cu un convertor BUCK-BOOST cu filtru la ieșire.
- imposibilitatea realizării condiției de curent cu pulsații nule cu bobine necuplate pentru convertoarele cu tensiune bipolară din familia ZSWJ, dar posibilitatea obținerii de astfel de convertoare cu bobine cuplate. Propunerea unei variante de convertor Watkins-Johnson care să prezinte curent de ieșire fără pulsații.
- demonstrarea, folosind analizei celulei de bază complete, că 6 din topologiile prezentate în [71] sunt de fapt versiuni a 2 topologii.
- obținerea de convertoare pătratice cu curenți de intrare/ieșire fără pulsații.

Autorul prognozează că în acest domeniu cercetarea în viitor se va canaliza pe:

- modificarea convenabilă minimală a topologiilor convertoarelor bipolare din familia ZSWJ în vederea obținerii de astfel de convertoare cu pulsații de curent nule și bobine necuplate.
- aplicarea noului model al transformatorului la alte topologii decât ĆUK, precum și în structurile cu izolare galvanică.
- un studiu comparativ mai aprofundat privind efortul de realizare a diverselor tipuri de convertoare cu pulsații nule de curent, cu indicarea categoriilor de aplicații cele mai potrivite pentru diversele topologii. Investigarea posibilității de realizare a amplificatoarelor de putere în comutație folosind topologii cu pulsații nule de curent [40].

## CAPITOLUL 3

### Sinteza convertoarelor cvasirezonante

Principala justificare a introducerii convertoarelor cu întrerupătoare rezonante și anume: convertoare cu comutare la curent nul (“zero-current switching” - ZCS), cu comutare la tensiune nulă (“zero-voltage switching” - ZVS) [106], [107], [108] a fost posibilitatea creșterii frecvenței de lucru comparativ cu corespondentele lor PWM, în condițiile menținerii randamentelor ridicate. Aceasta a permis reducerea elementelor reactive și construirea de convertoare mai ușoare și mai compacte. O generalizare a întrerupătoarelor rezonante a fost dată în [136], conducând la descoperirea a două noi clase de convertoare denumite convertoare ZCS sau ZVS cu undă cvasidreptunghiulară (“quasi-square wave - QSW”) [203]. Clasele de convertoare denumite multirezonante (“multirezonant converters”- MRC) conțin rețele rezonante mai complexe, cu mai mult de două elemente rezonante [180]. Toate aceste convertoare se numesc în literatură cu termenul generic de convertoare cvasirezonante (quasi-resonant converters - QRC).

Prezentul capitol se ocupă în primul rând de sinteza convertoarelor cvasirezonante funcționând la frecvență variabilă (cu comandă în frecvență), apoi se dezvoltă un procedeu de sinteză pentru convertoare cvasirezonante cu funcționare la frecvență constantă (cu comandă prin factorul de umplere.)

#### 3.1. Convertoare cvasirezonante. Definiții

Caracteristic tuturor convertoarelor QRC este faptul că ele derivă din convertoare PWM prin adăugarea de elemente rezonante. Dacă nu se precizează altceva, termenul de “elemente rezonante” va desemna bobine și capacități liniare. Conectarea elementelor rezonante se face de așa manieră încât formele de undă ale curentului și tensiunii pe elementele semiconductoare să se modifice, de la tranzițiile abrupte simultane în convertoarele PWM, în unde continue, cvasisinusoidale. Curentul prin sau tensiunea pe întrerupător sunt zero înainte și imediat după comutarea întrerupătorului, ceea ce conferă convertorului pierderi de comutație reduse și permite creșterea frecvenței.

Fie un convertor PWM cu o singură sursă de alimentare  $V_g$ , o sarcină  $R$  și două întrerupătoare generale  $S_1$  și  $S_2$ . Se presupune că în stare staționară curenții prin bobine și tensiunile pe capacități sunt mărimi cvasiconstante, cu pulsații neglijabile față de componentele continue suprapuse. Toate elementele de circuit se presupun ideale, deci randamentul convertorului este unitar.

*Definiția 1.* Prin circuit de curent continuu al unui convertor se înțelege circuitul în care toate bobinele sunt înlocuite cu scurtcircuite și toate capacitățile înlăturate, în timp ce sursele, sarcinile și întrerupătoarele sunt reținute.

*Definiția 2.* Se numesc prin termenul de “capacități PWM” capacitățile din convertorul părinte PWM, împreună cu sursa de alimentare  $V_g$ . Acesta pentru că termenul se referă la elemente pe care tensiunea poate fi admisă cvasiconstantă, cu o componentă continuă mult mai mare decât pulsațiile. Similar, bobinele din convertorul PWM părinte se numesc “bobine PWM”.

Are loc următoarea teoremă [114]:

*Teorema 1.* Dacă convertorului PWM descris mai sus i se adaugă elemente rezonante astfel încât în curent continuu circuitul convertorului rămâne nemodificat, atunci convertorul rezultat este QRC.



### 3.2 Sinteza unitară a convertoarelor cvasirezonante

Construcția convertoarelor QRC pleacă de la convertoare PWM prin inserarea elementelor rezonante. De aceea este necesară o caracterizare topologică a convertoarelor PWM, lucru care se poate face în termeni de teoria grafurilor, folosind definițiile și teorema din paragraful precedent.

Fie  $G$  graful asociat rețelei  $N$  a unui convertor oarecare și  $I$  un arbore al său - Fig. 3.1. Fie acum graful  $G^+$  care se construiește prin adăugarea unei ramuri la  $G$ . Există 4 posibilități de a adăuga această ramură, exemplificate în Fig. 3.2.

- ramura formează o buclă cu ea însăși.
- ramura nu se mai conectează la alte noduri din graful original.
- ramura se adaugă între două noduri din graful original
- ramura se conectează prin "spargerea" unui nod din graful original în alte două noduri, între cele două noduri rezultate.

Cazurile (a) și (b) dau un graf separabil și rețeaua  $N^+$  este complet echivalentă cu rețeaua  $N$ .

Cazurile (c) și (d) reprezintă însă modificări netriviiale ale rețelei originale. În cazul (c), în care ramura se adaugă între două noduri ale lui  $G$ , arborele  $I$  din  $G$  este arbore și pentru  $G^+$ , iar ramura adăugată face parte din co-arborele din  $G^+$ . În cazul (d), prin secționarea unui nod (nodul 2 se împarte în nodurile 2' și 2'') ramurile incidente la nodul inițial sunt și ele împărțite, unele legându-se la unul din cele două noduri rezultante, altele la celălalt (aici  $b, f$  la 2' iar  $g$  la 2''). În acest caz elementele arborelui original împreună cu ramura adăugată (aici  $i$ ) formează un arbore pentru graful rezultat  $G^+$ . Se pot astfel enunța următoarele două proprietăți [114]:

Proprietatea 1. Dacă ramura se adaugă între două noduri existente ale grafului  $G$  atunci orice arbore al lui  $G$  este arbore pentru  $G^+$ . Numim acest mod de inserție a ramurii *mod de tip A*.

Proprietatea 2. Dacă un nod din  $G$  este secționat în două noduri și ramura se adaugă legându-se între aceste două noduri atunci orice arbore din  $G$  împreună cu ramura adăugată este arbore pentru  $G^+$ . Numim acest mod de inserție a ramurii *mod de tip B*.

Proprietatea 3. Dacă inserția elementului în convertor este de tip A el formează o buclă (bucla fundamentală) cu ramuri din orice arbore al lui  $G$ . În Fig. 3.2c bucla fundamentală se formează cu ramura  $f$  a arborelui.

Proprietatea 4. Dacă inserția elementului este de tip B acesta formează o secțiune (secțiunea fundamentală) cu ramuri din rețeaua originală. În Fig. 3.2d secțiunea fundamentală se formează cu ramurile  $c$  și  $h$ .

În [114] se enunță următoarea proprietate:

Proprietatea 5. Pentru un convertor PWM cu două întrerupătoare, reprezentat prin graful  $G$ , avem:

- sursa de tensiune  $V_g$ , întrerupătorul  $S_1$  și capacitățile formează un arbore, notat  $I_{S_1}$ .
- sursa de tensiune  $V_g$ , întrerupătorul  $S_2$  și capacitățile formează un arbore, notat  $I_{S_2}$ .

Sinteza se bazează pe examinarea poziționării elementelor rezonante în raport cu  $I_{S_1}$  și  $I_{S_2}$  folosind proprietățile 1-4 și utilizează o parte din procedura expusă în [114] la care s-au adăugat unele demonstrații originale, s-au exclus unele situații neprecizate de autori și rezultatele s-au aplicat nu pe convertoare individuale ci pe celule de bază.

#### 3.2.1. Sinteza convertoarelor cvasirezonante cu două elemente rezonante

Se poate arăta că sunt necesare a se adăuga cel puțin două elemente rezonante - o bobină  $L_r$  și o capacitate  $C_r$  - la un convertor PWM pentru a se obține un convertor cvasirezonant. Se mai

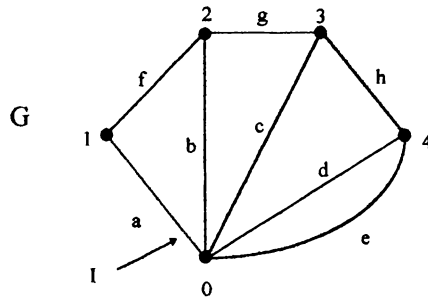
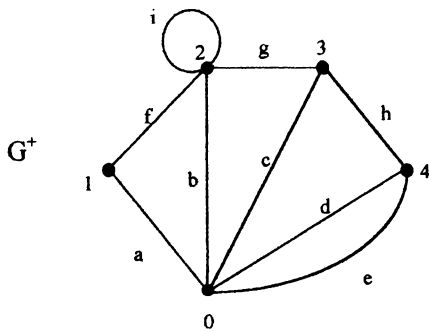
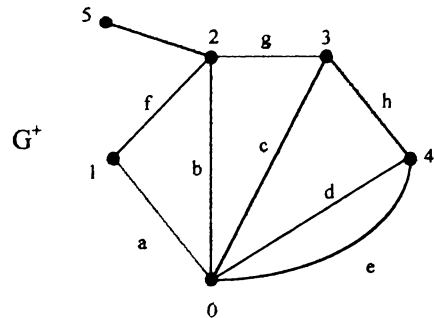


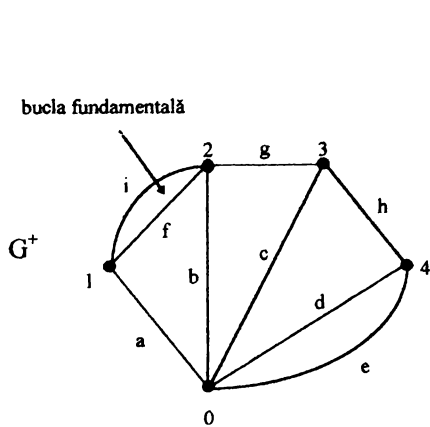
Fig. 3.1. Graful  $G$  asociat unui convertor PWM și un arbore  $I$  (roșu) al său.



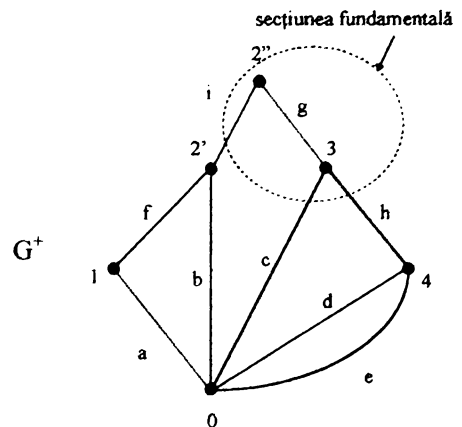
(a)



(b)



(c)



(d)

Fig. 3.2. Cele 4 posibilități de a adăuga o ramură (i) la graful  $G$  din Fig. 3.1 și obținerea lui  $G^+$

consideră că întotdeauna sarcina conține o capacitate  $C_o$  în paralel, astfel încât tensiunea și curentul prin  $R$  sunt practic constante.

Dacă condensatorul  $C_r$  este inserat în modul A atunci în curent continuu  $N^+ = N$ . Conform proprietății 3 el formează o buclă fundamentală cu ramuri din  $I_{S_1}$  și  $I_{S_2}$ . Conform proprietății 5,  $I_{S_1}$  și  $I_{S_2}$  conțin numai sursa de alimentare, capacități și un întrerupător. Împreună cu care dintre elementele din  $I_{S_1}$  și  $I_{S_2}$  se formează bucla avem trei situații:

- 1) bucla se formează numai cu un set nevid de capacități PWM. Dar capacitățile PWM au fost presupuse cu tensiune aproximativ constantă. Rezultă că și pe  $C_r$  tensiunea este constantă, deci el nu rezonază și practic formele de undă ale convertorului PWM rămân aceleași.
- 2) ( $CIS_1$ ): bucla se formează cu întrerupătorul  $S_1$  și un set, eventual vid, de capacități PWM.
- 3) ( $CIS_2$ ): bucla se formează cu întrerupătorul  $S_2$  și un set, eventual vid, de capacități PWM.

Conform proprietății 5 cazurile 2) și 3) sunt echivalente. Distincția dintre pozițiile  $CIS_1$  și  $CIS_2$  devine importantă numai după ce se adaugă al doilea element rezonant.

Notațiile  $CIS_1$  și  $CIS_2$  semnifică faptul că elementul rezonant este o capacitate ( $C$ ) și că este într-o buclă ( $I$ ) cu întrerupătorul  $S_1$  respectiv  $S_2$ .

Dacă condensatorul  $C_r$  este inserat în modul B atunci conform proprietății 4 el formează o secțiune cu ramuri din  $I_{S_1}$  și  $I_{S_2}$ . Față de inserția de tip A condiția  $N^+ = N$  are loc numai dacă  $C_r$  și un set nevid de capacități PWM (și nici un alt element) formează o secțiune în  $G^+$ . În această ipoteză singurele inserări eligibile sunt:

- 1)  $C_r$  și un set nevid de capacități PWM (și nici un alt element) formează o secțiune în  $G^+$  iar  $C_r$  formează o secțiune cu un set nevid de bobine PWM și eventual sarcina  $R$ . În această situație, conform celei de a doua părți a afirmației, pentru că prin sarcină și bobine curentul este cvasicontinuu, așa cum s-a presupus, rezultă că și prin  $C_r$  curentul este cvasicontinuu ceea ce ar însemna că el se încarcă la tensiune infinită sau deloc.
- 2)  $C_r$  și un set nevid de capacități PWM (și nici un alt element) formează o secțiune în  $G^+$  iar  $C_r$  formează o secțiune cu sarcina  $R$ , cu unul singur din întrerupătoarele  $S_1$  sau  $S_2$  și un set, eventual vid, de bobine PWM. În acest caz în starea topologică în care singurul întrerupător ce intră în secțiune cu  $C_r$  este deschis curentul prin sarcină este suma algebrică a curenților constanți ai bobinelor PWM ce intră în secțiune și curentul prin  $C_r$ . Ar rezulta că pulsațiile curentului prin  $C_r$  sunt și pulsațiile curentului de sarcină, deci și ale tensiunii, ceea ce contrazice presupunerea de pulsații reduse ale tensiunii pe sarcină.
- 3) ( $CcS_1$ ):  $C_r$  și un set nevid de capacități PWM (și nici un alt element) formează o secțiune în  $G^+$  iar  $C_r$  formează o secțiune cu  $S_1$  și un set nevid de bobine PWM.
- 4) ( $CcS_2$ ):  $C_r$  și un set nevid de capacități PWM (și nici un alt element) formează o secțiune în  $G^+$  iar  $C_r$  formează o secțiune cu  $S_2$  și un set nevid de bobine PWM.

Cazurile (3) și (4) sunt cele care rămân eligibile.

O analiză similară se poate face pentru posibilitățile de inserție ale bobinei. Mai simplu, acestea se deduc prin dualitate și se enunță astfel [114]:

Inductivitatea rezonantă  $L_r$  se poate insera astfel:

Inserția de tip A (dualitate cu inserția de tip B de la capacitate):

- 1)  $L_r$  și un set nevid de bobine PWM (și nici un alt element) formează o buclă în  $G^+$ , iar  $L_r$  formează o buclă cu un set nevid de capacități PWM și eventual sarcina  $R$ . În această situație, conform celei de a doua părți a afirmației, pentru că tensiunile pe capacitățile PWM și sarcină sunt cvasicontinui, așa cum s-a presupus, rezultă că și pe  $L_r$  tensiunea este cvasicontinuuă ceea ce ar însemna că prin  $L_r$  curentul crește infinit sau este nul.
- 2)  $L_r$  și un set nevid de bobine PWM (și nici un alt element) formează o buclă în  $G^+$ , iar  $L_r$  formează o buclă cu sarcina  $R$ , cu unul singur din întrerupătoarele  $S_1$  sau  $S_2$  și un set, eventual vid, de capacități PWM. În acest caz, în starea topologică în care singurul întrerupător ce intră în buclă cu  $L_r$  este în conducție, tensiunea pe sarcină este suma algebrică a tensiunilor constante ale capacităților PWM ce intră în buclă și tensiunea pe  $L_r$ . Ar rezulta că pulsațiile tensiunii pe  $L_r$  sunt și pulsațiile tensiunii pe sarcină, ceea ce contrazică presupunerea de pulsații reduse ale tensiunii pe sarcină.
- 3) ( $LIS_1$ ):  $L_r$  formează o buclă cu un set nevid de bobine PWM (și cu nici un alt element) și este în secțiune cu  $S_1$  și un set nevid de capacități PWM.
- 4) ( $LIS_2$ ):  $L_r$  formează o buclă cu un set nevid de bobine PWM (și cu nici un alt element) și este în secțiune cu  $S_2$  și un set nevid de capacități PWM.

Inserția de tip B (dualitate cu inserția de tip A de la capacități):

- 1)  $L_r$  formează o secțiune numai cu un set nevid de bobine PWM. Este o situație neeligibilă pentru că atunci curentul prin bobină rămâne cvasiconstant, nu rezonază și practic formele de undă ale convertorului rămân aceleași.
- 2) ( $LcS_1$ ):  $L_r$  formează o secțiune cu  $S_1$  și un set, eventual vid, de bobine PWM.
- 3) ( $LcS_2$ ):  $L_r$  formează o secțiune cu  $S_2$  și un set, eventual vid, de bobine PWM.

Cele de mai sus sunt sistematizate în Tab. 3.1.

Tab. 3.1. Modurile de inserție ale elementelor rezonante

MOD DE INSERȚIE	CAPACITATE $C_r$	BOBINĂ $L_r$
A	A1 Buclă cu set nevid de capacități PWM-NU. A2 ( $CIS$ ): Buclă cu <i>unul</i> din $S_1$ sau $S_2$ și cu un set (eventual vid) de capacități PWM - DA	A1 Buclă cu un set nevid de capacități PWM și eventual $R$ - NU. A2 Buclă cu <i>unul</i> din $S_1$ sau $S_2$ , $R$ și un set (eventual vid) de capacități PWM - NU. A3 ( $LIS$ ): Buclă cu un set nevid de bobine PWM și buclă cu <i>unul</i> din $S_1$ sau $S_2$ și cu un set nevid de capacități PWM - DA.
B	B1 Secțiune cu un set nevid de bobine PWM și eventual $R$ - NU. B2 Secțiune cu <i>unul</i> din $S_1$ sau $S_2$ , $R$ și un set (eventual vid) de bobine PWM-NU. B3 ( $CcS$ ): Secțiune cu un set nevid de capacități PWM și secțiune cu <i>unul</i> din $S_1$ sau $S_2$ și cu un set nevid de bobine PWM - DA.	B1 Secțiune cu set nevid de bobine PWM-NU. B2 ( $LcS$ ): Secțiune cu <i>unul</i> din $S_1$ sau $S_2$ și cu un set (eventual vid) de bobine PWM - DA.

Deoarece pentru capacitatea rezonantă  $C_r$  există 4 poziții de inserție ( $CIS_1$ ,  $CIS_2$ ,  $CcS_1$ ,  $CcS_2$ ), iar pentru bobina rezonantă  $L_r$  tot 4 poziții de inserție ( $LcS_1$ ,  $LcS_2$ ,  $LIS_1$ ,  $LIS_2$ ) rezultă cele două elemente se pot combina în 16 moduri. După ce combinațiile nefezabile și redundante sunt eliminate se obține setul complet al claselor distincte de convertoare QRC cu două întrerupătoare și două elemente rezonante. Acest lucru este sistematizat în Tab. 3.2, unde s-au menționat și definițiile topologice ale celor 3 perechi de clase ( $ZCS \leftrightarrow ZVS$ ,  $ZCS-QSW \leftrightarrow ZVS-QSW$ ,  $Q_n\text{-PWM} \leftrightarrow Q_r\text{-PWM}$ ).

Definiții topologice clare au fost date numai pentru clasele  $ZCS$  și  $ZVS$  în [42]. Suplimentar, procedura de sinteză generează două clase noi de convertoare denumite convertoare cvasi PWM "on-resonant" și "off-resonant" -  $Q_n\text{-PWM}$  respectiv  $Q_r\text{-PWM}$ .

Tab. 3.2. Clasele complete de convertoare QRC cu două elemente rezonante

	$LcS_1$	$LcS_2$	$LIS_1$	$LIS_2$
$CIS_1$	X	ZVS	ZVS-QSW	ZVS-QSW
$CIS_2$	ZCS	X	ZVS-QSW	ZVS-QSW
$CcS_1$	ZCS-QSW	ZCS-QSW	X	$Q_n\text{-PWM}$
$CcS_2$	ZCS-QSW	ZCS-QSW	$Q_r\text{-PWM}$	X

### 3.2.2. Variațiuni topologice

Pentru o clasă de convertoare dată, pozițiile lui  $L_r$  și  $C_r$  în convertorul PWM nu sunt unice, dar formele de undă ale tensiunii și curentului pe întrerupătoare sunt aceleași calitativ, iar caracteristicile de curent continuu identice. Numărul variațiunilor topologice este mare la convertoarele  $ZCS$  și  $ZVS$  pentru că cerința  $N^+ = N$  în curent continuu nu impune restricții în poziția elementelor. Numărul de variațiuni este cel mai mic în convertoarele  $Q\text{-PWM}$  unde există cele mai multe restricții.

### 3.2.3. Clase degenerare de convertoare QRC cu două elemente rezonante

Discuția precedentă a fost purtată, pentru fiecare element, relativ la arborii din convertorul PWM. Dar o dată primul element adăugat, rețeaua de la care se pleacă atunci când se inserează al doilea element nu mai este a convertorului PWM. Unul din arborele sau co-arborele original vor conține primul element rezonant adăugat.

Dacă primul element adăugat este  $L_r$  și acest lucru este făcut de manieră B ( $LcS_1$  sau  $LcS_2$ ), atunci pentru  $C_r$  în inserția de tip A pe lângă posibilitățile  $CIS_1$  și  $CIS_2$  mai există un caz de inserție de tip A și anume:

( $CIL$ ):  $C_r$  este într-o buclă cu  $L_r$  și un set, eventual vid, de capacități PWM.

Similar, dacă primul element adăugat este  $C_r$  și acest lucru este făcut de manieră A ( $CIS_1$  sau  $CIS_2$ ), atunci pentru  $L_r$  în inserția de tip B pe lângă posibilitățile  $LcS_1$  sau  $LcS_2$  mai există un caz de inserție de tip B și anume:

( $LcC$ ):  $L_r$  este într-o secțiune cu  $C_r$  și un set, eventual vid, de bobine PWM.

Aparent pozițiile duale  $CIL$  și  $LcC$  definesc, după tipicul din Tab. 3.1, o altă pereche de convertoare QRC. Se poate arăta că în aceste convertoare QRC degenerare nu este posibilă funcționarea cu tranziții  $ZCS/ZVS$  decât dacă atât  $S_1$  cât și  $S_2$  sunt întrerupătoare active. Clasa  $CIL$  este de tip  $ZVS$  iar clasa  $LcC$  de tip  $ZCS$ .

### 3.2.4. Sinteza convertoarelor cvasirezonante cu mai mult de două elemente rezonante

Aceste convertoare se vor construi recursiv, adăugând câte un element rezonant la convertorul QRC anterior. Examinarea poziționării elementului adăugat se va face în raport cu arborii și co-arborii din convertorul QRC original.

Dintre convertoarele QRC cu mai multe elemente rezonante cele rezultate din convertoare ZVS prin inserția de tip  $CS_1$  a unui nou condensator  $C_{r1}$  sunt chiar convertoarele ZVS multirezonante [180].

În general simplitatea rețelei convertorului este de dorit din punct de vedere al randamentului, dimensiunilor, greutateii și costului. De aceea introducerea a mai mult de două elemente rezonante trebuie bine motivată.

### 3.2.5. Implementarea întrerupătoarelor

Proprietățile și funcționarea convertoarelor QRC diferă substanțial pentru diferite modalități de implementare a întrerupătoarelor, vezi de exemplu modurile semiundă (“half-wave” - HW) și undă plină (“full-wave” - FW). Față de situația clasică, în care un întrerupător este activ și al doilea pasiv (diodă), ambele întrerupătoare pot fi active. O astfel de transformare a fost propusă pentru prima oară în convertoarele PWM în care întrerupătoarele comandate au înlocuit dioda de nul în scopul reducerii pierderilor în conducție. Denumirea dată acestui tip de comandă a fost de *redresare sincronă*. Ca și în convertoarele PWM, în convertoarele QRC un redresor sincron duce la noi moduri de funcționare și, foarte important, la posibilitatea funcționării la frecvență constantă [116].

Există 5 posibilități de combinare a întrerupătoarelor în convertoare QRC, astfel:

- un întrerupător activ și un întrerupător pasiv
  - CD: bidirecțional în curent + diodă
  - VD: bidirecțional în tensiune + diodă
- ambele întrerupătoare active
  - CC: bidirecțional în curent + bidirecțional în curent
  - VV: bidirecțional în tensiune + bidirecțional în tensiune
  - CV: bidirecțional în curent + bidirecțional în tensiune

unde bidirecțional în curent este un întrerupător de cadranele I+IV sau II+III iar bidirecțional în tensiune un întrerupător de cadranele I+II sau III+IV.

De menționat în final că pentru fiecare poziție posibilă a întrerupătoarelor convertorul se reduce la o rețea liniară. Spunem că este într-o *stare* de funcționare. Funcționarea periodică prin trecerea prin aceleași stări de funcționare formează un *mod* de funcționare. Pentru un convertor QRC dat și o anume implementare practică a întrerupătoarelor succesiunea stărilor nu este unică, cu alte cuvinte există mai multe moduri de funcționare. Deci față de convertoarele PWM, unde cea mai mare parte a caracteristicilor se pot deduce direct din topologie, în cazul convertoarelor QRC trebuie făcut un studiu al modurilor posibile de funcționare dacă se dorește o caracterizare completă. Desigur că de interes practic sunt acele moduri ce au toate comutările nedisipative (la tensiune sau la curent nul). O investigație a posibilelor moduri de funcționare este făcută în [61] prin analiza în planul stărilor, iar o excelentă comparație între clase de convertoare QRC în raport cu solicitările componentelor și a sensibilității la elemente parazite se găsește în [114].

### 3.2.6 Aplicații ale procedurii de sinteză

Se propune de către **autorul tezei** aplicarea procedurii de sinteză nu pe convertoare individuale, ci pe celule de bază complete cu 3 terminale. Se admite că celula de bază are  $n$  noduri, considerându-se ca noduri și terminalele celulei.

Inserarea capacității în modul A se poate face în  $C_n^2 = \frac{n(n-1)}{2}$  moduri. Presupunând că în paralel pe sarcină există întotdeauna o capacitate de filtraj,  $C_o$ , atunci conform cazului A1 rezultă că  $C_r$  nu se va putea conecta între două terminale ale celulei. Rămân astfel  $C_n^2 - C_3^2 = \frac{n(n-1)}{2} - 3$  moduri de a conecta capacitatea. De exemplu pentru celula ZSWJ din Fig. 3.3 avem  $n = 5$  și deci avem 7 posibilități de a conecta pe  $C_r$ : perechile de noduri 0-2, 0-3, 1-2, 1-3, 2-3, 2-4, 3-4. Dintre acestea perechea 2-3 este exclusă pentru că suntem în condițiile cazului A1 (buclă cu  $C_1$ ).

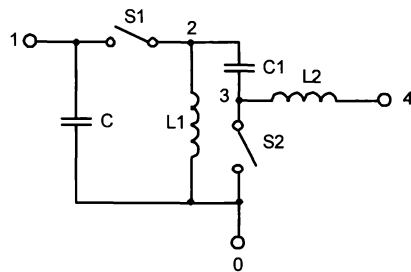
La inserarea capacității în modul B, pentru că o condiție necesară este ca el să formeze o secțiune numai cu capacități PWM și pentru că în celulă există numai o capacitate, rezultă că există numai două posibilități de inserare și anume în serie cu  $C_1$  sau în serie cu  $C$ . Plasarea în serie cu  $C$  este însă exclusă pentru că  $C_r$  se poate întotdeauna aduce să fie legat la borna lui  $R$  care nu este în comun cu  $V_g$ . Dacă această bornă este 1 se ajunge în cazul B1, dacă este 0 sau 4 se ajunge în cazul B2. Rămâne deci numai înserierea cu  $C_1$ . Pentru cele mai multe celule inserarea în mod A a capacității (pozițiile  $CcS_1$  și  $CcS_2$ ) conduce la plasarea capacității în serie cu o singură capacitate PWM. De aceea inserarea acestei capacități rezonante este echivalentă cu reducerea valorii capacității PWM.

Inserarea bobinei  $L_r$  în modul A conduce pentru celula în studiu la două posibilități: în paralel cu  $L_1$  sau în paralel cu  $L_2$ . Ca și la inserarea capacității în modul A, inserarea bobinei în modul B conduce de obicei la plasarea acesteia în paralel cu una din bobinele celulei, ceea ce este echivalent cu reducerea bobinei PWM.

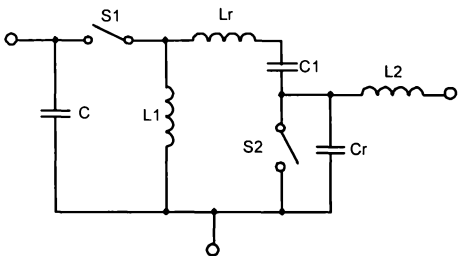
Inserarea în modul B a bobinei se studiază "spărgând" cele  $n$  noduri ale celulei (inclusiv terminalele pentru că sursa și sarcina nu aduc noi bobine) și eliminând pozițiile ce contrazic cazul B1. De exemplu pentru celula analizată bobina se poate insera în serie spre terminalul 1, în serie spre terminalul 2 sau în serie spre terminalul 3. De asemenea se poate sparge nodul 2 și lega  $L_r$  serie cu  $C_1$  și la fel pentru nodul 3.

Exemplele de celule din Fig. 3.3 ilustrează buclele și secțiunile în definițiile topologice din Tab. 3.2. Se remarcă identitatea celulelor Q-PWM cu celula PWM. Acest lucru se explică prin aceea că condensatorul PWM și cel rezonant sunt în serie, iar bobina PWM și cea rezonantă în paralel, absorbindu-se. Este interesant de subliniat că un convertor Q-PWM poate fi privit ca un convertor PWM ce funcționează simultan în conducție discontinuă din punct de vedere al curentului prin bobine (DICM) și în conducție discontinuă din punct de vedere a tensiunii pe capacitate (DCVM) [115]. În același mod, convertoarele QSW pot fi considerate convertoare PWM funcționând în DICM sau DCVM la care se adaugă un singur element rezonant. În sensul acestor interpretări numai convertoarele ZCS și ZVS reprezintă cu adevărat convertoare cvasirezonante cu două elemente rezonante.

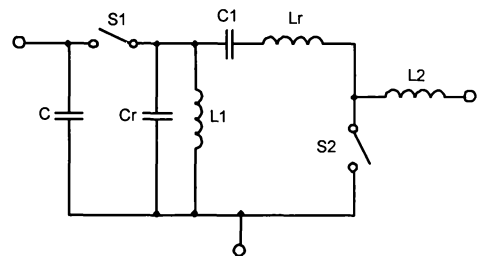
Având celulele QRC diversele tipuri de convertoare QRC se obțin legând celula la o sursă de tensiune continuă  $V_g$  și o sarcină  $R$  în paralel cu o capacitate  $C_o$ , având în comun borna minus a sursei cu ansamblul  $R - C_o$ , după procedeul expus în capitolul 1. În fig. 3.4 sunt ilustrate două exemple de celule ZSWJ QRC degenerate.



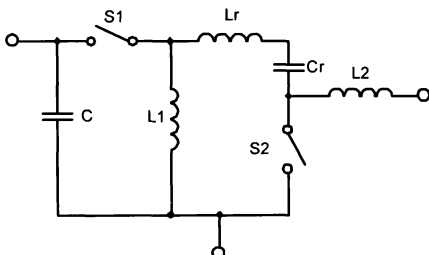
(a) ZSWJ - PWM



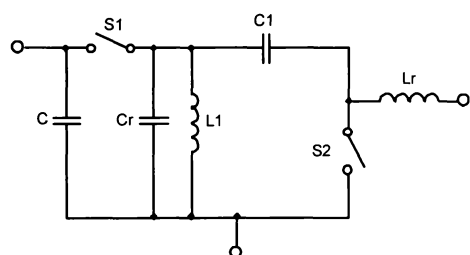
(a) ZC -  $CIS_2$  și  $LcS_1$



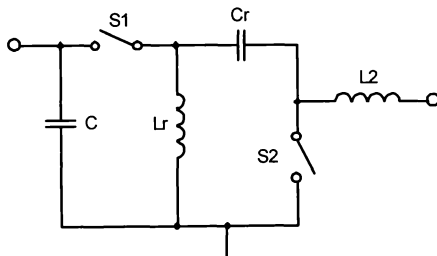
(b) ZV -  $CIS_1$  și  $LcS_2$



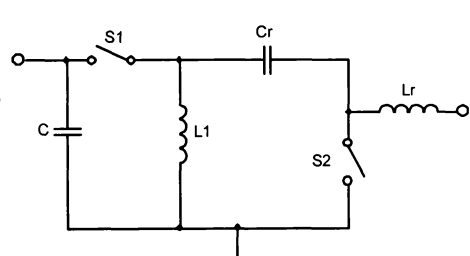
(c) ZC-QSW -  $CcS_2$  și  $LcS_1$



(d) ZV-QSW -  $CIS_1$  și  $LIS_2$



(e)  $Qr$ -PWM -  $CcS_2$  și  $LIS_1$



(f)  $Qn$ -PWM -  $CcS_1$  și  $LIS_2$

Fig. 3.3. Exemple de celule cvasirezonaante ZSWJ-QRC derivate din celula ZSWJ-PWM.



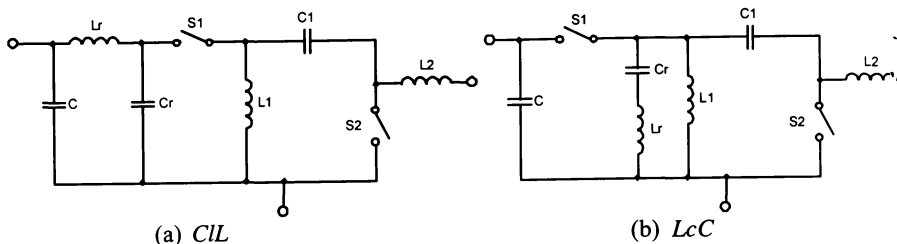


Fig. 3.4. Exemple de celule ZSWJ-QRC degenerate.

### 3.3 Comanda cu frecvență constantă a convertoarelor cvasirezonante și rezonante.

În comparație cu topologiile PWM din care provin, convertoarele QRC au o serie de dezavantaje binecunoscute:

- solicitări în curent și în tensiune ale întrerupătoarelor mai mari
- pierderi de conducție mai mari (date și de faptul că dispozitivele de tensiune mare au rezistențe de conducție mari).
- domeniu limitat al sarcinii pentru care se realizează comutarea nedisipativă.
- domeniu limitat al raportului static de conversie.
- funcționarea la frecvență variabilă atunci când se face stabilizarea tensiunii de ieșire în raport cu variațiile sarcinii sau ale tensiunii de alimentare, sau când se dorește o tensiune de ieșire modificabilă.

Acest ultim dezavantaj face ca dispozitivele magnetice să nu poată fi utilizate în mod optim și ca zgomotul de conducție și radiație generat să fie dificil de controlat. De asemenea nu este neobișnuit ca funcționarea la frecvență constantă să fie impusă de sistemul în care convertorul QRC se integrează.

Calitativ, mecanismul prin care se ajunge la funcționarea cu frecvență variabilă poate fi explicat după cum urmează. Într-un convertor QRC cu un întrerupător activ și o diodă aducerea în conducție (pentru ZCS) sau în blocare (pentru ZVS) se face cu un semnal extern. În schimb a doua comutare apare natural, impusă de condiția de anulare a curentului (pentru ZCS) sau tensiunii (pentru ZVS), deci a unei rezonante. Ca rezultat, se pierde un grad de libertate și deci modificarea frecvenței se impune.

S-au propus mai multe metode de funcționare la frecvență constantă a convertoarelor QRC. În toate este introdusă o comandă suplimentară independentă. Astfel în [49] se introduce un întrerupător adițional pentru a modifica valoarea aparentă a unui element rezonant. Similar conceptuală este metoda în care caracteristica neliniară a materialelor magnetice este folosită pentru a controla valoarea bobinei rezonante. O altă posibilitate este utilizarea structurilor rezonante în punte completă unde există o diversitate de strategii de comandă pentru cele patru întrerupătoare active. Exemple de astfel de convertoare rezonante în punte sunt descrise în [189] și [144]. În [190] funcționarea la frecvență constantă se obține prin utilizarea a două convertoare rezonante de tip paralel legate cu ieșirile în serie și reglând defazajul. Conceptul este însă mai vechi, fiind introdus de General Dynamics Corporation [44]. Aceiași autori introduc conceptul de funcționare pe mod limitat [191] în care se menține aceeași structură ca în [190], dar se modifică secvența de comandă a întrerupătoarelor celor două convertoare paralel.

În continuare metoda de funcționare la frecvență constantă prezentată se bazează pe înlocuirea diodei din convertoarele QRC clasice cu un întrerupător activ, după tipicul redresoarelor sincrone [116]. Numim acest element transformat element redresor comandat. În acest fel comutarea sa nu mai este dictată numai de formele de undă ale circuitului ci și de un semnal extern.

În noul convertor QRC astfel obținut, denumit convertor cvasirezonaant cu funcționare la frecvență constantă ("constant frequency quasi-resonant converter" - CF-QRC), există două comutări controlate extern. Care sunt aceste două comutări depinde de natura convertorului (ZCS, ZVS, QSW, MRC, etc.). Raportul static de conversie se poate modifica prin modificarea intervalului de timp dintre cele două comutări controlate, menținând frecvența constantă. Este interesant că în [50] s-a utilizat un astfel de redresor comandat pentru a obține funcționarea la frecvență constantă a unor categorii de convertoare ce pot fi clasificate ca membre ale clasei ZV-QSW.

Așa cum existența diodei în convertoarele QRC clasice și modul de implementare a întrerupătorului principal determină diferite moduri de funcționare (semiundă sau undă plină), nu apare surprinzător faptul că în convertoarele CF-QRC se întâlnesc o serie de *moduri de funcționare noi*, datorate faptului că acum ambele întrerupătoare sunt active. Așa cum s-a menționat în paragraful 3.2.5, în cazul în care ambele întrerupătoare sunt active există 3 posibilități distincte de implementare a întrerupătoarelor (în paranteză au fost trecute cadranele):

CC: bidirecțional în curent (I+IV sau II+III) + bidirecțional în tensiune (I+IV sau II+III).

VV: bidirecțional în tensiune (I+II sau III+IV) + bidirecțional în tensiune (I+II sau III+IV).

CV: bidirecțional în curent (I+IV sau II+III) + bidirecțional în tensiune (I+II sau III+IV).

Pentru fiecare structură de convertor și mod de implementare a întrerupătoarelor este necesar un studiu separat al diferitelor moduri posibile de funcționare. Din punct de vedere practic cea mai performantă modalitate de implementare este cu întrerupătoare bidirecționale în curent (adică unidirecționale în tensiune), din două motive:

- întrerupătoarele bidirecționale în tensiune au pierderi de conducție mai mari din cauza diodei suplimentare conectate în serie cu tranzistorul.
- comutarea la tensiune nulă (care se știe că este de preferat comutării la curent nul) a unui întrerupător bidirecțional în tensiune nu este posibilă într-un convertor practic pentru că dioda în serie cu tranzistorul nu permite descărcarea capacității parazite a tranzistorului.

Clasele de convertoare de interes pentru aplicații de înaltă frecvență sunt cele cu comutare la tensiune nulă: CF-ZVS, CF-ZVS-QSW și cele multirezonaante cu comutare la tensiune nulă CF-ZVS-MRC.

- Se va nota o *stare de funcționare* a convertorului printr-un dublet de numere ce semnifică starea întrerupătoarelor (0 - deschis, 1 - închis). Pentru convertoare cu două întrerupătoare există 4 stări posibile de funcționare: (0,0), (0,1), (1,0) și (1,1).
- Un *mod de funcționare* va fi descris prin succesiunea de stări de funcționare. De interes sunt desigur acele moduri de funcționare în care *toate* comutările se fac la tensiune nulă sau la curent nul și în care două comutări sunt controlabile extern pentru a permite funcționarea la frecvență constantă. Se definește factorul de umplere  $D$  al unui convertor CF-QRC ca fiind intervalul de timp dintre cele două comutări controlate raportat la perioada de comutație.
- În planul raport static de conversie - curent de sarcină normalizat (caracteristica de ieșire normalizată) se definește *regiunea de funcționare* ca fiind acea parte din planul caracteristicii de ieșire normalizate unde funcționarea într-un anumit mod este posibilă. În proiectarea convertoarelor granițele dintre diferite regiuni de funcționare trebuie să fie cunoscute, pentru că se impune ca pe domeniul de variație a sarcinii și a tensiunii de alimentare funcționarea să se facă în regiunea de funcționare dorită.

Un exemplu de celulă CF-ZVS este prezentat în Fig. 3.5b, în Fig. 3.5a arătându-se celula generală ZVS din care provine (generatoare a convertoarelor BUCK, BOOST și BUCK-BOOST). Tranzistorul din  $S_1$  este adus în conducție la tensiune nulă, ca în convertorul convențional. Există două moduri de interes:

I:  $\dots \rightarrow (0,0) \rightarrow (0,1) \rightarrow (1,1) \rightarrow (1,0) \rightarrow \dots$ ,

II:  $\dots \rightarrow (1,1) \rightarrow (0,1) \rightarrow (1,1) \rightarrow (1,0) \rightarrow \dots$ ,

Modul I este identic cu cel din convertoarele ZVS clasice. Dacă se utilizează întrerupătoare bidirecționale în curent, modul I este cunoscutul ZVS-HW. Singura diferență este că tranziția din starea (0,0) în starea (0,1) are loc atunci când  $S_2$  este adus în conducție, înainte ca tensiunea să ajungă la zero, în timp ce în convertorul clasic în modul HW dioda începe să conducă atunci când tensiunea pe ea atinge valoarea zero. Modul I este de mai puțin interes practic pentru că regiunea de funcționare în planul caracteristicii de ieșire este foarte restrânsă. Pe de altă parte modul II nu se întâlnește în convertoarele ZVS clasice cu comanda în frecvență și este modul cel mai indicat în funcționarea convertoarelor CF-ZVS.

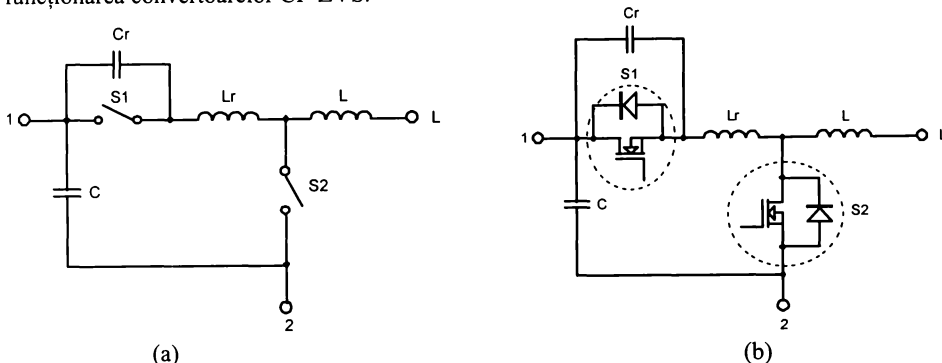


Fig. 3.5. O celulă de bază ZVS (în comanda cu frecvență variabilă  $S_2$  e o diodă) (a) și implementarea practică a întrerupătoarelor pentru ca celula să fie CF-ZVS (b)

În modul II aducerea în conducție a lui  $S_2$  și blocarea lui  $S_1$  sunt controlabile extern, în timp ce blocarea lui  $S_2$  și aducerea în conducție a lui  $S_1$  sunt comutări naturale. Tranzistorul din  $S_1$  trebuie adus în conducție înainte ca valoarea curentului prin  $S_1$  să devină pozitivă. Factorul de umplere se definește ca raportul dintre intervalul de timp măsurat de la aducerea în conducție a tranzistorului din  $S_2$  și blocarea tranzistorului din  $S_1$ .

**Observație.** Specificarea faptului că întrerupătoarele utilizate sunt bidirecționale în curent nu precizează însă și cum se conectează tranzistorul și dioda, ca și sens, în schemă. Aceasta pentru că un întrerupător bidirecțional în curent poate fi unul de cadranele (II, III) dar și unul de cadranele (I, IV). În Fig. 3.5.b sensul este cel din convertorul BUCK (cu borna plus a lui  $V_g$  în 1, borna minus în 2 și sarcina între 2 și L). Dacă se încearcă utilizarea aceleiași celule la sinteza convertorului BOOST (cu borna plus a lui  $V_g$  în L, borna minus în 2 și sarcina între 1 și 2) structura nu va funcționa. De aceea sinteza va pleca întotdeauna de la celula din Fig. 3.5.a iar studiul polarității tensiunii pe întrerupătoare, coroborat cu faptul că acestea sunt bidirecționale în curent, vor arăta cum anume se vor conecta în schemă (vor stabili cadranele de funcționare). Acest lucru este valabil și pentru celelalte celule care se vor analiza. **Autorul** propune ca regulă generală ca sensul de conectare al întrerupătoarelor bidirecționale în raport cu o mărime (tensiune sau curent), deci cu funcționare în două cadrane, se face estimând semnul mărimii duale (curent sau tensiune), în raport cu care întrerupătoarele sunt unidirecționale.

Convertoarele CF-ZVS-QSW moștenesc performanțele corespondentelor lor cu comandă în frecvență, din care cele mai importante sunt comutarea ambelor dispozitive la tensiune nulă și solicitările în tensiune care sunt aceleași cu cele din convertorul PWM din care provin. Un exemplu de celulă CF-ZVS-QSW este ilustrat în Fig. 3.6b. Un mod de funcționare cu toate comutările la tensiune nulă este definit de secvența:

$$I: \dots \rightarrow (0,0) \rightarrow (1,0) \rightarrow (0,0) \rightarrow (0,1) \rightarrow \dots,$$

În acest mod ambele blocări sunt controlabile extern, în timp ce aducerile în conducție sunt naturale. Factorul de umplere se definește ca intervalul de timp scurs de la blocarea tranzistorului din  $S_2$  și aducerea în conducție a tranzistorului din  $S_1$ .

Într-un convertor CF-ZVS-MR întrerupătoarele se blochează la tensiune nulă. În plus toate elementele parazite semnificative - capacitățile joncțiunilor dispozitivelor de comutație și inductivitatea parazită distribuită în bucla cu întrerupătoarele (inclusiv inductivitatea de scăpări a transformatoarelor, dacă există izolare galvanică) - se pot încorpora în rețeaua rezonantă. Ambele proprietăți menționate anterior se moștenesc de la convertoarele ZVS-MRC cu comandă în frecvență. O celulă BUCK CF-ZVS-MR este cea din Fig. 3.7b.

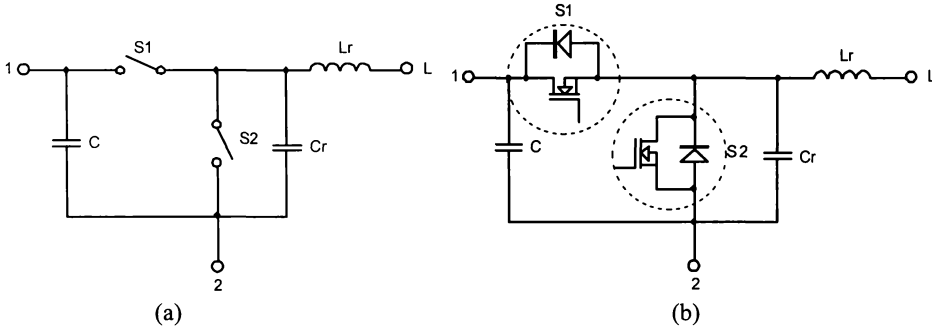


Fig. 3.6. O celulă de bază QSW (în comanda cu frecvență variabilă  $S_2$  e o diodă) (a) și implementarea practică a întrerupătoarelor pentru ca celula să fie CF-QSW (b).

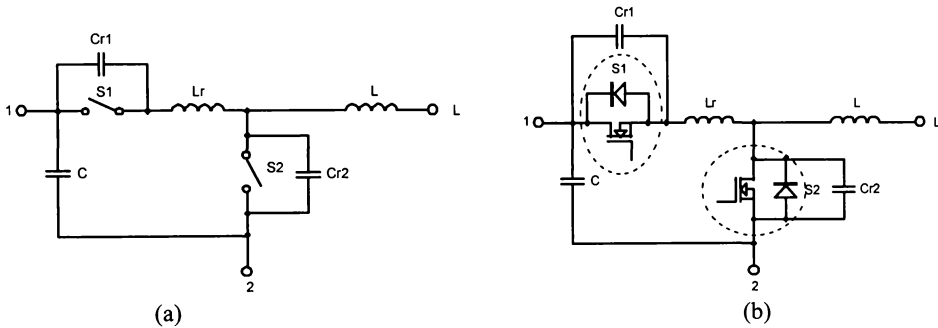


Fig. 3.7. O celulă de bază ZVS-MR (în comanda cu frecvență variabilă  $S_2$  e o diodă) (a) și implementarea practică a întrerupătoarelor pentru ca celula să fie CF-ZVS-MR (b).

Există patru moduri de interes:

- I:  $\dots \rightarrow (1,1) \rightarrow (1,0) \rightarrow (0,0) \rightarrow (0,1) \rightarrow \dots$ ,
- II:  $\dots \rightarrow (1,1) \rightarrow (1,0) \rightarrow (0,0) \rightarrow (1,0) \rightarrow \dots$ ,
- III:  $\dots \rightarrow (1,1) \rightarrow (0,1) \rightarrow (0,0) \rightarrow (0,1) \rightarrow \dots$ ,
- IV:  $\dots \rightarrow (1,1) \rightarrow (0,1) \rightarrow (1,1) \rightarrow (1,0) \rightarrow \dots$ ,

Factorul de umplere este definit ca fracțiunea din perioadă reprezentată de timpul scurs între blocarea tranzistorului din  $S_2$  și blocarea tranzistorului din  $S_1$ .

O ultimă observație se referă la faptul că modul de conectare a întrerupătoarelor în celulele prezentate poate fi și invers, în funcție de polaritatea tensiunilor aplicate între bornele celulei. Topologic însă și ca și tip de întrerupătoare, ea rămâne aceeași.

Pentru toate situațiile se poate face o analiză de curent continuu unitară. Acest lucru este efectuat în capitolul 4. O analiză a posibilelor moduri de funcționare pentru convertoarele rezonante serie sau paralel pe mod limitat, funcționând cu frecvență constantă este descrisă în [191], [22], [189] și [162].

### 3.4 Concluzii

Contribuția autorului în domeniul sintezei convertoarelor cvasirezonaante, fără pretenția de a fi fundamentală, se referă la:

- sistematizarea algoritmului de sinteză prezentat în paragraful 3.2, pe ideea expusă în [114], cu demonstrarea imposibilității de a avea cazurile B1, B2 pentru inserția capacității și cazurile A1, A2 pentru inserția bobinei rezonante.
- extinderea aplicării algoritmului de sinteză de la convertoare la celule de bază.
- sinteza convertoarelor cvasirezonaante cu celule de bază și precizarea clară a modului de alegere și conectare a întrerupătoarelor când acestea sunt bidirecționale într-o mărime și unidirecționale în cealaltă - paragraful 3.3.

Autorul consideră că ar fi deosebit de interesant și util să se găsească o procedură de sinteză în cazul unor convertoare cu trei sau mai multe întrerupătoare. Astfel de topologii au fost raportate în [52] și [43] și, deși numărul mai mare de întrerupătoare sugerează un randament mai slab, ele se bucură de unele proprietăți care le fac atractive în anumite aplicații.

## CAPITOLUL 4

### Analiza unitară de curent continuu a convertoarelor dc-dc

Analiza de curent continuu, sau *analiza statică* a convertoarelor dc-dc se referă în primul rând la deducerea caracteristicilor control-ieșire, caracteristicilor de ieșire, a solicitărilor de tensiune și curent ale componentelor și determinarea regiunilor de funcționare într-un anumit mod. Celula de bază permite o analiză unitară a caracteristicilor convertoarelor aparținând aceleiași familii. Ca și concept mai general se introduce noțiunea de mediu de comutație [Maksimović, 117]. Pentru a mări caracterul de generalitate rezultatele se vor prezenta sub formă unitară și normalizată. Totodată, din punct de vedere al modului de funcționare autorul introduce o nouă clasă de convertoare, denumită cu tensiune de intrare discontinuă (“discontinuous input voltage mode” - DIVM). Sunt de asemenea investigate în curent continuu și convertoarele pătratiche. Pe baza caracteristicilor statice deduse s-au scris programe de generare automată a caracteristicilor normalizate care se constituie într-un instrument util și necesar în proiectare.

#### 4.1 Notății utilizate și principiul analizei unitare de curent continuu

Pentru un convertor se notează cu  $M$  raportul dintre tensiunea medie de ieșire și tensiunea continuă de alimentare, numit *raport static de conversie*:

$$M = \frac{V_o}{V_g} \quad (1)$$

Notăm cu  $c$  *mărimea de comandă* a convertorului, de regulă factorul de umplere  $D$  al întrerupătorului (întrerupătoarelor) activ(e) sau frecvența de comutație  $f_s$ . Dependența tensiunii de ieșire de mărimea de comandă la sarcină constantă se numește *caracteristică statică control-ieșire*. Dependența tensiunii de ieșire de curentul mediu de ieșire la tensiune de intrare constantă se numește *caracteristică statică de ieșire*. De multe ori aceste caracteristici se dau în *formă normalizată*, iar atributul “static” se va omite pentru scurtarea denumirii. *Caracteristica statică control-ieșire normalizată* reprezintă dependența lui  $M$  de mărimea de comandă (care dacă este adimensională nu se modifică, în caz contrar normalizându-se și ea), iar *caracteristica de ieșire normalizată* reprezintă dependența lui  $M$  de curentul de sarcină normalizat. Evident normalizarea se face prin mărimi constante. În cele ce urmează toate tensiunile și curenții la care se va face referire vor fi valori medii, dacă nu se specifică altceva și se vor nota cu majuscule. Analiza unitară are la bază următoarele teoreme [115], [117], [118]:

**Teorema 1.** Pentru un convertor PWM cu două întrerupătoare, întrerupătoarele formează o buclă cu un set nenul de capacități PWM și o secțiune cu un set nenul de bobine PWM.

Prima parte a concluziei rezultă și din proprietatea 5 din capitolul 3.

Consecința acestei teoreme este că din punct de vedere al comutației, *pentru un punct static de funcționare*, pentru orice convertor PWM cu două întrerupătoare, schema echivalentă este cea din Fig. 4.1. denumită *mediul de comutație*.

Sensurile pentru curenții și tensiunile în discuție se aleg astfel încât să obținem numai relații sub formă de sumă, nu și sub formă de diferență, adică:

$$v_{off} = v_{S_1} + v_{S_2} \quad (2)$$

$$i_{on} = i_{S_1} + i_{S_2} \quad (3)$$

unde indicii “off” și “on” se referă la faptul că respectivele mărimi sunt chiar solicitările de tensiune și de curent când unul din întrerupătoare conduce și celălalt este blocat. Acesta este unul din criteriile de identificare a lor în circuite.

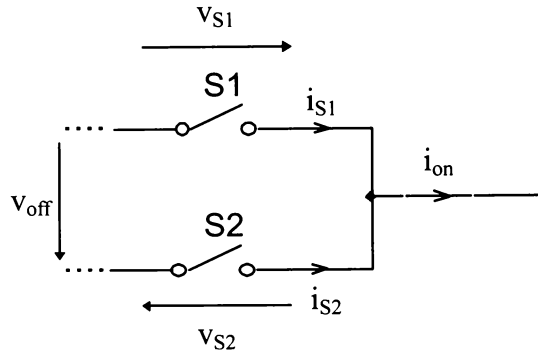


Fig. 4.1. Mediul de comutație într-un convertor PWM dc-dc cu două întrerupătoare și definirea sensurilor pentru mărimile de interes.

Pentru convertoarele PWM cu 2 întrerupătoare în [115] este efectuată o analiză completă a posibilelor moduri de funcționare și sunt date excelente definiții, în termenii cei mai generali posibili, pentru aceste moduri astfel:

- modul de conducție continuă (CCM) este acel mod în care tensiunea  $v_{off}$  și curentul  $i_{on}$  sunt constante pe întreaga durată a unei perioade. O condiție suficientă pentru ca acest mod să aibă loc este ca ipoteza de pulsații reduse să fie respectată de tensiunea pe fiecare capacitate din bucla de capacități ce cuprinde și întrerupătoarele și de curentul prin fiecare bobină a secțiunii ce cuprinde și întrerupătoarele. Succesiunea stărilor într-o perioadă pentru  $v_{off}$  și  $i_{on}$  pozitive este:

$$\dots \rightarrow (1,0) \rightarrow (0,1) \rightarrow \dots,$$

- modul de conducție discontinuă din punct de vedere al curentului inductiv (DICM) se instalează atunci când  $v_{off}$  este constantă pe întreaga durată a unei perioade, dar  $i_{on} = 0$  pe durata unui interval de timp dintr-o perioadă. Ipoteza de pulsații reduse nu este respectată pentru cel puțin o bobină din secțiunea ce cuprinde și întrerupătoarele. Succesiunea stărilor într-o perioadă pentru  $v_{off}$  și  $i_{on}$  pozitive este:

$$\dots \rightarrow (1,0) \rightarrow (0,1) \rightarrow (0,0) \rightarrow \dots,$$

- modul de conducție discontinuă din punct de vedere al tensiunii capacitive (DCVM) are loc atunci când  $i_{on}$  este constant pe întreaga durată a unei perioade, dar  $v_{off} = 0$  pe durata unui interval de timp dintr-o perioadă. Ipoteza de pulsații reduse nu este respectată pentru cel puțin o capacitate din bucla de capacități care cuprinde și întrerupătoarele. Succesiunea stărilor într-o perioadă pentru  $v_{off}$  și  $i_{on}$  pozitive este:

$$\dots \rightarrow (1,0) \rightarrow (1,1) \rightarrow (0,1) \rightarrow \dots,$$

- modul discontinuu cvasirezonaant (“discontinuous quasi-resonant mode” - DQRM) este cel în care nici una dintre mărimi nu-și schimbă semnul și există într-o perioadă de comutație un interval în care  $v_{off} = 0$  și un interval în care  $i_{on} = 0$ . Ipoteza pulsațiilor mici nu este

îndeplinită pentru cel puțin o capacitate din bucla ce cuprinde întrerupătoarele și pentru cel puțin o bobină din secțiunea în care intră și întrerupătoarele. Succesiunea stărilor într-o perioadă pentru  $v_{off}$  și  $i_{on}$  pozitive este:

$$\dots \rightarrow (1,0) \rightarrow (1,1) \rightarrow (0,1) \rightarrow (0,0) \rightarrow \dots,$$

Deși nu se referă explicit la natura întrerupătoarelor, definițiile de mai sus, coroborate cu schema din Fig. 4.1, stabilesc și natura practică a acestora. De exemplu, dacă analizăm modul DICM ipoteza pulsațiilor mici pentru  $v_{off}$  ne permite să admitem că ea este constantă. Indiferent care ar fi polaritatea acestei tensiuni, din schemă rezultă imediat că, pentru sensurile adoptate, produsul  $v_{S_1} \cdot v_{S_2} < 0$ . Pe de altă parte  $i_{on}$  nu poate fi constant decât luând valoarea zero. Într-adevăr, dacă  $i_{on}$  ar fi constant la o valoare nenulă atunci cel puțin  $S_1$  sau  $S_2$  trebuie să fie în conducție pentru a prelua acest curent. Ambele însă nu pot conduce pentru că s-ar scurtcircuita  $v_{off}$ . Înseamnă că doar un întrerupător conduce. Dar conducția unui singur întrerupător se știe că este totdeauna asociată cu o variație a lui  $i_{on}$ , ceea ce contrazice ipoteza constanței sale. Revenind, existența unei porțiuni de anulare în curentul  $i_{on}$  arată de fapt că el este unipolar, pentru că celelalte două intervale topologice sunt asociate cu creșterea și descreșterea sa. Cum curentul  $i_{on}$  trece prin ambele întrerupătoare când acestea conduc, semnul curentului prin ele este același. Deci  $i_{S_1} \cdot i_{S_2} > 0$ . Examinând cele 4 combinații posibile legate de semnele lui  $v_{off}$  și  $i_{on}$  rezultă că în mod corespunzător combinațiile de cadrane pentru  $S_1$  și  $S_2$  sunt: (I, II), (IV, III), (II, I), (III, IV) (prima poziție se referă la  $S_1$ , iar a doua la  $S_2$ ).

Concluzia este că, pentru un punct static de funcționare dat, un întrerupător va fi tranzistor iar celălalt diodă, conectate adecvat. Aceeași concluzie se poate trage și dacă se analizează celelalte moduri. Aceasta nu contrazice felul în care s-au implementat întrerupătoarele pentru unele convertoare (vezi de exemplu varianta de Watkins-Johnson invers) unde acestea apar și de cadrane multiple. Explicația este că alegerea lor s-a făcut având în vedere că factorul de umplere parcurge tot domeniul  $[0, 1]$  permis. Dacă însă se examinează polaritățile concrete pentru un factor de umplere fixat se obține tot o implementare cu un tranzistor și o diodă.

Se notează cu  $\mu$  raportul:

$$\mu = \left| \frac{V_D}{V_{off}} \right| \tag{4}$$

denumit factor de umplere echivalent, unde  $v_D$  reprezintă tensiunea pe diodă (una dintre  $v_{S_1}$  sau  $v_{S_2}$ ). În [115] și [117] sunt date următoarele trei teoreme:

**Teorema 2.** Pentru orice convertor PWM rapoartele  $\left| \frac{V_{S_2}}{V_{off}} \right|$  și  $\left| \frac{V_{S_1}}{V_{off}} \right|$  depind numai de mărimea de comandă  $c$  și de raportul  $\left| \frac{V_{off}}{I_{on}} \right|$ .

**Teorema 3.** În orice convertor PWM cu două întrerupătoare și într-un mod de funcționare,  $\mu$  are aceeași expresie în orice topologie.

**Teorema 4.** În orice convertor PWM, indiferent de modul de funcționare, avem egalitatea:

$$\left| \frac{V_{off}}{I_{on}} \right| = \left| \frac{V_g}{I_o} \right| \tag{5}$$

Deoarece



$$\left| \frac{V_g}{I_o} \right| = \left| \frac{V_g}{\frac{V_o}{R}} \right| = \left| \frac{V_g}{\frac{MV_g}{R}} \right| = \frac{R}{|M|} \quad (6)$$

are loc următoarea

Consecința 1. Rapoartele  $\left| \frac{V_{S1}}{V_{off}} \right|$  și  $\left| \frac{V_{S2}}{V_{off}} \right|$  depind numai de mărimea de comandă  $c$  și de raportul  $\frac{|M|}{R}$ .

Teorema 4. Pentru orice convertor PWM  $\mu$  este același și depinde numai de mărimea de comandă  $c$  și raportul  $\left| \frac{V_{off}}{I_{on}} \right|$ , sau, echivalent, numai de mărimea de comandă  $c$  și de raportul  $\left| \frac{V_g}{I_o} \right|$ , sau de mărimea de comandă și raportul  $\frac{|M|}{R}$ .

Demonstrație. Din definiția dată de (1) rezultă că  $\mu$  este fie  $\left| \frac{V_{S1}}{V_{off}} \right|$ , fie  $\left| \frac{V_{S2}}{V_{off}} \right|$ , în funcție care dintre întrerupătoarele  $S_1$  sau  $S_2$  este implementat practic în convertor cu o diodă. Conform teoremei 2, aceste rapoarte depind numai de mărimea de comandă  $c$  și de raportul  $\left| \frac{V_{off}}{I_{on}} \right|$ , sau conform teoremei 4 și consecinței 1, numai de  $c$  și  $\left| \frac{V_g}{I_o} \right|$ , sau numai de  $c$  și raportul  $\frac{|M|}{R}$ . Pe de altă parte, în virtutea teoremei 3, expresia lui  $\mu$  este aceeași în orice topologie, ceea ce finalizează demonstrația.

În virtutea teoremei 4 strategia de analiză în curent continuu a familiilor de convertoare PWM, dezvoltată de **autorul tezei** după cele expuse de Maksimović în [117] este următoarea:

1. Pentru o celulă dată, deoarece expresia lui  $\mu$  este aceeași pentru orice convertor, se identifică mărimile  $v_{off}$  și  $i_{on}$  și se calculează raportul  $\left| \frac{V_D}{V_{off}} \right|$  pentru o structură de celulă la care întrerupătoarele au fost implementate practic. Aceasta se va face examinând formele de undă corespunzătoare modului analizat.
2. Pentru fiecare convertor analizat se exprimă  $M$  funcție de  $\mu$ . Acest lucru se realizează înlocuind pe  $D$  cu  $\mu$  în dependența funcțională a lui  $M$  de  $D$  în CCM. Apoi se scoate dependența inversă, a lui  $\mu$  în funcție de  $M$ . Acest fapt este în general posibil pentru că la majoritatea convertoarelor dependența lui  $M$  funcție de  $\mu$  în CCM este o funcție rațională, ceea ce conduce la o ecuație algebrică pentru găsirea lui  $\mu$ .
3. Se egalează expresia lui  $\mu$  găsită la pct. 1 cu cea a lui găsită la pct. 2, obținându-se astfel ecuația căutată pentru  $M$ . Cu această ecuație se pot reprezenta și calcula caracteristicile căutate făcându-se, dacă se dorește, normalizări.
4. Pentru că circuitele convertoarelor QRC și CF-QRC sunt identice în curent continuu cu cele ale convertoarelor PWM, metoda de analiză expusă în punctele 1-3 se aplică identic și convertoarelor QRC și CF-QRC.

Să accentuăm că modul în care funcționează convertorul influențează ecuația lui  $M$  prin cele obținute la pasul 1. De asemenea, gradul de generalitate se menține doar până la punctul 1 inclusiv, până la care se operează numai la nivel de mediu generalizat. Abia la punctul 2 intervine specificitatea convertorului în studiu, care de fapt se reflectă prin dependența lui  $M$  de  $\mu$ . De aceea găsirea lui  $\mu$  este decisivă și suficientă pentru a trage concluzii asupra întregii familii de convertoare.

Observația 1. Găsirea lui  $\mu$  la pasul 1 se poate face examinând orice structură particulară de celulă rezultată din celula de bază. De asemenea dependențele lui  $\mu$  de  $M$  pentru structurile particulare de convertoare se calculează o singură dată (pentru CCM), putând apoi fi utilizate la analiza oricărui mod.

Observația 2. Forma dependenței lui  $\mu$  de  $M$  determinată la pct. 1 poate să determine o ecuație algebrică sau una transcendentă pentru  $M$  la pct. 3. De exemplu, dacă dependența calculată la pct. 1 este o funcție rațională și la fel cea de la pct. 2 (ceea ce la convertoarele PWM uzuale are loc întotdeauna), atunci la pct. 3 pentru  $M$  se poate găsi o expresie în formă închisă în funcție de  $R$ .

Observația 3. Algoritmii de mai sus sugerează și modul în care se vor face normalizările. Este clar că atât  $M$  cât și  $\mu$  sunt deja mărimi normalizate, fiind cele de interes. Pentru că  $\mu$  depinde de

$\frac{|M|}{R} = \left| \frac{I_o}{V_g} \right|$  rezultă că dacă se doresc așa numitele *caracteristici de ieșire normalizate ale lui  $\mu$* ,

adică dependențele  $\mu = \mu(I_o \text{ norm})$  la  $V_g = \text{constant}$  și cu  $c$  parametru, atunci se va înlocui în

expresia lui  $\mu$  de la punctul 4 raportul  $\frac{|M|}{R}$  cu  $\left| \frac{I_o}{V_g} \right|$  și se va norma  $|I_o|$  cu  $V_g$  raportat la o

impedanță dată numai de elemente reactive ale celulei. Acest lucru este totdeauna posibil analitic pentru că  $\mu$  este adimensional. Similar, dacă se doresc *caracteristici de ieșire normale*, adică dependențele  $M = M(I_o \text{ norm})$  cu  $V_g = \text{constant}$  și cu  $c$  parametru, atunci în expresia

$M = M(c, R)$  se scrie  $R = \frac{V_o}{I_o} = \frac{MV_g}{I_o}$  și din aceasta se poate scoate  $M$  în funcție de  $I_o$ . Din nou

normarea se va face cu  $V_g$  raportat la o impedanță dată de elementele reactive ale celulei.

Dacă însă  $M$  este dat de o ecuație transcendentă atunci caracteristicile se vor trasa și calcula rezolvând punctual ecuația prin metode numerice.

Din motive de spațiu nu vom aplica metoda unitară de analiză prezentată pentru toate celulele studiate în capitolele precedente ci vom da câteva exemple semnificative.

Noțiunea de conducție discontinuă se poate generaliza pentru un convertor PWM cu oricâte întrerupătoare în felul următor [118]:

Definiția 1. Un mod de funcționare al unui convertor oarecare se numește *discontinuu* dacă comutarea (aducerea în conducție sau blocarea) unui *întrerupător pasiv* (diode) din convertor *nu este sincronă* cu comutarea (aducerea în conducție sau blocarea) unui *întrerupător activ* (tranzistor).

Reamintim că diferitele moduri de funcționare în care poate evolua un convertor sunt date de:

- punctul static de funcționare, care influențează modalitatea concretă în care se materializează practic întrerupătoarele.
- ipotezele privitoare la natura pulsațiilor mărimilor de stare aferente mediului de comutație.

## 4.2 Analiza unitară a convertoarelor PWM

Pentru modurile CCM și DICM se consideră ca și studiu de caz celula BBB particularizată, în care  $S_1$  este un întrerupător de cadranul I, adică un tranzistor bipolar sau MOS, iar  $S_2$  un întrerupător de cadranul II, adică o diodă - Fig. 4.2 - implementarea pentru convertorul BUCK, unde se observă că  $v_{off} > 0$ ,  $i_{on} > 0$ . Așa cum s-a precizat,  $D$  este factorul de umplere aferent întrerupătorului  $S_1$ . Se mai notează cu  $D_2$  factorul de umplere al întrerupătorului  $S_2$ . Pentru convertoarele rezultate din celula BBB expresiile lui  $M$  funcție de  $\mu$  și  $\mu$  funcție de  $M$  sunt date în Tab 4.1.

Tab. 4.1.

CONVERTORUL	$M = M(\mu)$	$\mu = \mu(M)$
BUCK	$\mu$	$M$
BOOST	$\frac{1}{1-\mu}$	$-\frac{1-M}{M}$
BUCK-BOOST	$-\frac{\mu}{1-\mu}$	$-\frac{M}{1-M}$

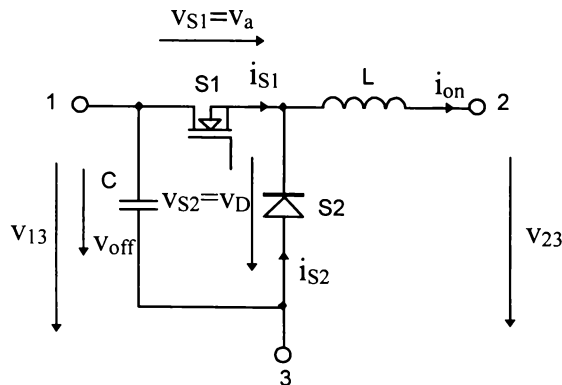


Fig. 4.2. Celula BBB cu întrerupătoarele materializate și specificarea sensurilor mărimilor de interes.

### 4.2.1 Regimul de conducție continuă (CCM)

În acest regim formele de undă sunt cele din Fig. 4.3. Se constată fără dificultate că:

$$V_D = DV_{off} \quad (7)$$

deci

$$\mu = D \quad (8)$$

expresiile raportului static de conversie fiind cele cunoscute. Se remarcă faptul că  $\mu$  nu depinde de

raportul  $\frac{V_{off}}{I_{on}}$ , deci nici de  $\frac{M}{R}$ . Deci rezistența de ieșire a convertoarelor în acest mod va fi zero.

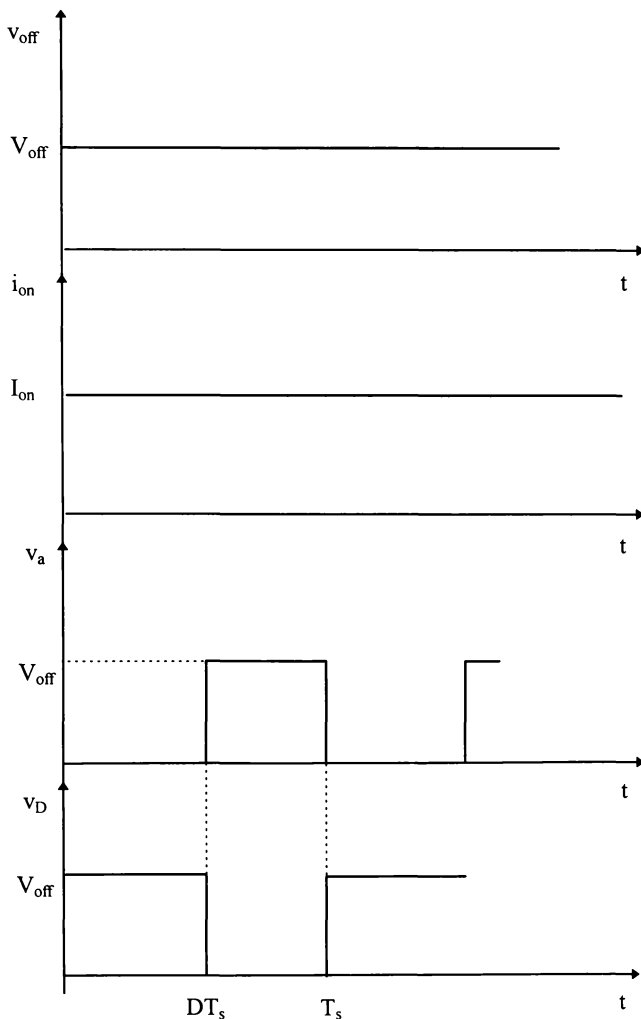


Fig. 4.3. Principalele forme de undă în CCM.

#### 4.2.2 Regimul de conducție discontinuă din punct de vedere al curentului inductiv (DICM)

##### 4.2.2.1 Regimul DICM pentru convertoare PWM cu două întrerupătoare

Formele de undă caracteristice sunt cele din Fig. 4.4, unde  $v_a$  este tensiunea pe întrerupătorul activ.

Strategia propusă de autor pentru a exprima  $\mu$  funcție de  $D$  și raportul  $\frac{V_{off}}{I_{on}}$  este următoarea:

- se exprimă valoarea medie a tensiunii pe diodă din examinarea formei de undă corespunzătoare și se calculează raportul  $\frac{V_D}{V_{off}}$ .
- se scrie că valoarea medie a tensiunii pe bobină este nulă.
- se exprimă valoarea medie,  $I_{on}$ , din graficul acestei mărimi și din funcționarea circuitului.

Setul de 3 ecuații care se obține este:

$$V_D = DV_{off} + (1 - D - D_2)(V_{off} - V_{23}) \quad (9)$$

$$D(V_{off} - V_{23}) + D_2(-V_{23}) = 0 \quad (10)$$

$$I_{on} = \frac{1}{2} \frac{V_{off} - V_{23}}{L} D(D + D_2) T_s \quad (11)$$

Împărțind ecuația (9) cu  $V_{off}$  pentru a pune în evidență raportul  $\frac{V_D}{V_{off}}$  căutat, se obține:

$$\frac{V_D}{V_{off}} = D + (1 - D - D_2) \left(1 - \frac{V_{23}}{V_{off}}\right) \quad (12)$$

ceea ce arată că trebuie căutate exprimări ale lui  $D_2$  și  $\frac{V_{23}}{V_{off}}$  în funcție de  $\frac{V_{off}}{I_{on}}$ .

Acest lucru se obține din ecuațiile (10) și (11), găsindu-se:

$$D_2 = \frac{1}{D} \frac{I_{on}}{V_{off}} \frac{2L}{T_s} \quad (13)$$

$$\frac{V_{23}}{V_{off}} = \frac{D^2}{D^2 + \frac{I_{on}}{V_{off}} \frac{2L}{T_s}} \quad (14)$$

Înlocuind (13) și (14) în (12) rezultă:

$$\mu = \left| \frac{V_D}{V_{off}} \right| = \frac{D^2}{D^2 + \left| \frac{I_{on}}{V_{off}} \right| 2Lf_s} \quad (15)$$

sau

$$\mu = \frac{D^2}{D^2 + \frac{|M|}{R} 2Lf_s} \quad (16)$$

Graficele din Fig. 4.4 sugerează și condiția de funcționare în DICM și anume ca intervalul de anulare al curentului să existe și să fie mai mic decât perioada:

$$1 \geq 1 - D - D_2 \geq 0 \quad (17)$$

Prima inegalitate este întotdeauna adevărată, în timp ce folosind (17) și (21) a doua conduce la:

$$\mu \geq D \quad (18)$$

Dacă se dorește expresia analitică a caracteristicii de ieșire normale a lui  $\mu$ , folosind (15) găsim:

$$\mu = \frac{D^2}{D^2 + \left| \frac{I_{on}}{V_{off}} \right| 2Lf_s} = \frac{D^2}{D^2 + \frac{|I_o|}{V_g} 2Lf_s} = \frac{D^2}{D^2 + \frac{|I_o|}{I_\alpha}} = \frac{D^2}{D^2 + I_{onorm}} \quad (19)$$

unde  $I_{onorm}$  este curentul de ieșire normalizat prin curentul

$$I_{\alpha} = \frac{V_g}{2Lf_s} \tag{20}$$

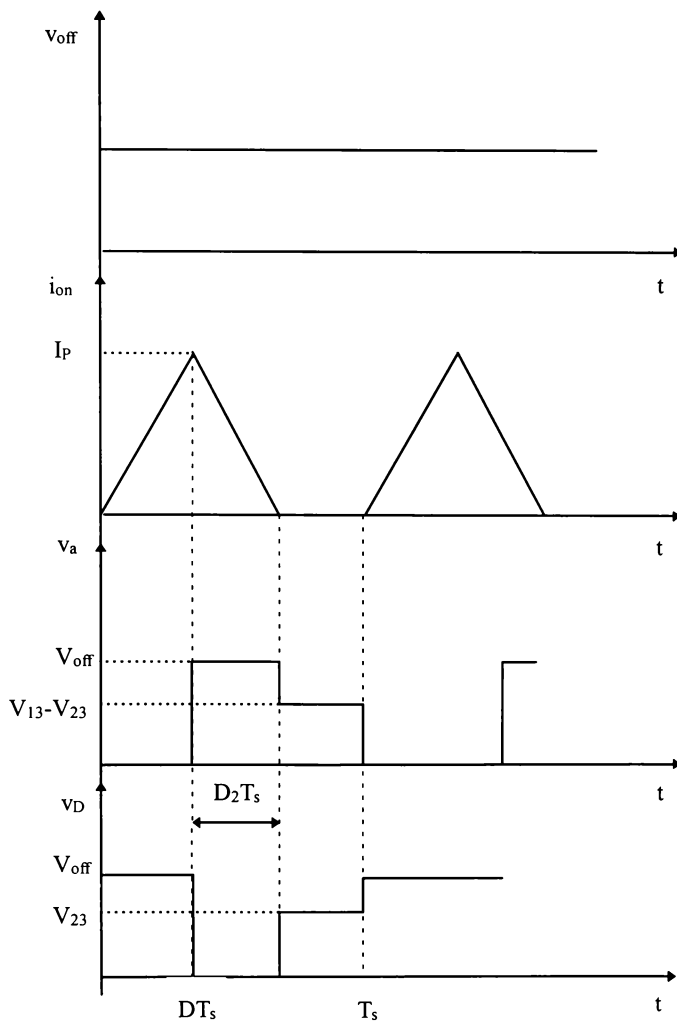


Fig. 4.4. Principalele forme de undă în DICM.

Relația (19) arată că  $\mu$  scade cu  $I_{onorm}$ . Caracteristicile de ieșire normate ale lui  $\mu$  în CCM și DICM sunt prezentate în Fig. 4.5, programul MATLAB de trasare fiind dat în Anexa A4. Cum în modul toate cele 3 funcții din coloana a doua din Tab. 4.1, care dau dependența lui  $M$  de  $\mu$  pentru convertoarele familiei, sunt strict crescătoare în  $\mu$ , rezultă caracterul căzător al caracteristicilor de ieșire. Dacă se notează

$$k_i = \frac{2Lf_s}{R} \tag{21}$$

în acest caz din (16)  $\mu$  se poate scrie și:

$$\mu = \frac{D^2}{D^2 + k_i |M|} \quad (22)$$

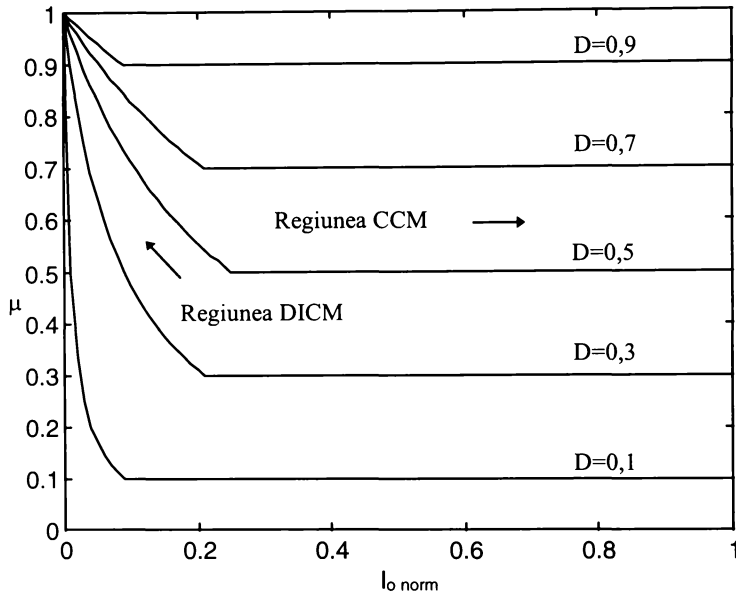


Fig. 4.5. Caracteristicile de ieșire normalizate ale lui  $\mu$  în CCM și DICM.

Fie  $\mu = \mu(M)$  dependența lui  $\mu$  de  $M$  a unui convertor oarecare din familie (vezi a treia coloană din Tab. 4.1). Atunci din (22) se obține ecuația pentru găsirea lui  $M$  la factor de umplere, frecvență, rezistență de sarcină și inductivitate fixată:

$$\mu(M) = \frac{D^2}{D^2 + k_i |M|} \quad (23)$$

precum și factorul de umplere necesar pentru a avea un raport de conversie dorit la frecvență, sarcină și bobină precizate:

$$D = \sqrt{\frac{k_i |M| \mu(M)}{1 - \mu(M)}} \quad (24)$$

Funcționarea acestei familii de convertoare în DCVM nu prezintă interes deoarece pentru această familie  $v_{off}$  este chiar tensiunea de la o pereche de borne ale celulei. Cum modul DCVM reclamă înlăturarea ipotezei de pulsații reduse în  $v_{off}$  ar însemna că și tensiunea de ieșire va avea pulsații semnificative. Deși posibil, bineînțeles că acest lucru nu este de dorit în practică.

#### 4.2.2.2 Regimul DICM pentru convertoare pătratic

Convertoarele pătratic conțin un tranzistor și trei diode, aplicându-se definiția generalizată de conducție discontinuă care se asociază cu oricare dintre diode. Pentru că ne referim la DICM presupunem că pulsațiile curenților prin bobine sunt acum semnificative, în timp ce tensiunile pe

capacități sunt cvasiconstante. Similar cu notațiile introduse la convertoarele studiate la paragraful 4.2.2.1 se definesc parametrii  $k_{i1}$  și  $k_{i2}$  prin relațiile:

$$k_{i1} = \frac{2L_1 f_s}{R} \qquad k_{i2} = \frac{2L f_s}{R} \qquad (25)$$

**Autorul tezei** propune mai jos o analiză unitară nu studiind convertoarele separat, ci direct pe celula de bază completă din care provin. Fie de exemplu celula  $\beta_2$  din Fig. 4.6. Pentru toate convertoarele din familie  $S_1$  este tranzistor, notat  $T_1$ , iar  $S_1', S_2, S_2'$  sunt diode, notate  $D_1', D_2$  respectiv  $D_2'$ .

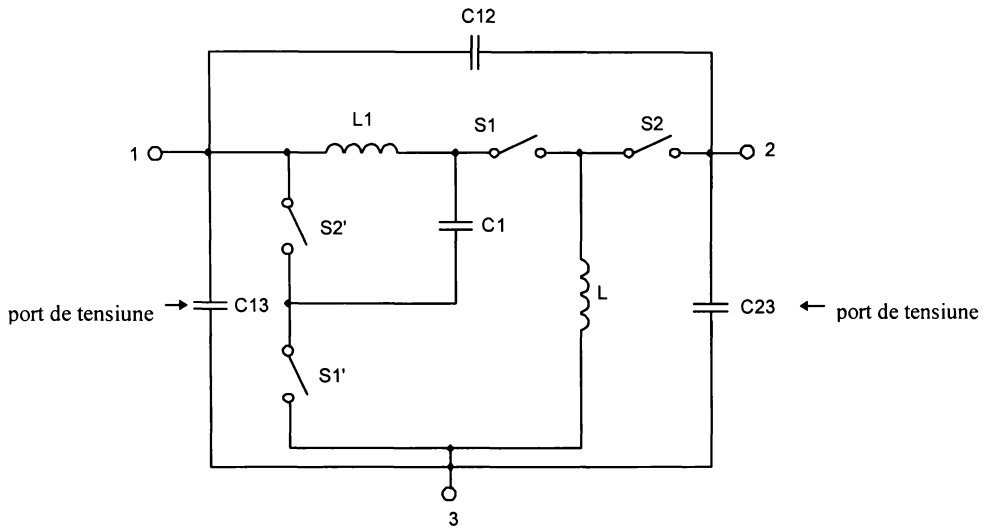


Fig. 4.6. Celula de bază  $\beta_2$ .

Se aleg sensuri pentru curenții prin bobine astfel încât aceștia să rezulte pozitivi. Moduri DICM ordinară sunt legate de diodele  $D_2$  și  $D_2'$ . De exemplu când tranzistorul este blocat  $D_2$  este parcursă de curentul  $i_L$ . Curentul prin diodă are o variație liniară și pentru funcționare CCM valoarea minimă a curentului  $i_L$  trebuie să fie pozitivă:

$$I_L - \frac{\Delta I_L}{2} > 0 \qquad (26)$$

unde  $I_L$  și  $\Delta I_L$  reprezintă valoarea medie respectiv valoarea vârf la vârf a pulsațiilor curentului prin bobina  $L$ . Similar, pentru că  $D_2'$  este parcursă de curentul  $i_{L_1}$ , condiția de funcționare CCM pentru această diodă este:

$$I_{L_1} - \frac{\Delta I_{L_1}}{2} > 0 \qquad (27)$$

O situație diferită apare pentru dioda  $D_1'$ . Atunci când conduce ea este parcursă de curentul  $i_L - i_{L_1}$ . Pentru convertoarele generate cu celula  $\beta_2$ ,  $i_L$  și  $i_{L_1}$  au aceeași monotonie. De aceea condiția necesară și suficientă pentru funcționare CCM din punct de vedere al diodei  $D_1'$  este:

$$(i_L - i_{L_1})_{\min} > 0 \qquad (i_L - i_{L_1})_{\max} > 0 \qquad (28)$$



scrisă și în forma:

$$(I_L - I_{L_1}) - \left(\frac{\Delta I_L}{2} - \frac{\Delta I_{L_1}}{2}\right) > 0 \quad (I_L - I_{L_1}) + \left(\frac{\Delta I_L}{2} - \frac{\Delta I_{L_1}}{2}\right) > 0 \quad (29)$$

sau

$$\frac{\Delta I_L}{2} - \frac{\Delta I_{L_1}}{2} < I_L - I_{L_1} \quad \frac{\Delta I_{L_1}}{2} - \frac{\Delta I_L}{2} < I_L - I_{L_1} \quad (30)$$

Ținând cont că pentru celula  $\beta_2$  există relația:

$$I_{L_1} = D I_L \quad (31)$$

care se poate demonstra fără dificultate, (30) are forma echivalentă

$$\left| \frac{\Delta I_{L_1}}{2} - \frac{\Delta I_L}{2} \right| < (1-D) I_L \quad (32)$$

Pulsățiile vârf la vârf ale curenților prin bobine se exprimă numai în funcție de tensiunea  $V_{13}$  de intrare în celulă, factorul de umplere și mărimile de circuit:

$$\Delta I_L = D^2 \frac{|V_{13}|}{L f_s} \quad \Delta I_{L_1} = D(1-D) \frac{|V_{13}|}{L_1 f_s} \quad (33)$$

Relațiile (33) înlocuite în (26), (27) și (32) și folosind definițiile (25) duc la următoarele relații generale pentru funcționare CCM:

$$k_{i2} > D^2 \frac{|V_{13}|}{I_L R} \quad k_{i1} > (1-D) \frac{|V_{13}|}{I_L R} \quad \left| \frac{1}{k_{i1}} - \frac{D}{1-D} \frac{1}{k_{i2}} \right| < \frac{1}{D} \frac{I_L R}{|V_{13}|} \quad (34)$$

Examinând (34) rezultă că trebuie calculat raportul  $\frac{|V_{13}|}{I_L R}$ . Acest lucru se realizează pe topologia particulară analizată exprimând  $V_{13}$  și  $I_L$  numai în funcție de  $V_g$ ,  $D$  și  $R$ . De exemplu pentru convertorul  $\beta_2$ -BOOST:

$$V_{13} = -V_g \quad I_L = \frac{1}{1-D} I_{D_2} = \frac{1}{1-D} (I_I - I_o) = \frac{1}{1-D} (1-M) M \frac{V_g}{R} = \frac{D^2}{(D^2 - D + 1)^2} \frac{V_g}{R} \quad (35)$$

obținându-se

$$\frac{|V_{13}|}{I_L R} = \frac{(1-D)(D^2 - D + 1)}{D^2} \quad (36)$$

Înlocuind (36) în (34) se obțin relațiile finale necesare pentru funcționarea CCM:

$$k_{i1} > \frac{D^2(1-D)^2}{D^2 - D + 1}, \quad k_{i2} > \frac{D^4(1-D)}{D^2 - D + 1}, \quad \left| \frac{1}{k_{i1}} - \frac{D}{1-D} \frac{1}{k_{i2}} \right| < \frac{D^2 - D + 1}{D^3(1-D)} \quad (37)$$

Analiza pentru toate celelalte celule de conversie pătratice a fost efectuată de **autorul tezei** în mod similar, rezultatele fiind sintetizate în Tab. 4.2. De notat că aceste condiții pot fi mult diferite chiar pentru convertoare care au același raport static de conversie în CCM.

Tab. 4.2. Condițiile de funcționare în conducție continuă pentru convertoarele pătratice.

CONVERTORUL	CONDIȚIILE PENTRU FUNCȚIONARE CCM
$\alpha_1$ -BUCK	$k_{i1} > \frac{1-D}{D^2}$ , $k_{i2} > 1-D$ , $\left  \frac{1}{D k_{i1}} - \frac{1}{k_{i2}} \right  < 1$
$\alpha_1$ -BOOST	$k_{i1} > (1-D)(1-D^2)$ , $k_{i2} > D^2(1-D)(1-D^2)$ ,

	$\left  \frac{1}{Dk_{i1}} - \frac{1}{k_{i2}} \right  < \frac{1}{D^2(1-D^2)}$
$\alpha_1$ -BUCK-BOOST	$k_{i1} > \frac{(1-D)(1-D^2)}{D^2}$ , $k_{i2} > (1-D)(1-D^2)$ , $\left  \frac{1}{Dk_{i1}} - \frac{1}{k_{i2}} \right  < \frac{1}{(1-D^2)}$
$\alpha_2$ -BUCK	$k_{i1} > \frac{1}{D}$ , $\frac{1+D}{k_{i2}} + \frac{1}{k_{i1}} < \frac{D^2}{1-D}$
$\alpha_2$ -BOOST	$k_{i1} > D(1-D^2)$ , $\frac{1+D}{k_{i2}} + \frac{1}{k_{i1}} < \frac{1}{(1-D)(1-D^2)}$
$\alpha_2$ -BUCK-BOOST	$k_{i1} > \frac{1-D^2}{D}$ , $\frac{1+D}{k_{i2}} + \frac{1}{k_{i1}} < \frac{D^2}{(1-D)(1-D^2)}$
$\beta_1$ -BUCK	$k_{i1} > \frac{(1-D)(D^2-D+1)}{D^2}$ , $k_{i2} > D^2-D+1$
$\beta_1$ -BOOST	$k_{i1} > \frac{(1-D)^2}{D^2-D+1}$ , $k_{i2} > \frac{D^2(1-D)}{D^2-D+1}$
$\beta_1$ -BUCK-BOOST	$k_{i1} > \frac{(1-D)^2}{D^2}$ , $k_{i2} > 1-D$
$\beta_2$ -BUCK	$k_{i1} > \frac{(1-D)^2(D^2-D+1)}{D^2}$ , $k_{i2} > (1-D)(D^2-D+1)$ , $\left  \frac{1}{k_{i1}} - \frac{D}{1-D} \frac{1}{k_{i2}} \right  < \frac{D}{(1-D)(D^2-D+1)}$
$\beta_2$ -BOOST	$k_{i1} > \frac{D^2(1-D)^2}{D^2-D+1}$ , $k_{i2} > \frac{D^4(1-D)}{D^2-D+1}$ , $\left  \frac{1}{k_{i1}} - \frac{D}{1-D} \frac{1}{k_{i2}} \right  < \frac{D^2-D+1}{D^3(1-D)}$
$\beta_2$ -BUCK-BOOST	$k_{i1} > \frac{(1-D)^3}{D^2}$ , $k_{i2} > (1-D)^2$ , $\left  \frac{1}{k_{i1}} - \frac{D}{1-D} \frac{1}{k_{i2}} \right  < \frac{D}{(1-D)^2}$
$\beta_3$ -BUCK	$k_{i2} > \frac{(1-D)(D^2-D+1)}{D}$ , $\frac{1}{k_{i1}} + \frac{1-D}{k_{i2}} < \frac{D^2}{(1-D)(D^2-D+1)}$
$\beta_3$ -BOOST	$k_{i2} > \frac{D(1-D)^2}{D^2-D+1}$ , $\frac{1}{k_{i1}} + \frac{1-D}{k_{i2}} < \frac{D^2-D+1}{(1-D)^2}$
$\beta_3$ -BUCK-BOOST	$k_{i2} > \frac{(1-D)^2}{D}$ , $\frac{1}{k_{i1}} + \frac{1-D}{k_{i2}} < \frac{D^2}{(1-D)^2}$
$\delta_1$ -BUCK	$k_{i1} > \frac{1}{2-D}$ , $k_{i2} > \frac{(1-D)^2}{2-D}$
$\delta_1$ -BOOST	$k_{i1} > D(1-D)^2$ , $k_{i2} > D(1-D)^4$
$\delta_1$ -BUCK-BOOST	$k_{i1} > \frac{(1-D)^2}{2-D}$ , $k_{i2} > \frac{(1-D)^4}{2-D}$

$\gamma_1$ -BUCK	$k_{i1} > \frac{(1-D)^4}{D^2}, \quad k_{i2} > (1-D)^2, \quad D \leq 0,5$
$\gamma_1$ -BOOST	$k_{i1} > 1-2D, \quad k_{i2} > \frac{D^2(1-2D)}{(1-D)^2}, \quad D \leq 0,5$
$\gamma_1$ -BUCK-BOOST	$k_{i1} > \frac{(1-D)^2(1-2D)}{D^2}, \quad k_{i2} > 1-2D, \quad D \leq 0,5$

Fie de exemplu convertoarele  $\alpha_1$ -BUCK și  $\alpha_2$ -BUCK care au același raport de conversie  $M(D) = D^2$ . Se presupun aceleași condiții de funcționare și un factor de umplere de valoare 0,5 pentru a avea curenți medii prin bobine identici. Dorim să vedem în care convertor sunt necesare inductivități mai mici pentru funcționarea CCM. Din Tab. 4.2 pentru convertorul  $\alpha_1$ -BUCK condițiile de funcționare CCM sunt:

$$k_{i1} > 2 \quad k_{i2} > 2 \quad \left| \frac{1}{k_{i2}} - \frac{2}{k_{i1}} \right| < 1 \quad (38)$$

Valoarea minimă a energiei stocate, sau, echivalent, valorile minime ale bobinelor se realizează dacă suma  $k_{i1} + k_{i2}$  este minimă cu respectarea condițiilor (32). Acest lucru are loc la limită pentru  $k_{i1} = k_{i2} = 2$ , deci

$$k_{i1} + k_{i2} = 4 \quad (39)$$

Pentru convertorul  $\alpha_2$ -BUCK condițiile funcționării CCM sunt:

$$k_{i1} > 2 \quad \frac{2}{k_{i1}} + \frac{3}{k_{i2}} < 1 \quad (40)$$

pentru care soluția  $k_{i1} = 4,45$ ,  $k_{i2} = 5,45$  duce la:

$$k_{i1} + k_{i2} = 9,9 \quad (41)$$

deci pentru convertorul  $\alpha_2$ -BUCK sunt necesare inductivități de peste două ori mai mari decât la convertorul  $\alpha_1$ -BUCK.

#### 4.2.3. Regimul de conducție discontinuă din punct de vedere al tensiunii capacitive

Studiul regimului DCVM îl vom face pe o nouă clasă de convertoare introduse de o manieră unificată și la modul cel mai general posibil [Lascu, 74]. Ideea de la care s-a pornit nu este aceea de a introduce o nouă celulă de bază ci de a modifica toate celulele de bază prin *modificarea mediului de comutație* din Fig. 4.1 ca în Fig. 4.6. Practic se introduc două elemente noi,  $L_i$  și  $C_i$  unde indicele "i" derivă de la faptul că acestea sunt elementele nou introduse. În Fig. 4.7  $v'_{off}$  este aceeași cu tensiunea  $v_{off}$  din Fig. 4.1. Se observă că în noul mediu de comutație bucla care conține capacități PWM și întrerupătoare este impusă de modificarea făcută, fiind formată din  $C_i - S_1 - S_2$ , în timp ce secțiunea ce cuprinde întrerupătoarele și bobinele rămâne aceeași, deci  $i_{on}$  este același. În schimb noua tensiune  $v_{off}$  devine tensiunea de pe  $C_i$ .

Toate elementele reactive din mediul de comutație original respectă ipoteza pulsațiilor mici. Bobina  $L_i$  se alege suficient de mare pentru ca ipoteza pulsațiilor mici de curent să fie și pentru ea respectată. În analiză se poate deci admite că  $i_{L_i}$  este constant, cum de altfel sunt și  $i_{on}$  și  $v'_{off}$ , motiv pentru care vom folosi direct majuscule în desemnarea lor în continuare. În schimb  $v_{off}$  va avea o porțiune de anulare dacă  $C_i$  se alege suficient de mic.

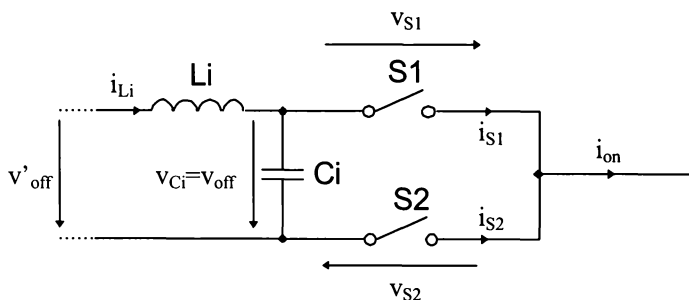


Fig. 4.7. Modificarea mediului de comutație prin introducerea unei capacități și a unei bobine.

Se impune a se sublinia că *modificarea mediului de comutație* este echivalentă cu modificarea tuturor convertoarelor PWM. Este *cel mai general mod de a găsi noi convertoare*.

O posibilă implementare practică a mediului de comutație, corespunzătoare lui  $v_{off} > 0$  și  $i_{on} > 0$  este dată în Fig. 4.8.

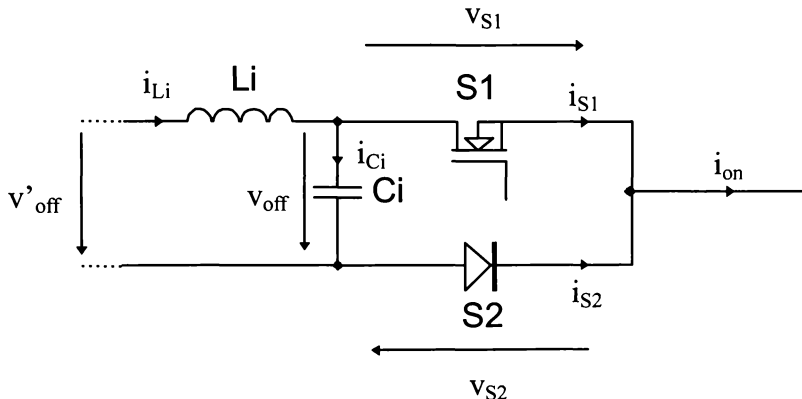


Fig. 4.8. O posibilă implementare practică a întrerupătoarelor din mediul de comutație modificat pentru generarea convertoarelor DIVM

Principalele forme de undă sunt redată în Fig. 4.9. Din Fig. 4.7 este evident că  $I_{L_i} = I_{S_1}$ , valoarea medie a curentului prin  $C_i$  fiind bineînțeles zero. Cum în valori medii  $I_{S_1} + I_{S_2} = I_{on}$  și toate sensurile mărimilor au fost alese astfel încât ele să fie pozitive, rezultă imediat că

$$I_{L_i} < I_{on} \tag{42}$$

Acest fapt ajută la explicarea formelor de undă din Fig. 4.8. La începutul perioadei, când  $S_1$  este închis, curentul prin condensator este:

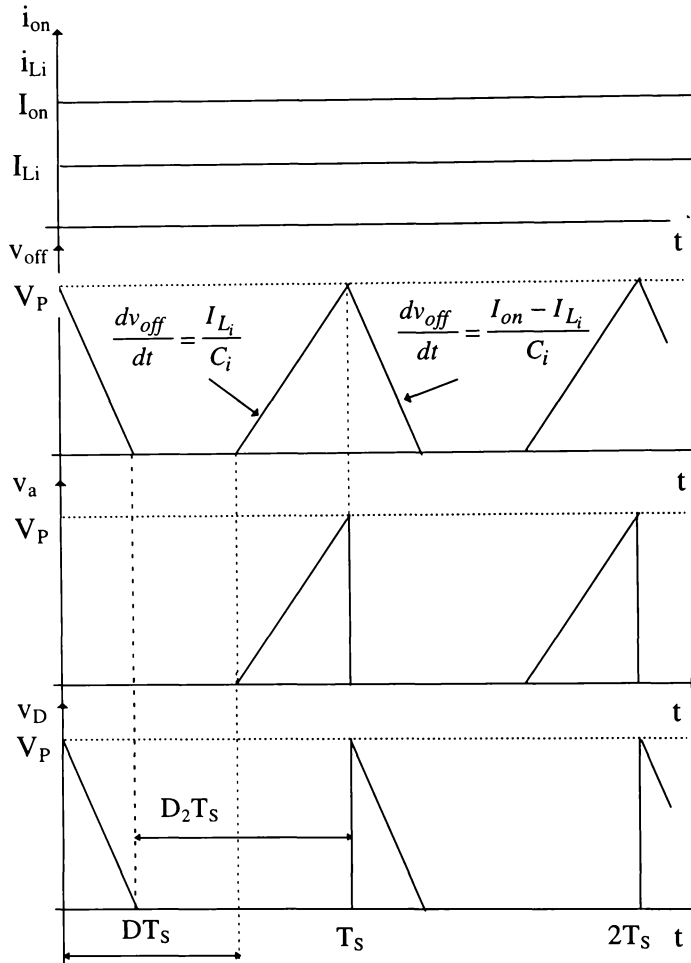


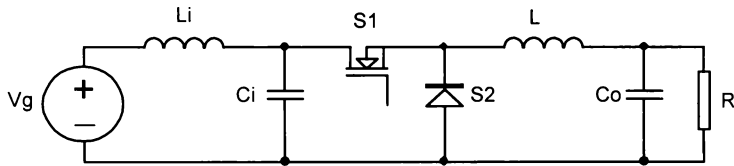
Fig. 4.9. Principalele forme de undă ale mediului de comutație specifice convertoarelor DIVM.

$$i_{C_i} = I_{L_i} - I_{on} < 0 \quad (43)$$

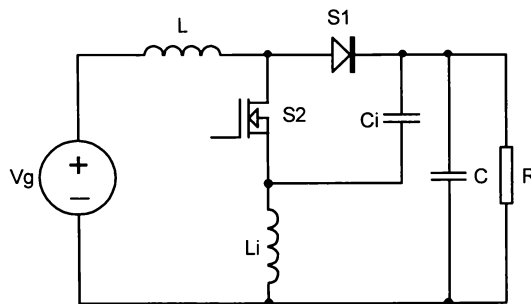
valoarea negativă justificându-se din (42). Condensatorul  $C_i$  se descarcă deci liniar. În tot acest timp tensiunea de pe condensator se aplică și diodei. În momentul în care tensiunea pe condensator se anulează și tinde să devină negativă dioda este polarizată direct și intră în conducție.  $S_1$  și  $S_2$  conduc simultan și tensiunea pe condensator rămâne zero. În această stare topologică prin  $S_1$  trece  $i_{L_i}$ , iar prin diodă  $i_{on} - i_{L_i}$ . După  $D_2 T_s$  unități de timp de la începutul perioadei  $S_1$  este blocat din exterior și în consecință tot curentul  $i_{L_i}$  trece acum prin  $C_i$  care este încărcat liniar. Prin diodă trece curentul  $i_{on}$ , dioda conducând până la începutul perioadei următoare, când din nou  $S_1$  este adus în conducție și fenomenele se repetă. Forma tipică a tensiunii  $v_{off}$ , care acum este tensiunea pe o capacitate ( $C_i$ ) și nu este cvasicontinuă a dat denumirea modului de funcționare a convertoarelor rezultante: modul de conducție discontinuă din punct de vedere al tensiunii de

intrare (“discontinuous input voltage mode” - DIVM), tensiunea de pe  $C_i$  fiind privită ca tensiune de intrare în bucla ce conține întrerupătoarele.

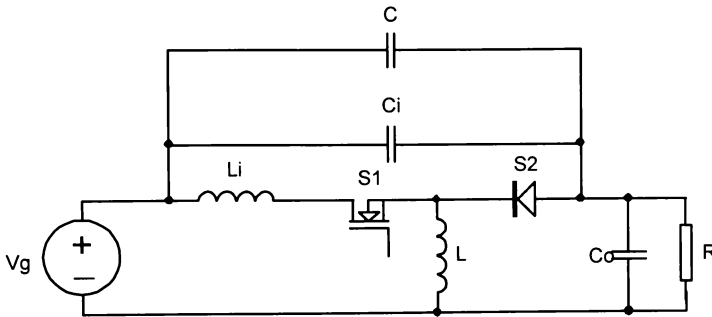
Principalele convertoare DIVM sunt prezentate în Fig. 4.10. Întrerupătoarele au fost notate ca în celulele de bază.



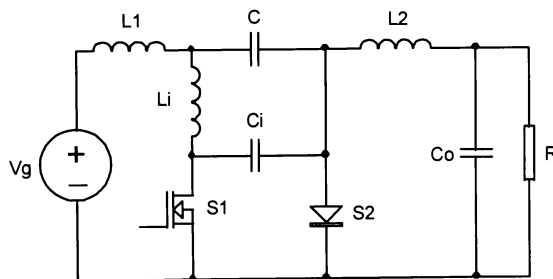
(a) BUCK



(b) BOOST



(c) BUCK-BOOST



(d) ĆUK

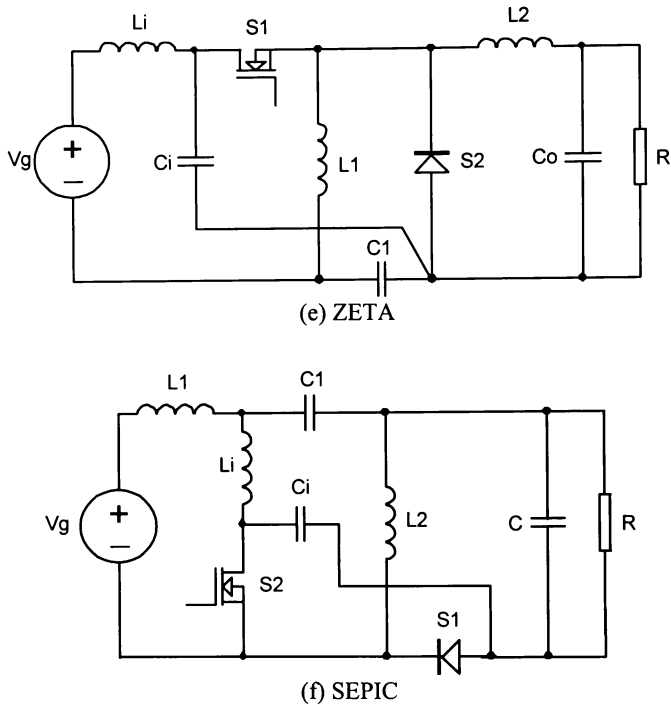


Fig. 4.10. Principalele convertoare de bază DIVM.

Pentru analiză fie celula ĆUK DIVM materializată ca în Fig. 4.11. Pentru rezolvarea modului DICM se procedează astfel (în fond procedura se obține prin dualitate din strategia de rezolvare a modului DIVM, expusă în paragraful 4.2.2.1):

- se exprimă valoarea medie a tensiunii pe diodă din examinarea formei de undă corespunzătoare și se calculează  $\frac{V_D}{V_{off}}$ .
- se egalează cu zero valoarea medie a curentului prin capacitatea  $C_i$ .
- se exprimă valoarea medie,  $V_{off}$ , din graficul acestei tensiuni numai în funcție de curenți.

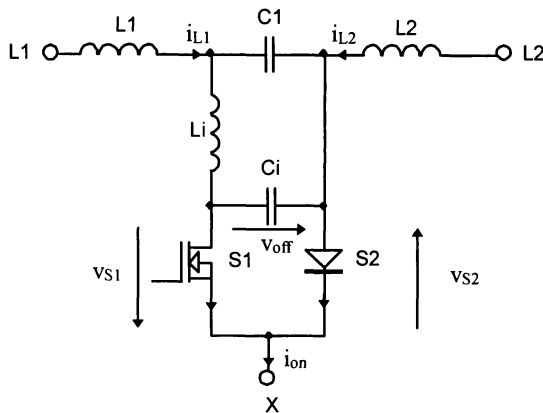


Fig. 4.11. Celula ĆUK DIVM.

Se obțin astfel ecuațiile:

$$V_D = \frac{1}{2C_i f_s} (1-D_2)(1-D) \frac{I_{L_i}}{V_{off}} \quad (44)$$

$$(1-D_2)(I_{L_i} - I_{on}) + (1-D)I_{L_i} = 0 \quad (45)$$

$$V_{off} = \frac{1}{2C_i f_s} (1-D)(2-D-D_2)I_{L_i} \quad (46)$$

de unde prin rezolvare se găsește:

$$\mu = \frac{2C_i f_s \left| \frac{V_{off}}{I_{on}} \right|}{2C_i f_s \left| \frac{V_{off}}{I_{on}} \right| + (1-D)^2} \quad (47)$$

sau, în alte forme echivalente,

$$\mu = \frac{2C_i f_s \frac{|M|}{R}}{2C_i f_s \frac{|M|}{R} + (1-D)^2} \quad (48)$$

$$\mu = \frac{\frac{I_\beta}{|I_o|}}{\frac{I_\beta}{|I_o|} + (1-D)^2} = \frac{1}{1 + (1-D)^2 I_o \text{ norm}} \quad (49)$$

$$I_o \text{ norm} = \frac{|I_o|}{I_\beta} \quad (50)$$

$$I_\beta = 2V_g C_i f_s \quad (51)$$

Caracteristicile de ieșire normate ale lui  $\mu$  în CCM și DIVM sunt înfățișate în Fig. 4.12 arătând că în DIVM caracteristicile de ieșire ale convertoarelor sunt căzătoare, dat fiind că  $|M|$  este strict crescător cu  $\mu$ . Programul MATLAB de trasare se găsește în anexa A5.

Condiția de funcționare DIVM se găsește examinând Fig. 4.8 și este:



$$0 \leq D_2 - (1 - D) \leq 1 \quad (52)$$

ceea ce conduce la:

$$\mu \leq D \quad (53)$$

Pentru convertoarele prezentate în Tab. 4.3 se dau expresiile raportului static de conversie, condiția individuală de funcționare DIVM și solicitările în tensiune ale tranzistorului și diodei, [Lascu, 74] introducându-se parametrul

$$k_v = 2C_i R f_s \quad (54)$$

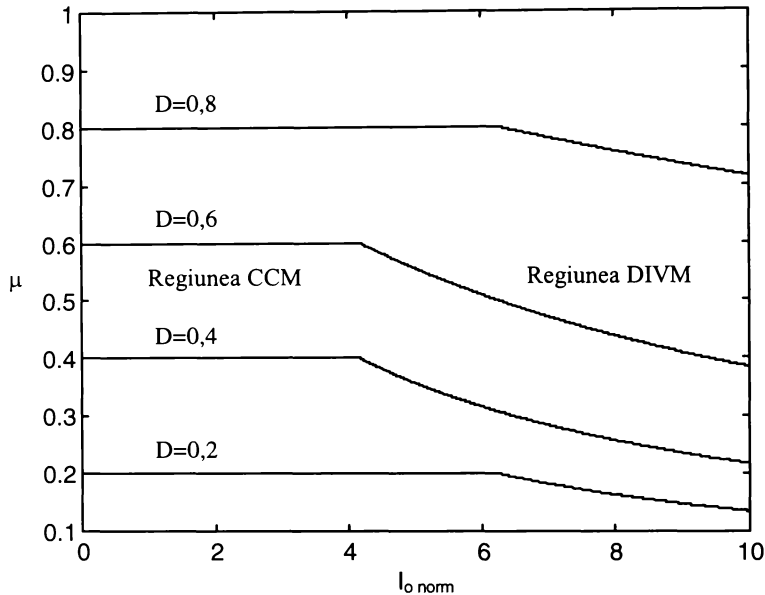


Fig. 4.12. Caracteristicile de ieșire normale ale lui  $\mu$  în CCM și DIVM.

Examinarea funcționării tuturor acestor convertoare arată că, în ciuda faptului că se utilizează două elemente de circuit adiționale ( $L_i$  și  $C_i$ ), o valoare ușor crescută pentru  $L_i$  și a unor solicitări de tensiune mai mari decât în CCM, familia de convertoare DIVM prezintă și o serie de avantaje:

- pierderi mici la blocarea tranzistorului întrucât aceasta se produce la tensiune zero, ceea ce permite creșterea frecvenței de lucru și deci greutatea și gabaritul mai mici pentru elementele reactive.
- funcționarea la frecvență constantă, deci o utilizare optimă a dispozitivelor magnetice.
- curent de intrare cu pulsații foarte mici față de convertoarele corespunzătoare PWM clasice (cu excepția convertorului BOOST unde pulsațiile sunt aceleași), deci proprietăți EMI superioare.
- solicitările în tensiune ale componentelor nu depind de sarcină (cu excepția convertorului BUCK unde însă dependența este slabă)
- este posibilă o analiză unitară atât la semnal mare cât și la semnal mic.
- se pretează a fi utilizate ca și circuite de corecție a factorului de putere. O astfel de aplicație este dată în capitolul 7, paragraful 7.2.

Tab. 4.3.

CONVERTORUL	$M = \frac{V_o}{V_g}$	Condiția de funcționare DIVM	Solicitarea în tensiune a semiconductoarelor $ V_P $
BUCK	$\frac{2}{1 + \sqrt{1 + 4 \frac{(1-D)^2}{k_v}}}$	$k_v \leq D^2(1-D)$	$\frac{2V_g}{1-D}(1-M)$
BOOST	$\frac{1 + \sqrt{1 + 4 \frac{k_v}{(1-D)^2}}}{2}$	$k_v \leq D$	$\frac{2V_g}{1-D}$
BUCK-BOOST, ĆUK,	$\frac{\sqrt{k_v}}{1-D}$	$k_v \leq D^2$	$\frac{2V_g}{1-D}$
SEPIC, ZETA	$\frac{\sqrt{k_v}}{1-D}$	$k_v \leq D^2$	$\frac{2V_g}{1-D}$

Verificarea analizelor teoretice de mai sus s-a făcut pe un montaj practic de convertor BOOST, similar cu cel din Fig. 4.10 (b) având următorii parametri:

$$V_g = 10\text{ V}; L = 3,41\text{ mH}; L_1 = 11\text{ mH}; C = 4,7\text{ }\mu\text{F}; C_1 = 48,9\text{ nF}; R = 50\text{ }\Omega; f_s = 50\text{ kHz}; D = 0,4;$$

Întâi convertorul a fost simulat cu pachetul CASPOC (Simulation Research), rezultatele simulării fiind cele din Fig. 4.13. Pentru program vezi Anexa A6. Formele de undă măsurate experimental sunt prezentate în Fig. 4.14, constatându-se o concordanță perfectă cu anticipările teoretice.

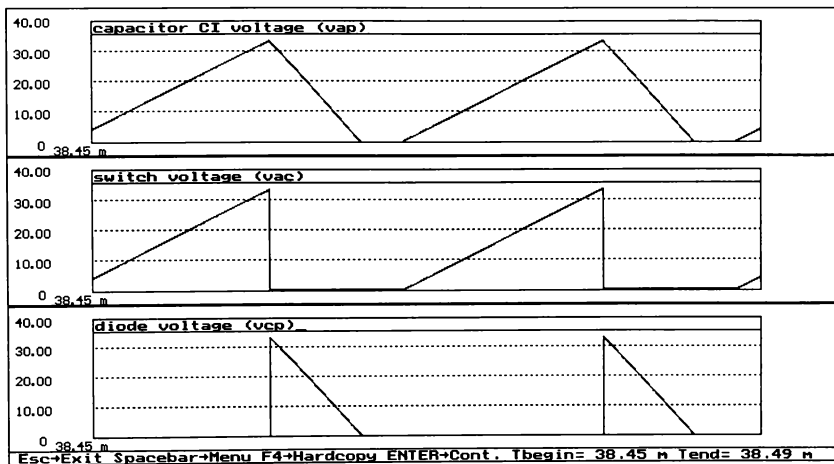


Fig. 4.13. Formele de undă aferente convertorului BOOST DIVM (simulator CASPOC). De sus în jos: tensiunea pe capacitatea  $C_i$  ( $v_{off}$ ), tensiunea pe tranzistor, tensiunea pe diodă.

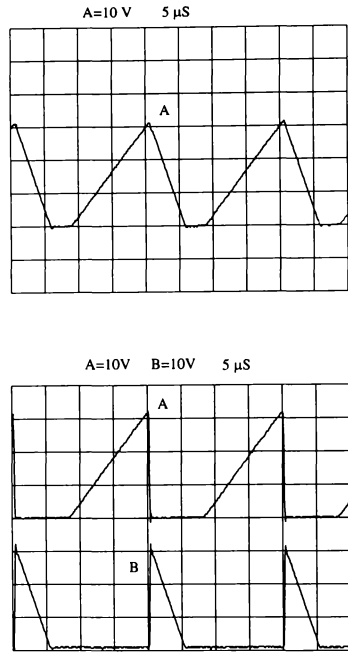


Fig. 4.14. Formele de undă măsurate experimental pentru convertorul BOOST DIVM. De sus în jos: tensiunea pe capacitatea  $C_i$  ( $v_{off}$ ), tensiunea pe tranzistor, tensiunea pe diodă.

### 4.3. Analiza unitară de curent continuu a convertoarelor cvasirezonante

#### 4.3.1. Analiza unitară a convertoarelor ZCS și ZVS

Un studiu pe baza unor întrerupătoare rezonante generalizate este efectuat în [204], [205]. Se poate găsi și pentru convertoarele QRC de tip ZCS și ZVS un mediu de comutație, așa cum este reprezentat simbolic în Fig. 4.15. Întrerupătorul activ poate fi într-un cadran sau altul, dând astfel modul de funcționare al convertorului (HW sau FW), iar elementele rezonante  $L_r$  și  $C_r$ , după modul în care sunt conectate dau tipul de convertor (ZCS sau ZVS). În orice convertor QRC de tip ZCS sau ZVS pentru  $v_{off}$  și  $i_{on}$  este adevărată ipoteza pulsațiilor mici. Presupunând aceste mărimi constante se poate și în acest caz exprima factorul de umplere echivalent  $\mu$  numai în funcție de

raportul  $\frac{|V_{off}|}{|I_{on}|} = \frac{V_g}{|I_o|} = \frac{|M|}{R}$  într-o formă unitară astfel:

Fie

$$f_0 = \frac{1}{2\pi\sqrt{L_r C_r}} \quad (55)$$

$$Z_0 = \sqrt{\frac{L_r}{C_r}} \quad (56)$$

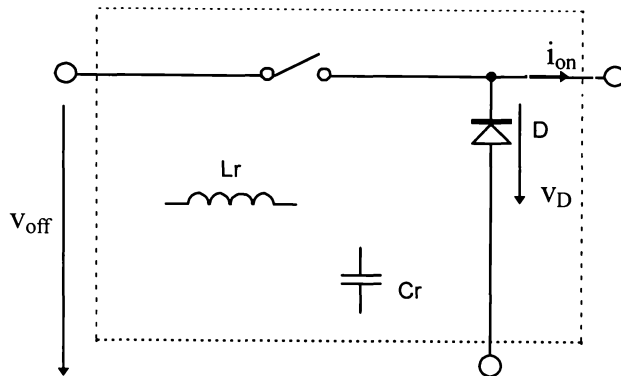


Fig. 4.15. Reprezentarea simbolică a mediului de comutație pentru un convertor QRC.

frecvența de rezonanță respectiv impedența caracteristică.

Se notează

$$\alpha = \left| \frac{I_{on} Z_0}{V_{off}} \right| = \left| \frac{I_o}{V_g} \right| Z_0 = \frac{|M|}{R} Z_0 = \frac{|M|}{R_N} \quad (57)$$

unde

$$R_N = \frac{R_L}{Z_0} \quad (58)$$

este rezistența de sarcină normalizată, caz în care au loc relațiile

$$\mu_{ZCS} = \frac{1}{2\pi} \frac{f_s}{f_0} q(\alpha) \quad (59)$$

$$\mu_{ZVS} = 1 - \frac{1}{2\pi} \frac{f_s}{f_0} q\left(\frac{1}{\alpha}\right) \quad (60)$$

unde

$$q(\alpha) = \frac{\alpha}{2} + n\pi + (-1)^{n-1} \arcsin \alpha + \frac{1}{\alpha} \left[ 1 + (-1)^{n-1} \sqrt{1 - \alpha^2} \right] \quad n = \begin{cases} 1 \text{ pt. HW} \\ 2 \text{ pt. FW} \end{cases} \quad (61)$$

este așa numită *funcție cvasirezonanță*. Se preferă normalizarea frecvenței de lucru prin frecvența de rezonanță

$$f_N = \frac{f_s}{f_0} \quad (62)$$

Pentru modurile cu undă plină, adică  $n = 2$ , funcția cvasirezonanță poate fi bine aproximată prin  $2\pi$ , astfel că  $\mu$  va depinde numai de frecvența normalizată:

Pentru structurile ZCS  $\alpha$  trebuie să fie subunitar,  $\alpha \leq 1$ , în timp ce pentru cele ZVS supraunitar,  $\alpha \geq 1$ . În modul I de funcționare (cazul clasic) există o condiție suplimentară care trebuie respectată și anume ca pentru convertoarele ZCS tensiunea pe capacitatea rezonantă să revină la valoarea constantă de la începutul perioadei, iar pentru modul ZVS curentul prin bobina rezonantă să revină la valoarea constantă de la începutul perioadei. Aceste două condiții se scriu matematic astfel:

$$f_s \leq \frac{2\pi f_0}{q(\alpha) + \frac{\alpha}{2}} \quad \text{pentru convertoare ZCS} \quad (63)$$

$$f_s \leq \frac{2\pi f_0}{q\left(\frac{1}{\alpha}\right) + \frac{1}{2\alpha}} \quad \text{pentru convertoare ZVS} \quad (64)$$

și care în fond restricționează superior frecvența de comutație normată.

Deși dependențele  $\mu = \mu(M)$  și  $M = M(\mu)$  rămân aceleași de la convertoarele PWM, expresiile (59) și (60) în care apare funcția cvasirezonantă fac ca ecuația din care se obține  $|M|$  să fie transcendentă. Astfel familiile de curbe care dau dependențele  $|M| = g(f_N)$ , cu  $R_N$  parametru, se obțin din relațiile:

$$\mu(M) = \frac{1}{2\pi} f_N q\left(\frac{|M|}{R_N}\right) \quad (65)$$

$$f_N \leq \frac{2\pi}{q\left(\frac{|M|}{R_N}\right) + \frac{|M|}{2R_N}} \quad (66)$$

pentru ZCS și

$$\mu(M) = 1 - \frac{1}{2\pi} f_N q\left(\frac{R_N}{|M|}\right) \quad (67)$$

$$f_N \leq \frac{2\pi}{q\left(\frac{R_N}{|M|}\right) + \frac{R_N}{2|M|}} \quad (68)$$

pentru ZVS

Observație. În modurile FW, pentru că  $q(\alpha) \approx 2\pi$ , relațiile (65) și (67) devin:

$$\mu(M) = f_N \quad (70)$$

respectiv

$$\mu(M) = 1 - f_N \quad (71)$$

arătând că în modul cu undă plină convertoarele cvasirezonante se comportă ca niște convertoare PWM CCM în care singura deosebire este că factorul de umplere  $D$  este înlocuit cu  $f_N$  pentru ZCS și  $1 - f_N$  pentru ZVS.

Pentru reprezentarea caracteristicilor control-ieșire au fost scrise programe publicate de autor în [78], care rezolvă numeric ecuațiile corespunzătoare pentru toate convertoarele. Programele sunt interactive, în sensul că i se cere utilizatorului să specifice convertorul dorit a se analiza, tipul (ZCS sau ZVS), modul (HW sau FW) și valorile rezistențelor normalizate. De asemenea programele returnează mesaje de eroare în cazul în care anumiți parametri au fost introduși incorect. Dificultatea rezolvării ecuațiilor transcendente o constituie punctul de plecare în iterație. În programe s-a folosit metoda înjumătățirii intervalului. Deși cu o convergență mai slabă decât alte metode s-a preferat această metodă pentru că pentru fiecare convertor au putut fi găsite punctele de plecare în iterație care să asigure convergența algoritmului. Pentru că expunerea s-ar prelungi foarte mult dacă s-ar demonstra aici alegerea acestor puncte, sugerăm examinarea programelor din Anexa A7, care sunt edificatoare în justificarea modului lor de alegere.

Ca și exemple de aplicare a programului în Fig. 4.16 s-au trasat caracteristicile control ieșire normalizate pentru un convertor BUCK ZCS HW, în Fig. 4.17 pentru un convertor BOOST ZCS HW, în Fig. 4.18 pentru un convertor BUCK-BOOST ZCS HW, , în Fig. 4.19 pentru un convertor BUCK ZVS HW, , în Fig. 4.20 pentru un convertor BOOST ZVS HW, iar în Fig. 4.21 pentru un convertor BOOST ZVS FW. Parametrul indicat pe fiecare curbă este rezistența de sarcină normalizată. Concluziile sunt de acum binecunoscute:

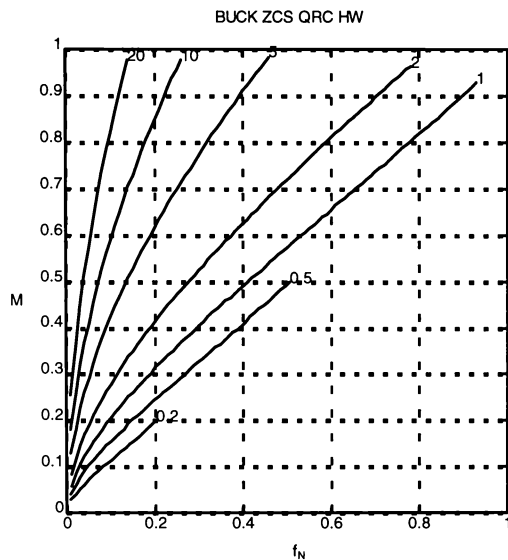


Fig. 4.16. Caracteristicile control-ieșire normalizate pentru un convertor BUCK ZCS QRC HW.

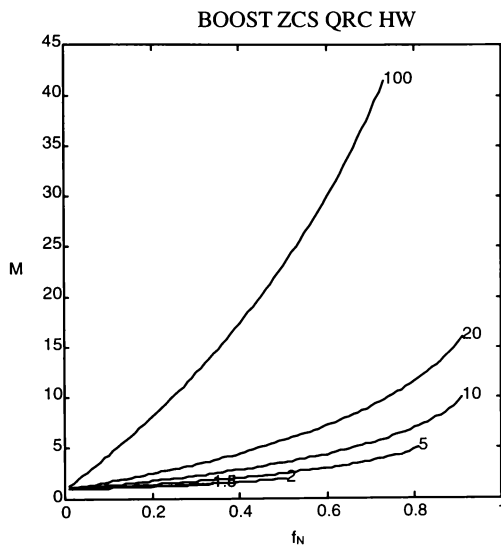


Fig. 4.17. Caracteristicile control-ieșire normalizate pentru un convertor BOOST ZCS QRC HW.

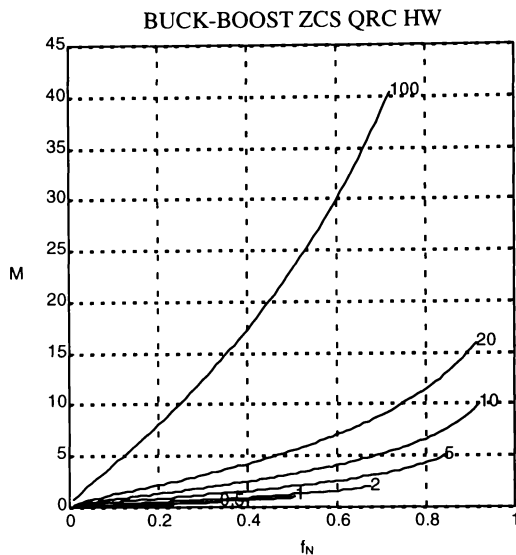


Fig. 4.18. Caracteristicile control-ieșire normalizate pentru un convertor BUCK-BOOST ZCS QRC HW.

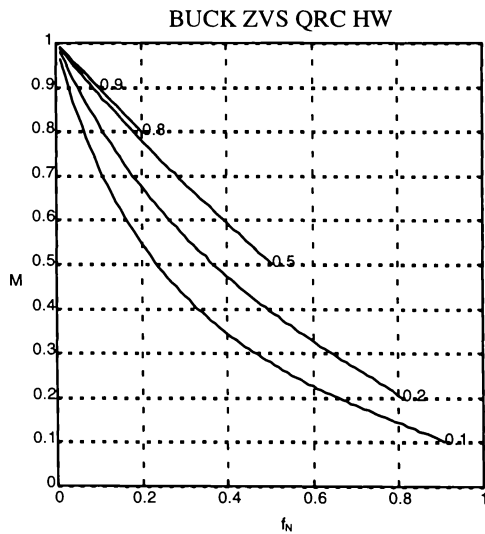


Fig. 4.19. Caracteristicile control-ieșire normalizate pentru un convertor BUCK ZVS QRC HW

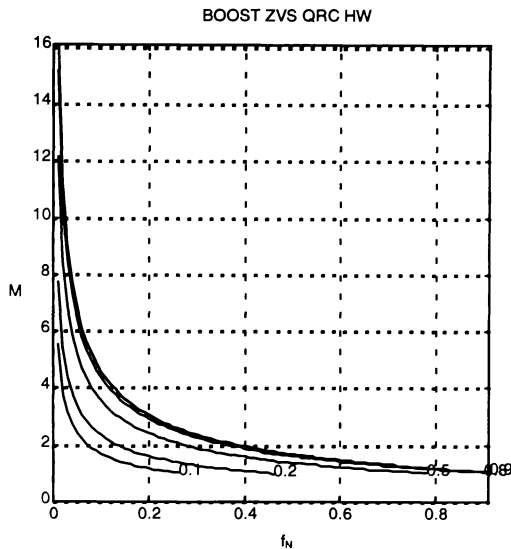


Fig. 4.20. Caracteristicile control-ieșire normalizate pentru un convertor BOOST ZVS QRC HW.

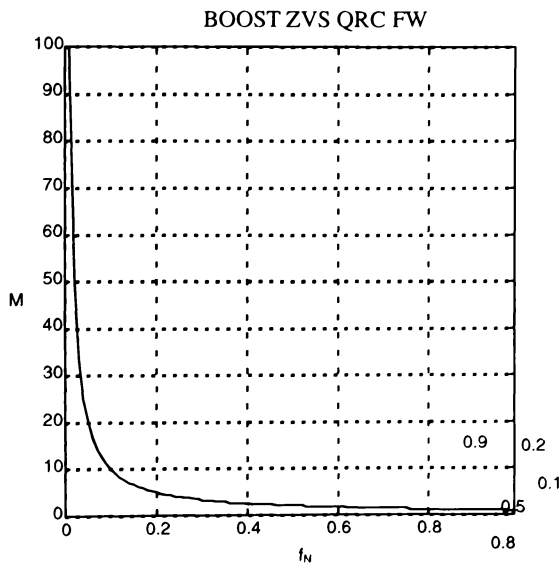


Fig. 4.21. Caracteristicile control-ieșire normalizate pentru un convertor BOOST ZCS QRC HW.

- convertoarele ZCS au un raport static de conversie strict crescător cu frecvența de comutație, în timp ce la convertoarele ZVS acest raport este strict descrescător.
- la funcționarea în mod semiundă caracteristicile sunt puternic dependente de sarcină; în modul cu undă plină caracteristicile sunt practic independente de sarcină.



- sunt moștenite proprietățile convertoarelor PWM din care provin: raportul static de conversie subunitar la convertoarele BUCK, supraunitar la cele BOOST și poate fi, în modul, fie subunitar, fie supraunitar pentru convertoarele BUCK-BOOST.

Programele dezvoltate își găsesc aplicația în analiza și proiectarea oricărui convertor cvasirezonant.

#### 4.3.2 Analiza unitară a convertoarelor cvasirezonante cu funcționare la frecvență constantă

În analiza convertoarelor CF-QRC semnificația notațiilor este aceeași din paragraful precedent.

1. Pentru convertoarele CF-ZVS în modul II de funcționare (vezi paragraful 3.3, Fig. 3.5), la care se reamintește că factorul de umplere este raportul dintre intervalul de timp măsurat de la aducerea în conducție a tranzistorului din  $S_2$  și blocarea tranzistorului din  $S_1$ , factorul de umplere echivalent are expresia [116]:

$$\mu = 1 - \frac{f_N}{\pi} \left[ \pi - \arctg\left(\alpha + D \frac{2\pi}{f_N}\right) + \left(\alpha + D \frac{2\pi}{f_N}\right) \right] \quad (71)$$

Pentru frecvențe joase regiunea de funcționare se extinde atât în direcția lui  $\mu$ , cât și a lui  $\alpha$ . Spre deosebire de convertoarele ZVS cu frecvență variabilă este posibilă funcționarea în gol. Totuși, domeniul rapoartelor de conversie obținabile este mai restrâns. Pe baza inegalității  $\arctg(x) < x$ , adevărată pentru  $x > 0$ , se arată ușor că  $\mu$  nu poate depăși  $1 - f_N$ . Dacă factorul de umplere echivalent necesar,  $\mu^*$ , s-a determinat din specificațiile de proiectare. Atunci  $f_N$  trebuie să fie:

$$f_N < 1 - \mu^* \quad (72)$$

Alegerea lui  $f_N$  se face ca un compromis între solicitările în tensiune și în curent ale lui  $S_1$ . Astfel dacă  $f_N \rightarrow 1 - \mu^*$  solicitarea în tensiune a lui  $S_1$  tinde la valoarea sa minimă, egală cu  $2V_g$ , dar solicitarea în curent devine infinită. Dacă  $f_N \rightarrow 0$  solicitarea în curent tinde la  $\frac{V_g}{Z_0}$  (ca în convertorul PWM generator), dar solicitarea în tensiune crește mult. Deci între aceste limite trebuie ales  $f_N$  pentru a avea ambele solicitări rezonabile. Indiferent de alegerea lui  $f_N$  solicitarea în tensiune a lui  $S_2$  este egală cu cea din convertorul PWM generator.

În Fig. 4.22 și Fig. 4.23 sunt prezentate caracteristicile de ieșire respectiv caracteristicile control-ieșire ale convertoarelor CF-ZVS, programele corespunzătoare fiind expuse în Anexa A8..

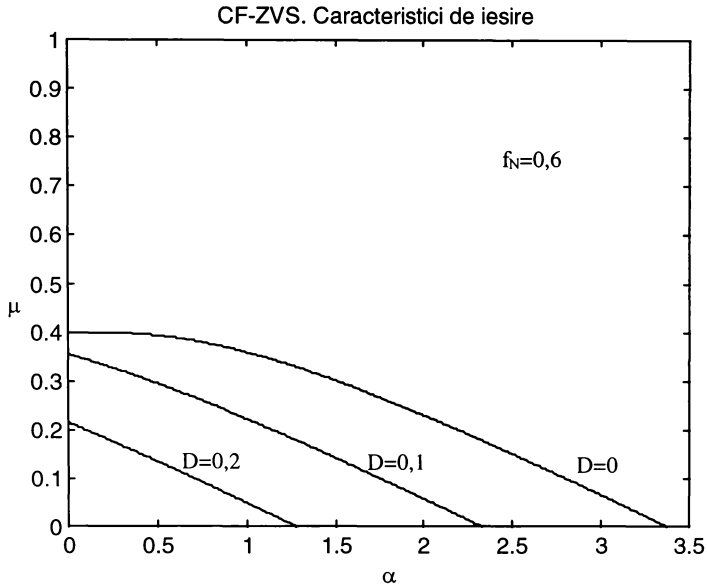


Fig. 4.22. Caracteristicile de ieșire ale convertoarelor CF-ZVS în modul II de funcționare.

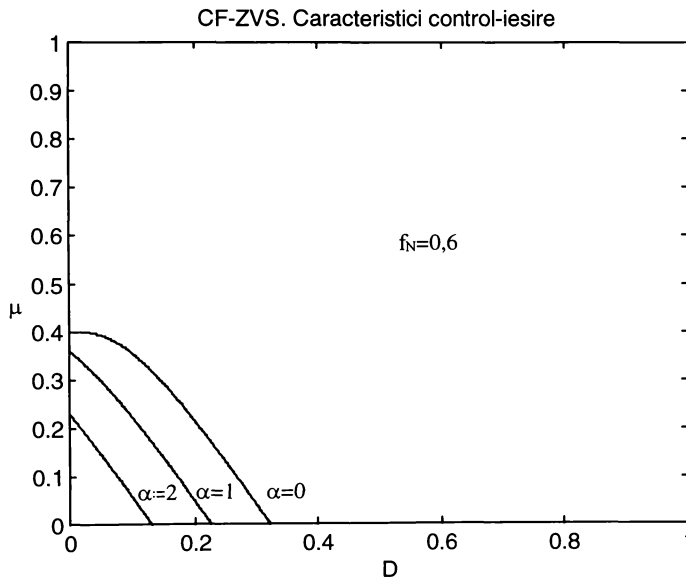


Fig. 4.23. Caracteristicile control-ieșire ale convertoarelor CF-ZVS în modul II de funcționare.

O analiză similară efectuată asupra convertoarelor CF-ZV-QSW arată că ambele întrerupătoare sunt comutate la tensiune nulă și au solicitări de tensiune identice cu cele ale convertoarelor PWM din care provin. În modul I regiunea de funcționare este simetrică față de axa

$\mu = 0,5$  și cuprinde și o parte din axa de sarcină nulă. Acesta este un avantaj important față de convertoarele ZVS-QSW comandate în frecvență, la care regiunea de funcționare era limitată la  $\mu > 0,5$  [203]. Alegerea frecvenței normalizate este dictată de domeniul accesibil cerut pentru factorul de umplere echivalent și de compromisul dintre solicitările în curent și durata stărilor topologice. Caracteristicile de curent continuu arată că, exceptând limitarea din regiunea de funcționare, ele sunt similare cu cele ale convertoarelor PWM generatoare, cu caracteristici de ieșire aproape horizontale, deci cu o rezistență de ieșire mică.

Pentru convertoarele CF-ZVS-MR, ambele întrerupătoare sunt comutate în conducție și blocare la tensiune zero. Se poate arăta că în modurile I, II, III, dacă  $f_N < 1$ , toți factorii de umplere echivalenți sunt accesibili de la sarcină zero la o sarcină maximă. Cu cât  $f_N$  scade, cu atât sarcina maximă crește, solicitările în curent scad, dar solicitările de tensiune cresc. O proprietate remarcabilă a convertoarelor CF-ZVS-MR este aceea că ele se comportă ca o sursă de curent constant după ce se atinge curentul maxim de ieșire. Curentul de ieșire în scurtcircuit este numai cu puțin mai mare decât curentul nominal maxim. Dacă factorul de umplere maxim este limitat convertorul este imun la condițiile de funcționare în gol și scurtcircuit, deci nu sunt necesare protecții adiționale.

#### 4.4. Concluzii

Analiza efectuată în acest capitol arată că în curent continuu convertoarele PWM cu două întrerupătoare și convertoarele cvasirezonante se pot analiza cu tehnici identice. Acest lucru nu este surprinzător dacă ne gândim că ultimele provin din convertoarele PWM, având circuite identice în curent continuu. În fond similitudinile comportării în curent continuu sunt consecințe ale similitudinilor topologice. De aceea *modelul de curent continuu general* al unui convertor PWM sau cvasirezonant este același, fiind exemplificat în Fig. 4.24.

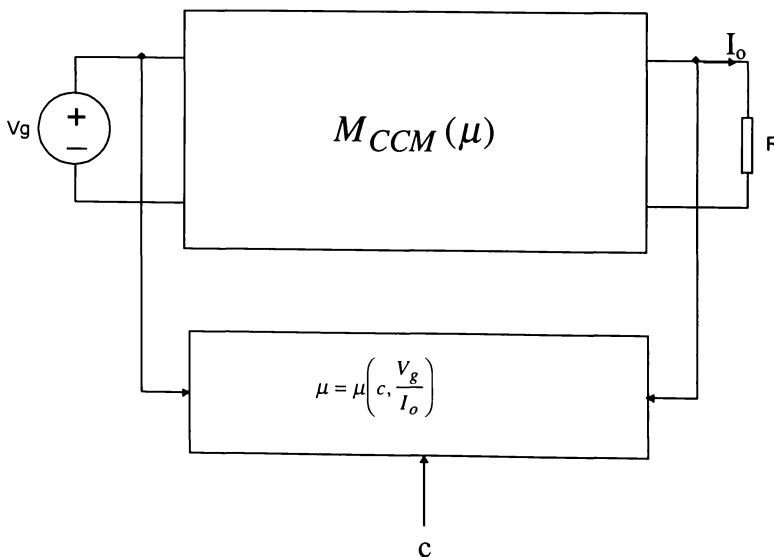


Fig. 4.24. Modelul de curent continuu generalizat al convertoarelor PWM și QRC cu două întrerupătoare.

Diferitele moduri de funcționare ale convertoarelor sunt influențate de modalitățile practice de implementare a întrerupătoarelor și de ipotezele de pulsații mici făcute asupra unor curenți sau tensiuni. Pentru convertoare PWM condiția funcționării într-un anumit mod se exprimă printr-o singură inegalitate ce se referă la factorul de umplere echivalent.

Pentru convertoarele cu mai multe întrerupătoare modelele sunt diferite, exemplul convertoarelor pătratice fiind elocvent. La aceste convertoare condiția funcționării într-un anumit mod este dată în fapt de un set de inegalități rezultate din condiții de funcționare corespunzătoare fiecărui întrerupător pasiv al schemei.

Contribuțiile originale ale autorului în acest domeniu se referă la:

- sistematizarea procedurii de deducere a caracteristicilor statice pentru modul DICM și DCVM (paragrafele 4.1 și 4.2.2.1).
- introducerea noii clase de convertoare DIVM și efectuarea analizei unitare de curent continuu a lor (paragraful 4.3.2).
- stabilirea condițiilor de funcționare CCM pentru convertoarele pătratice printr-o analiză a celulei de bază complete, deci într-un mod unitar, caracterizând complet familia și nu convertorul individual (paragraful 4.2.2.2).
- elaborarea programelor de trasare automată a caracteristicilor control-ieșire și a caracteristicilor de ieșire pentru convertoarele QRC și CF-QRC ca instrumente de analiză și proiectare (paragraful 4.3).

## CAPITOLUL 5

## Metode de comandă a convertoarelor dc-dc

În acest capitol este efectuată o trecere în revistă a principalelor metode de comandă pentru convertoarele PWM, altele decât comanda tradițională în factor de umplere. Este prezentată apoi o metodă nouă, propusă de autor și denumită comanda mixtă. În final sunt expuse rezultatele teoretice și experimentale întreprinse de doctorand privind comanda cu rețele neuronale și comanda fuzzy a convertoarelor dc-dc.

## 5.1 Comanda în curent

În literatură s-au propus numeroase tehnici de comandă în curent [32], [160], [129] care determină funcționarea la frecvență variabilă (de ex. comanda prin timpul de blocare sau comanda cu bandă de toleranță) sau la frecvență constantă. Cele cu funcționare la frecvență constantă sunt de departe cele mai răspândite și pentru care s-au dezvoltat și o serie de circuite integrate dedicate.

Cea mai cunoscută tehnică de comandă în curent la frecvență constantă pentru convertoare PWM cu două întrerupătoare are schema bloc din Fig. 5.1. Ieșirea convertorului este controlată prin vârful de curent al curentului  $i_S$  ce trece prin întrerupătorul activ. Tehnica este numită și comandă prin injecție de curent (current injection control - CIC) [181], [182]. Deci factorul de umplere devine o mărime intermediară de comandă, depinzând acum nu numai de semnalul extern de comandă,  $v_{ref}$ , ci și, într-un caz mai general, de curenții prin bobinele din secțiunea mediului de comutație ce conține întrerupătoarele,  $i_{on}$ , deci și de  $V_g$  și  $i_o$ .

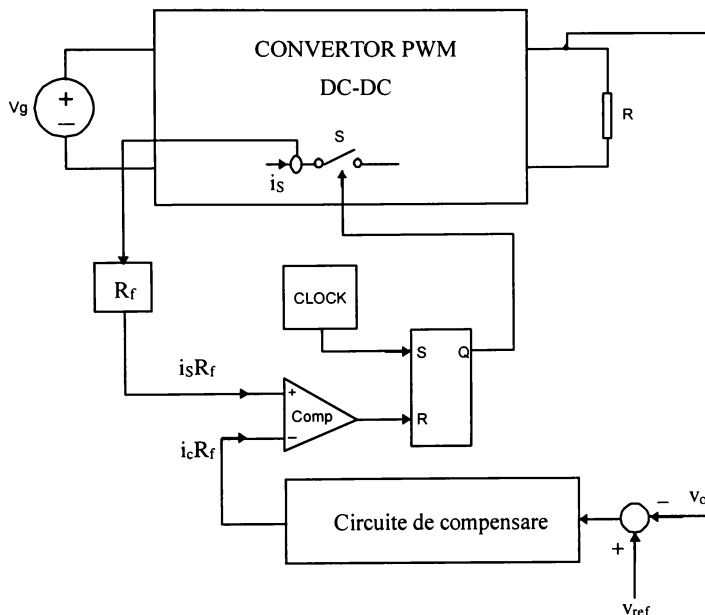


Fig. 5.1. Principiul comenzii în curent.

Factorul de umplere din a  $k$ -a perioadă,  $d_k$ , este dat de ecuația:

$$i_S(d_k T_S) = v_{ref}(k T_S) \quad (1)$$

unde originea timpului este luată la începutul perioadei  $k$  de comutație

Avantajele comenzii în curent sunt:

- dinamică mai bună decât la comanda în factor de umplere. Funcția de transfer (model mediat) la semnal mic  $\frac{\hat{v}_o}{\hat{i}_c}$  conține cu un pol mai puțin decât funcția  $\frac{\hat{v}_o}{\hat{d}}$ , unde  $\hat{d}$  reprezintă perturbația de semnal mic din factorul de umplere continuu. La semnal mare rejecția perturbațiilor din tensiunea de alimentare este mai bună și timpul de răspuns mai scurt.
- nu necesită circuite de protecție la supracurent (scurtcircuit)
- pentru anumite convertoare permite paralelizarea ieșirilor, fără alte precauții de împerechere.
- reducerea sau eliminarea problemelor de saturare a transformatorului la convertoare în punte sau push-pull.
- fabricarea unei game largi de circuite integrate de cost redus pentru implementarea comenzii.

Totuși comanda în curent este însoțită și de o serie de dezavantaje:

- apariția oscilațiilor de frecvență egală cu jumătate din frecvența de comutație la factori de umplere mai mari de 0,5. Înlăturarea acestei oscilații se face cu așa numita tehnică de compensare de pantă, unde o rampă descrescătoare externă (artificială) se adaugă peste semnalul de comandă pentru a mări panta netă.
- sensibilitate la zgomot în  $i_S$  sau  $i_c$ , zgomote care pot comuta fals bistabilul. De aceea este necesară o filtrare suplimentară a curentului sesizat, în acest fel înlăturându-se spike-urile de curent de la aducerea în conducție a tranzistorului cauzate de sarcina stocată în diodă.
- nu se pretează la aplicații în care mărimea de interes este curentul mediu, cum ar fi încărcarea acumulatorilor sau corecția factorului de putere.

Tehnica de comandă în curent a fost extinsă și la funcționarea în DICM

## 5.2 Tehnici de comandă integrative

### 5.2.1 Comanda prin sarcină

Prima tehnică de comandă integrativă raportată a fost comanda prin sarcină ("charge control") [182]. Așa cum se observă în Fig. 5.2 (în care tactul care impune frecvența de comutație, bistabilul și bucla externă de tensiune prin care se face stabilizarea au fost omise, subînțelegându-se) situația este asemănătoare cu comanda în curent, cu excepția faptului că mărimea comparată cu tensiunea de comandă este o tensiune proporțională cu integrala curentului prin întrerupător. Aceasta face ca la comanda prin sarcină să apară o serie de avantaje față de comanda în curent:

- rampa externă necesară pentru a conferi stabilitate schemei la factori de umplere peste 0,5 nu mai este necesară la un domeniu dat de variație pentru tensiunea de alimentare și sarcină.
- excelentă imunitate la zgomote datorită fenomenului de integrare prealabil comparării.
- se poate aplica și convertoarelor la care blocarea întrerupătorului activ se produce în momentul în care curentul nu atinge maximum. De aceea teoretic comanda prin sarcină se poate aplica și convertoarelor cvasirezonaante [183].
- timp de răspuns bun la perturbații în tensiunea de alimentare cu menținerea proprietăților de rejecție ale comenzii în curent.

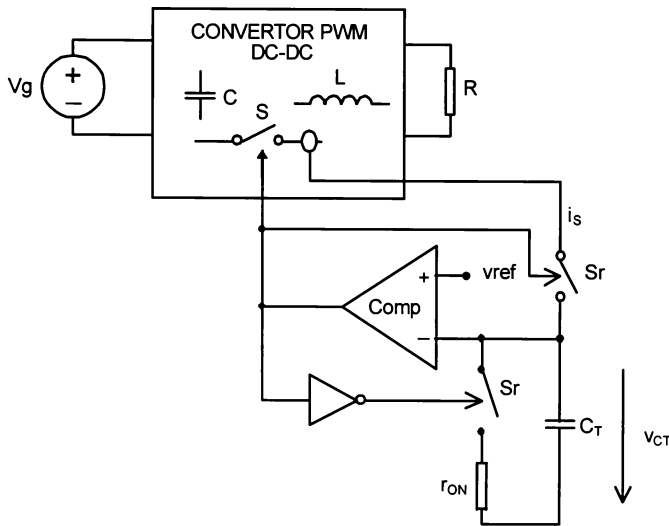


Fig. 5.2. Principiul comenzii prin sarcină.

Factorul de umplere,  $d_k$ , aferent celei de a  $k$ - a perioade, este date de ecuația:

$$\frac{1}{\tau} \int_0^{d_k T_s} i_S(t) dt = v_{ref} (d_k T_s) \quad (2)$$

### 5.2.2 Comanda integrativă după o variabilă

Smedley și Ćuk [171], [172], au arătat că de fapt comanda prin sarcină este un caz particular al așa numitei comenzi integrative după o variabilă. Observația care se face este că, fiind dat un întrerupător, prin comutarea acestuia practic o variabilă de intrare în întrerupător  $z(t)$  este convertită într-o așa numită variabilă de ieșire comutată,  $u(t)$ , ca în Fig. 5.3. Formele de undă aferente sunt prezentate în Fig. 5.4. Luând originea la începutul celei de a  $k$ - a perioade, factorul de umplere,  $d_k$ , corespunzător perioadei  $k$ , este dat de ecuația:

$$\frac{1}{\tau} \int_0^{d_k T_s} u(t) dt = v_{ref} (d_k T_s) \quad (3)$$

Se observă fără dificultate că dacă  $z(t) = i_L(t)$  și  $u(t) = i_S(t)$  atunci comanda integrativă după o variabilă este chiar comanda prin sarcină. În esență observația făcută de autori este aceea că în loc de a integra o variabilă comutată (aici  $u(t)$ ) se poate integra variabila de intrare  $z(t)$  în întrerupător, variabilă care de regulă este o mărime cu pulsații reduse, ceea ce facilitează modelarea [170], [73].

Comanda integrativă după o variabilă asigură o serie de avantaje, unele din ele fiind deja amintite la comanda prin sarcină:

- capacitatea de a rejecta complet perturbațiile din sursa de alimentare.
- valoarea mediată a variabilei comutate urmărește mărimea de referință într-o perioadă de comutație. Nu există eroare staționară sau eroare dinamică între variabila de comutație mediată și mărimea de referință.

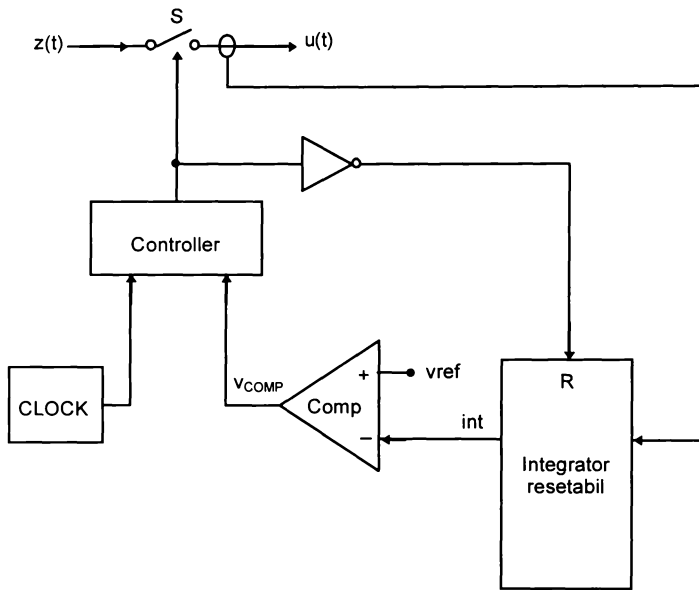


Fig. 5.3. Comanda integrativă după o variabilă.

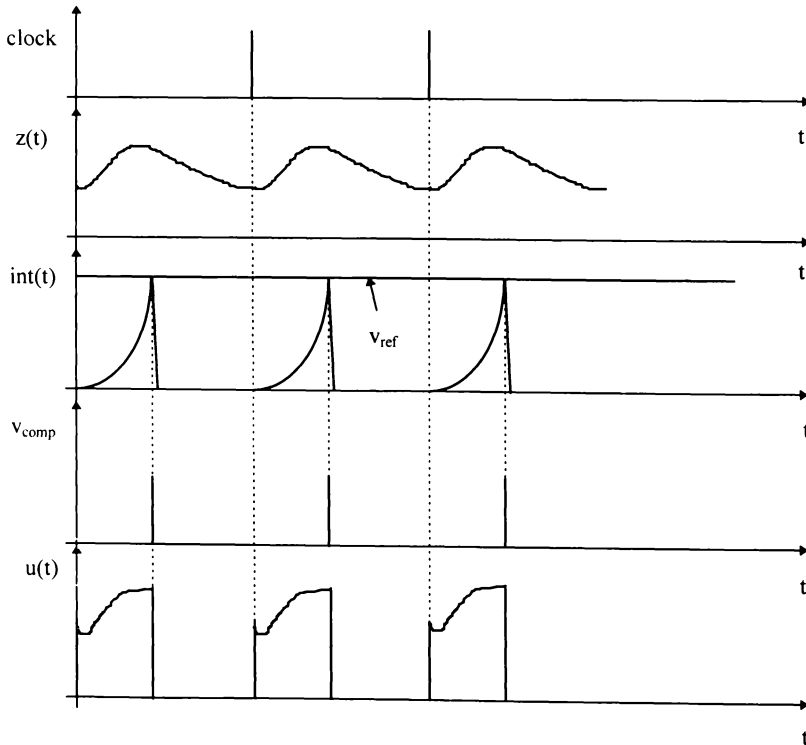


Fig. 5.4. Formele de undă tipice pentru comanda integrativă după o variabilă.



- eroarea de comutație este automat corectată într-o perioadă.
- tehnica are un mare grad de generalitate, putând fi aplicată la convertoare PWM, rezonante, invertoare, redresoare, fie pentru comanda în tensiune, fie pentru comanda în curent, în conducție continuă sau în conducție discontinuă. De fapt tehnica se poate utiliza în comanda oricărei variabile fizice sau semnal abstract care are forma unei variabile comutate sau care poate fi convertit în forma unei variabile comutate.
- modalitate de implementare practică simplă.

### 5.2.3 Comanda mixtă

Comanda integrativă după o variabilă se poate la rândul său generaliza în sensul înlăturării restricției că cea care se integrează este intrarea în întrerupător, efectuându-se operația de integrare asupra unei combinații liniare de variabile, [Lascu, 81]. Comanda mixtă este legată de mediul de comutație introdus în capitolul 4, factorul de umplere din perioada  $k$  fiind dat de ecuația (din nou originea la începutul perioadei):

$$\frac{1}{\tau} \int_0^{d_k T_s} v_{off}(t) dt + \frac{1}{C_T} \int_0^{d_k T_s} i_{on}(t) dt = v_{ref}(d_k T_s) \quad (4)$$

Relația (4) arată că implementarea practică este simplă, deși are dezavantajul minor că cere un integrator suplimentar și un sumator față de celelalte tehnici integrative menționate. Principalele avantaje ale acestei tehnici de comandă sunt:

- ca și la comanda prin sarcină sau comanda integrativă după o variabilă se poate utiliza și în convertoare în care întrerupătorul activ se blochează atunci când curentul prin el nu atinge valoarea maximă. Acest lucru este posibil întrucât tensiunea de la ieșirea sumatorului este strict crescătoare, cele două integrări efectuându-se asupra a două mărimi pozitive.
- nu este necesară compensarea de pantă, deși în anumite situații prezența acesteia îmbunătățește substanțial audiosusceptibilitatea.
- răspunsul la perturbații în tensiunea de alimentare este mai rapid și calitativ mai bun comparativ cu comanda prin sarcină, dar mai lent decât la comanda integrativă după o variabilă.
- răspunsul la salt treaptă în mărimea de comandă este mai bun decât la comanda integrativă după o variabilă dar mai slab decât la comanda prin sarcină.
- imunitate la zgomote datorată proceselor de integrare.

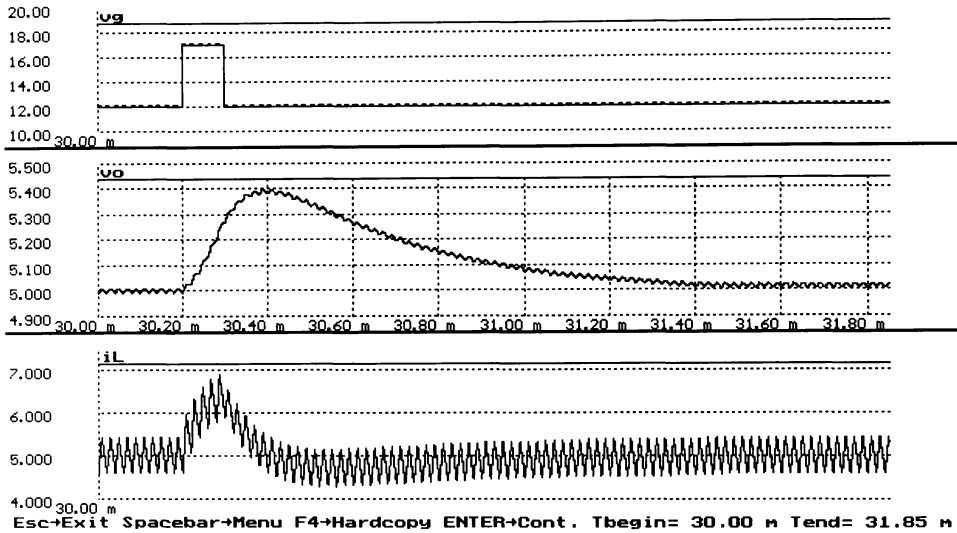
Cele de mai sus sunt ilustrate de simulările din Fig. 5.5 unde este prezentată comparativ comportarea la semnal mare la salt treaptă în tensiunea de alimentare, pentru comandă în sarcină (a) și pentru comandă mixtă (b) pentru un convertor BUCK cu următorii parametri:

$$L = 70 \mu\text{H}; \quad C = 390 \mu\text{F}; \quad R_c = 20 \text{ m}\Omega; \quad I_o = 5 \text{ A}; \quad f_s = 50 \text{ kHz}$$

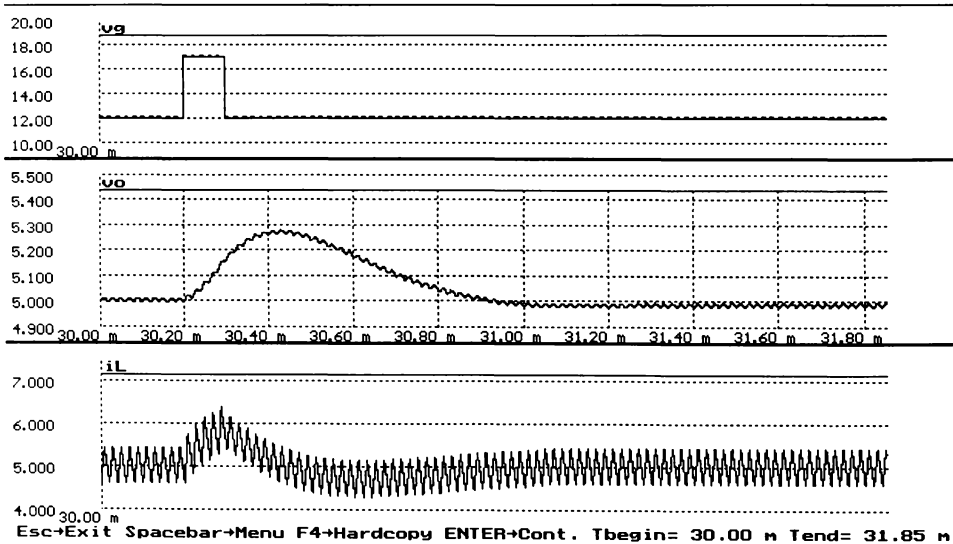
În ambele cazuri  $C_T = 10 \mu\text{F}$ , iar pentru comanda mixtă  $\tau = 24 \mu\text{s}$ , astfel încât tensiunea de ieșire să fie de 5V. Saltul în tensiunea de alimentare este de la 12V la 17V.

Figura 5.6 prezintă comportarea comparativă la semnal mare pentru salt în tensiunea de comandă, pentru comanda integrativă după o variabilă (a) și pentru comanda mixtă (b), având în studiu același convertor. Pentru o comparație corectă în ambele cazuri tensiunea de ieșire are aceeași variație: de la 2V la 5V, în condițiile în care ambele scheme au același  $\tau = 24 \mu\text{s}$ . Se remarcă răspunsul oscilant în cazul comenzii integrative după o variabilă față de amortizarea mult mai bună și timpul de răspuns mult mai scurt în cazul comenzii mixte. Programele aferente se găsesc în Anexa A9

Modelele de semnal mic, studiul stabilității și indicațiile de proiectare complete se găsesc în [Lascu 81]. O parte din aceste aspecte vor fi reluate în capitolul 6.



(a)

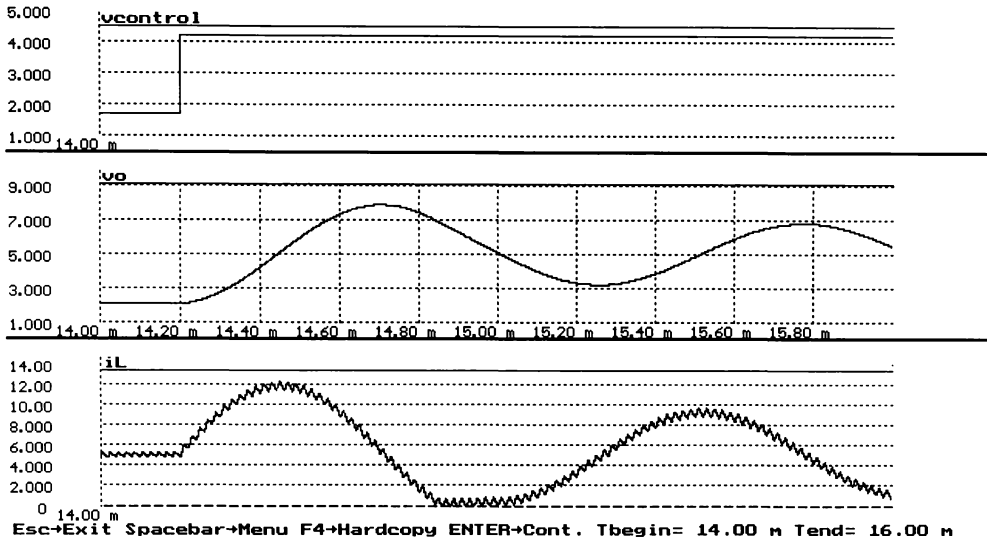


(b)

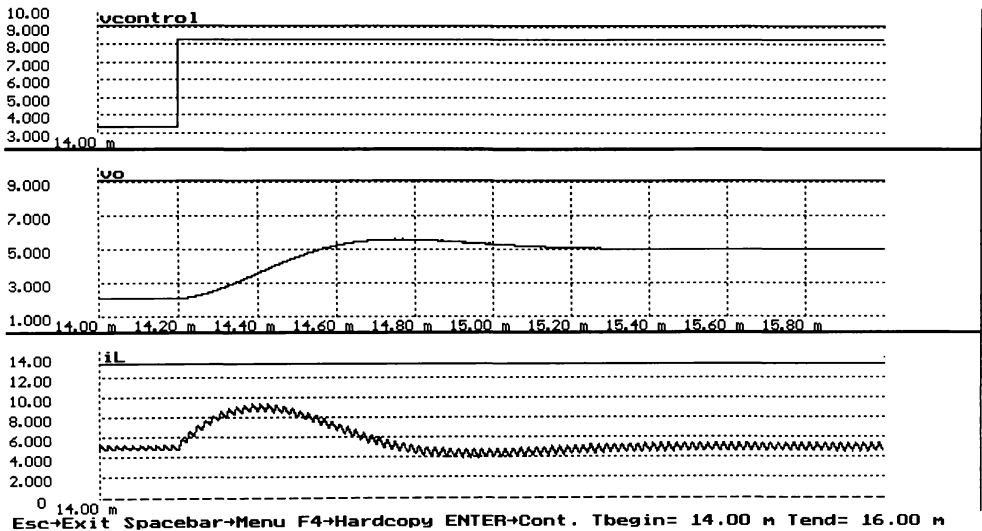
Fig. 5.5. Răspunsul la perturbații de semnal mare în tensiunea de alimentare.

(a) - comanda prin sarcină. (b) - comanda mixtă.

Simulator: CASPOC - Simulation Research.



(a)



(b)

Fig. 5.6. Răspunsul la perturbații de semnal mare în tensiunea de comandă.

(a) - comanda integrativă după o variabilă. (b) - comanda mixtă.

Simulator: CASPOC - Simulation Research.

### 5.3. Comanda fuzzy

Alegerea metodei de comandă este crucială în funcționarea unui convertor dc-dc. Dificultatea care apare în analiza, modelarea și elaborarea controllerului este dată de caracterul *neliniar* al sistemului reprezentat de convertor.

Metodele de comandă prezentate până acum presupun o bună cunoaștere a sistemului și o acordare precisă pentru a obține performanțele dinamice dorite. Ele se bazează în ultimă instanță pe un model liniar de semnal mic al părții de putere. Datorită faptului că modelele de semnal mic au validitate limitată, elaborarea controllerului este adesea un compromis între simplitate și performanțe. Performanțele lor sunt puternic dependente de punctul de funcționare în jurul căruia se face liniarizarea, motiv pentru care prezența elementelor parazite, sarcinile dependente de timp sau tensiunile de alimentare variabile fac alegerea parametrilor controllerului dificilă.

Stabilitatea de semnal mare reclamă de regulă o reducere a benzii de frecvență, deci o dinamică mai slabă. Pe de altă parte aplicarea metodelor de comandă tradiționale la convertoare de ordin superior (de ex. ĆUK sau SEPIC) poate duce la dificultăți de proiectare și o stabilitate dificil de obținut. În plus, modelele matematice devin din ce în ce mai complexe, iar algoritmi de reglare numerică complicați și improprie pentru o implementare practică.

O abordare complet diferită este oferită de logica de comandă fuzzy ("fuzzy logic control" - FLC), care nu necesită o modelare matematică precisă și nici calcule complexe. Tehnica de comandă fuzzy se bazează pe capabilitatea proiectantului de a înțelege calitativ funcționarea sistemului și de a folosi euristic aceste cunoștințe. Ea utilizează proprietățile fizice de bază ale sistemului și potențial poate realiza comanda și în acele condiții de funcționare în care tehnicile liniare eșuează, de exemplu în cazul dinamicii de semnal mare sau în cazul unor variații în game largi ale parametrilor.

Desigur, controllerele fuzzy nu pot realiza, în general, un răspuns de semnal mic mai bun decât cel realizat de controllerele standard. Totuși, întrucât se bazează pe reguli euristice, aplicarea unor legi de comandă neliniare care să contracareze caracterul neliniar al convertoarelor dc-dc este ușoară. Regulile euristice sunt deduse de obicei din caracteristicile statice, deci din acest punct de vedere sistemul este unul benign [63]. Mai mult, aceleași legi de comandă pot fi aplicate mai multor convertoare dc-dc. Cei care se modifică în funcție de topologia convertorului și de parametrii acestuia și deci trebuie accordați, sunt câțiva factori de scalare. Rezultatele proiectării controllerului pot fi memorate în memorii EPROM de tip "lookup table" sau se pot utiliza sisteme cu procesoare de semnal, având în vedere apariția procesoarelor de semnal de tip fuzzy.

Autorul a studiat un convertor BOOST cu comandă fuzzy având schema bloc din Fig. 5.7 folosind un modul EVM30. Parametrii convertorului sunt:

$$L = 3,704 \text{ mH}; R_L = 0,47 \text{ } \Omega; C = 32,1 \text{ } \mu\text{F}; R_C = 0,356 \text{ } \Omega; R = 67,7 \text{ } \Omega, T = BD239, D = SB540$$

Sistemul a fost construit în jurul modului EVM30, cu un procesor de semnal TMS320C30/30MHz. Intrarea analogică a modului EVM30 este folosită pentru măsurarea tensiunii de ieșire a convertorului dc-dc. Întrerupătorul activ este comandat de semnalul XF1 al procesorului. Intrarea analogică este eșantionată cu o frecvență  $f_{s\text{amp}} = 18,6 \text{ kHz}$ , pe 14 biți. În cazul prelevării unui singur eșantion din tensiunea de ieșire eșantionarea se face pe intervalul de blocare al întrerupătorului activ, pentru a nu fi afectați de zgomotul de comutație. Frecvența de comutație a convertorului este  $f_s = \frac{f_{s\text{amp}}}{3}$ , pentru că interfața analogică (AIC) are nevoie de 3 perioade de eșantionare pentru a trimite eșantionul către procesor. Aceasta permite prelevarea și medierea a 3 eșantioane (semnale de reacție) în fiecare perioadă de comutație, cu efecte benefice în calitatea stabilizării. Regulatorul fuzzy este de tip Mamdani, folosind implicația min în metoda de inferență. Defuzificarea se bazează pe metoda centrului de greutate. Programele au fost scrise integral în C. În fiecare perioadă de comutație a convertorului sunt necesari 495 de cicli pentru a efectua calculele, în vederea obținerii factorului de umplere.

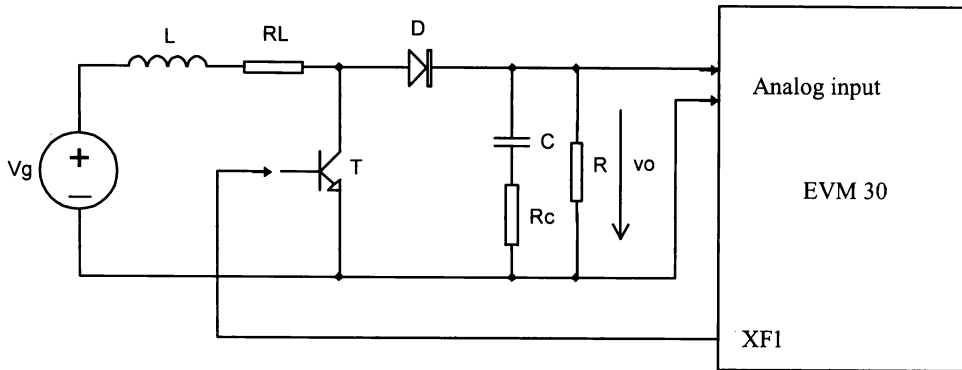


Fig.5.7. Schema bloc a controllerului fuzzy pentru un convertor BOOST.

Aceasta permite creșterea frecvenței de comutație până la 60 kHz dacă convertorul analog-numeric de pe modul este înlocuit cu un convertor analog-numeric mai rapid și dacă se utilizează un procesor de 60 MHz.

Regulatorul implementat este asemănător cu cel din [174], corespunzător unui analogic de tip PI. Intrările în controllerul fuzzy sunt eroarea,  $e$ , și derivata erorii,  $de$ , definite astfel:

$$e = V_o - V_{ref} \quad (5)$$

$$de = e_k - e_{k-1} \quad (6)$$

unde  $V_o$  este tensiunea de ieșire actuală,  $V_{ref}$  este tensiunea de ieșire, iar indicii  $k$  se referă la valori din a  $k$ -a perioadă de comutație. Ieșirea controllerului este factorul de umplere, dat de:

$$d_k = d_{k-1} + \eta \cdot \delta d_k \quad (7)$$

unde  $\delta d_k$  este variația factorului de umplere obținută prin inferență, iar  $\eta$  este câștigul regulatorului. Intrările  $e$  și  $de$  ale regulatorului sunt normalizate prin factorii  $\beta_e$  respectiv  $\beta_{de}$ .

Fiecare univers a fost împărțit în 5 subseturi fuzzy: mare pozitiv (PB), mic pozitiv (PS), zero (Z), mic negativ (NS) și mare negativ (NB). Forma funcțiilor de apartenență este una simetrică și echidistantă, cum se obișnuiește frecvent. Deducerea regulilor de comandă fuzzy este euristică și se bazează pe următoarele criterii:

- 1) Când ieșirea convertorului este departe de valoarea de referință variația factorului de umplere trebuie să fie mare, pentru a aduce rapid tensiunea de ieșire la valoarea dorită.
- 2) Când ieșirea convertorului se apropie de valoarea dorită variația factorului de umplere trebuie să fie nenuă dar mică.
- 3) Când ieșirea convertorului este aproape de valoarea de referință și se apropie rapid de ea factorul de umplere va fi menținut constant, pentru a preveni supracreșterile.
- 4) Când valoarea dorită este atinsă și ieșirea încă tinde să se modifice factorul de umplere se va modifica puțin, pentru a preveni devierea ieșirii de a valoarea atinsă.
- 5) Când valoarea de referință este atinsă și tensiunea de ieșire nu tinde să se modifice factorul de umplere rămâne nemodificat.
- 6) Când ieșirea este peste valoarea de referință semnul variației factorului de umplere este negativ și invers.

De asemenea, după calculul cu relația (7) factorul de umplere a fost limitat inferior și superior la valorile  $D_{min}$  respectiv  $D_{max}$ .

Experimentele au început cu un tabel de reguli la care elementele sunt singleton-uri, ca în Tab. 5.1. Această bază de reguli va fi folosită până ce schimbarea ei va fi specificată.

Tab. 5.1. Tabelul de inferență al regulatorului fuzzy.

e	de	NB	NS	Z	PS	PB
PB		-0,3	-0,35	-0,45	-0,65	-1
PS		0	-0,1	-0,2	-0,35	-0,5
Z		0,2	0,1	0	-0,1	-0,2
NS		0,5	0,35	0,2	0,1	0
NB		1	0,65	0,45	0,35	0,3

Factorii de normare au valorile  $\beta_e = 20$ ,  $\beta_{de} = 20$ , iar  $\eta = 0,011$ ,  $D_{\max} = 0,85$ ,  $D_{\min} = 0,1$  și s-a prelevat un singur eșantion. Pentru o tensiune de alimentare de  $V_g = 10V$  comportarea la salt treaptă în mărimea de comandă între  $V_{ref1} = 20V$  și  $V_{ref2} = 30V$  este prezentată în Fig. 5.8.

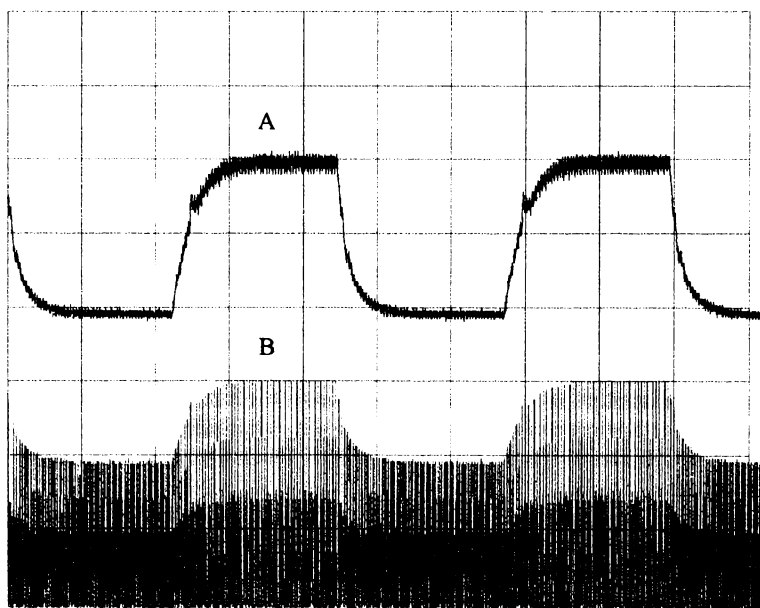


Fig. 5.8. Răspunsul convertorului BOOST, cu regulator fuzzy având tabelul de inferență cu singleton-uri, la salt în mărimea de comandă:

Trasa A - tensiunea de ieșire, 5V/div. Trasa B - tensiunea pe tranzistor, 10V/div. X=50 ms/div.

Linia de potențial 0 coincide în ambele cazuri cu ultima linie orizontală inferioară a caroiajului ecranului.

Se remarcă răspunsul fără supracreșteri și cu timp de răspuns acceptabil. Fig. 5.9. prezintă aceleași forme de undă din convertor, cu același regulator, cu evidențierea a două perioade de comutație, în situația în care tensiunea de referință este constantă,  $V_{ref} = 20V$  (starea staționară). Se observă

lipsa oricărei oscilații de înaltă frecvență. În continuare, în Tab. 5.2 sunt prezentate valorile medii ale tensiunii de ieșire pentru o gamă largă a tensiunii de intrare. Se remarcă un factor de stabilizare bun. Tensiunea medie de ieșire nu se stabilizează chiar la 20 V pentru că cea care se stabilizează este valoarea eșantionată, care nu coincide cu valoarea medie. Factorul de stabilizare nenul se datorează și zgomotului de comutație

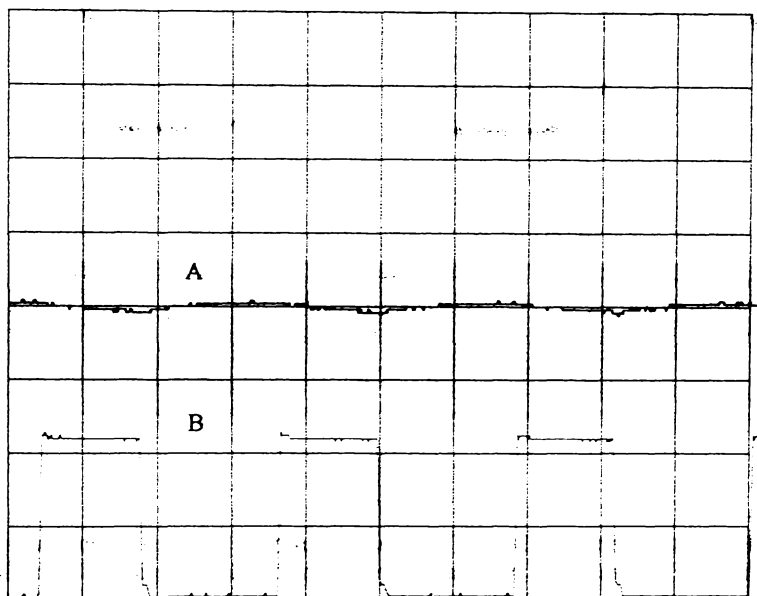


Fig. 5.9. Comportarea convertorului BOOST cu regulator fuzzy la mărime de comandă constantă (stare staționară):

Trasă A - tensiunea de ieșire, 5V/div. Trasă B - tensiunea pe tranzistor, 5V/div. X=50 μs/div.  
Linia de potențial 0 coincide în ambele cazuri cu ultima linie orizontală inferioară a caroiajului ecranului.

Tab. 5.2. Valorile medii ale tensiunii de ieșire funcție de valorile medii ale tensiunii de alimentare

$V_g$ [V]	6	7	8	9	10	11	12	13	14	15	16	17	18
$V_o$ [V]	20,35	20,3	20,3	20,2	20,15	20,1	20	20	19,94	19,93	19,92	19,92	19,91

În Fig. 5.10 se reia experimental comportării la salt în mărimea de comandă, singura modificare fiind reducerea lui  $D_{min}$  la  $D_{min} = 0,01$ . Se remarcă dispariția spike-ului de pe frontul ridicător.

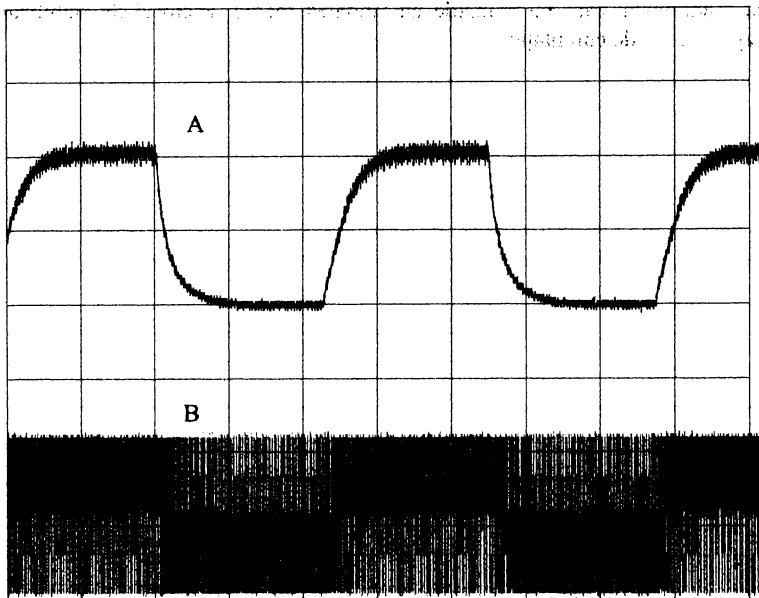


Fig. 5.10. Răspunsul convertorului BOOST cu regulator fuzzy cu tabel de inferență cu singleton-uri la salt în mărimea de comandă. Limitarea inferioară a factorului de umplere a fost mult redusă.

Trasa A - tensiunea de ieșire, 5V/div. Trasa B - tensiunea pe tranzistor, 10V/div. X=50 ms/div  
Linia de potențial 0 coincide în ambele cazuri cu ultima linie orizontală inferioară a caroiajului ecranului.

S-a studiat de asemenea comportarea regulatorului la salt în tensiunea de alimentare, între 7V și 17V, la tensiune de referință fixată,  $V_{ref} = 20V$ , rezultatele fiind cele din Fig. 5.11. Din păcate saltul în tensiunea de alimentare nu a putut fi obținut cu fronturi foarte abrupte, dar și așa se remarcă timpul de răspuns bun al regulatorului și răspunsul neoscilant. Nu se poate constata care dintre timpi este mai scurt și nici la care dintre salturi apar spike-uri pe fronturi, întrucât nu s-au putut realiza fronturi egale în tensiunea de alimentare.

Stabilizarea în raport cu sarcină s-a verificat cu schema din Fig. 5.12, la  $V_g = 10V$  și  $V_{ref} = 20V$ , rezultatele fiind cele din Fig. 5.13 și Fig. 5.14. GI este un generator de impulsuri.



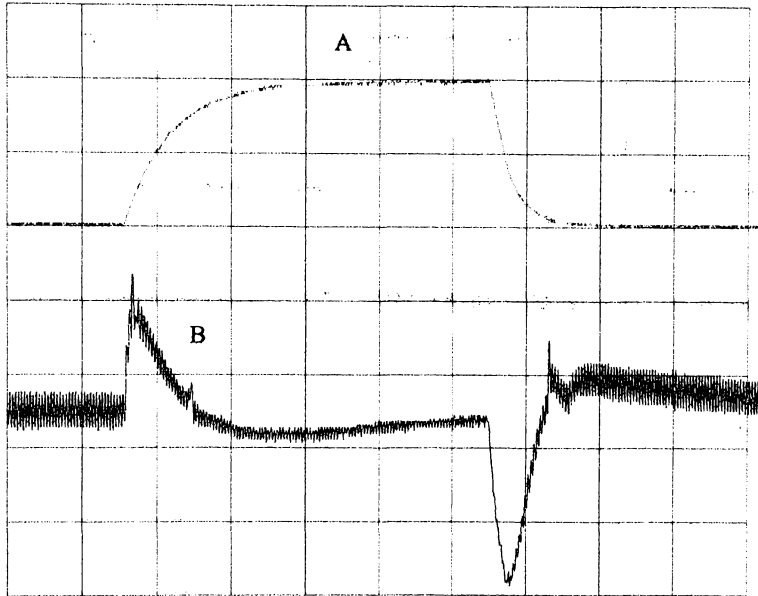


Fig. 5.11: Răspunsul convertorului BOOST cu regulator fuzzy la salt în tensiunea de alimentare (tabel de inferență cu singleton-uri):

Trasa A - tensiunea de alimentare, 5V/div. Trasa B - tensiunea de ieșire, 2V/div, fără componentă continuă. X=100 ms/div.

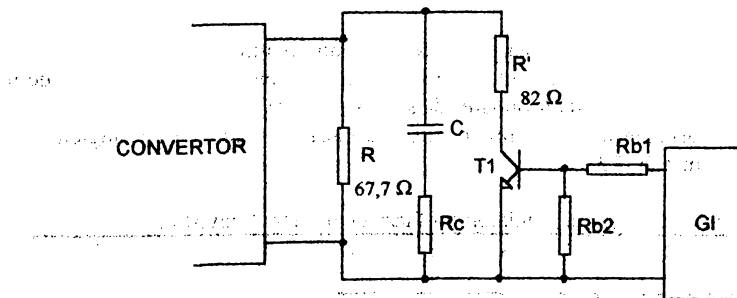


Fig. 5.12. Schema de principiu pentru studiul dinamic al stabilizării în raport cu sarcina.

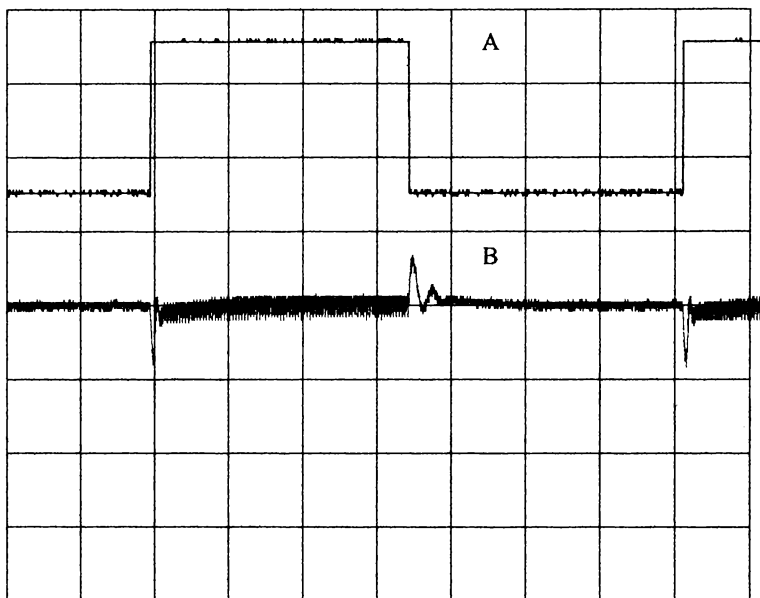


Fig. 5.13: Răspunsul convertorului BOOST cu regulator fuzzy (tabel de inferență cu singleton-uri) la salt treaptă în sarcină:

- Trasa A - tensiunea de comandă, 5V/div, "1" logic înseamnă  $T_1$  în conducție (curent mare)
  - Trasa B - tensiunea de ieșire, 5V/div. Pentru această tensiune linia de potențial 0 coincide cu ultima linie orizontală inferioară a caroiajului ecranului.
- X=20 ms/div.

La comutarea de la curent de sarcină mare la curent de sarcină mic se remarcă un ușor caracter oscilant și un timp de răspuns ușor crescut. Ambii timpi de răspuns sunt însă suficient de mici iar erorile tranzitorii sunt mici în raport cu valoarea de stare staționară.

S-a studiat de asemenea un controller fuzzy cu baza de reguli nu cu singleton-uri ci cu termeni lingvistici, ca în Tab. 5.3.

Tab. 5.3. Tabelul de reguli al regulatorului fuzzy cu termeni lingvistici.

de	NB	NS	Z	PS	PB
NB	PB	PS	Z	Z	NS
NS	PS	PS	Z	Z	NS
Z	PS	PS	Z	NS	NS
PS	PS	Z	Z	NS	NS
PB	PS	Z	Z	NS	NB

Suprafața de reglare corespunzătoare este cea din Fig. 5.15, trasarea ei făcându-se cu programul MATLAB din anexa A10.

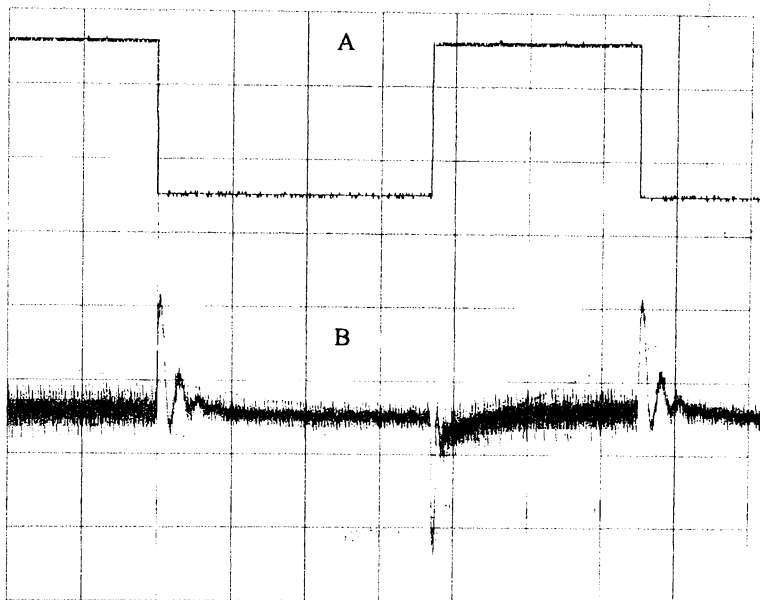


Fig. 5.14. Răspunsul convertorului BOOST cu regulator fuzzy (tabel de inferență cu singleton-uri) la salt treaptă în sarcină (detaliu):

Trasa A - tensiunea de comandă, 5V/div, "1" logic înseamnă  $T_1$  în conducție (curent mare)

Trasa B - tensiunea de ieșire, 2V/div., fără componentă continuă.

X=20 ms/div.

Cu o astfel de bază de reguli a fost reluat experimentul comportării la salt în mărirea de comandă între  $V_{ref1} = 20V$  și  $V_{ref2} = 30V$ , având  $D_{min} = 0,01$ , restul parametrilor fiind aceiași.

Rezultatele sunt date în Fig. 5.16, evidențiind un timp de răspuns mai scurt (aproximativ de 2 ori) decât în cazul utilizării bazei de reguli cu singleton-uri (vezi Fig. 5.8). De asemenea s-au prelevat 3 eșantioane din tensiunea de ieșire, observându-se o stabilizare mai bună a valorii medii datorită medierii celor trei eșantioane. Prelevarea a 3 eșantioane nu a modificat însă timpul de răspuns. Față de situația cu baza de reguli cu singleton-uri se remarcă și posibilitatea de a crește mai mult câștigul regulatorului,  $\eta$ , fără ca sistemul să devină instabil. Creșterea lui  $\eta$  duce la micșorarea și mai mult a timpilor de răspuns, dar este însoțită de fenomene oscilante și supracreșteri ce se accentuează cu creșterea lui  $\eta$ . Acest lucru este de altfel previzibil dacă se face o analogie cu teoria sistemelor liniare cu regulator PI analogic.

În Fig. 5.17 este prezentat răspunsul aceluiași regulator cu tabel de inferență cu termeni lingvistici la salturi treaptă în sarcină pentru  $V_g = 10V$  și  $V_{ref} = 20V$ . Și în acest caz timpul de răspuns este mai scurt decât în cazul regulatorului cu bază de reguli cu singleton-uri (vezi Fig. 5.12).

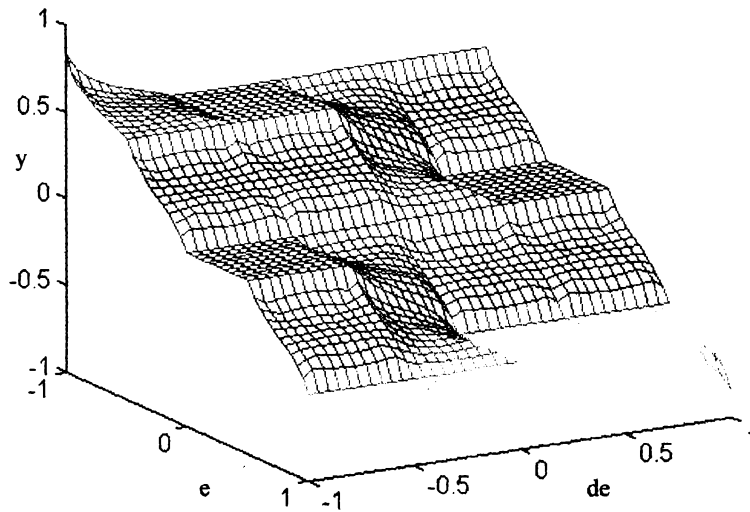


Fig. 5.15. Suprafața de reglare a regulatorului în care baza de reguli are termeni lingvistici.

În concluzie se poate afirma că utilizarea unei baze de reguli cu variabile lingvistice duce la performanțe dinamice (timp de răspuns în primul rând) mai bune. De asemenea mediile eșantioanelor sunt binevenite, ducând la o valoare medie mai apropiată de valoarea de referință în condițiile unor pulsații în tensiunea de ieșire. Limitarea artificială a factorului de umplere poate duce la distorsionarea fronturilor prin apariția unor spike-uri nedorite. Stabilizarea în raport cu tensiunea de alimentare se face pe un domeniu foarte larg și cu performanțe dinamice bune. Stabilizarea în raport cu sarcina are timpi de răspuns foarte scurți și prezintă un caracter oscilant nesemnificativ.

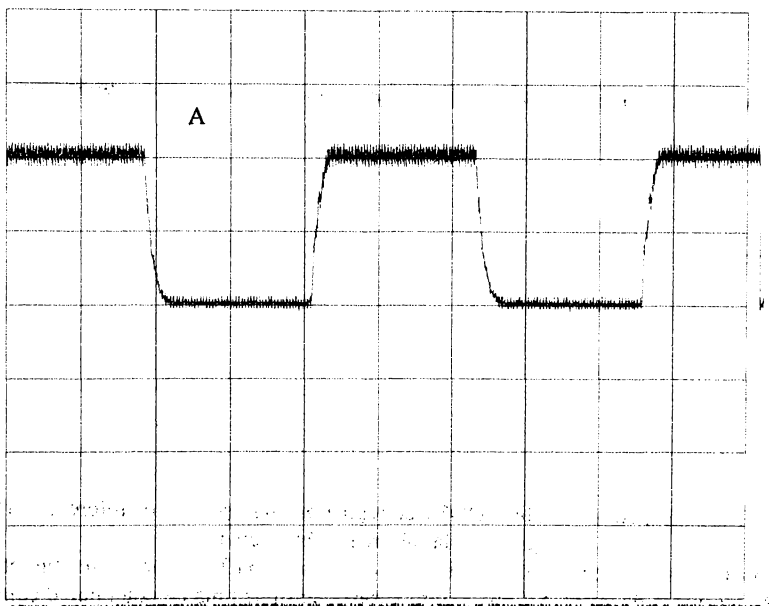


Fig. 5.16. Răspunsul convertorului BOOST cu regulator fuzzy cu bază de reguli cu termeni lingvistici, la salt treaptă în mărimea de comandă.  
 Trasa A - tensiunea de ieșire, 5V/div., X=50 ms/div. Linia de potențial 0 coincide cu ultima linie orizontală inferioară a caroiajului ecranului.  
 Trasa B - neutilizată.

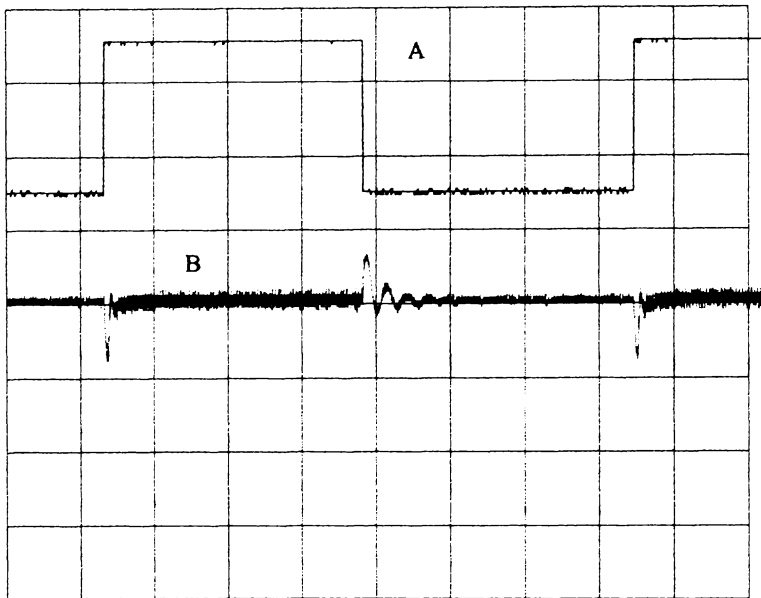


Fig. 5.17. Răspunsul convertorului BOOST cu regulator fuzzy și tabel de inferență cu termeni lingvistici, la salt treaptă în sarcină:

Trasa A - tensiunea de comandă, 5V/div, "1" logic înseamnă  $T_1$  în conducție (curent mare)  
 Trasa B - tensiunea de ieșire, 5V/div. Pentru această tensiune linia de potențial 0 coincide cu ultima linie orizontală inferioară a caroiajului ecranului.  
 $X=20$  ms/div.

#### 5.4 Comanda cu rețele neuronale

Ca și comanda fuzzy, comanda cu rețele neuronale este abia la începuturi în domeniul convertoarelor dc-dc, deși în alte domenii ale electronicii de putere ea a penetrat semnificativ. Autorul a investigat prin simulare comportarea unui convertor rezonant serie (SRC) comandat cu o rețea neuronală. Convertorul rezonant serie în semipunte este cel din Fig. 5.18. S-a plecat de la comanda acestui convertor pe traiectorie optimală [142] [143], presupusă cunoscută. O traiectorie în spațiul stărilor este reprezentată în Fig. 5.19. Ea se compune din 4 arce de cerc având centrele în punctele  $\pm 1 \pm V_{oN}$ . Normarea tensiunilor s-a făcut cu  $V_g$ , iar a curenților cu  $\frac{V_g}{Z_0}$  unde  $Z_0 = \sqrt{\frac{L_r}{C_r}}$ ,

toate mărimile fiind raportate la primar (fapt ce face ca pentru normarea mărimilor din secundar să apară raportul de transformare  $n$ ). Comanda pe traiectorie optimală se realizează astfel: După blocarea naturală a unuia dintre tranzistoare este monitorizată distanța până la traiectorie măsurată din centrul cercului aferent celuilalt tranzistor. Când această distanță scade sub valoarea dorită a

razei pentru traiectoria de tranzistor,  $R_{ref}$ , se dă comanda de aducere în conducție pentru celălalt tranzistor.

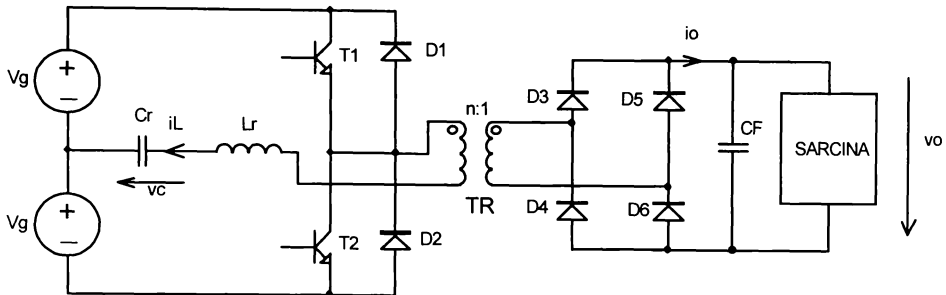


Fig. 5.18. Convertorul SRC comandat cu o rețea neuronală.

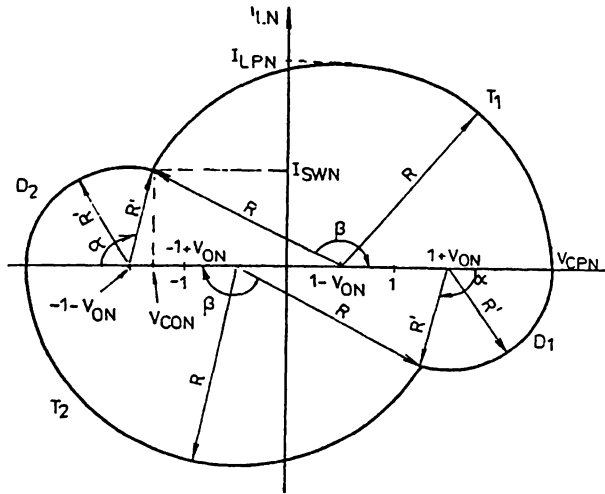


Fig. 5.19. Traiectoria tipică în spațiul stărilor.

Se poate demonstra ușor că în acest fel comutarea unuia dintre tranzistoare este dată de semnul curentului prin  $L_r$  și că, dacă se iau valorile în modul, comutarea se reduce la un singur cadran. Astfel toate mărimile sunt convertite la cadranul II, corespunzător aducerii în conducție a lui  $T_2$ . Practic se lucrează cu mărimile în regiunea  $|v_{CN}| - |i_{LN}|$ . Fie tensiunea de alimentare  $V_g$  constantă și fie  $V_o$  tensiunea de ieșire dorită. Fișierele MATLAB cu programele de antrenare și simulare sunt prezentate în Anexa A10.

În comanda cu rețeaua neuronală spațiul stărilor este împărțit în mod echidistant pe orizontală și verticală. Fie  $[-V_{Cmax}, V_{Cmax}]$  domeniul simetric ales pentru tensiunea pe capacitatea rezonantă și  $[-I_{Lmax}, I_{Lmax}]$  cel pentru curentul rezonant și fie  $\Delta V_C$  și  $\Delta I_L$  pasul pentru domeniul tensiunii respectiv cel pentru domeniul curentului. De asemenea același lucru se face pentru domeniul curentului de sarcină,  $[I_{o min}, I_{o max}]$  cu pasul  $\Delta I_o$ .

În primă fază, pentru rețeaua de puncte astfel creată în planul stărilor și pentru valorile de curent de ieșire rezultate prin împărțirea domeniului curentului de ieșire se stabilesc stările în care trebuie să se afle întrerupătoarele (fișierul *train.m*). În acest scop, pentru tensiunea de ieșire dorită și curentul de sarcină  $I_o$  curent (una din valorile din setul rezultat din împărțirea domeniului de variație al curentului) se calculează raza de referință curentă,  $R_c$ . Se ține cont că pentru o traiectorie oarecare de rază  $R$  avem (notațiile sunt cele din Fig. 5.18):

$$V_{CPN} = R + 1 - V_{oN} \quad (8)$$

$$I_{LAVN} = \frac{2V_{CPN}}{\alpha + \beta} \quad (9)$$

Dar

$$\alpha = \arccos\left(\frac{RV_{oN} - V_{oN}^2 - 1}{R - 2V_{oN}}\right) \quad \beta = \arccos\left(\frac{V_{oN}^2 - RV_{oN} - 1}{R}\right) \quad (10)$$

$$I_{LAVN} = \frac{\frac{I_o}{V_g}}{\frac{Z_0}{Z_0}} = \frac{\frac{V_o}{nR_{sarc}}}{\frac{V_g}{Z_0}} = \frac{V_o}{nV_g} \frac{Z_0}{R_{sarc}} = \frac{V_{oN}}{R_{sarcN}} \quad (11)$$

unde

$$V_{oN} = \frac{nV_o}{V_g}, \quad R_{sarcN} = \frac{R_{sarc}}{Z_0} \quad (12)$$

Substituind relațiile (10) și relația (11) în (9), exprimând din relația astfel obținută pe  $V_{CPN}$  și egalându-l cu cel dat de relația (8) **autorul tezei** obține ecuația razei traiectoriei în forma:

$$2(R + 1 - V_{oN}) - \frac{V_{oN}}{R_{sarcN}} \left( \arccos\left(\frac{RV_{oN} - V_{oN}^2 - 1}{R - 2V_{oN}}\right) + \arccos\left(\frac{V_{oN}^2 - RV_{oN} - 1}{R}\right) \right) \quad (13)$$

Rezolvarea acestei ecuații se face în fișierul *train.m* prin metoda Newton-Raphson în care procesul iterativ este întotdeauna convergent dacă, așa cum propune **autorul** prima soluție se alege:

$$R_{c0} = 1 + V_{oN} + \varepsilon \quad (14)$$

unde  $\varepsilon$  este un număr pozitiv suficient de mic (în program s-a luat egal cu  $10^{-5}$ ). După calculul razei curente corespunzătoare traiectoriei cu  $V_o$  dorit și  $I_o$  curent se compară această rază cu raza determinată de perechea de puncte curente  $v_{CN} - i_{LN}$  din rețeaua generată în planul stărilor. Întrucât s-a raportat totul la cadranul II distanța se va măsura față de centrul traiectoriei lui  $T_2$ , adică punctul  $(-1 + V_{oN}, 0)$ , având expresia:

$$d = \sqrt{(v_{CN} + 1 - V_{oN})^2 + i_{LN}^2} \quad (15)$$

Dacă această distanță este mai mică decât raza curentă atunci conform principiului comenzii pe traiectorie optimă tranzistorul trebuie adus în conducție, în caz contrar va rămâne în continuare blocat. În acest fel semnalul  $Q$ , prin care se marchează conducția tranzistorului, va avea valorile 1 sau 0. Efectuând acest lucru pentru toate tripletele  $v_C - i_L - I_o$  se obține starea de conducție a tranzistorului. Pentru că programul Neural Graphics, prin care se va antrena rețeaua, impune ca intrările să fie subunitare, intervalul tensiunii pe condensatorul rezonant se normalizează cu  $V_{Cmax}$ , al curentului prin bobina rezonantă cu  $I_{Lmax}$ , iar cel al curentului de sarcină cu  $I_{o,max}$ . Fișierul *train.m* generează fișierul *srcclass.dat*, în formatul acceptat de programul Neural Graphics. În urma antrenării rețelei rezultă fișierul de ponderi și praguri *srcclass.wei*. În continuare fișierul *neur\_net.m* este cel care calculează răspunsul rețelei antrenate la o tripletă oarecare normată  $v_C - i_L - I_o$ . El va fi apelat din programul de simulare *src\_neur.m* care simulează comportarea controllerului neuronal la un salt treaptă în rezistența de sarcină.



Rețeaua neuronală aleasă are 3 intrări, 2 ieșiri și 2 straturi ascunse, unul cu 7 neuroni, celălalt cu 3 neuroni - Fig. 5.20.

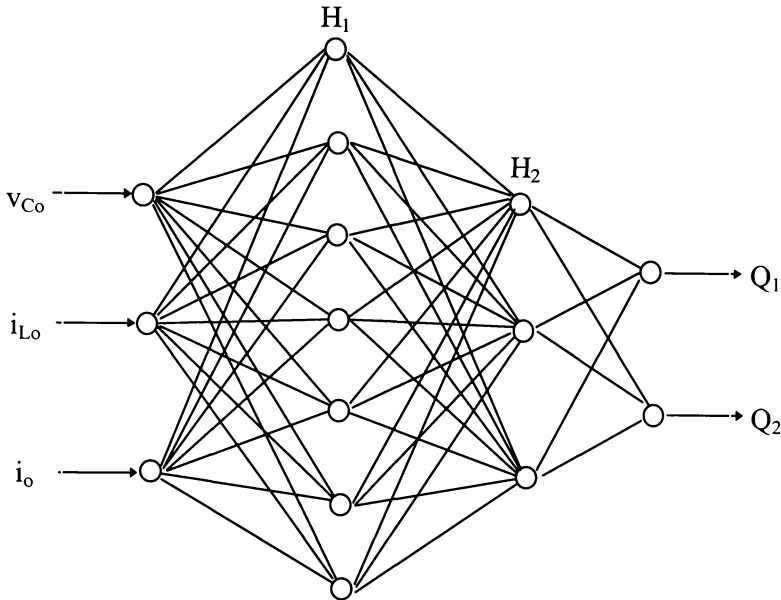


Fig. 5.20. Rețeaua neuronală utilizată în controller.

Metoda de antrenare utilizată a fost cu propagarea înapoi a erorii ("back propagation"). Rezultatele simulării sunt prezentate în Fig. 5.21 și Fig. 5.22. Convertorul are următorii parametri:

$$V_g = 25V; \quad L_r = 81,8 \mu H; \quad C_r = 125 nF; \quad C_f = 20 \mu F; \quad V_{oref} = 12V; \quad R_{sarc} = 10 \rightarrow 4 \Omega$$

În Fig. 5.21 se observă evoluția sistemului în planul stărilor, remarcându-se un fenomen tranzitoriu de numai două traiectorii. Faptul că se realizează o bună stabilizare și un timp de răspuns bun la o variație de 250% a sarcinii se observă și în Fig. 5.21, Fig. 5.22 și Fig. 5.23.

Desigur aceste estimări teoretice vor trebui confirmate și în practică, unde controllerul trebuie transpus fără a avea o schemă deosebit de complexă. Se mai remarcă faptul că sunt necesare trei traductoare pentru citirea celor trei mărimi de intrare în rețeaua neuronală, ceea ce, pe lângă multe alte considerente, face ca alegerea controllerului neuronal sau clasic să se facă cu multă precauție.

## 5.5 Concluzii

Capitolul de mai sus și-a propus trecerea în revistă a principalelor tehnici de comandă analogice și un studiu al implementării de controllere fuzzy și neuronale în convertoarele dc-dc. Deși aplicația de bază a controllerelor fuzzy rămâne comanda servomotoarelor ele pot fi utilizate cu succes și în convertoarele dc-dc, deschizătorul de drum în acest domeniu fiind da Silva [168]. În privința utilizării rețelelor neuronale, deși acestea își găsesc aplicații în special în probleme de estimare și identificare, ele pot fi cu succes utilizate și în comanda convertoarelor dc-dc.

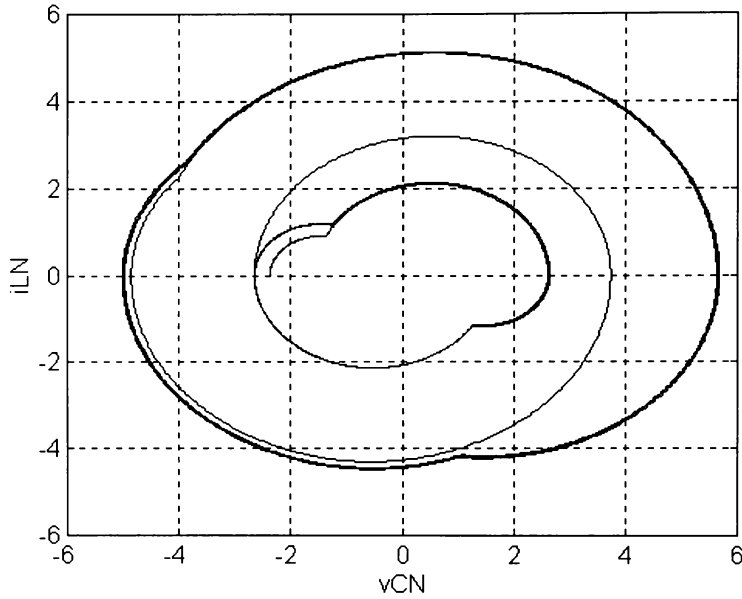


Fig. 5.21. Evoluția în spațiul stărilor a convertorului SRC cu controller neuronal.

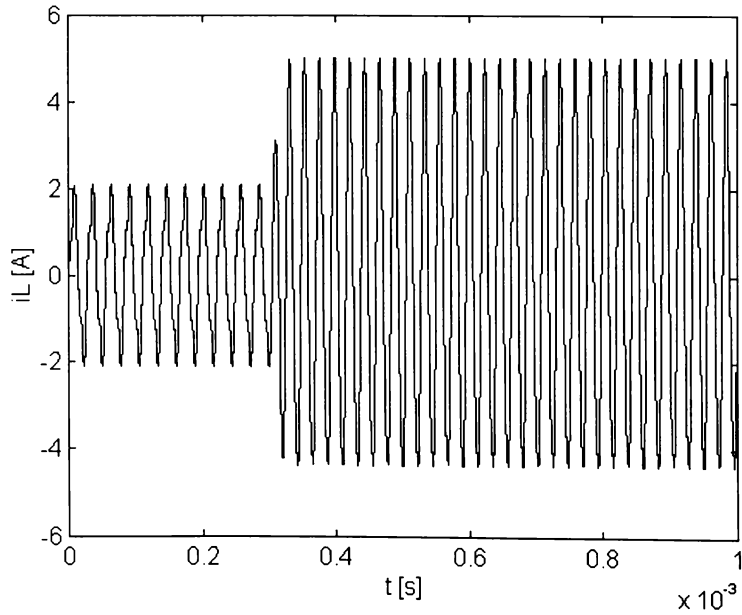


Fig. 5.22. Evoluția curentului prin elementele rezonante.

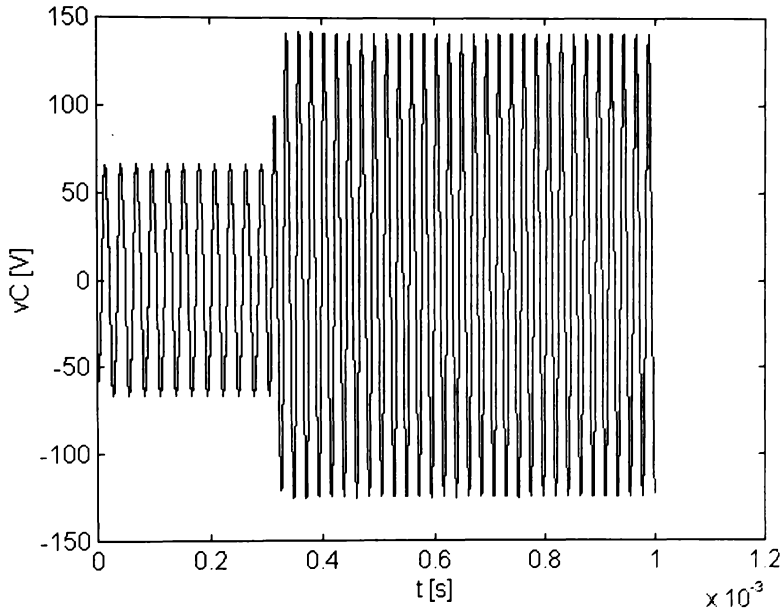


Fig. 5.23. Evoluția tensiunii pe capacitatea rezonantă.

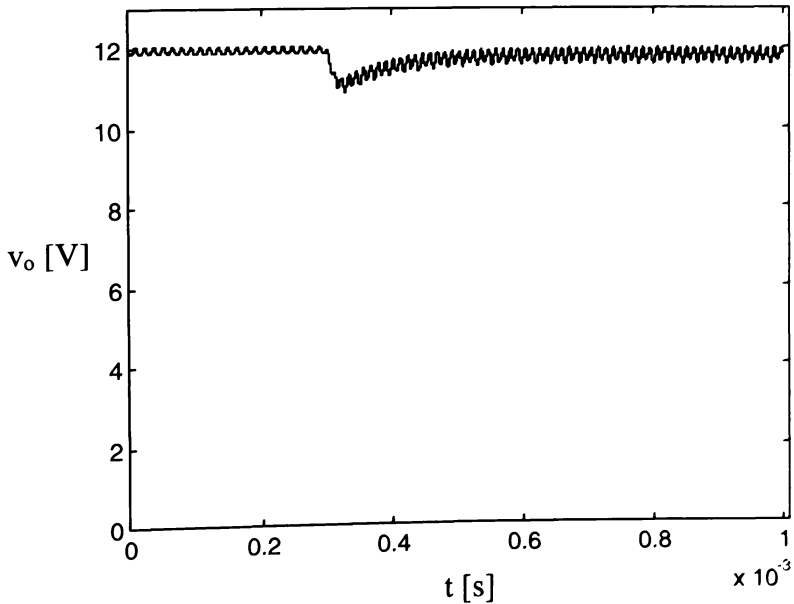


Fig. 5.24. Evoluția tensiunii de ieșire a convertorului SRC cu controller neuronal la salt treaptă în sarcină.

Contribuțiile autorului privind tehnicile de comandă a convertoarelor dc-dc sunt legate de:

- propunerea tehnicii integrative de comandă mixte (paragraful 5.2.3) ca generalizare a comenzii prin sarcină și a comenzii integrative după o variabilă.
- un studiu, însoțit de experimente, asupra comenzii fuzzy a unui convertor PWM dc-dc cu evidențierea posibilității de realizare de controllere fuzzy performante (paragraful 5.3).
- un studiu prin simulare legat de fezabilitatea comenzii cu controller neuronal a unui convertor rezonant serie (paragraful 5.4).

Autorul este conștient de faptul că mai există numeroase aspecte de aprofundat, legate de exemplu de alegerea judicioasă a funcțiilor de apartenență, alegerea optimă și generală a coeficienților de normare, a numărului de straturi ascunse și a numărului de neuroni din acestea, etc.

În situația în care domeniul electronicii de putere a devenit deja atât de complex și interdisciplinar, în conjuncție cu domeniul exploziv al procesoarelor de semnal, este de așteptat ca tehnicile fuzzy și neuronale să aibă pe viitor un impact puternic, oferind numeroase provocări comunității inginerilor specialiști în electronica de putere.

## CAPITOLUL 6

### Modelarea convertoarelor dc-dc

Așa cum se cunoaște, problematica modelării convertoarelor de putere este extrem de vastă și de dinamică. Spațiul nu permite un studiu profund de prezentare, clasificare și comparare a tehnicilor de modelare existente, deși acest lucru ar fi extraordinar de benefic în acest moment.

În primul paragraf al capitolului se va face o sumară trecere în revistă a principalelor tehnici de modelare, urmând ca toate celelalte paragrafe, de la 6.2 până la 6.8, să prezinte numai contribuțiile autorului în domeniul modelării și analizei convertoarelor dc-dc.

#### 6.1 O scurtă trecere în revistă a principalelor tehnici de modelare

Cel mai exact model dinamic pentru un convertor în comutație se obține din setul de ecuații diferențiale ce îi descriu funcționarea și propagarea puterii în sistem. Datorită fenomenelor de comutație aceste ecuații diferențiale își modifică forma (se spune, oarecum impropriu, că sunt discontinue), ceea ce le face dificil de rezolvat. Simulatoarele, privite într-o abordare generală, sunt singurele care rezolvă numeric, dar încă aproximativ, aceste ecuații diferențiale, ținând cont de discontinuitățile acestora. Cu toate acestea, deși simulatoarele sunt aproape un etalon în stabilirea gradului de precizie al unui model sau altul, atât la semnal mare cât și la semnal mic, au dezavantajul că nu oferă modele analitice, cu formule închise, care să poată fi ulterior utilizate în analiză. De aceea s-au elaborat modele noi, continue sau discrete, bazate pe anumite simplificări.

În Fig. 6.1 este arătat un model general al unui sistem dinamic. Acesta este caracterizat printr-un set de variabile de stare,  $x$ , un set de variabile de intrare,  $u$ , reprezentând acțiunea mediului în care se află sistemul și ieșirile  $y$ . Acest tip de model dinamic poate fi utilizat în 3 moduri:

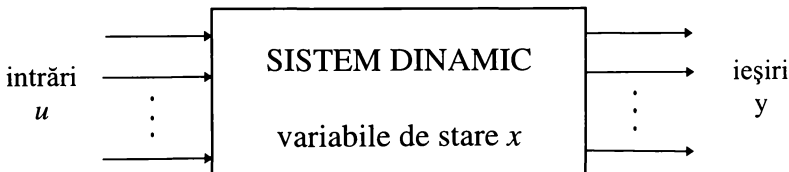


Fig. 6.1. Modelul general al unui sistem dinamic.

1. *Analiză.* Pentru un vector de intrare  $u$  dat, cunoscând starea prezentă a vectorului de stare  $x$ , evoluția viitoare a sistemului (deci și a ieșirilor, care sunt elementele de interes) este cunoscută.
2. *Identificare.* Pentru o istoricitate temporală a intrărilor și ieșirilor cunoscută, de regulă obținută experimental, se poate găsi pentru sistem un model (deci și un set de variabile de stare  $x$ ) consistent cu  $u$  și  $y$ . Aceasta este de fapt esența experimentelor științifice; un model exact și valoros fiind acel model care descrie corect comportarea pentru o gamă largă de vectori  $u$  și  $y$ .
3. *Sinteză.* Pentru intrări  $u$  precizate și pentru anumite ieșiri  $y$  dorite se caută un model astfel încât modelul având ca intrări pe  $u$  să producă ieșirile  $y$  dorite. Sinteza se face în convertoarele dc-dc într-un context limitat, acolo unde metodele de sinteză sunt aplicabile direct.

Majoritatea metodelor de analiză dinamică a convertoarelor dc-dc se bazează pe mediere și/sau liniarizare [63]. Mediarea elimină influența pulsațiilor de frecvențe înalte (peste jumătate din frecvența de comutație), dar furnizează relații analitice între mărimile mediate, simplificând astfel analiza. Cu toate acestea modelul obținut prin mediere este de multe ori neliniar. Examinarea stabilității și proiectarea de controllere pentru sistemele neliniare prezintă serioase dificultăți. Liniarizarea este aceea care furnizează un model liniar care descrie aproximativ, dar cu suficientă precizie, comportarea sistemului la perturbații și deviații mici din funcționarea nominală. Stabilitatea modelelor de semnal mic garantează stabilitatea sistemului cel puțin la perturbații mici.

Tehnicile de modelare pot fi analitice (orientate pe ecuații) sau orientate pe circuit. Acestea din urmă sunt mai ilustrative, dar într-un context mai larg ele sunt echivalente cu cele orientate pe ecuații.

Literatura de specialitate abundă în tehnici de modelare, cu mai multe sau mai puține puncte comune. Fără a avea pretenția de a le fi epuizat pe toate, se vor prezenta succint mai jos cele mai reprezentative, unanim acceptate și utilizate dintre ele.

1. *Modelarea orientată pe dispozitivele neliniare.* Este o metodă de modelare care "izolează" din convertoare grupul de întrerupătoare care conferă neliniaritate acestuia. Implicit, în obținerea modelului, pentru ca acesta să fie simplu, se face și o operație de mediere. Avantajele acestei tehnici de modelare sunt simplitatea modelului și aplicarea imediată, fără dificultăți, în analiza convertoarelor. Dezavantajele sunt legate de valabilitatea numai pentru anumite moduri de funcționare și inaplicabilitatea în cazul în care se dorește luarea în considerație și a elementelor parazite. În [204] se prezintă modele pentru întrerupătoare ideale din convertoare rezonante și PWM, privind rețeaua de comutație ca un dispozitiv cu trei terminale și efectuând o analiză asemănătoare cu cea din modelarea tranzistorului. În [34] este efectuată o analiză similară, numai pentru convertoare PWM, considerând însă și elementele parazite ale dispozitivelor reactive. Un model de întrerupător PWM corelat cu frecvența de interes în analiză este raportat în [196].
2. *Metoda extinsă a injecției-absorbției de curent a lui Kislovski și Chetty* [18], [19], [20], [21],[69] oferă un model de semnal mic și de frecvență joasă pentru că implicit se admite existența unei relații liniare între perturbațiile de semnal mic. Privind convertorul ca un diport - Fig. 6.2 - curentul mediat injectat,  $i_i$  și curentul mediat absorbit,  $i_o$ , sunt dați de relațiile:

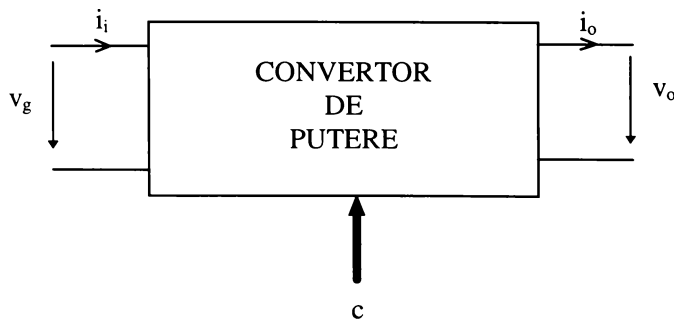


Fig. 6.2. Explicativă la metoda extinsă a injecției-absorbției de curent.

$$i_i = i_i(x, v_g, v_o) \quad (1)$$

$$i_o = i_o(x, v_g, v_o) \quad (2)$$

$$\hat{i}_i = \frac{\partial i_i}{\partial c} \hat{x} + \frac{\partial i_i}{\partial v_g} \hat{v}_g + \frac{\partial i_i}{\partial v_o} \hat{v}_o \quad (3)$$

$$\hat{i}_o = \frac{\partial i_o}{\partial c} \hat{x} + \frac{\partial i_o}{\partial v_g} \hat{v}_g + \frac{\partial i_o}{\partial v_o} \hat{v}_o \quad (4)$$

unde simbolul “^” desemnează perturbațiile de semnal mic, iar  $c$  mărimile de comandă.

Derivatele parțiale se evaluează în punctul static de funcționare și sunt de regulă constante, ceea ce face ca relațiile (3) și (4) să fie liniare. Din aceste ecuații liniare obținute se poate găsi un model al celulelor de comutație și funcțiile de transfer de interes.

În cazul în care curentul de intrare sau de ieșire depind și de o mărime suplimentară (de exemplu curentul printr-o bobină internă din convertor) este necesară o ecuație suplimentară. Acest lucru se întâmplă în CCM, ecuația adițională fiind legată de energia stocată. Aplicarea metodei la convertoare cvasirezonante este raportată în [51].

3. *Medierea în spațiul stărilor*, datorată lui Čuk și Middlebrook [126], [127], [128], are calitatea unei abordări dintr-o reprezentare standardizată și generalizată a sistemului, constând în ecuațiile în spațiul stărilor pentru fiecare stare topologică a circuitului. Ideea este de a înlocui cele două sau mai multe ecuații în spațiul stărilor, aferente stărilor topologice, cu una singură care caracterizează global comportamentul mediu într-o perioadă de comutație. Cea mai uzitată este scrierea matricială a ecuațiilor de stare în forma:

$$\begin{cases} \frac{dx}{dt} = A_i x + B_i u \\ y = E_i x + F_i u \end{cases} \quad i = \overline{1, N_s} \quad (5)$$

unde  $N_s$  este numărul stărilor topologice dintr-o perioadă. De exemplu într-un convertor dc-dc CCM comandat în factor de umplere avem  $i = 2$ ,  $u = v_g$  iar matricile mediate sunt date de:

$$\begin{cases} A = dA_1 + (1-d)A_2 \\ B = dB_1 + (1-d)B_2 \\ E = dE_1 + (1-d)E_2 \\ F = dF_1 + (1-d)F_2 \end{cases} \quad (6)$$

unde aici  $d$  reprezintă factorul de umplere continuu, adică mediata funcției de comutație,  $\delta(t)$ :

$$d = \bar{\delta}(t) \quad (7)$$

(bara orizontală superioară denotă medierea), iar ecuațiile modelului mediat în spațiul stărilor vor fi:

$$\begin{cases} \frac{d\bar{x}}{dt} = A\bar{x} + B\bar{v}_g \\ \bar{y} = E\bar{x} + F\bar{v}_g \end{cases} \quad (8)$$

Soluția stării staționare este constantă fiind dată de (8) cu  $d = D$ ,  $v_g = V_g$  și anulând derivata:

$$\begin{cases} X = -A_D^{-1} B_D V_g \\ Y = (F - E_D A_D^{-1} B_D) V_g \end{cases} \quad (9)$$

unde  $A_D = A_{i/d=D}$ , etc. Soluția de semnal mic se obține perturbând (8), ținând cont de (6) și făcând ipoteza de semnal mic. După reducerea termenilor corespunzători stării staționare și neglijarea termenilor de ordin mai mare sau egal cu 2 se obține:

$$\begin{cases} \frac{d\hat{x}}{dt} = [(A_1 - A_2)X + (B_1 - B_2)V_g] \hat{d} + A_D \hat{x} + B_D \hat{v}_g \\ \hat{y} = [(E_1 - E_2)X + (F_1 - F_2)V_g] \hat{d} + E_D \hat{x} + F_D \hat{v}_g \end{cases} \quad (10)$$

Se notează:

$$\begin{cases} \xi_D = (A_1 - A_2)X + (B_1 - B_2)V_g \\ \zeta_D = (E_1 - E_2)X + (F_1 - F_2)V_g \end{cases} \quad (11)$$

și luând în (10) transformata Laplace se obține:

$$\hat{y} = \left[ E_D(sI - A_D)^{-1} \xi_D + \zeta_D \right] \hat{d} + \left[ E_D(sI - A_D)^{-1} B_D + F_D \right] \hat{v}_g \quad (12)$$

de unde rezultă imediat funcțiile de transfer dorite.

Analiza de semnal mic a convertoarelor rezonante [200] apelează și ea la modele în spațiul stărilor. De data aceasta însă circuitul rezonant are perioada de oscilație apropiată de perioada de comutație, ceea ce face ca ipoteza pulsațiilor lente și a pulsațiilor mici să nu mai poată fi aplicată. Analiza de semnal mic a acestor convertoare se bazează pe următoarea relație:

$$e^{At_s} = e^{A(T_s + \hat{t}_s)} = e^{AT_s} e^{A\hat{t}_s} \cong e^{AT_s} (I + A\hat{t}_s) \quad (13)$$

Modelele dezvoltate rămân însă cu destule limitări, motiv pentru care elaborarea de modele de semnal mic performante pentru convertoare rezonante rămâne încă o provocare importantă, la care primii pași au fost făcuți de câțiva autori: King și Stuart [67], Nakahara ș. a. [132], Witulski, Hernandez [211], [212], Vorperian [201].

În concluzie, medierea este o operație care se face în ipoteza unor variații lente ale mărimilor de stare pe durata unei perioade de comutație, transformând modelul nelinier și variant în timp (mai concret, liniar pe porțiuni) al convertorului într-unul invariant în timp, dar de obicei tot nelinier. Aceasta, împreună cu definiția factorului de umplere continuu, face ca valabilitatea modelului să fie restricționată până la jumătate din frecvența de comutație. Dacă modelului mediat  $i$  se aplică și liniarizarea, în ipoteza de semnal mic, atunci restricției în frecvență  $i$  se adaugă și aplicabilitatea numai pentru perturbații mici, dar cu avantajul unui model liniar și invariant în timp (LTI). Modele mediate mai precise, dar și mai greu de manipulat sunt date în [104]. Generalizarea operației de mediere este efectuată elegant în [164].

4. *Modelele discrete.* Sunt de fapt modele eșantionate, dezvoltate pentru prima oară de Verghese, Elbuluk și Kassakian [198] pentru determinarea răspunsului în frecvență la convertoarele cu comandă în curent. Modelul eșantionat descrie evoluția variabilelor de stare, stabilind o legătură între valoarea acestora de la începutul unei perioade de comutație și valoarea de la începutul perioadei următoare. Modul de obținere a unor astfel de modele este excelent descris în [63] și are la bază următoarea relație:

$$x(t_{i+1}) = e^{A_i(t_{i+1}-t_i)} x(t_i) + \int_{t_i}^{t_{i+1}} e^{A_i(t_{i+1}-\tau)} B_i u_i(\tau) d\tau \quad (14)$$

unde s-a considerat că starea topologică  $i$  începe la  $t_i$ , se termină la  $t_{i+1}$  și este caracterizată de matricile  $A_i$  și  $B_i$ , modelul discret obținându-se prin invocarea continuității variabilelor de stare, ceea ce permite "legarea" stărilor topologice între ele. Modelul de semnal mic se obține dezvoltând (14) în serie Taylor și reținând numai termenii până la ordinul I inclusiv. Se obține în final o ecuație cu diferențe finite liniară și invariantă în timp, cu perioada de eșantionare egală cu perioada de comutație. O abordare cu caracter mai general, dar similară în multe privințe cu cele expuse, este efectuată de Visser și van den Bosch în [199]. Caracterul general al modelelor discrete cu eșantionare le face aplicabile la orice convertor cu traectorie ciclică, așa cum arată Elbuluk ș.a. în [35], unde se dezvoltă modele și tehnici de comandă discrete pentru convertoare rezonante. Eventuala dificultate care ar putea apare ar fi necesitatea obținerii unei soluții nominale periodice ca prim pas în analiză.

5. *Modelele în planul stărilor,* introduse de Oruganti și Lee [142], [141], [61], reprezintă modele continuele aplicabile convertoarelor rezonante. Comanda pe traiectorie optimală bazată pe un astfel de model s-a dovedit deosebit de performantă [143], [140]. Modelele în planul stărilor au însă o limitare decisivă, aceea că nu se pot aplica decât sistemelor de ordinul 2. Este dificil să se includă variabilele de stare adiționale legate de filtrele asociate cu sursa și sarcina (caz în care discuția s-ar purta într-un hiperplan al stărilor). Totuși o extensie a metodei a fost raportată de



Cheron și Foch în [17]. Luarea în considerație a snubber-elor duce la complicații dificil de depășit.

6. *Modelarea prin grafuri de fluentă* a fost propusă pentru prima oară de Smedley [173] pentru convertoare PWM funcționând în CCM. Se introduc conceptele de ramuri comutate și se elaborează corespundențele acestor ramuri comutate în modelul de semnal mare (sunt înlocuite cu multiplicatoare) și în modelul de semnal mic (sumare de ramuri). O dată aceste reguli stabilite, avantajele esențiale ale acestei tehnici de modelare sunt translatarea directă a convertorului în modelul dinamic grafic corespunzător și posibilitatea utilizării formulei lui Mason pe grafurile de stare staționară și de semnal mic în vederea obținerii directe a caracteristicilor statice și a funcțiilor de transfer. Modelarea prin grafuri de fluentă a fost extinsă și la convertoare cu mai mult de două stări [112]. Deși promițătoare la început, această tehnică nu poate fi aplicată convertoarelor cu timpi de comutație comandați indirect (de exemplu DICM, DCVM), ceea ce a redus din entuziasmul inițial. De fapt ea presupune o mediere implicită atunci când, plecând de la ramurile comutate, se obțin ramurile de semnal mare și de aceea este în fond o reprezentare grafică a modelelor mediate, păstrând însă avantajul rezolvării simple prin aplicarea formulei lui Mason.

### 6.2. Modele canonice. Modele canonice generalizate. Problema filtrului de intrare.

Este de așteptat ca acele convertoare care au proprietăți fizice asemănătoare să aibă și modele echivalente, similare calitativ. S-a arătat [126], [127] că într-adevăr se poate construi un model mediat, numit *model canonic* pentru convertoare PWM CCM a cărei arhitectură este cea din Fig. 6.3 De remarcat că modelul este atât de curent continuu cât și de semnal mic.

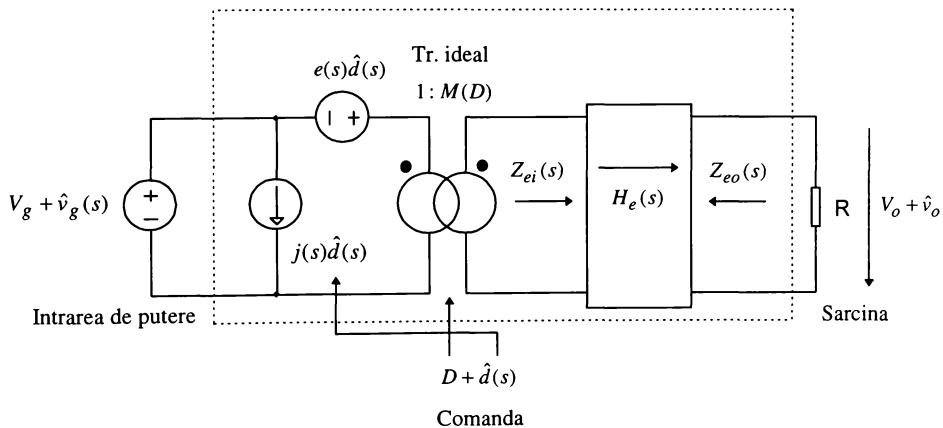


Fig. 6.3. Modelul canonic pentru convertoare PWM.

De exemplu pentru convertoare ideale cu o bobină și o capacitate (BUCK, BOOST, BUCK-BOOST), filtrul din model este de ordinul 2 și modelul echivalent devine cel din Fig. 6.4 (s-a presupus că pentru convertorul BUCK-BOOST capacitatea este conectată tot în paralel cu sarcina). Orice circuit de semnal mic al unui astfel de convertor poate fi adus la această formă, tehnica de transformare fiind dată în [ 39, paragraful 7.6.2]. Acest model canonic este utilizat pentru analiza fenomenelor din convertoare de o manieră generală, fără o referire specifică la un convertor anume.

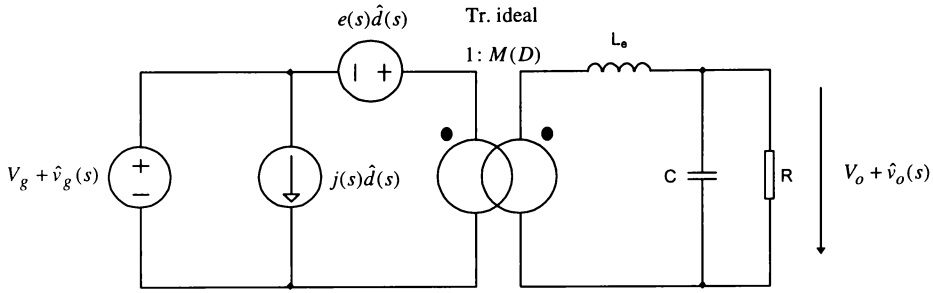


Fig. 6.4. Modelul canonic pentru convertore PWM ideale cu o bobină și un condensator.

Deși forma modelului canonic este aceeași, parametrii modelului depind de topologia fiecărui convertor, pentru cele trei convertore din familie fiind specificați în Tab. 6.1.

Tab. 6.1. Parametrii modelului canonic pentru convertore cu o bobină și un condensator.

CONVERTORUL	$M(D)$	$L_e$	$e(s)$	$j(s)$
BUCK	$D$	$L$	$\frac{V_g}{D}$	$\frac{DV_g}{R}$
BOOST	$\frac{1}{1-D}$	$\frac{L}{(1-D)^2}$	$\frac{V_g}{1-D} \left( 1 - \frac{sL}{R(1-D)^2} \right)$	$\frac{V_g}{R(1-D)^3}$
BUCK-BOOST	$-\frac{D}{1-D}$	$\frac{L}{(1-D)^2}$	$-\frac{V_g}{D(1-D)} \left( 1 - \frac{sDL}{R(1-D)^2} \right)$	$-\frac{DV_g}{R(1-D)^3}$

De subliniat că elementele filtrului trece-jos depind de punctul static de funcționare și de aceea și funcțiile de transfer și impedanțele de intrare și de ieșire depind și ele de punctul static de funcționare. Versiunile cu transformator ale convertoarelor din această familie, de exemplu convertorele FORWARD, FLYBACK sau PUNTE COMPLETĂ se modelează identic, făcându-se numai o corecție multiplicativă prin raportul de transformare.

Modelul canonic se poate extinde [210] și pentru convertore PWM funcționând în alte moduri (DICM, DCVM) precum și la convertore rezonante, având în vedere că acestea provin din convertorele PWM. Elementul cheie care realizează acest lucru este factorul de umplere echivalent  $\mu$  (vezi capitolul 4) care se cunoaște că depinde numai de mărimea de comandă,  $c$  și raportul

$$\frac{v_{off}}{i_{on}}$$

$$\mu = \mu\left(\frac{v_{off}}{i_{on}}, c\right) \tag{15a}$$

și deci

$$\hat{\mu} = \frac{\partial \mu}{\partial v_{off}} \Big|_{PSF} \cdot \hat{v}_{off} + \frac{\partial \mu}{\partial i_{on}} \Big|_{PSF} \cdot \hat{i}_{on} + \frac{\partial \mu}{\partial c} \Big|_{PSF} \cdot \hat{c} \tag{15b}$$

Practic *modelul canonic generalizat* la convertore PWM funcționând în diverse moduri și la convertore QRC, are aceeași arhitectură cu modelul de la convertorele PWM CCM, exceptând

faptul că factorul de umplere  $d$  se înlocuiește cu factorul de umplere echivalent  $\mu$ , ca în Fig. 6.5, unde  $\mu_0$  semnifică factorul de umplere echivalent din starea staționară.

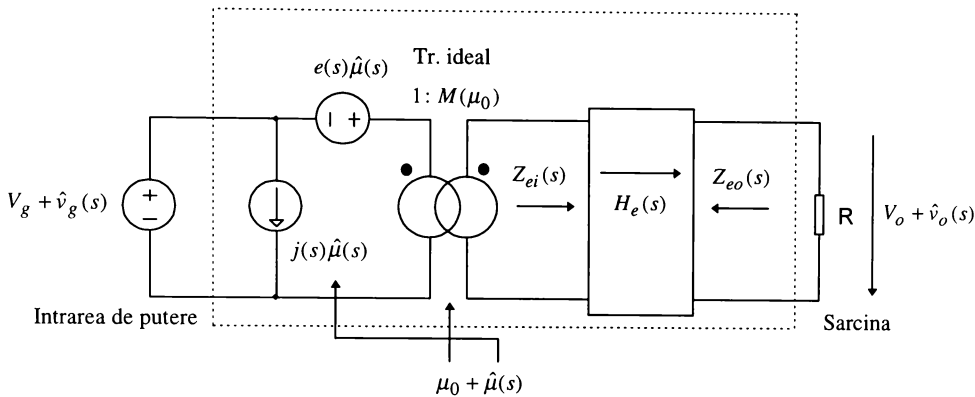


Fig. 6.5. Modelul canonic generalizat.

Este interesant că, spre deosebire de modelul din Fig. 6.4 modelul canonic generalizat are drept mărime de comandă factorul de umplere echivalent. Conform (15) acesta depinde de mărimea de comandă propriu-zisă (de obicei factorul de umplere sau frecvența) dar și de mărimi de stare din convertor, prin intermediul variabilelor  $v_{off}$  și  $i_{on}$ . Deci, deși nu apar desenate în schemă, există o buclă internă de reacție de tensiune și o buclă internă de reacție de curent. Bineînțeles că aceste bucle nu sunt accesibile pentru măsurare sau comandă dar reprezintă dependențe fizice din convertor, afectând câștigurile de curent continuu, amortizările și frecvențele de tăiere.

Parametrii modelului rămân aceiași pentru o topologie dată (de exemplu BUCK PWM CCM, BUCK PWM DICM, BUCK ZCS QRC, BUCK ZVS QRC, etc.), ei fiind dați de Tab. 6.2, care provine din Tab. 6.1 substituind pe  $D$  cu  $\mu_0$ .

Tab. 6.2. Parametrii modelului canonic generalizat.

CONVERTORUL	$M(\mu_0)$	$L_e$	$e(s)$	$j(s)$
BUCK	$\mu_0$	$L$	$\frac{V_g}{\mu_0}$	$\frac{\mu_0 V_g}{R}$
BOOST	$\frac{1}{1-\mu_0}$	$\frac{L}{(1-\mu_0)^2}$	$\frac{V_g}{1-\mu_0} \left( 1 - \frac{sL}{R(1-\mu_0)^2} \right)$	$\frac{V_g}{R(1-\mu_0)^3}$
BUCK-BOOST	$-\frac{\mu_0}{1-\mu_0}$	$\frac{L}{(1-\mu_0)^2}$	$-\frac{V_g}{\mu_0(1-\mu_0)} \left( 1 - \frac{s\mu_0 L}{R(1-\mu_0)^2} \right)$	$-\frac{\mu_0 V_g}{R(1-\mu_0)^3}$

Plecând de la modelul din Fig. 6.5 Witulski [209] face un studiu dinamic comparativ și dă ecuații de proiectare pentru toate topologiile de tip BUCK. Ulterior Popescu și Lascu respectiv Negoitescu și Lascu, plecând de la același model general, efectuează analize similare pentru topologiile BOOST [100] și respectiv BUCK-BOOST [97].

În cele ce urmează ne propunem soluționarea așa numitei *probleme a filtrului de intrare*. Să presupunem că s-a reușit proiectarea unui convertor în comutație care satisface condițiile legate de:

- răspunsul tranzitoriu - suficient de rapid și de bine amortizat. Acesta este legat nemijlocit de funcția de transfer control ieșire  $\frac{v_o}{\hat{c}}(s)$ .
- impedanța de ieșire - suficient de mică într-o gamă largă a frecvenței de comutație.
- audiosusceptibilitatea  $\frac{v_o}{\hat{v}_g}(s)$  - suficient de mică.

Cu toate acestea, de cele mai multe ori convertoarele produc fenomene de interferență electromagnetică (EMI), în primul rând din cauza curenților pulsatorii (de multe ori cu fronturi abrupte) de valori mari extrași din tensiunea de alimentare  $V_g$ . În noua conjunctură, în care condițiile de compatibilitate electromagnetică au devenit foarte severe, o soluție simplă și firească este înscrierea cu sursa de alimentare a unui filtru  $L_f - C_f$ . Prezența acestuia însă degradează răspunsul tranzitoriu, crește impedanța de ieșire și audiosusceptibilitatea. Problema filtrului de intrare se enunță astfel:

Să se proiecteze un filtru de intrare care să atenueze perturbațiile electromagnetice dar care să nu modifice funcțiile de transfer control ieșire,  $\frac{v_o}{\hat{c}}(s)$ , audiosusceptibilitatea,  $\frac{v_o}{\hat{v}_g}(s)$  și impedanța de ieșire  $Z_{out}$  ale convertorului inițial fără filtru.

Modelul canonic generalizat permite găsierea unei soluții generale a problemei, *valabilă pentru orice convertor și în orice mod de funcționare*. Vom lua în considerație numai convertoare la care  $\mu$  nu depinde decât de mărimea de comandă (adică PWM CCM și QRC-FW). Din relația (15b) rezultă că pentru aceste convertoare  $\hat{\mu} = \left. \frac{\partial \mu}{\partial c} \right|_{PSF} \cdot \hat{c}$ , astfel că din modelul canonic generalizat din Fig. 6.5 (fără filtru), pentru convertoarele PWM cu o bobină și un condensator și a celor derivate din ele se găsește:

$$\frac{\hat{v}_o}{\hat{c}} = \frac{\hat{v}_o}{\hat{\mu}} \cdot \frac{\hat{\mu}}{\hat{c}} = e(s) M(\mu_0) H_e(s) \cdot \left. \frac{\partial \mu}{\partial c} \right|_{PSF} \quad (16)$$

$$Z_{out} = (sL_e) \parallel (R \parallel \frac{1}{sC}) \quad (17)$$

$$\frac{\hat{v}_o}{\hat{v}_g} = M(\mu_0) H_e(s) \quad (18)$$

În deducerea lui  $Z_{out}$  și  $\frac{\hat{v}_o}{\hat{v}_g}$  s-a ținut cont că  $\hat{c} = 0$  implică și  $\hat{\mu} = 0$  pentru că s-au presupus în studiu convertoare la care  $\mu$  depinde numai de mărimea de comandă. De remarcat prezența încărcării  $R$  în expresia impedanței de ieșire, justificat prin aceea că impedanța de ieșire va reflecta modificările tensiunii de ieșire la variațiile curentului de sarcină (în model mai apare o sursă de curent în paralel cu  $R$ ) în jurul punctului static de funcționare, punct static care depinde de  $R$ .

O primă cerință a prezenței filtrului este *să nu modifice modul de funcționare* al convertorului (CCM, DICM, ZVS-HW, etc.). În al doilea rând, pentru a obține funcții de transfer și impedanță de ieșire identice, fiind vorba de mărimi de semnal mic, ai căror parametri depind de punctul static de funcționare, este necesar ca *punctele statice de funcționare în modelul mediat să fie identice* (în fond aceleași  $V_{off}$ ,  $I_{on}$  și deci același  $\mu_0$ ). Într-adevăr acest lucru este adevărat pentru că în curent continuu modelul din Fig. 6.5 coincide cu modelul cu filtrul inclus deoarece filtrul conține numai elemente reactive.

În al treilea rând, *expresiile funcțiilor de transfer trebuie să fie identice*. Pentru a obține modelul de semnal mic al convertorului cu filtru de intrare trebuie doar adăugat filtrul în modelul canonic generalizat - Fig. 6.6. Acest lucru se poate demonstra folosind medierea în spațiul stărilor sau modele mediate pentru ansamblul întrerupătoarelor. Din schema din Fig. 6.6 se găsesc funcția de transfer control-ieșire, impedanța de ieșire și audiosusceptibilitatea în prezența filtrului ca având următoarele expresii:

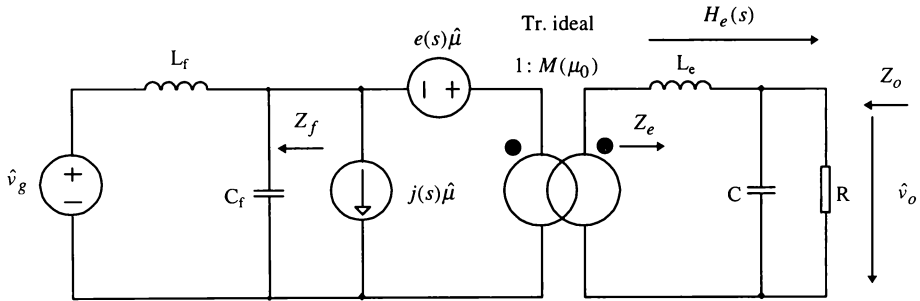


Fig. 6.6. Modelul canonic extins conținând filtrul de intrare.

$$\left(\frac{\hat{v}_o}{\hat{c}}\right)_f = \left(\frac{\hat{v}_o}{\hat{c}}\right) \cdot \begin{bmatrix} 1 - \frac{Z_f}{e(s)} \\ j(s) \\ 1 + \frac{Z_f}{Z_e} \\ M^2(\mu_0) \end{bmatrix} \quad (19)$$

$$(Z_{out})_f = Z_{out} \cdot \begin{bmatrix} 1 + \frac{Z_f}{sL_e} \\ M^2(\mu_0) \\ \frac{Z_f}{1 + \frac{Z_e}{M^2(\mu_0)}} \end{bmatrix} \quad (20)$$

$$\left(\frac{\hat{v}_o}{\hat{v}_g}\right)_f = \left(\frac{\hat{v}_o}{\hat{v}_g}\right) \cdot \begin{bmatrix} \frac{1}{1 + s^2 L_f C_f} \cdot \frac{1}{1 + \frac{Z_f}{Z_e}} \\ M^2(\mu_0) \end{bmatrix} \quad (21)$$

unde

$$Z_f = sL_f \parallel \frac{1}{sC_f} \quad Z_e = sL_e + (R \parallel \frac{1}{sC}) \quad (22)$$

reprezintă impedanța filtrului inserat, respectiv impedanța filtrului echivalent de ordinul 2, cu încărcare, de la ieșire. Este clar că pentru a evita modificarea funcțiilor de transfer și a impedanței

de ieșire este necesar ca parantezele pătrate din membrii drepti să fie unitare. Se obține astfel următorul set de inegalități necesare a se realiza simultan:

$$\left\{ \begin{array}{l} \|Z_f\| \ll \left\| \frac{e(s)}{j(s)} \right\| \\ \|Z_f\| \ll \left\| \frac{Z_e}{M^2(\mu_0)} \right\| \\ \|Z_f\| \ll \left\| \frac{sL_e}{M^2(\mu_0)} \right\| \\ \|Z_f\| \ll \|Z_i\| \\ f \ll \frac{1}{2\pi\sqrt{L_f C_f}} \end{array} \right. \quad (23)$$

Ultima relație provine din condiția de audiosusceptibilitate (prima fracție din paranteza pătrată) și este totdeauna îndeplinită dat fiind că filtrul trebuie să suprimă pulsațiile curentului de intrare care au frecvența de comutație, iar pe de altă parte, modelul fiind mediat, frecvențele de interes sunt până la jumătate din frecvența de comutație. Celelalte patru relații se referă la impedența de ieșire a filtrului,  $Z_f$ , care trebuie să fie mult mai mică decât diverse impedanțe legate de filtrul echivalent de la ieșire reflectate în primar (acolo unde este plasat filtrul) și mult mai mică decât impedența efectivă  $\frac{e(s)}{j(s)}$  care se evaluează din Tab. 6.2.

### Aplicație

Fie un convertor BUCK ZCS FW la care dorim ca filtrul de intrare să nu modifice funcția de transfer control-ieșire. Pentru aceasta primele două condiții din (23) trebuie îndeplinite. Ținând cont că într-un convertor BUCK, indiferent de tip, avem:

$$L_e = L; \quad e(s) = \frac{V_g}{\mu_0}; \quad j(s) = \mu_0 \frac{V_g}{R}; \quad M(\mu_0) = \mu_0 \quad (24)$$

se obține

$$\frac{e(s)}{j(s)} = \frac{R}{\mu_0^2} = \frac{R}{M^2(\mu_0)} \quad (25)$$

În ceea ce privește evaluarea impedanțelor, o metodă simplă și eficientă pe care o vom utiliza este cea a "algebrei pe grafic" ("doing algebra on the graph") prezentată în [39, paragraful 8.3]. În urma unei astfel de analize rezultă că alura impedanțelor este cea din Fig. 6.7 unde:

$$\omega_f = \frac{1}{2\pi\sqrt{L_f C_f}}, \quad \omega_1 = \frac{1}{RC}; \quad \omega_2 = \frac{1}{\sqrt{LC}} \quad (26)$$

Dacă elementele de circuit sunt alese astfel încât ordonarea frecvențelor să fie cea din Fig. 6.7 atunci cele două inegalități sunt respectate peste tot, mai puțin în jurul pulsației  $\omega_f$ , (unde  $Q_f$  este mare, teoretic infinit) și eventual la pulsația  $\omega_2$ , pentru rezistență de sarcină minimă când filtrul de la ieșire devine slab amortizat. Oricum ideea de a alege filtrul astfel încât  $\omega_f = \omega_2$ , cum poate este prima tentăție, este cea mai puțin inspirată.

Amortizarea necesară pentru a micșora  $Q_f$  se poate obține cu o schemă de genul celei din Fig. 6.8.

$C_b$  are rolul blocării componentei continue de curent prin  $R_f$  și se alege astfel încât  $\frac{1}{\omega_f C_b} \ll R_b$ .

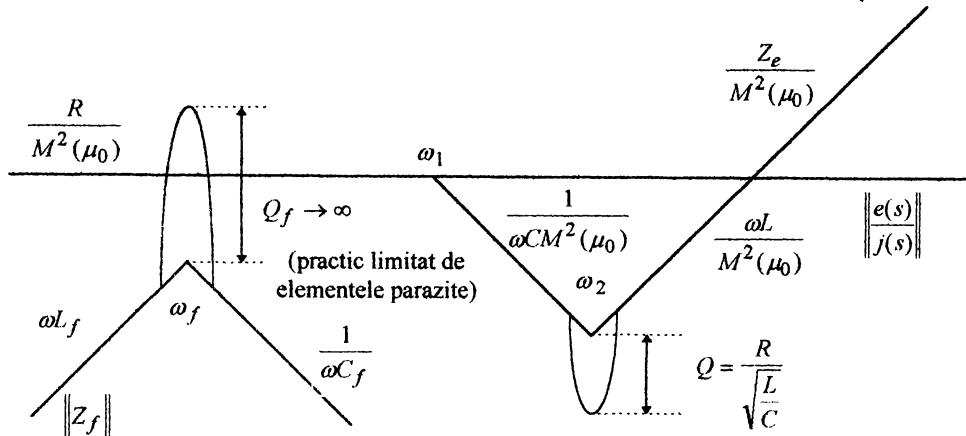


Fig. 6.7. Alura modulelor impedanțelor în urma folosirii metodei “algebrei pe grafic”.

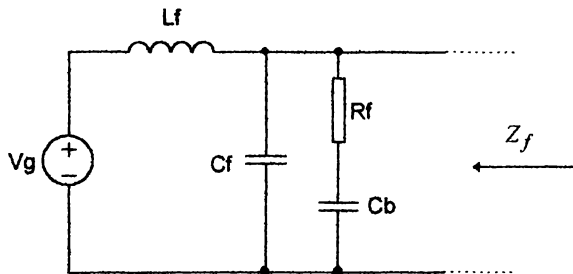


Fig. 6.8. Structura filtrului, modificată pentru a mări amortizarea.

Pe de altă parte însă  $C_b$  nu poate fi mai prea mare în raport cu  $C_f$  pentru că frecvența de frângere se deplasează nepermis. Fie

$$C_b = mC_f \tag{27}$$

Dacă  $R_f \rightarrow \infty$   $C_b$  este practic deconectat și deci  $Z_f = sL_f \parallel \frac{1}{sC_f}$  iar  $Q_f \rightarrow \infty$ .

Pentru  $R_f \rightarrow 0$   $C_b$  apare direct în paralel cu  $C_f$  și  $Z_f = sL_f \parallel \frac{1}{s(C_f + C_b)}$  și din nou  $Q_f \rightarrow \infty$ .

Între aceste extreme  $R_f$  amortizează circuitul. Dându-se  $C_b$  și  $C_f$  există o valoare optimă a lui  $R_f$ , notată  $R_{f\text{opt}}$ , pentru care circuitul are amortizare minimă:

$$R_{f\text{opt}} = \sqrt{\frac{L_f}{C_f}} \cdot \sqrt{\frac{(4+3m)(2+m)}{2m^2(4+m)}} \quad Q_{\text{min}} = \frac{R_{f\text{opt}}}{\sqrt{\frac{L_f}{C_f}}} \tag{28}$$

Cu aceste precizări proiectarea filtrului rămâne o simplă problemă de alegere de componente care să îndeplinească cerințele enunțate mai sus.

### 6.3. Modele unificate pentru convertoarele rezonante serie

#### 6.3.1 O metodă generală de mediere

Sun și Grotstollen [178], [179] propun o metodă de mediere bazată pe partiționarea convertorului studiat în două subsisteme: un sistem lent și un sistem rapid. Subsistemul lent conține atât principalele elemente reactive pentru stocarea energiei, cât și intrarea și ieșirea convertorului. Subsistemul rapid are în componență dispozitivele de comutație și elementele rezonante, sau elementele reactive la care mărimile de stare nu respectă ipoteza pulsațiilor lente (în raport cu perioada de comutație). *Scopul urmărit este studiul dinamicii sistemului lent.* Fie  $u_s$  vectorul de intrare în sistemul total,  $v_s$  vectorul de intrare în sistemul lent produs de sistemul rapid (deci care este și ieșire din sistemul rapid),  $v_f$  vectorul de ieșire din sistemul lent, care este intrare în sistemul rapid și  $y_s$  ieșirea din sistemul total (care este și ieșire din sistemul lent). Se mai notează cu  $x_s$  variabilele de stare ale sistemului lent și cu  $x_f$  variabilele de stare din sistemul rapid. Fig. 6.9 exemplifică mărimile definite mai sus.

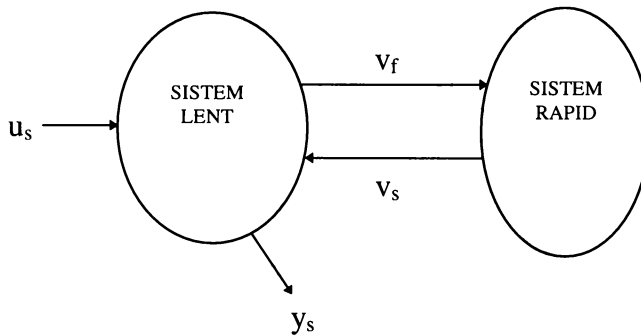


Fig. 6.9. Împărțirea unui convertor într-un sistem lent și un sistem rapid.

Se consideră ecuațiile de stare corespunzătoare sistemului lent:

$$\frac{dx_s}{dt} = f(x_s, u_s, v_s) \quad (29)$$

și ale sistemului rapid:

$$\frac{dx_f}{dt} = g_i(x_s, u_s, v_s) \quad i = \overline{1, N_s} \quad (30)$$

Se observă că sistemul rapid, conținând întrerupătoarele, este guvernat de  $N_s$  legi de variație distincte, aferente celor  $N_s$  stări topologice, determinate de comutarea întrerupătoarelor. Acest lucru nu se întâmplă în sistemul lent unde legea de variație este unică. Cuplarea dintre cele două sisteme este dată de relațiile:

$$v_f = p(x_s, u_s) \quad v_s = q_i(x_f, v_f) \quad i = \overline{1, N_s} \quad (31)$$

observându-se că  $v_f$ , fiind ieșire din sistemul lent, se presupune a avea o variație lentă, determinată numai de  $x_s$  și  $u_s$ , nu și de  $v_s$ . Influența lui  $v_s$  apare în  $v_f$  indirect, prin  $x_s$ , care e influențat la rândul său de  $v_s$  prin ecuația de stare (29).

Într-o astfel de partiționare un model mediat se obține în 3 pași:



1. *Relaxarea*. Se analizează funcționarea sistemului lent presupunând intrarea acestuia,  $v_f$ , constantă. Această presupunere se justifică prin variațiile lente ale lui  $v_f$ . Sunt astfel determinate legile de variație ale variabilelor de stare rapide într-o perioadă:

$$\tilde{x}_f = h(v_f, t) \quad (32)$$

unde simbolul “ $\sim$ ” denotă aici o mărime relaxată, iar  $h$  nu are obligatoriu o expresie unică în cele  $N_s$  stări topologice.

2. *Decuplarea*. Folosind (17) și (18) se poate acum obține răspunsul relaxat al ieșirii,  $\tilde{v}_s$ , a sistemului lent:

$$\tilde{v}_s = q_i(h(v_f, t), v_f) = z(v_f, t) \quad (33)$$

Dinamica sistemului lent se poate acum deduce înlocuind în ecuația de stare (29) a sistemului lent pe  $v_s$  cu soluția relaxată,  $\tilde{v}_s$ , dată de (33). În acest fel sistemul rapid, este înlocuit printr-o rețea rezistivă dependentă de timp.

3. *Medierea*. Pentru că perioada de comutație  $T_s$  este mică în raport cu constantele de timp ale sistemului lent, răspunsul acestuia la  $\tilde{v}_s$  este dat în cea mai mare proporție de componenta continuă (valoarea medie) a acesteia în raport cu timpul:

$$\bar{v} = \bar{z}(v_f) = \frac{1}{T_s} \int_{t_0}^{t_0+T_s} \tilde{v}(t) dt = \int_{t_0}^{t_0+T_s} z(v_f, \tau) d\tau \quad (34)$$

În acest fel modelul obținut la pasul de decuplare devine invariant în timp, rămânând însă neliniar:

$$\frac{dx_s}{dt} = f(x_s, u_s, \bar{v}_f) = f(x_s, u_s, \bar{z}(p(x_s, u_s))) = f_0(x_s, u_s) \quad (35)$$

Cu o astfel de tehnică au fost deduse modele pentru convertoare PWM CCM și convertoare cvasirezonaante.

**Autorul prezentei teze** a arătat [87] că tehnica se poate aplica cu succes și convertoarelor PWM DICM, PWM DCVM și convertoarelor rezonante serie și paralel. Dintre acestea se va prezenta mai jos numai un model unificat pentru convertorul rezonant serie.

### 6.3.2 Modele unificate pentru convertorul rezonant serie

Pentru convertorul serie din Fig. 6.10 sistemul lent este compus din capacitatea  $C_F$ , sarcina  $R$  și tensiunea de intrare  $v_g$ , în timp ce sistemul rapid este alcătuit din circuitul rezonant și dispozitivele de comutație. Vectorii de legătură între sisteme sunt:  $v_f = [v_{C_F} \ v_g]^T$  și  $v_s = [i_r]$ .

Notațiile pentru  $f_N$  și  $Z_0$  fiind cele clasice, folosite și la convertoarele cvasirezonaante, aplicarea tehnicii expusă în paragraful precedent duce, după calcule laborioase, la următoarele expresii:

$$i_r = \frac{2f_N}{\pi} \cdot \frac{\sqrt{v_{C_F}^2 + \frac{2(v_g^2 - v_{C_F}^2)}{1 + \cos(\frac{\pi}{f_N})} - v_g \cdot \text{sign}(f_N - 1)}}{Z_0} \quad (36)$$

În acest fel modelul de semnal mare al convertorului se obține în forma:

$$\frac{dv_o}{dt} = 4 \frac{C}{C_F} f_0 f_N \left[ \sqrt{v_o^2 + \frac{2}{\cos(\frac{\pi}{f_N})} (v_g^2 - v_o^2) - v_g \cdot \text{sign}(f_N - 1)} \right] - \frac{1}{RC_F} v_o \quad (37)$$

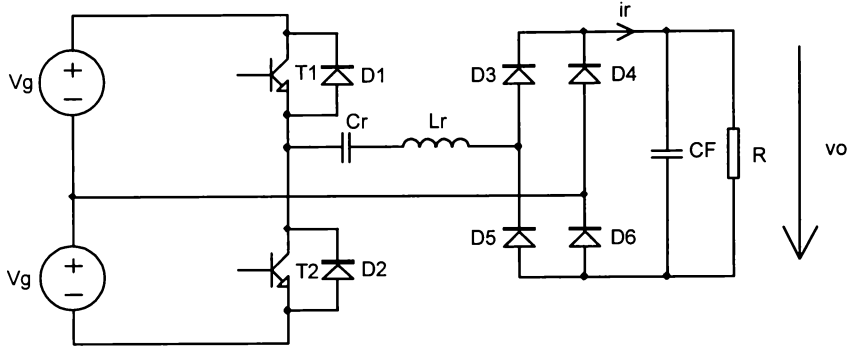


Fig. 6.10. Convertorul rezonant serie studiat.

unde în (37) s-a preferat să se noteze variabila de stare cu  $v_o$ , pentru că  $v_{C_F} = v_o$ , o astfel de notație fiind utilă la scrierea expresiilor caracteristicilor statice. Caracteristicile statice control-ieșire se obțin din (37) anulând derivata. În formă normalizată (prin  $V_g$  pentru tensiunea de ieșire) ele au expresia analitică:

$$V_{ON} = \frac{\sqrt{2 \left( \frac{\pi Z_0}{2f_N R} \right)^2 \left( 1 + \cos\left(\frac{\pi}{f_N}\right) \right) + \left( 1 - \cos\left(\frac{\pi}{f_N}\right) \right)^2 - \text{sign}(f_N - 1) \frac{\pi Z_0}{2f_N R} \left( 1 + \cos\left(\frac{\pi}{f_N}\right) \right)}{1 - \cos\left(\frac{\pi}{f_N}\right) + \left( \frac{\pi Z_0}{2f_N R} \right)^2 \left( 1 + \cos\left(\frac{\pi}{f_N}\right) \right)} \quad (38)$$

Aceste caracteristici sunt reprezentate în Fig. 6.11 având rezistența de sarcină normalizată

$$R_N = \frac{R}{Z_0} \quad (39)$$

ca și parametru. Tot din ecuația (38) se deduce și expresia caracteristicilor de ieșire normalizate:

$$V_{ON} = \sqrt{\frac{2}{1 - \cos\left(\frac{\pi}{f_N}\right)} - \frac{1 + \cos\left(\frac{\pi}{f_N}\right)}{1 - \cos\left(\frac{\pi}{f_N}\right)} \cdot \left[ \frac{\pi}{2f_N} I_{ON} + \text{sign}(f_N - 1) \right]^2} \quad (40)$$

unde curentul s-a normalizat prin factorul  $\frac{V_g}{Z_0}$ . Reprezentarea grafică a acestor caracteristici este

prezentată în Fig. 6.12, programele aferente fiind cele din Anexa A12.

Trebuie subliniat că termenul “unificat” se referă la faptul că aceste caracteristici sunt *valabile atât sub cât și peste frecvența de rezonanță*, expresiile analitice sintetizând ambele situații. În situația în care este utilizat și un transformator de izolare expresiile, curbele și toate rezultatele rămân absolut aceleași, singura deosebire fiind aceea că raportul de transformare apare în mărimile normalizate, după regulile enunțate în paragraful 5.4.

Legat de aceste caracteristici statice **autorul** a propus în [79] o analiză statică cu ajutorul calculatorului a convertoarelor rezonante serie. Organigrama algoritmului este dată în Fig. 6.13. Principalii pași parcurși în derularea proiectării asistate de calculator sunt următorii:

*Pasul 1.* Introducerea datelor de intrare:

- tensiunea de alimentare nominală, minimă și maximă:  $V_{g\text{nom}}$ ,  $V_{g\text{min}}$  respectiv  $V_{g\text{max}}$ .

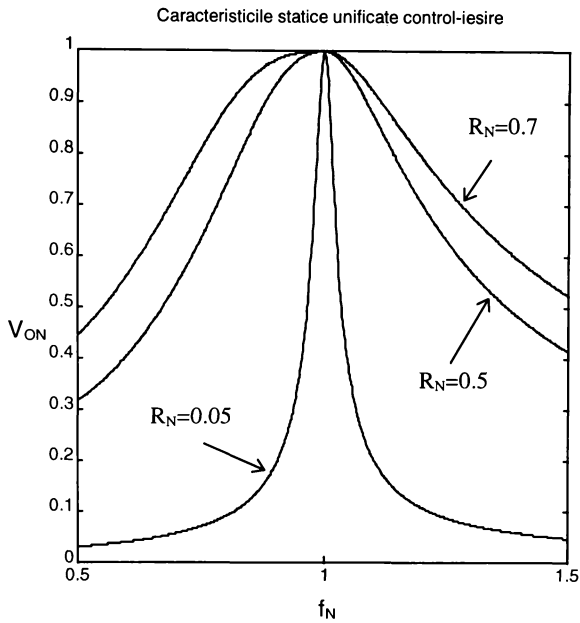


Fig. 6.11. Caracteristicile statice unificate de tip control-ieșire ale convertoarelor SRC.

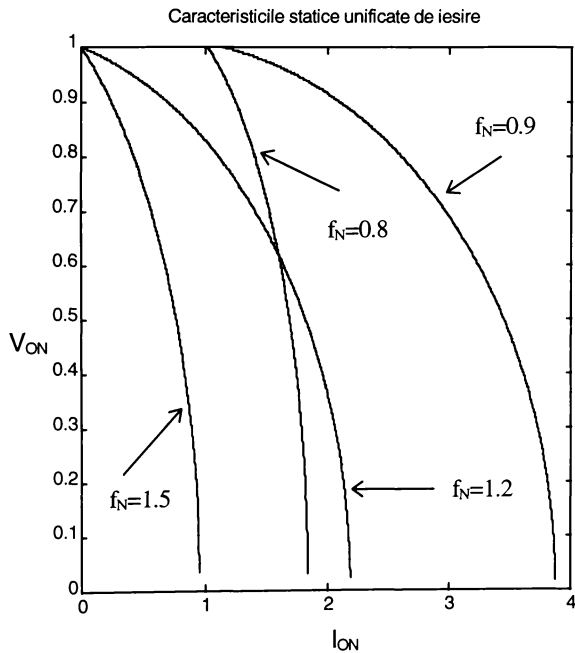


Fig. 6.12. Caracteristicile unificate de ieșire ale convertoarelor SRC.

- curentul de sarcină minim și maxim,  $I_{o \min}$  respectiv  $I_{o \max}$ .
- valoarea tensiunii de ieșire stabilizate,  $V_o$

*Pasul 2. Alegerea raportului de transformare.*

În eventualitatea folosirii unui transformator programul calculează valoarea optimă a raportului de transformare și o afișază ca valoare recomandată. Această valoare rezultă din transferul maxim de putere fiind:

$$n_{rec} = \frac{1}{2} \frac{V_{gnom}}{V_o} \quad (41)$$

De asemenea raportul de transformare pentru care  $V_{oN \max} = \frac{nV_o}{V_{g \min}} = 1$  este și el afișat ca valoare

limitativă în alegerea lui  $n$ . Utilizatorul poate să modifice acest raport în concordanță cu opțiunile sale, reamintindu-i-se că la  $n$  mare tensiunea de ieșire dorită nu va putea fi obținută datorită pierderilor inerente și răspunsul convertorului va fi lent. Pe de altă parte, la  $n$  mic curenții prin tranzistoare și tensiunea pe capacitatea rezonantă vor fi mari fără nici un sens. De aceea utilizatorului i se furnizează curbele normalizate de solicitare  $I_{LPN} = f(f_N)$ ,  $V_{CPN} = f(f_N)$ , cu  $V_{oN}$  parametru. Se mai reamintește și faptul că valoarea raportului de transformare afectează și elementele rezonante dacă  $f_0$  și  $f_{s \min}$  sau  $f_{s \max}$  sunt fixate:  $L_r$  crește proporțional cu  $n$ , în timp ce  $C_r$  variază invers proporțional cu  $n$ . Evident, impedanța caracteristică  $Z_0$  crește cu  $n$ .

*Pasul 3. Alegerea frecvenței de rezonanță.*

Desigur este de dorit o frecvență de rezonanță  $f_0$  cât mai mare, pentru a obține dimensiuni și greutate mică a convertorului, dar trebuie ținut cont că pierderile de comutație cresc cu frecvența.

*Pasul 4. Alegerea frecvenței de comutație minime sau maxime.*

În acest punct utilizatorul decide dacă convertorul va funcționa sub sau peste frecvența de comutație. Dacă decizia este  $f_s < f_0$  atunci frecvența maximă,  $f_{s \max}$  se va obține la  $V_{g \min}$  și la  $I_{o \max}$ , iar frecvența minimă,  $f_{s \min}$ , la  $V_{g \max}$  și  $I_{o \min}$ . Dacă decizia luată este  $f_s > f_0$  atunci lucrurile stau exact invers:  $f_{s \max}$  se va obține la  $V_{g \max}$  și  $I_{o \min}$ , iar  $f_{s \min}$  la  $V_{g \min}$  și  $I_{o \max}$ .

La funcționare sub frecvența de rezonanță, dacă  $f_{s \min}$  este crescut  $L_r$  va crește,  $C_r$  va scădea și tensiunea maximă pe capacitate va crește. Pe de altă parte, dacă  $f_{s \max}$  este scăzut este posibilă funcționarea în conducție discontinuă, care nu este de dorit.

La funcționare peste frecvența de rezonanță dacă  $f_{s \min}$  este crescut  $L_r$  va scădea,  $C_r$  va crește și tensiunea maximă pe capacitate va fi mai mică. Dar în acest caz domeniul frecvențelor de funcționare va fi larg ceea ce face controlul (sub aspectul stabilității) dificil. Dacă  $f_{s \max}$  este scăzut  $L_r$  va crește, domeniul frecvențelor de funcționare se va îngusta, dar tensiunea maximă pe capacitate va crește.

Concluzia este că trebuie făcut un compromis între funcționarea sub sau peste  $f_0$  și solicitările componentelor. Toate considerațiile de mai sus sunt furnizate utilizatorului de către programul de proiectare.

*Pasul 5. Calculul elementelor rezonante  $L_r$  și  $C_r$ .*

Calculul este efectuat pe baza traiectoriei în planul stărilor și a ecuațiilor caracteristicilor statice deduse anterior.

*Pasul 6. Execută analiza de curent continuu pentru combinațiile extreme dintre  $V_g$  și  $I_o$*

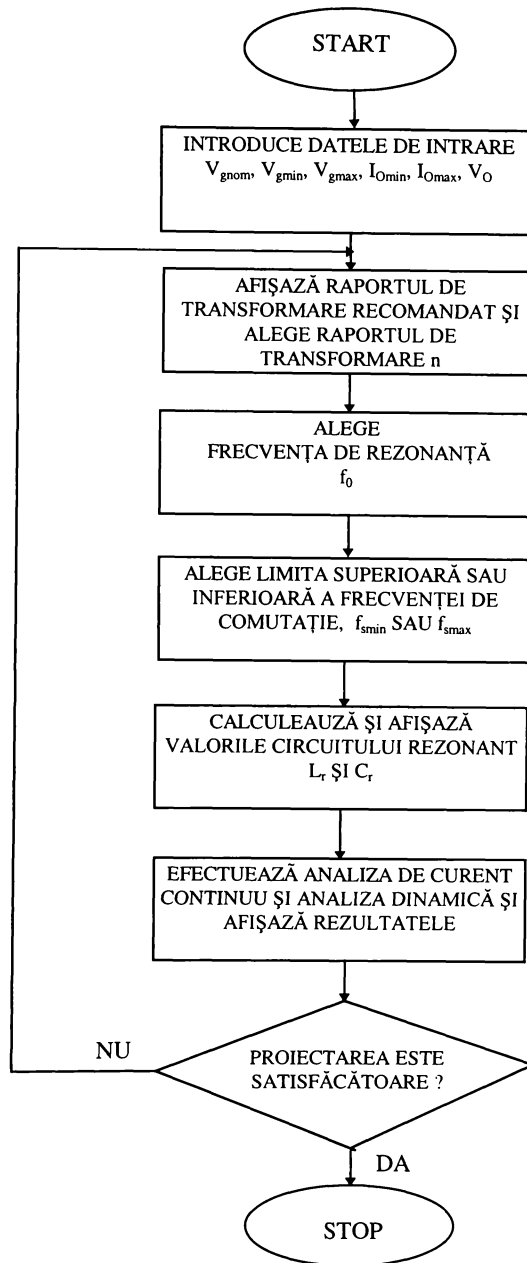


Fig. 6.13. Organigrama programului de proiectare automată a convertoarelor rezonante serie.

Se calculează solicitările de curent și tensiune  $I_{LP}$ ,  $V_{CP}$ ,  $I_{SW}$ ,  $V_{CO}$  (pentru notații vezi Fig. 5.19), curentul redresat mediu,  $I_{LAV}$  și frecvențele corespunzătoare. De asemenea este efectuată o analiză dinamică în sensul că se presupune o buclă de reacție cu timp de răspuns mare și se admite o variație treaptă într-un dintre mărimi (de ex. la  $f_{s\min}$ ,  $V_{g\max}$  și  $I_{o\min}$  tensiunea de alimentare

variază brusc de la  $V_{g \min}$ , dar pentru câteva perioade frecvența de comutație rămâne tot  $f_{s \min}$ ). Evident, acestea reprezintă situațiile cele mai defavorabile.

*Pasul 7. Eventuala iterare a proiectării*

Dacă anumite rezultate sunt nesatisfăcătoare se poate relua proiectarea de la pasul 2 ținând cont de curbele afișate în timpul derulării programului.

Exemplu. Se dorește proiectarea unui convertor rezonant serie cu următoarele date:

$$V_{gnom} = 80 \text{ V}; \quad V_{g \min} = 60 \text{ V}; \quad V_{g \max} = 100 \text{ V}; \quad V_o = 5 \text{ V}; \quad I_{o \min} = 4 \text{ A}; \quad I_{o \max} = 6 \text{ A};$$

și se alege  $f_0 = 80 \text{ kHz}$  și  $f_{s \min} = 50 \text{ kHz}$ . Curbele normalizate afișate de program sunt cele din Fig. 6.14. Valorile calculate pentru elementele rezonante sunt:

$$L_r = 348,5 \mu\text{H}; \quad C_r = 11 \text{ nF}$$

Rezultatele analizei de curent continuu pentru combinații extreme între tensiunea de alimentare și curentul de sarcină sunt cele de mai jos:

$V_g$ [V]	$I_o$ [A]	$f_s$ [kHz]	$I_{LAV}$ [A]	$I_{LI}$ [A]	$V_{CP}$ [V]
60.0000	4.0000	62.8842	0.5000	0.8850	175.0363
60.0000	6.0000	68.7375	0.7500	1.2569	240.1914
100.0000	4.0000	49.9998	0.5000	0.9141	220.1412
100.0000	6.0000	59.1704	0.7500	1.2503	279.0294

Domeniul de variație al principalelor mărimi de interes este:

$f_s$ [kHz]	ISW[A]	VC0[V]	ILP[A]	VCP[V]
49.9998	0.3484	-88.0565	0.8850	175.0363
68.7375	0.7769	-160.1276	1.2569	279.0294

În urma analizei dinamice se obțin următoarele valori:

VC0[V]	ILP[A]	VCP[V]
85.4884	0.6178	128.2326
184.9135	2.2964	462.2839

Pentru datele de proiectare menționate mai sus valorile de vârf ale curentului prin bobina rezonantă și a tensiunii pe capacitatea rezonantă depind de frecvență conform curbelor din Fig. 6.15.

Programul interactiv de proiectare asistată este scris în MATLAB, fișierele fiind prezentate în Anexa A13.

În continuare, modelul de semnal mare dat de relația (37) a fost introdus în MATLAB sub mediul SIMULINK - vezi Anexa A14 și comparat cu rezultatele simulării cu simulatorul CASPOC (Simulation Research) - Anexa A15. Obiectul de studiu a fost un convertor SRC cu următorii parametri (același convertor cu cel studiat în [202], unde există și rezultate experimentale, pentru a avea un element de comparație în ceea ce privește modelul de semnal mic):

$$V_g = 50 \text{ V}; \quad L_r = 197 \mu\text{H}; \quad C_r = 51 \text{ nF}; \quad C_F = 32 \mu\text{F}; \quad R = 60 \Omega.$$

A fost studiată comportarea la semnal mare pentru un salt treaptă în frecvența de comutație, la 15 ms, de la 30,127 KHz ( $0,6f_0$ ) la 42,68 KHz ( $0,85f_0$ ) și apoi, la 20 ms, revenire tot cu salt treaptă la 30,127 KHz ( $0,6f_0$ ). Rezultatele obținute prin simulare sunt cele din Fig. 6.16, iar cele pronosticate de modelul de semnal mare cele din Fig. 6.17. Coincidența este excelentă.

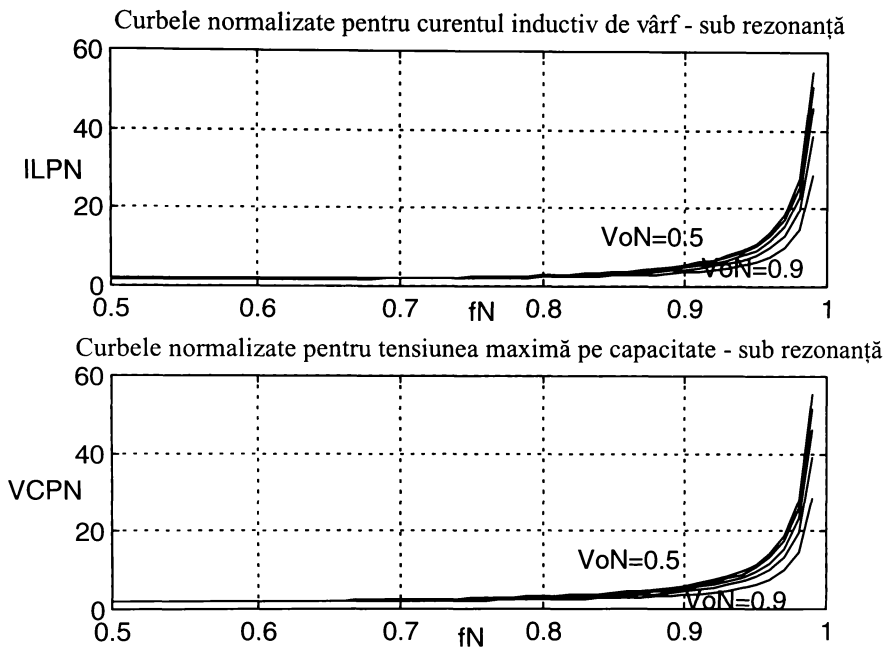


Fig. 6.14. Principalele solicitări normalizate.

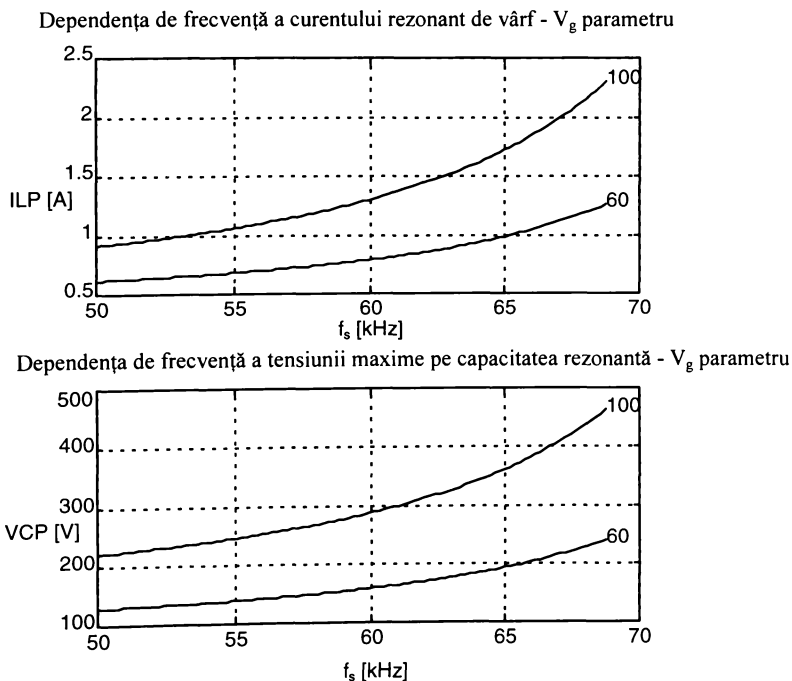


Fig. 6.15. Dependența de frecvență a principalelor solicitări absolute ale componentelor la diferite valori ale tensiunii de alimentare.

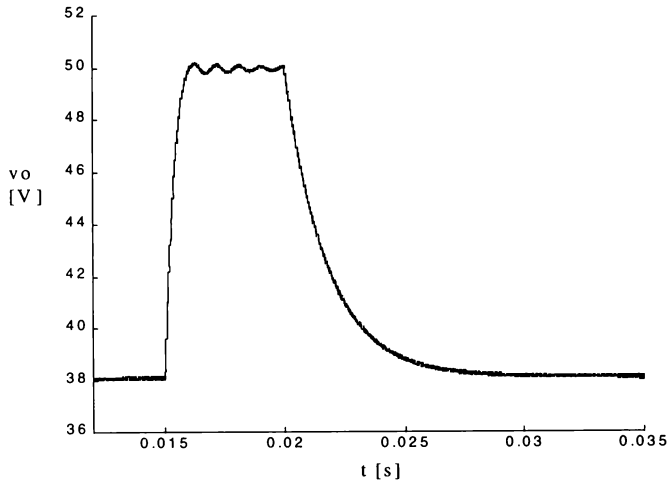


Fig. 6.16. Comportarea la semnal mare a convertorului SRC - rezultatele simulării (simulator CASPOC - Simulation Research).

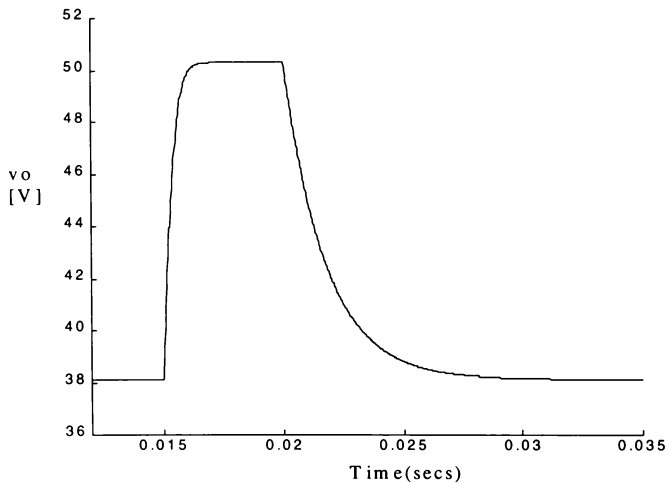


Fig. 6.17. Prognostarea comportării la semnal mare a convertorului SRC conform modelului de semnal mare dat în ecuația (37) - implementarea modelului s-a făcut în MATLAB/SIMULINK.

Modelul de semnal mare se poate liniariza cu metodele cunoscute, obținându-se modelul de semnal mic. Punctul de funcționare calculat prin caracteristicile statice este util pentru că parametrii modelului de semnal mic sunt dependenți de punctul static de funcționare. Întrucât sistemul lent din partiția efectuată inițial pentru convertor este de ordinul 1 și funcțiile de transfer de semnal mic vor fi de același ordin, deci:



$$\hat{v}_o = A_C \frac{1}{1 + \frac{s}{\omega_c}} \cdot \hat{f}_s + A_g \frac{1}{1 + \frac{s}{\omega_c}} \cdot \hat{v}_g \quad (42)$$

unde  $A_C$ ,  $A_g$  și  $\omega_c$  se evaluează funcție de mărimile din punctul static de funcționare, deja determinate. Expresiile lor fiind lungi nu vor mai fi reproduse aici. Pentru convertorul menționat anterior, la care sarcina este de  $R = 10,3\Omega$ , funcția de transfer control ieșire și audiosusceptibilitatea la semnal mic sunt reprezentate în Fig. 6.18 și respectiv Fig. 6.19 folosind programele MATLAB din Anexa A16. Rezultatele sunt identice cu cele obținute în [202], coincizând cu experimentele. Aceasta confirmă încă o dată valabilitatea modelului dezvoltat.

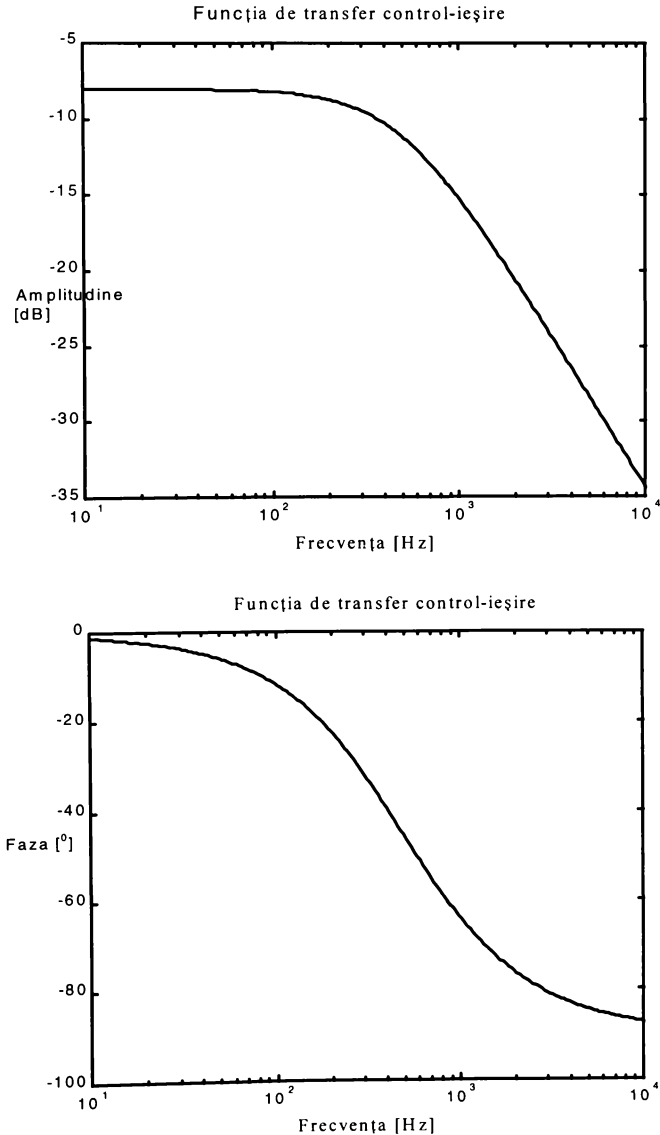


Fig. 6.18. Funcția de transfer control-ieșire.

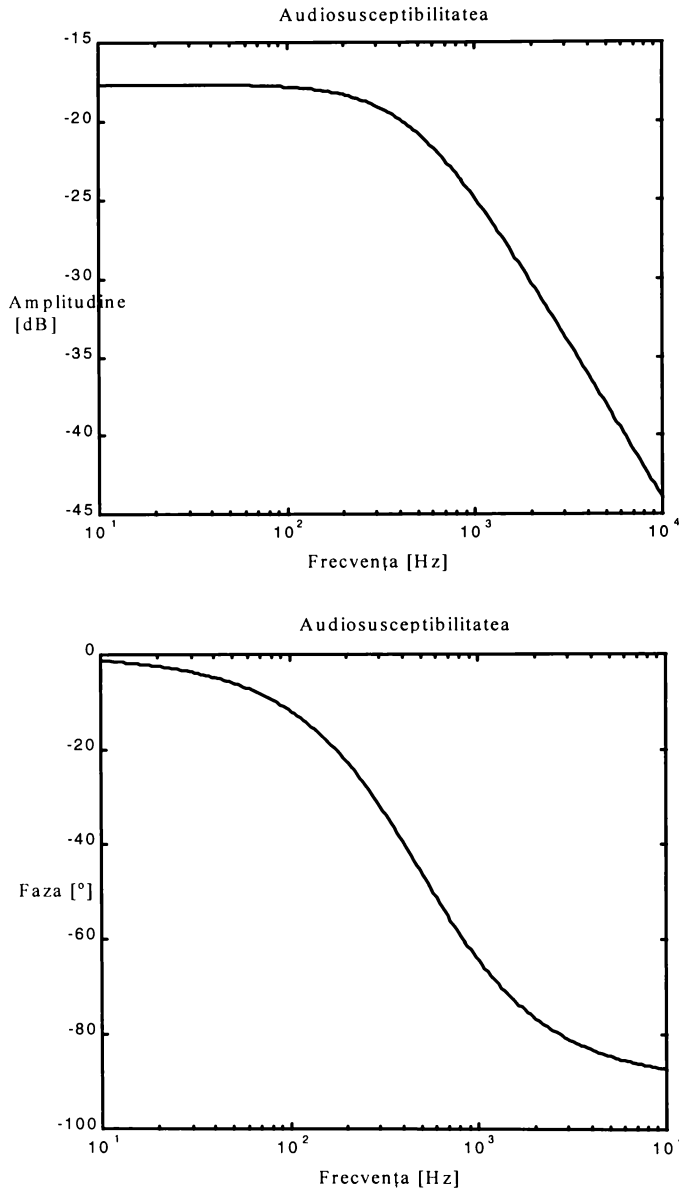


Fig. 6.19. Audiosusceptibilitatea.

Avantajele modelului propus sunt:

- caracterul unitar, valabil atât sub cât și peste frecvența de rezonanță.
- obținerea unitară a caracteristicilor statice și utilizarea lor în proiectarea asistată.
- implementare ușoară în orice program de simulare de sistem, cu simulare mult mai rapidă a comportării la semnal mare față de cazul utilizării unui simulator de circuit.
- obținerea funcțiilor de transfer de semnal mic în formă rațională, ceea ce permite proiectarea controllerului cu metodele teoriei sistemelor liniare.

### 6.4. Un model de semnal mic în spațiul stărilor pentru comanda prin sarcină

Dinamica convertoarelor cu comandă prin sarcină a fost studiată în [182] folosind o tehnică de mediere și dezvoltând un model de semnal mic sub forma schemei bloc din Fig. 6.20.

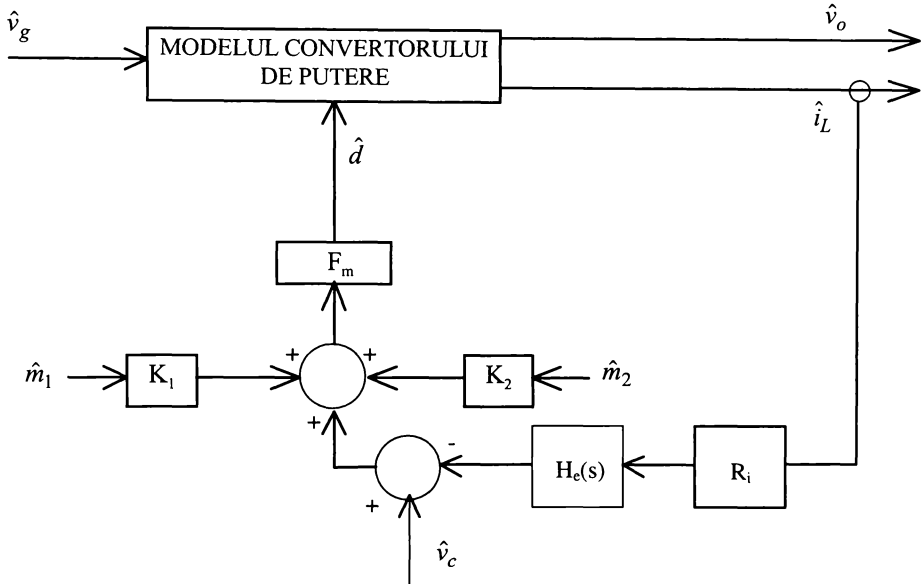


Fig. 6.20. Modelul de semnal mic, sub formă de schemă bloc, al unui convertor cu comandă prin sarcină.

Aici  $\hat{m}_1$  și  $\hat{m}_2$  reprezintă perturbațiile din pantele ridicătoare respectiv coborâtoare ale curentului inductiv. Ceilalți parametri ai modelului au următoarele expresii:

$$H_e(s) = 1 + \frac{1}{\omega_n Q_z} s + \frac{1}{\omega_n^2} s^2 \quad \omega_n = \frac{\pi}{T_S} \quad Q_z = -\frac{2}{\pi} \quad (43)$$

$$F_m = \frac{C_T}{I_L (DT_S) T_S} \quad R_i = \frac{DT_S}{C_T} \quad (44)$$

$$K_1 = -\frac{D^2 D' T_S^2}{2C_T} \quad K_2 = \frac{DD'^2 T_S^2}{2C_T} \quad (45)$$

Acest model este punctul de plecare pentru obținerea unui model în spațiul stărilor pentru acest tip de comandă, folosind o tehnică asemănătoare cu cea din [194]. Deducerea in extenso a modelului este publicată de **autorul tezei** în [88]. Se pleacă de la modelul în spațiul stărilor al convertorului de putere cu comanda în factor de umplere, dat de relațiile binecunoscute:

$$\frac{d\hat{x}}{dt} = A_D \hat{x} + B_D \hat{v}_g + \xi \hat{d} \quad (46)$$

$$\hat{y} = E_D \hat{x} + F_D \hat{v}_g + \zeta \hat{d} \quad (47)$$

unde  $A_D, E_D, F_D, \xi_D$ , și  $\zeta_D$  au fost definite în paragraful 6.1 (relațiile (6)-(11)). Acest model pentru partea de putere sugerează transformarea modelului din Fig. 6.20 în așa fel încât blocurile de câștig să multiplice tot pe  $\hat{x}$  și  $\hat{v}_g$ . Noul model, încă sub forma unei scheme bloc, este

cel din Fig. 6.21. De observat că acum câștigul  $h_x$  este un vector. Comparând Fig. 6.20 cu Fig. 6.21 și folosind ecuațiile specifice fiecărei topologii, prin identificare se obține:

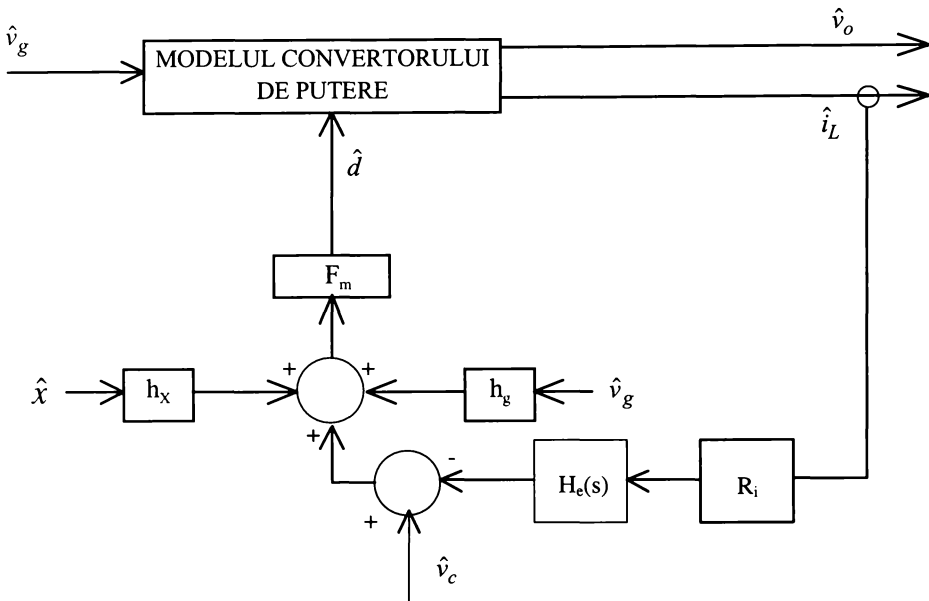


Fig. 6.21. Noul model sub formă de schemă bloc cu vectorul de stare și tensiunea de alimentare ca intrări în blocurile de câștig.

$$h_x = e_1(K_1 A_1 - K_2 A_2) \qquad h_g = e_1(K_1 B_1 - K_2 B_2) \qquad (48)$$

unde vectorul  $e_1 = [1 \ 0]$  extrage curentul  $i_L$  din vectorul de stare  $x = [i_L \ v_C]^T$ . Expresiile de mai sus pentru  $e_1$  și  $x$  au admis un convertor cu o singură bobină și un singur condensator, dar expresiile sunt aceleași pentru orice alt tip de convertor PWM CCM.

Din examinarea Fig. 6.21 se remarcă existența blocului  $H_e(s)$ , deci pentru a obține reprezentarea în spațiul stărilor dorită trebuie găsită o reprezentare în spațiul stărilor pentru funcția:

$$G(s) = \frac{1}{H_e(s)} \qquad (49)$$

Acest lucru se realizează folosind un observator canonic, prin introducerea unui vector de stare suplimentar,  $x'$  și are forma:

$$\begin{cases} \frac{dx'}{dt} = A' x' + B' u \\ y = E' x' + B' u \end{cases} \qquad (50)$$

unde

$$A' = \begin{bmatrix} -\frac{\omega_n}{Q_z} & 1 \\ -\omega_n^2 & 0 \end{bmatrix} \qquad B' = \begin{bmatrix} 0 \\ \omega_n^2 \end{bmatrix} \qquad E' = [1 \ 0] \qquad F' = [0] \qquad (51)$$

De remarcat că semnificația fizică a componentelor vectorului de stare  $x'$  nu are importanță deoarece se dorește obținerea funcțiilor de transfer, care se știe că sunt independente de stare. Definind vectorul de stare extins:

$$x = [\hat{x} \quad x']^T \quad (52)$$

folosind relațiile (45), (48) și modelul din Fig. 6.21 se obține modelul dorit în spațiul stărilor:

$$\begin{cases} \frac{dx}{dt} = Ax + B\hat{v}_g + J\hat{v}_c \\ y = Ex + F\hat{v}_g + H\hat{v}_c \end{cases} \quad (53)$$

cu

$$A = \begin{bmatrix} \left[ I_n - \frac{\xi_D R_i e_1}{R_i e_1 \xi_D} \right] A_D & \frac{\xi_D E' A'}{R_i e_1 \xi_D} \\ B' \left[ h_x + \frac{R_i e_1 A_D}{F_m R_i e_1 \xi_D} \right] & \left[ I_2 - \frac{B' E'}{F_m R_i e_1 \xi_D} \right] A' \end{bmatrix} \quad B = \begin{bmatrix} \left[ I_n - \frac{\xi_D R_i e_1}{R_i e_1 \xi_D} \right] B_D \\ B' \left[ h_g + \frac{R_i e_1 B_D}{F_m R_i e_1 \xi_D} \right] \end{bmatrix} \quad (54)$$

$$J = \begin{bmatrix} 0_{n \times 1} \\ B' \end{bmatrix} \quad E = \begin{bmatrix} E_D - \frac{\xi_D R_i e_1 A_D}{R_i e_1 \xi_D} & \frac{\xi_D E' A'}{R_i e_1 \xi_D} \end{bmatrix} \quad F = \begin{bmatrix} F_D - \frac{\xi_D R_i e_1 B_D}{R_i e_1 \xi_D} \end{bmatrix}$$

$$H = [0]$$

unde  $I_n$  și  $I_2$  sunt matricile unitate de ordin  $n$  respectiv 2, iar  $n$  este ordinul convertorului de putere.

Verificarea și aplicațiile modelului în spațiul stărilor obținut s-a făcut pe același convertor BUCK cu cel studiat în [182], pentru a avea elemente de comparație. Parametrii convertorului sunt:

$$V_g = 12 \text{ V}; \quad V_O = 5 \text{ V}; \quad L = 37.5 \mu\text{H} \quad C = 380 \text{ nF}; \quad R_C = 20 \text{ m}\Omega; \quad f_S = 90 \text{ kHz}; \quad R = 10\Omega$$

Pentru că modelul în spațiul stărilor obținut utilizează matrici, programul MATLAB este adecvat pentru analiză. Autorul a elaborat un program general, dezvoltat în Anexa A17, care analizează orice tip de convertor comandat prin sarcină. Amplitudinea funcției de transfer control-ieșire este prezentată în Fig. 6.22, iar faza în Fig. 6.23, coincidența fiind, așa cum era de așteptat, perfectă cu aceleași caracteristici din [182].

Tot în MATLAB, cu ajutorul toolbox-ului “Control System” polii și zerourile funcției de transfer rezultă:

$$p_1 = 0; \quad p_2 = -83; \quad p_3 = (-0.9 + 4.4i) \times 10^4 \quad p_4 = (-0.9 - 4.4i) \times 10^4 \\ z_1 = -2 \times 10^4; \quad z_2 = -11 \times 10^{-4}$$

Poziția polilor arată că este vorba de un convertor stabil. Dacă însă  $R$  este crescută la 20  $\Omega$  polii sistemului sunt:

$$p_1 = -41; \quad p_2 = 0; \quad p_3 = (0.29 + 4.5i) \times 10^4 \quad p_4 = (0.29 - 4.5i) \times 10^4$$

și polii complecși conjugați fiind în semiplanul drept indică un sistem instabil, partea lor imaginară arătând instabilitate pe jumătate din frecvența de comutație. Acest rezultat este în concordanță cu cele raportate în [182], unde se prezice instabilitate pe jumătate din frecvența de comutație dacă:

$$\frac{L f_S}{R} \leq \frac{D}{2} \quad (55)$$

Acest lucru se și verifică pentru valorile elementelor de circuit. Desigur, prin calcul simbolic se pot obține o serie de relații utile din modelul în spațiul stărilor, modelul fiind util în determinarea funcțiilor de transfer de interes, determinarea polilor și zerourilor prin implementarea sa facilă în programe de analiză de sistem de uz general.

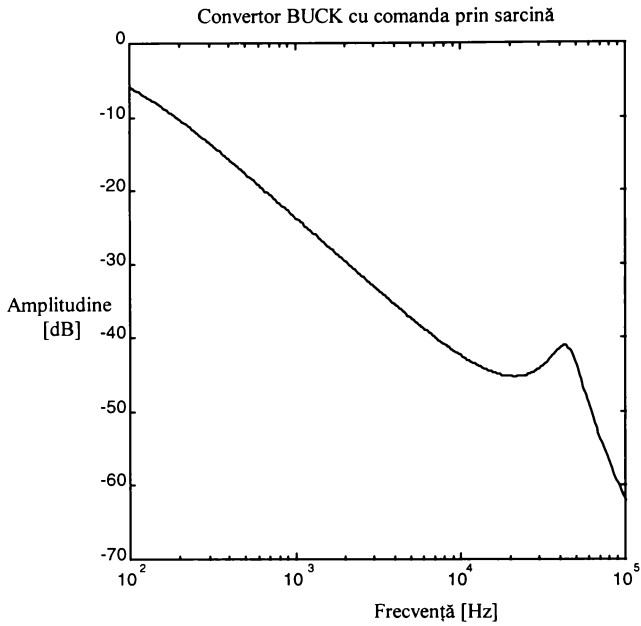


Fig. 6.22. Caracteristica de amplitudine a funcției de transfer control-ieșire..

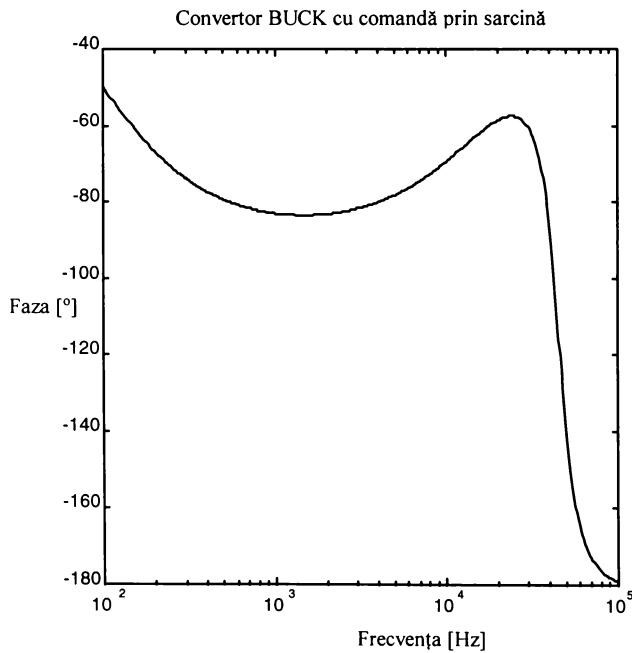


Fig. 6.23. Caracteristica de fază a funcției de transfer control-ieșire..

### 6.5. Un model de semnal mic îmbunătățit pentru comanda integrativă după o variabilă

Smedley și Čuk [170] au arătat că atunci când mărimea care se integrează este o intrare în convertorul cu comandă integrativă după o variabilă, acesta este stabil. În schimb, atunci când mărimea integrată este funcție de variabilele de stare, convertorul cu comandă integrativă după o variabilă poate deveni instabil. Acest lucru are loc de exemplu în convertorul ČUK - Fig. 6.24 - în care factorul de umplere depinde de tensiunea pe de condensatorul  $C_1$ :

$$\frac{1}{\tau} \int_0^{d_k T_s} v_{C_1}(t) dt = v_C \quad (56)$$

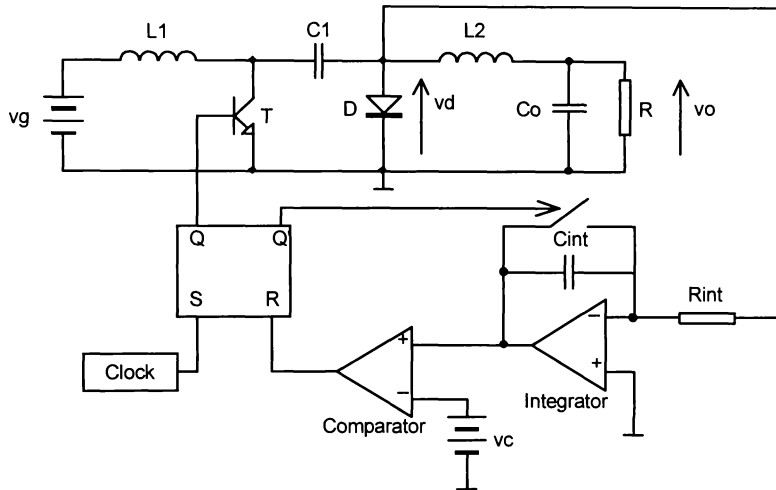


Fig. 6.24. Convertor ČUK cu comandă integrativă după o variabilă.

În [170] cei doi autori, folosind o tehnică de modelare bazată pe grafuri de fluentă, propun un model de semnal mic pentru acest sistem bazat pe următoarea relație între mărimile mediate:

$$d = \frac{v_C}{v_{C_1}} \quad (57)$$

care perturbată duce la :

$$\hat{d} = \frac{1}{V_{C_1}} \hat{v}_C - \frac{D}{V_{C_1}} \hat{v}_{C_1} \quad (58)$$

Din păcate în acest model se neglijează efectul de eșantionare-memorare prezent în perturbațiile din tensiunea  $v_{C_1}$ . **Autorul tezei** propune în [77] un model îmbunătățit, care ține cont și de natura eșantionată și discretă a sistemului. Tehnica aplicată este inspirată din [161], de la modelarea comenzii în curent. Ideea de la care se pleacă este că, examinând (56), rezultă că  $d_k$  depinde de  $v_C$  și  $v_{C_1}$ . Într-un convertor ČUK CCM  $v_{C_1}$  are o variație liniară în cele două stări topologice. Prin urmare ea va fi complet definită de cele două pante și de valoarea din punctul  $t = t_k + DT_s$ , unde  $t_k = kT_s$ . În concluzie factorul de umplere perturbat,  $\hat{d}_k$ , va depinde de trei parametri:  $\hat{v}_C$ , perturbațiile din cele două pante ale lui  $v_{C_1}$  și perturbația din  $v_{C_1}$  de la momentul  $t = t_k + DT_s$ , notată  $\hat{v}_{C_1}(k)$ . Efectul fiecărei mărimii de semnal mic din cele trei menționate este analizat prin superpoziție. Natura eșantionată a sistemului apare atunci când se studiază influența lui  $\hat{v}_{C_1}(k)$ :

pantele fiind presupuse constante, perturbația treaptă aplicată la  $t = t_k + DT_S$  menține tensiunea  $\hat{v}_{C_1}(t)$  perturbată dar constantă până la momentul  $t = t_{k+1} + d_{k+1}T_S$ , astfel încât, cu bună aproximație, legătura dintre  $\hat{v}_{C_1}[k]$  și  $\hat{v}_{C_1}(t)$  este dată de un memorator de ordinul 0, vezi Fig. 6.25.

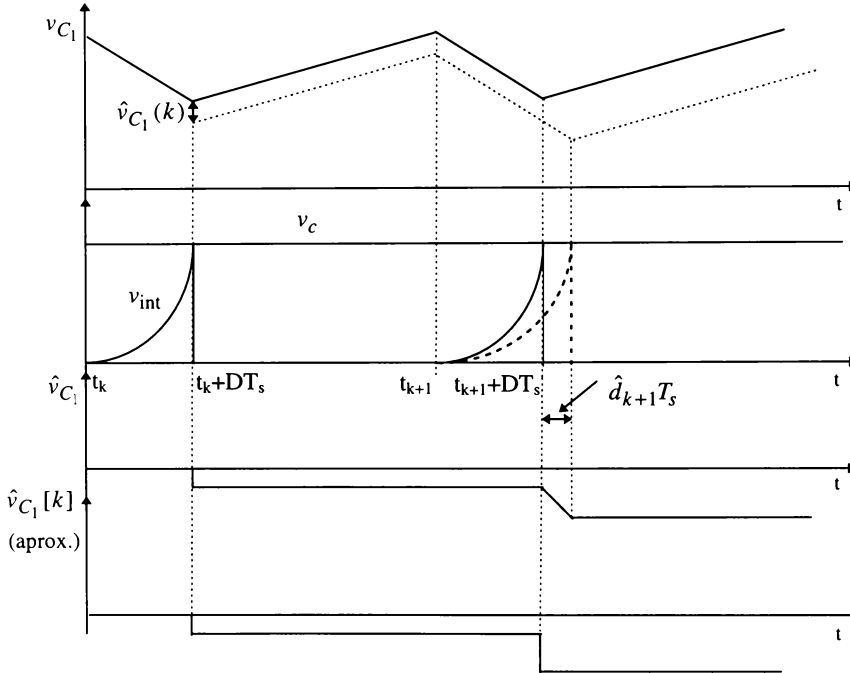


Fig. 6.25. Efectul de eșantionare în tensiunea  $\hat{v}_{C_1}$ .

Modelul obținut are schema bloc din Fig. 6.26. Deducerea sa a fost efectuată în articolul publicat de **autorul tezei** în [77]. Parametrii modelului sunt:

$$F_m = \frac{\tau}{T_S V_{C_1} (DT_S)} \quad A_v = \frac{DT_S}{\tau} \quad (59)$$

$$k_{ON} = \frac{D^2 (1-D) T_S^2}{2C_1 \tau} \quad k_{OFF} = -\frac{D(1-D)^2 T_S^2}{2C_1 \tau} \quad (60)$$

în timp ce  $H_e(s)$  este aceeași cu cea dată de relațiile (43). Evident, modelul este de ordinul 6.

Pentru comparație s-a determinat funcția de transfer control-ieșire în două moduri: cu noul model propus și folosind vechiul model utilizat de Smedley și Čuk. Programul de determinare și afișare a celor două funcții de transfer, scris în MATLAB, este dat în Anexa A18. Convertorul investigat are următorii parametri:

$$V_g = 12 \text{ V}; \quad D = 0,25; \quad L_1 = 0,62 \text{ mH}; \quad R_{L1} = 30 \text{ m}\Omega; \quad L_2 = 0,3 \text{ mH} \quad R_{L2} = 10 \text{ m}\Omega; \quad C_1 = 4,7 \mu\text{F};$$

$$C_2 = 2,2 \mu\text{F}; \quad R_{C2} = 80 \text{ m}\Omega; \quad R = 10 \Omega; \quad f_S = 40 \text{ kHz}; \quad \tau = 25 \mu\text{s}$$

Caracteristicile amplitudine-frecvență sunt trasate în Fig. 6.27, iar cele fază-frecvență în Fig. 6.28. Diferențele se observă că apar la frecvențe înalte, fiind mai pronunțate în caracteristica de fază. Din expresia analitică a funcției de transfer control-ieșire au fost calculați polii și zerourile (valorile sunt exprimate în hertzi):



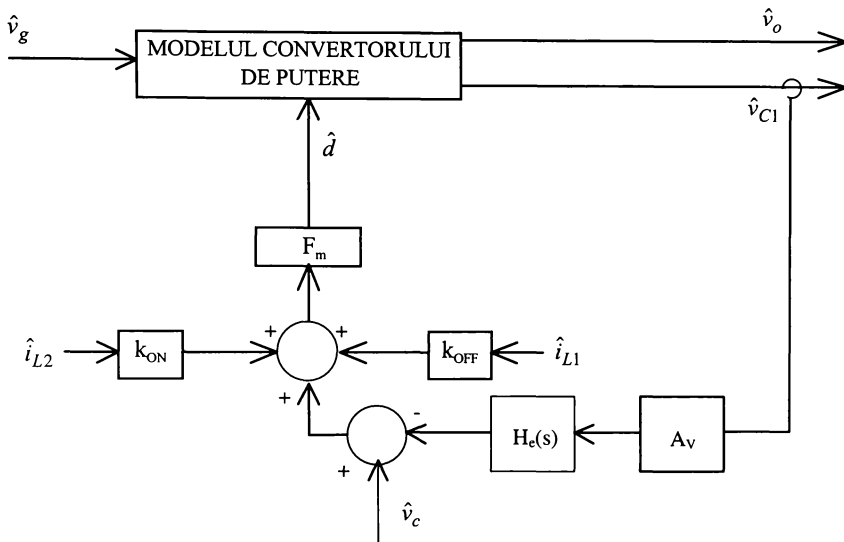


Fig. 6.26. Modelul general mediat de semnal mic al convertorului ĆUK cu comandă integrativă după o variabilă.

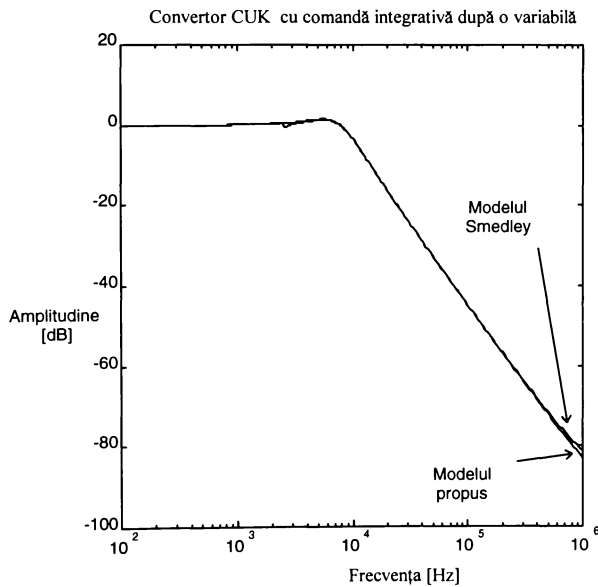


Fig. 6.27. Caracteristica de amplitudine a funcției de transfer control-ieșire..

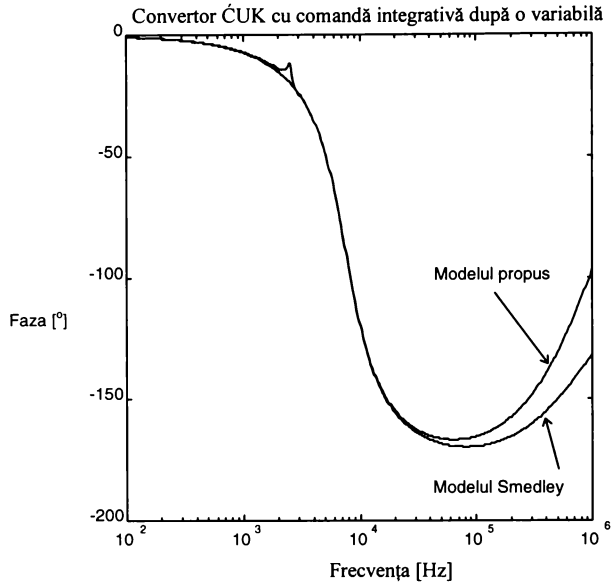


Fig. 6.28. Caracteristica de fază a funcției de transfer control-ieșire..

$$p_1 = 1,42 \times 10^6; \quad p_2 = 146,8 + 253i; \quad p_3 = 146,8 - 253i;$$

$$p_4 = -3637 + 6665i; \quad p_5 = -3637 - 6665i; \quad p_6 = 0;$$

$$z_1 = -9 \times 10^5; \quad z_2 = 0,016; \quad z_3 = 137,1 + 2549i; \quad z_4 = 137,1 - 2549i;$$

Examinarea poziției polilor prognozează instabilitate pe o frecvență de 2,531 kHz. Pentru verificare convertorul a fost simulat cu programul CASPOC - Anexa A19. Formele de undă rezultate în urma simulării fiind cele din Fig. 6.29. Instabilitatea prognozată este întocmai verificată.

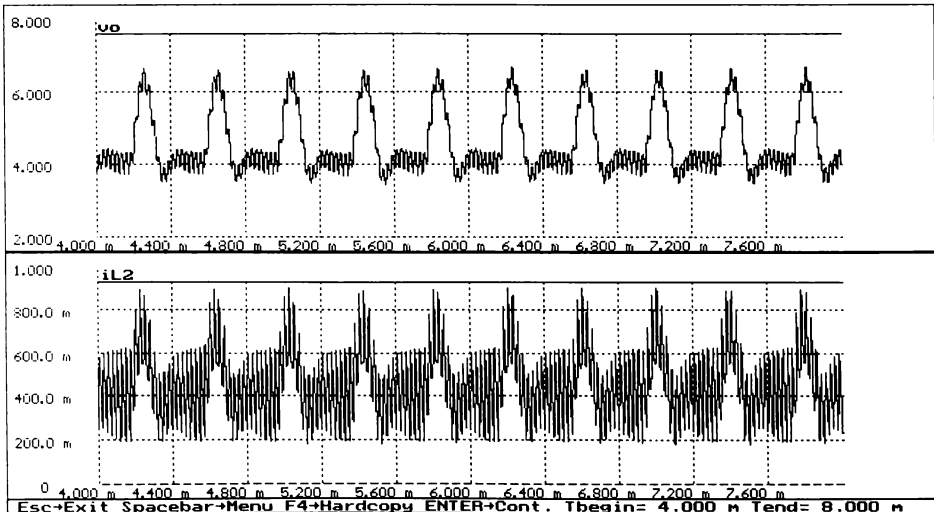


Fig. 6.29. Rezultatele simulării convertorului ĆUK care relevă instabilitatea prognozată.

### 6.6. Noi modele de semnal mare și de semnal mic pentru convertoare cvasirezonaante

Aceste modele unitare au fost publicate de **autorul tezei** în [84]. Obținerea acestor modele pleacă de la două observații privind formele de undă tipice dintr-un convertor cvasirezonaant. Se consideră mediul de comutație al unui convertor QRC [209], [210] din Fig. 6.30. Pentru convertoarele QRC în sensul clasic al cuvântului,  $v_{off}$  și  $i_{on}$  au pulsații reduse, putând fi admise constante într-o perioadă de comutație, astfel încât formele de undă au aspecte tipice, aceleași în orice convertor. În Fig. 6.31 se prezintă tensiunea pe diodă pentru convertoare ZVS.

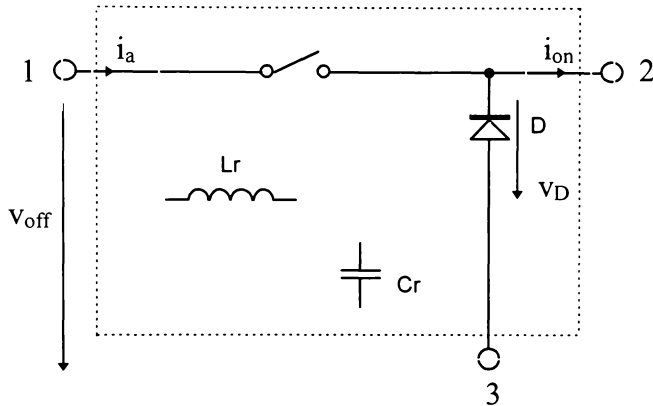


Fig. 6.30. Mediul de comutație pentru un convertor QRC.

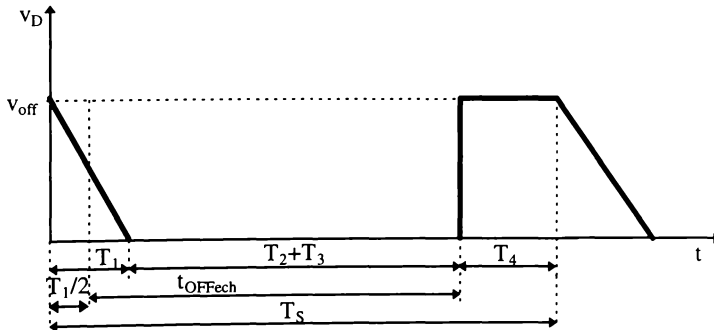


Fig. 6.31. Tensiunea pe diodă,  $v_D$ , într-un convertor ZVS.

Examinarea acestei forme de undă arată că celula cvasirezonaantă ZVS conectează tensiunea  $v_{off}$  între bornele 2 și 3 pe o durată echivalentă egală cu:

$$t_{ONechZVS} = \frac{T_1}{2} + T_4 \quad (61)$$

Un raționament similar, care se poate face invocând principiul dualității, arată că în convertoarele ZCS celula cvasirezonaantă conectează curentul  $i_{on}$  la borna 1 pe o durată echivalentă:

$$t_{ONechZCS} = \frac{T_1}{2} + T_2 + T_3 \quad (62)$$

Timpii  $T_1$ ,  $T_2$ ,  $T_3$  și  $T_4$  fiind cunoscuți ca și expresii în fiecare tip de convertor, rezultă că celula cvasirezonaantă se comportă ca o celulă PWM, având timpul de conectare  $t_{ONech}$  dependent de

tipul de convertor și, mai important, dependent de  $i_{on}$  și  $v_{off}$ . Se poate defini deci și pentru convertoare cvasirezonaante un factor de umplere  $d$  care depinde de  $i_{on}$  și  $v_{off}$ , având expresiile:

$$d_{ZCS} = \frac{t_{ONechZCS}}{T_S} = \frac{1}{2\pi} \frac{f_s}{f_0} q\left(\frac{i_{on}Z_0}{v_{off}}, n\right) \quad (63)$$

$$d_{ZVS} = \frac{t_{ONechZVS}}{T_S} = 1 - \frac{1}{2\pi} \frac{f_s}{f_0} q\left(\frac{v_{off}}{i_{on}Z_0}, n\right) \quad (64)$$

unde  $q(x)$  este funcția cvasirezonaantă definită în paragraful 4.3.1, relația (61).

Rezultă deci că modelul neliniar de semnal mare pentru orice convertor QRC este de forma:

$$\begin{cases} \frac{dx}{dt} = [dA_1 + (1-d)A_2]x + [dB_1 + (1-d)B_2]v_g \\ v_o = [dE_1 + (1-d)E_2]x + [dF_1 + (1-d)F_2]v_g \end{cases} \quad (65)$$

unde  $x = \begin{bmatrix} v_{off} \\ i_{on} \end{bmatrix}$ , iar matricile  $A_1, A_2, B_1, B_2, E_1, E_2, F_1, F_2$  sunt matricile ce caracterizează

convertorul PWM CCM din care provine convertorul cvasirezonaant. Evident modelul este unul mediat, simbolul de mediere fiind omis pentru simplitatea scrierii. De reținut că  $d$  depinde puternic și profund neliniar de vectorul de stare.

Modelul de semnal mic se obține liniarizând modelul de semnal mare (literele mari semnifică mărimile aferente punctului static de funcționare, celelalte notații fiind cele date de relațiile (6)-(11)). Pentru convertoare ZCS el este:

$$\begin{cases} \frac{d\hat{x}}{dt} = A_D\hat{x} + B_D\hat{v}_g + \xi\hat{d} \\ \hat{v}_o = E_D\hat{x} + F_D\hat{v}_g + \zeta\hat{d} \end{cases} \quad (66)$$

cu

$$\hat{d} = \frac{1}{2\pi f_0} \left[ q\left(\frac{I_C Z_0}{V_{ap}}, n\right) \hat{f}_S + F_S Z_0 q'\left(\frac{I_C Z_0}{V_{ap}}, n\right) \frac{V_{ap} \hat{I}_C - I_C \hat{v}_{ap}}{V_{ap}^2} \right] \quad (67)$$

și

$$D = \frac{1}{2\pi} \frac{F_S}{f_0} q\left(\frac{I_C Z_0}{V_{ap}}\right) \quad (68)$$

Atât modelul de semnal mare cât și cel de semnal mic sunt *modele generale, unitare, valabile pentru orice fel de convertor*. Informația despre tipul convertorului este conținută în matricile  $A_1, A_2, B_1, B_2, E_1, E_2, F_1, F_2$ , în timp ce tipul comutației își face simțită prezența prin expresia caracteristică pentru  $d$ . Din acest model de semnal mic se pot deduce expresiile unitare ale funcțiilor de transfer, care însă nu vor mai fi reproduse aici din cauza lungimii lor. Singura dificultate este că rezolvarea stării staționare conduce la ecuații transcendentente. Acest lucru a fost însă rezolvat prin metode numerice, așa cum s-a arătat în paragraful 4.3.1.

Un caz interesant îl reprezintă modul cu undă plină, în care se poate aproxima  $q(x,2) \cong 2\pi$ . În aceste condiții

$$D = \frac{F_S}{f_0} \quad \hat{d} = \frac{\hat{f}_S}{f_0} \quad (69)$$

și modelul devine identic cu cel al convertorului PWM CCM, cu observația că  $\hat{d}$  este substituit cu  $\frac{\hat{f}_S}{f_0}$ . Considerații identice sunt adevărate și pentru convertoarele ZVS. Este de subliniat faptul că atât timp cât ipotezele făcute sunt respectate modelul rămâne valabil și în condițiile existenței unor

elemente parazite în elementele convertorului PWM părinte (de exemplu rezistențele serie ale bobinelor convertorului PWM din care provine).

Se dau mai jos două exemple de aplicare a modelului. Primul dintre ele îl reprezintă simularea accelerată la semnal mare. Convertorul studiat este un BOOST ZVS QRC HW al cărui model a fost asimilat în mediul MATLAB/SIMULINK. Programul este dat în Anexa A20. Parametrii convertorului sunt:

$$V_g = 15V; \quad L = 400 \mu H; \quad C = 100 \mu F; \quad L_r = 160 nH \quad C_r = 64 nF; \quad R = 1.26 \Omega$$

scopul studiului fiind comportarea tensiunii de ieșire la salt treaptă în frecvența de comutație,  $f_S$ , la momentul  $4 ms$ , de la  $0,2f_0$  la  $0,6f_0$ . În Fig. 6.32 se prezintă rezultatele folosind modelul propus, iar în Fig. 6.33 rezultatele folosind simulatorul PETS - Anexa A21.

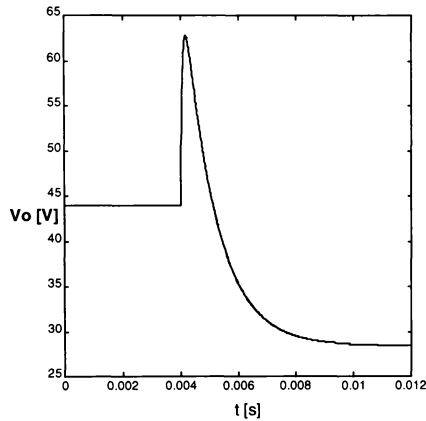


Fig. 6.32. Răspunsul convertorului folosind modelul propus.

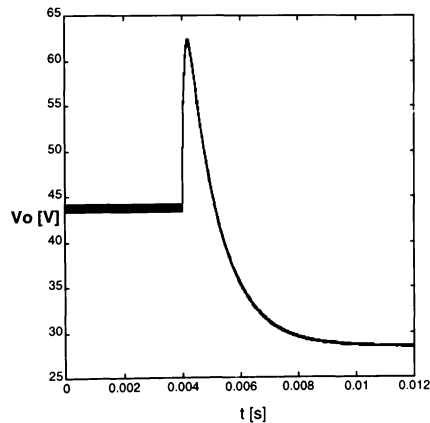


Fig. 6.33. Răspunsul convertorului folosind simulatorul dedicat PETS (Boulder University Colorado).

Se observă precizia excelentă a modelului în condițiile în care  *timpul de simulare este aproximativ cu două ordine de mărime mai mic.*

Al doilea exemplu își propune calcularea funcției de transfer control-ieșire pentru un convertor BOOST ZCS QRC FW având parametrii:

$$V_g = 15V; \quad L = 500 \mu H; \quad C = 100 \mu F; \quad L_r = 3.47 \mu H; \quad C_r = 430 nF; \quad R = 10 \Omega; \quad F_S = 30 kHz$$

identici cu cei din exemplul dat în [204], pentru a avea un element de comparație. Matricile implicate în model pentru convertorul BOOST sunt:

$$A_1 = \begin{bmatrix} 0 & 0 \\ 0 & -\frac{1}{RC} \end{bmatrix} \quad B_1 = \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} \quad E_1 = [0 \ 1] \quad F_1 = [0]$$

$$A_2 = \begin{bmatrix} 0 & -\frac{1}{L} \\ \frac{1}{C} & -\frac{1}{RC} \end{bmatrix} \quad B_2 = \begin{bmatrix} \frac{1}{L} \\ 0 \end{bmatrix} \quad E_2 = [0 \ 1] \quad F_2 = [0]$$

care înlocuite în (66) și folosind (69) duc la expresia căutăată:

$$\frac{\hat{v}_o}{\hat{f}_S} = \frac{1}{f_0} \frac{V_g}{\left(1 - \frac{F_S}{f_0}\right)^2} \cdot \frac{1 - \frac{L}{R \left(1 - \frac{F_S}{f_0}\right)^2} s}{1 + \frac{L}{R \left(1 - \frac{F_S}{f_0}\right)^2} s + \frac{LC}{\left(1 - \frac{F_S}{f_0}\right)^2} s^2} \quad (70)$$

adică exact expresia găsită în [204]. Amplitudinea și faza funcției de transfer sunt reprezentate în Fig. 6.34. Programul MATLAB aferent este prezentat în Anexa A 22.

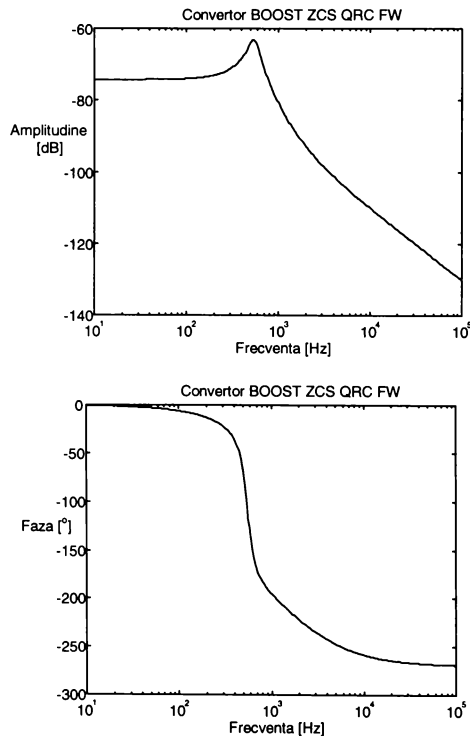


Fig. 6.34. Caracteristicile de amplitudine și de fază.

### 6.7 Două modele discrete noi pentru convertoare în comutație

Modelele discrete ce vor fi propuse în continuare au apărut ca o necesitate de a îmbunătăți modelul discret elaborat de Kassakian ș.a. în [63]. Acest model se bazează pe dezvoltarea în serie Taylor a vectorului de stare și exprimarea în acest fel a valorii vectorului într-un moment în care are loc o comutație în funcție de valoarea de la momentul de comutație anterior. Astfel, mergând regresiv, se exprimă vectorul de stare de la sfârșitul perioadei de comutație în funcție de valoarea de la începutul perioadei de comutație, construindu-se o ecuație cu diferențe în care variabila discretă este indexată după numărul perioadei de comutație.

Deși modelul dă satisfacție în anumite situații, el suferă de un mare dezavantaj. Chiar dacă nu se precizează explicit, practic el se bazează pe o metodă de integrare numerică de tip “forward Euler”, care se cunoaște că nu este stabilă. Cum se va vedea mai jos el prognozează instabilitate în convertoare stabile, ceea ce este inadmisibil.

Modelele propuse de **autorul tezei** se bazează pe aceeași abordare recursiv-regresivă în valorile vectorului de stare, însă relația utilizată folosește o integrare numerică (aproximare) “backward Euler” (cu diferență regresivă) respectiv “trapezoidală”. Modelele se vor prezenta aplicându-le pe convertoare PWM CCM, dar extensia la alte moduri sau convertoare este absolut identică.

Pentru un convertor PWM CCM se consideră o perioadă oarecare care începe de la  $kT_s$  și se termină la  $(k+1)T_s$ , comutarea între cele două stări topologice făcându-se la momentul  $kT_s + d_k T_s$  - Fig. 6.35. Se pleacă deci de la modelul comutat al convertorului.

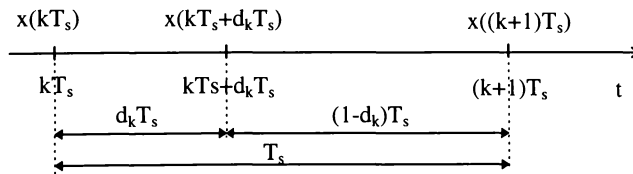


Fig. 6.35. Momentele de comutație dintr-o perioadă.

Se cunoaște că pentru ecuația

$$\frac{dx}{dt} = f(x(t), t) \tag{71}$$

algoritmul de integrare numerică de tip “backward Euler” este dat de:

$$x_{n+1} = x_n + hf(x_{n+1}, t_{n+1}) \tag{72}$$

unde  $h$  este pasul de integrare numerică iar  $(t_n, x_n)$  și  $(t_{n+1}, x_{n+1})$  două puncte succesive din rețeaua de puncte pe care se face integrarea, cu notația  $x_n = x(t_n)$ .

Cu aceleași notații, metoda trapezoidală este tot un algoritm în 2 puncte dat de:

$$x_{n+1} = \frac{1}{2}h[f(x_n, t_n) + f(x_{n+1}, t_{n+1})] \tag{73}$$

Deoarece convertoarele PWM CCM sunt sisteme LTI pe porțiuni se pot aplica cele două aproximări pentru începutul și sfârșitul celor două stări topologice, ținând cont și de ecuațiile matriciale de stare. De exemplu pentru aproximarea trapezoidală se obține:

$$x_{n+1} = x_n + \frac{1}{2}h[A_i x_n + B_i v_{gn} + A_i x_{n+1} + B_i v_{gn+1}] \tag{74}$$

Pentru prima stare topologică  $h = d_k T_s$ ,  $i = 1$ ,  $x_n = x(kT_s)$ ,  $x_{n+1} = x(kT_s + d_k T_s)$ , iar pentru a doua  $h = (1 - d_k) T_s$ ,  $i = 2$ ,  $x_n = x(kT_s + d_k T_s)$ ,  $x_{n+1} = x((k+1)T_s)$ . Eliminând pe

$x(kT_s + d_k T_s)$  din cele două relații și notând  $x(kT_s) = x[k]$  și  $x((k+1)T_s) = x[k+1]$  se obține modelul de semnal mare în forma:

$$\begin{aligned} x[k+1] = & \left[ I - \frac{1}{2}(1-d_k)T_s A_2 \right]^{-1} \left[ I + \frac{1}{2}(1-d_k)T_s A_2 \right] \left[ I - \frac{1}{2}d_k T_s A_1 \right]^{-1} \left[ I + \frac{1}{2}d_k T_s A_1 \right] x[k] + \\ & + \left[ I - \frac{1}{2}(1-d_k)T_s A_2 \right]^{-1} \left[ I + \frac{1}{2}(1-d_k)T_s A_2 \right] \left[ I - \frac{1}{2}d_k T_s A_1 \right]^{-1} d_k T_s B_1 v_g[k] + \\ & + \left[ I - \frac{1}{2}(1-d_k)T_s A_2 \right]^{-1} (1-d_k)T_s B_2 v_g[k] \end{aligned} \quad (75)$$

Modelul de semnal mic se obține liniarizând (75). Deși posibil, această operație este anevoioasă datorită inverselor de matrici care apar. Mai simplu este să se liniarizeze individual cele două ecuații de semnal mare obținute prin aproximarea trapezoidală, din care apoi se elimină  $\hat{x}(kT_s + d_k T_s)$ .

Modelul de semnal mic rezultat este descris prin:

$$\begin{aligned} \hat{x}[k+1] = & \left[ I - \frac{1}{2}(1-D)T_s A_2 \right]^{-1} \left[ I + \frac{1}{2}(1-D)T_s A_2 \right] \left[ I - \frac{1}{2}DT_s A_1 \right]^{-1} \left[ I + \frac{1}{2}DT_s A_1 \right] \hat{x}[k] + \\ & + \left[ I - \frac{1}{2}(1-D)T_s A_2 \right]^{-1} \left[ I + \frac{1}{2}(1-D)T_s A_2 \right] \left[ I - \frac{1}{2}DT_s A_1 \right]^{-1} DT_s B_1 + \\ & + \left[ I - \frac{1}{2}(1-D)T_s A_2 \right]^{-1} (1-D)T_s B_2 \hat{v}_g[k] + \\ & + \left[ I - \frac{1}{2}(1-D)T_s A_2 \right]^{-1} \left\{ -\frac{1}{2}T_s A_2 X(kT_s) - \frac{1}{2}T_s A_2 X(kT_s + DT_s) + \right. \\ & + \left. \left[ I + \frac{1}{2}(1-D)T_s A_2 \right] \left[ I - \frac{1}{2}(1-D)T_s A_2 \right]^{-1} \left[ \frac{1}{2}T_s A_1 X(kT_s + DT_s) + \frac{1}{2}T_s A_1 X(kT_s) + \right. \right. \\ & \left. \left. + T_s B_1 V_g \right] - T_s B_2 B_g \right\} \hat{d}_k \end{aligned} \quad (76)$$

unde  $X(kT_s)$  se obține din (75) făcând  $x[k] = x[k+1] = X(kT_s)$ ,  $d_k = D$  și  $v_g[k] = V_g$ , iar

$$X(kT_s + DT_s) = \left[ I - \frac{1}{2}DT_s A_1 \right]^{-1} \left[ I + \frac{1}{2}DT_s A_1 \right] X(kT_s) + \left[ I - \frac{1}{2}DT_s A_1 \right]^{-1} DT_s B_2 V_g \quad (77)$$

Deși par complicate ca și expresii, cele două modele se pot asimila foarte ușor în MATLAB, 20SIM sau alte programe de simulare de sistem, accelerând substanțial simularea. Modelul de semnal mic se implementează o singură dată, după care se poate studia stabilitatea oricărui convertor numai modificând matricile.

Dacă perioada de comutație este mult mai mică decât constantele de timp ale convertorului se poate dezvolta o variantă de model discret simplificată. Pentru aceasta se pleacă nu de la modelul comutat al convertorului ci de la cel mediat continuu dat de :

$$\frac{d\bar{x}}{dt} = [dA_1 + (1-d)A_2] \bar{x} + [dB_1 + (1-d)B_2] \bar{v}_g \quad (78)$$

Eșantionând acest model cu perioada  $T_s$  și aplicând aproximația trapezoidală se obține modelul discret de semnal mare (în scriere s-a omis simbolul de mediere, subînțelegându-se):

$$\begin{aligned} x[k+1] = & \left[ I - \frac{1}{2}T_s [d[k]A_1 + (1-d[k])A_2] \right]^{-1} \left[ I + \frac{1}{2}T_s [d[k]A_1 + (1-d[k])A_2] \right] x[k] + \\ & + \left[ I - \frac{1}{2}T_s [d[k]A_1 + (1-d[k])A_2] \right]^{-1} T_s [d[k]B_1 + (1-d[k])B_2] v_g[k] \end{aligned} \quad (79)$$



și modelul discret de semnal mic:

$$\hat{x}[k+1] = \left[ I - \frac{1}{2} T_s A_D \right]^{-1} \left[ I + \frac{1}{2} T_s A_D \right] \hat{x}[k] + \left[ I - \frac{1}{2} T_s A_D \right]^{-1} T_s B_D \hat{v}_g[k] + \left[ I - \frac{1}{2} T_s A_D \right]^{-1} T_s \xi \hat{d}[k] \quad (80)$$

unde  $d[k] = \bar{q}(kT_s)$  cu  $q(t)$  funcția de comutație asociată convertorului.

Aplicând același procedeu pentru aproximarea “backward Euler” plecând de la modelul comutat al convertorului se obține următorul model de semnal mare:

$$x[k+1] = \left[ I - d_k T_s A_1 - (1-d_k) T_s A_2 + d_k (1-d_k) T_s^2 A_1 A_2 \right] x[k] + \left[ I - d_k T_s A_1 - (1-d_k) T_s A_2 + d_k (1-d_k) T_s^2 A_1 A_2 \right]^{-1} \cdot \left[ d_k T_s B_1 + (1-d_k) T_s B_2 - d_k (1-d_k) T_s^2 A_1 B_2 \right] v_g[k] \quad (81)$$

și modelul de semnal mic:

$$\hat{x}[k+1] = \left[ I - T_s A_D + D(1-D) T_s^2 A_1 A_2 \right]^{-1} \hat{x}[k] + \left[ I - T_s A_D + D(1-D) T_s^2 A_1 A_2 \right]^{-1} \cdot \left[ T_s B_D - D(1-D) T_s^2 A_1 B_2 \right] \hat{v}_g[k] + \left[ I - T_s A_D + D(1-D) T_s^2 A_1 A_2 \right]^{-1} \cdot \left\{ T_s \left[ (A_1 - A_2) X + (B_1 - B_2) V_g \right] - (1-2D) T_s^2 A_1 (A_2 X + B_2 V_g) \right\} \hat{d}_k \quad (82)$$

cu

$$X = - \left[ T_s A_D - D(1-D) T_s^2 A_1 A_2 \right]^{-1} \left[ T_s B_D - D(1-D) T_s^2 A_1 B_2 \right] V_g \quad (83)$$

Dacă se pleacă de la modelul mediat se obține următorul model de semnal mare:

$$x[k+1] = \left[ I - d[k] T_s A_1 + (1-d[k]) T_s A_2 \right]^{-1} x[k] + \left[ I - d[k] T_s A_1 + (1-d[k]) T_s A_2 \right]^{-1} T_s [d[k] B_1 + (1-d[k]) B_2] v_g[k] \quad (84)$$

modelul de semnal mic fiind descris de:

$$\hat{x}[k+1] = \left[ I - T_s A_D \right]^{-1} \hat{x}[k] + \left[ I - T_s A_D \right]^{-1} T_s B_D \hat{v}_g[k] + \left[ I - T_s A_D \right]^{-1} T_s \xi \hat{d}[k] \quad (85)$$

Pentru a pune în evidență gradul de performanță al modelelor se va studia comportarea unui convertor BOOST PWM CCM cu următorii parametri:

$$V_g = 10 \text{ V}; \quad L = 3,704 \text{ mH}; \quad C = 32,1 \mu\text{F}; \quad R = 180 \Omega; \quad D = 0,6; \quad f_s = 6,25 \text{ kHz}$$

Fișierul MATLAB utilizat în acest scop este *compar.m*, prezentat în Anexa A23.

Se poate verifica ușor că pentru valorile de mai sus ale elementelor de circuit convertorul funcționează într-adevăr în CCM.

Aplicând modelul discret propus de Kassakian în [63] se obțin polii și zerourile:

$$p_f = 0,9862 \pm 0,1851i; \quad z_f = 2,2441; \quad |p_f| = 1,0034$$

Aplicând modelul backward Euler polii și zerourile rezultă:

$$p_b = 0,9545 \pm 0,1743i; \quad z_b = -4,0973; \quad |p_b| = 0,9703$$

iar cu modelul trapezoidal

$$p_t = 0,9696 \pm 0,1810i; \quad z_t = 4,2914; \quad |p_t| = 0,9864$$

Pentru comparație se dau polii și zerourile modelului continuu convertit în model discret cu memorator de ordinul zero:

$$p_c = 0,9694 \pm 0,1815i; \quad z_c = 4,3422; \quad |p_c| = 0,9862$$

Se observă că modelul din [63] prognozează un sistem instabil, ceea ce evident nu poate avea loc fiind vorba de un convertor banal, comandat în factor de umplere, fără buclă de reacție. De

asemenea este clar că cel mai aproape de modelul continuu, considerat etalon, este modelul trapezoidal, deși și modelul “backward Euler” prognozează corect stabilitatea. Fig. 6.36, unde se

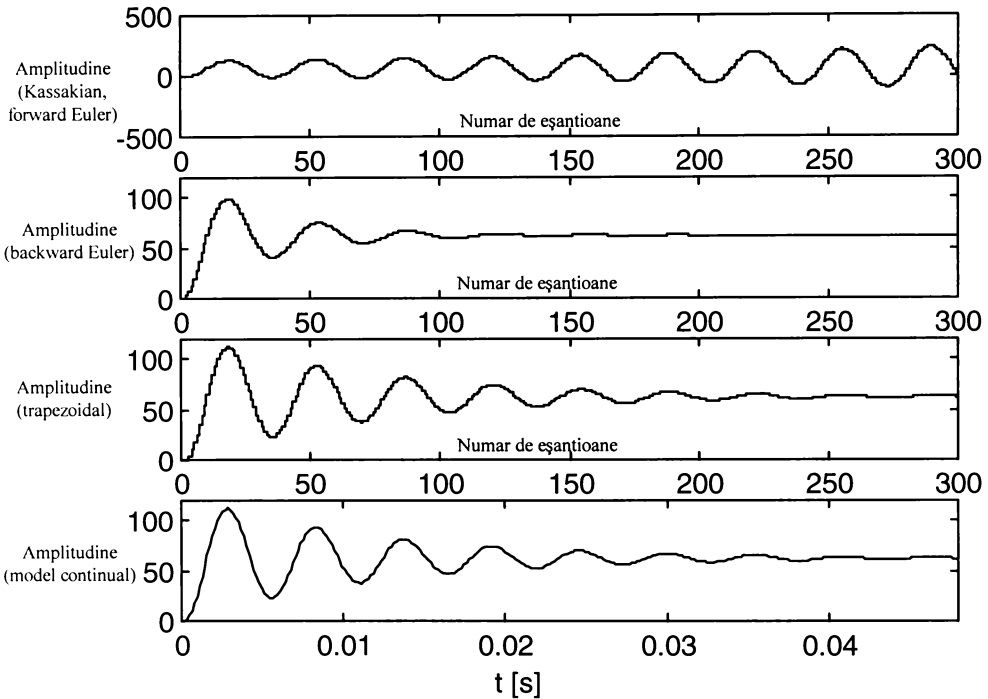


Fig. 6.36.

prezintă răspunsurile la semnal treaptă ale modelelor analizate, ilustrează foarte bine acest lucru. Se remarcă coincidența foarte bună între modelul trapezoidal și cel continuu, eroarea ceva mai mare a modelului “backward Euler” care însă compensează prin simplitate și erorile inadmisibile ale modelului din [63].

A fost de asemenea efectuată o verificare experimentală a modelului trapezoidal. Topologia studiată a fost un BOOST în care elementele de circuit au valorile:

$$V_g = 10V; L = 3,704 \text{ mH}; R_L = 0,47 \Omega; C = 32 \text{ ,}\mu\text{F}; R_c = 0,356 \Omega; R = 67,7 \Omega; f_s = 6,25 \text{ kHz}$$

S-au aplicat salturi treaptă în factorul de umplere de la  $D_1 = 0,56$  la  $D_2 = 0,54$  și invers, variația tensiunii de ieșire anticipată de modelul discret de semnal mic provenit din modelul mediat și folosind aproximația trapezoidală fiind cea din Fig. 6.37. Fișierele utilizate au fost *trapez.m* (fișier MATLAB) și *zt\_model.m* (fișier SIMULINK), prezentate în Anexa A24. Variația practică, reală a tensiunii de ieșire este prezentată în Fig. 6.38, aceasta fiind obținută prin comanda cu un modul EVM30. Concordanța formelor de undă prognozate de modelul trapezoidal este perfectă, graficele diferind doar prin pulsațiile de înaltă frecvență pe care modelul discret, dat fiind că provine dintr-un model mediat, le neglijează. În schimb timpii de răspuns, supracreșterile și perioadele oscilațiilor amortizate coincid cu precizie.

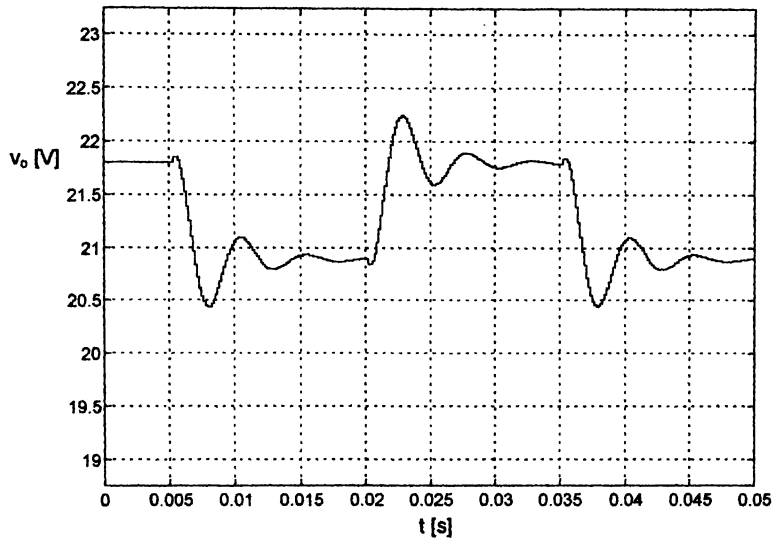


Fig. 6.37. Forma de undă a tensiunii de ieșire prognozată de modelul discret cu aproximare trapezoidală provenit din modelul mediat.

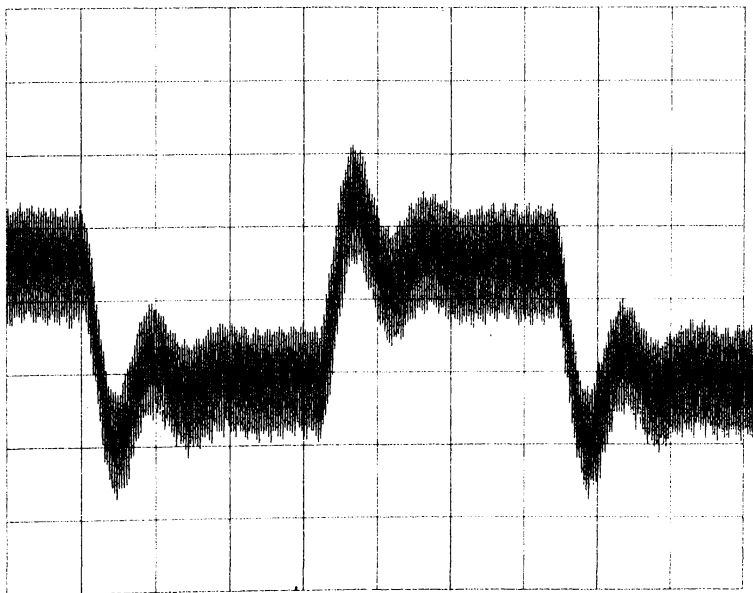


Fig. 6.38. Forma de undă a tensiunii de ieșire obținută experimental.  $Y=0,5V/div$ ;  $X=5ms/div$ .

### 6.8. Analiza de semnal mic exactă pentru convertoare dc-dc cu funcționare la frecvență constantă

S-au raportat numeroase moduri de calcul sau de reprezentare a funcțiilor de transfer control-ieșire [36], [133], unele dintre ele de o manieră generală sau unificată [134], [138]. Toate însă se bazează pe o mediere explicită sau implicită, care le limitează în frecvență la jumătatea frecvenței de comutație. Recent a fost expusă o metodă promițătoare [120] de reprezentare a funcțiilor de transfer bazate pe răspunsul la impuls al convertorului folosind un simulator. Lucrarea nu precizează însă dependența erorii de pasul de simulare, de felul cum se aproximează în simulare impulsul și nu oferă a formulă în formă analitică pe care ulterior să se poată face unele considerații sau din care să se obțină expresii aproximative în formă de funcție rațională.

Convertoarele în comutație sunt sisteme variabile în timp la care comanda se realizează, în ultimă instanță prin modularea intervalelor de timp asociate diverselor stări topologice. Pentru analiza sistemelor variabile în timp Zadeh [217] propune o funcție de transfer dependentă de timp. Plecând de la remarcă binecunoscută că convertoarele dc-dc în comutație sunt sisteme variabile în timp dar sunt liniare pe porțiuni (LTI pe porțiuni), Tymerski [193] dezvoltă un procedeu de deducere a *funcției de transfer control-ieșire exacte* pentru rețele de comutație, aplicând-o la convertoare PWM CCM cu comanda în factor de umplere. Ulterior tot Tymerski deduce [192] funcțiile de transfer control ieșire exacte pentru convertoare PWM DICM, convertoare PWM CCM cu comandă în curent și convertoare PWM DICM cu comandă în curent. Între timp procedeul de calcul exact al funcției de transfer propus de Tymerski a devenit un standard cu care se compară toate celelalte modele dezvoltate ulterior. El este sintetizat sumar în cele ce urmează.

#### 6.8.1. Algoritmul Tymerski de calcul exact al funcției de transfer control-ieșire pentru convertoare dc-dc cu funcționare la frecvență constantă

Algoritmul ce va fi prezentat mai jos se bazează în mare măsură pe rezultatele publicate de Tymerski în [193] și [192]. Ceea ce autorul tezei aduce nou este o sistematizare a rezultatelor și prezentarea acestora sub forma unui algoritm aplicabil oricărui convertor dc-dc în comutație.

Dându-se un convertor dc-dc cu perioada de comutație constantă  $T_S$ , alimentat de la o sursă de tensiune continuă fixă  $V_g$ , a cărei funcționare se cicleză prin  $N_S$  stări topologice liniare și invariabile în timp, atunci sistemul reprezentat de convertor este descris de următorul set de ecuații de stare și de ieșire:

$$\frac{dx}{dt} = A_i x(t) + B_i V_g \quad (86a)$$

$$y(t) = E_i x(t) + F_i V_g \quad (86b)$$

pentru  $t_i < t < t_{i+1}$  și  $i \in \{1, 2, \dots, N_S\}$ .  $t_i$  și  $t_{i+1}$  reprezintă momentele de început respectiv de sfârșit ale celei de a  $i$ -a topologii. Pentru o analiză prin liniarizare vectorul de stare,  $x(t)$ , se consideră format dintr-o componentă aferentă stării staționare,  $X(t)$  și o perturbație de ordinul întâi,  $\hat{x}(t)$  și la fel pentru  $t_i$ :

$$x(t) = X(t) + \hat{x}(t) \quad (87a)$$

$$t_i = T_i + \hat{t}_i \quad (87b)$$

În [193] se arată că între cele două componente din (87) există următoarele relații:

$$X(T_{i+1}) = \Phi_i X(T_i) + \Psi_i V_g \quad (88a)$$

$$\hat{x}(t_{i+1}) = \Phi_i \hat{x}(t_i) + \gamma_i (\hat{t}_{i+1} - \hat{t}_i) \quad (88b)$$

unde

$$\Phi_i = e^{A_i (T_{i+1} - T_i)} \quad (89a)$$

$$\Psi_i = A_i^{-1}[e^{A_i(T_{i+1}-T_i)} - I]B_i \quad (89b)$$

$$\gamma_i = A_i X(T_{i+1}) + B_i V_g \quad (89c)$$

pentru  $\det A_i \neq 0$  (ceea ce are loc întotdeauna dacă se adoptă pentru întrerupătoare un model cu rezistențe nenule în conducție și neînfinită în blocare) și  $I$  matricea unitate de ordin corespunzător. Atunci componenta de semnal mic din ieșire este dată de:

$$\hat{y}(j\omega) = 2H_0(j\omega) \quad (90)$$

cu

$$H_0(j\omega) = \frac{1}{NT_S} \sum_{k=0}^{N-1} \sum_{i=1}^{N_S} e^{-j\omega T_i} [\alpha_i \hat{x}(t_i) + \tau_i \hat{t}_i + \lambda_i \hat{t}_{i+1}] \quad (91)$$

și

$$\alpha_i = E_i(j\omega I - A_i)^{-1}[I - e^{-(j\omega I - A_i)(T_{i+1}-T_i)}] \quad (92a)$$

$$\tau_i = E_i(j\omega I - A_i)^{-1} \{e^{-(j\omega I - A_i)(T_{i+1}-T_i)} [A_i X(T_i) + B_i V_g] - [j\omega X(T_i) + B_i V_g]\} - F_i V_g \quad (92b)$$

$$\lambda_i = [E_i X(T_{i+1}) + F_i V_g] e^{-j\omega(T_{i+1}-T_i)} \quad (92c)$$

$N$  este numitorul numărului rațional egal cu raportul dintre pulsația perturbației de semnal mic,  $\omega$ , și pulsația de comutație,  $\omega_s$ .

Suplimentar, pentru simplificarea scrierii, se fac următoarele notații:

$$\beta_i = e^{-(j\omega I - A_i)(T_{i+1}-T_i)} \quad (93a)$$

$$\xi_i = (A_i - A_{i+1})X(T_{i+1}) + (B_i - B_{i+1})V_g \quad (93b)$$

$$\zeta_i = (E_i - E_{i+1})X(T_{i+1}) + (F_i - F_{i+1})V_g \quad (93c)$$

$$\mu_i = \frac{1}{h_i \gamma_i} \quad (93d)$$

$$\Gamma_i = I - \xi_i \mu_i h_i \quad (93e)$$

$$\Gamma_i' = I - \gamma_i \mu_i h_i \quad (93f)$$

unde  $h_i$  este o matrice numită *extractoare* (din motive ce vor reieși în paragrafele următoare), de dimensiuni adecvate, care intervine într-o constrângere ce determină momentul de comutație  $t_{i+1}$ .

Următoarele identități sunt utile în efectuarea calculelor necesare determinării funcției de transfer exacte:

$$\phi_{i+1} \gamma_i - \gamma_{i+1} = \Phi_{i+1} \xi_i \quad (94a)$$

$$\beta_i = e^{-j\omega(T_{i+1}-T_i)} \Phi_i \quad (94b)$$

$$\xi_i = \gamma_i - [A_{i+1} X(T_{i+1}) + B_{i+1} V_g] \quad (94c)$$

$$\lambda_i + e^{-j\omega(T_{i+1}-T_i)} \tau_{i+1} = [\zeta_i + \alpha_{i+1} (\xi_i - \gamma_i)] e^{-j\omega(T_{i+1}-T_i)} \quad (94d)$$

$$\Gamma_i = \Gamma_i' + [A_{i+1} X(T_{i+1}) + B_{i+1} V_g] \mu_i h_i \quad (94e)$$

Pașii care trebuie parcurși în deducerea expresiei analitice exacte a funcției de transfer control-ieșire sunt:

**Pasul 1.** Determinarea valorilor *exacte* aferente stării staționare.

Din (91) și (92) este clar că sunt necesare valorile din starea staționară  $T_{i+1} - T_i$  și  $X(T_i)$ . Acest lucru este absolut natural, întrucât se cunoaște că starea staționară (punctul static de funcționare) este cea în jurul căreia se face liniarizarea, iar pe de altă parte parametrii modelului liniarizat depind de punctul static de funcționare. Ceea ce trebuie remarcat este faptul că sunt necesare valorile exacte ale duratelor stărilor topologice și ale vectorului de stare în momentele de comutație. Aceasta complică deducerea atunci când nu toți timpii de comutație sunt prescriși extern de convertor, unii dintre ei depinzând de stare. Metode și algoritmi de rezolvare exactă și accelerată a stării staționare sunt prezentați în paragraful 6.8.2.

Pasul 2. Se exprimă toate mărimile de semnal mic din (91) numai funcție de  $\hat{x}(t_1)$  și mărimea de comandă de semnal mic, fie ea  $\hat{c}_k$ .

Aceasta se face utilizând ecuația de semnal mic (88b) și eventualele ecuații de semnal mic ale constrângerilor. Se admite o mărime de comandă cu variație cosinusoidală:

$$\hat{c}_k = c_p \cos(\omega k T_s + \varphi) \quad (95)$$

Pasul 3. Folosind rezultatele de la pasul 2 se calculează  $H_0(j\omega)$ .

În acest scop se dă factor comun forțat  $e^{-j\omega T_1} = e^{-j\omega k T_s}$  și după câteva calcule se obține:

$$H_0(j\omega) = \frac{1}{NT_s} \left[ a \sum_{k=0}^{N-1} \hat{x}(t_1) e^{-j\omega k T_s} + b \sum_{k=0}^{N-1} \hat{c}_k e^{-j\omega k T_s} \right] \quad (96)$$

în care  $a$  și  $b$  nu depind de  $k$ . În acest fel în (96) apar chiar transformatele Fourier discrete ale lui  $\hat{x}(t_1)$  și  $\hat{c}_k$ .

Pasul 4. Se calculează transformatele Fourier discrete pentru  $\hat{x}(t_1)$  și  $\hat{c}_k$ .

Transformata Fourier discretă lui  $\hat{c}_k$  a fost evaluată în [192], rezultatul final fiind:

$$\sum_{k=0}^N \hat{c}_k e^{-j\omega T_1} = N e^{j\varphi} \frac{\hat{c}_p}{2} \quad \frac{\omega}{\omega_s} \neq 0.5, 1, 1.5, \dots \quad (97)$$

Pentru calculul transformatei Fourier discrete a lui  $\hat{x}(t_1)$  este necesară ecuația cu diferențe asociată. Aceasta se obține din toate ecuațiile de semnal mic (88b) aferente tuturor stărilor topologice, în care se ține cont că  $\hat{x}(t_1) = \hat{x}[k]$  și  $\hat{x}_{t_{N_s+1}} = \hat{x}[k+1]$  deoarece  $k$  indexează perioada. Mergând recursiv spre începutul perioadei ecuația cu diferențe obținută va fi de forma:

$$\hat{x}[(k+1)] = \Phi \hat{x}[k] + \gamma \hat{c}_k \quad (98)$$

a cărei soluție este [192]:

$$\sum_{k=0}^N \hat{x}(t_1) e^{-j\omega T_1} = N (I - e^{-j\omega T_s} \Phi)^{-1} \cdot e^{-j(\omega T_s - \varphi)} \gamma \frac{\hat{c}_p}{2} \quad (99)$$

$$\frac{\omega}{\omega_s} \neq 0.5, 1, 1.5, \dots$$

Pasul 5. Se înlocuiesc transformatele Fourier calculate la pasul 4 în expresia lui  $H_0(j\omega)$ , se efectuează calculele finale pentru această mărime și apoi se evaluează funcția de transfer dorită,

$$\frac{\hat{y}}{\hat{c}_p}(j\omega) \text{ folosind (90)}$$

De remarcat că, deoarece utilizează ecuațiile de stare, rezultatul final este formal același pentru orice convertor, ceea ce diferă de la un convertor la altul fiind expresiile matricilor  $A_i, B_i, E_i, F_i$  și eventual matricile extractoare.

Un alt aspect demn de luat în seamă este acela că deși Tymerski propune o modalitate de calcul a funcției de transfer control-ieșire pentru rețele de comutație, rezultatele sunt valabile numai pentru convertoare cu funcționare la frecvență constantă. Deși nemenționată explicit, această concluzie reiese din considerațiile făcute la deducerea expresiei de calcul (91) a lui  $H_0(j\omega)$ .

Așa cum s-a precizat, calculul exact al stării staționare este unul din elementele decisive în determinarea funcției de transfer control-ieșire exacte. Dar, într-un context mai general, starea staționară este importantă și pentru că numeroase specificații de proiectare se dau în termeni de caracteristici de stare staționară. Autorul tezei aduce câteva contribuții în acest sens în paragraful ce urmează.

### 6.8.2. Contribuții la calculul exact, accelerat și automat al stării staționare într-un convertor dc-dc

Una dintre primele încercări de calcul accelerat al stării staționare este cea raportată de Aprille și Trick [4] care utilizează algoritmul backward Euler și procedeul iterativ Newton-Raphson. Nakhla și Branin [135] propun apoi o metodă de optimizare pe care au denumit-o metoda gradientului. Ulterior au fost publicate numeroase studii [214], [72], [7], [64], [159], [166], [139], [47] privind calcul accelerat al stării staționare, majoritatea bazate pe formularea cu ecuații de stare și folosind algoritmul iterativ Newton-Raphson.

Se prezintă în continuare un procedeu de calcul exact și rapid al stării staționare bazat pe o metodă Newton-Raphson vectorială, procedeul propus de **autorul tezei** și prezentat în [83], apoi o serie de metode de calcul automat accelerat care leagă programe de calcul matematic (simulatoare de sistem) cu programe de simulare de uz general.

#### 6.8.2.1 Calculul stării staționare prin metoda Newton-Raphson vectorială

Cu notațiile de la paragraful 6.8.1 se definesc următoarele mărimi:

– duratele stărilor topologice:

$$T_i' = T_{i+1} - T_i \quad i = \overline{1, N_s} \quad (100)$$

– *argumentul vectorial extins*, ca fiind vectorul coloană de dimensiune  $2N_s + 1$  dat de:

$$X_e = \left[ X(T_1) \quad X(T_2) \quad \dots \quad X(T_{N_s+1}) \quad T_1' \quad T_2' \quad \dots \quad T_{N_s}' \right]^T \quad (101)$$

– funcția vectorială de argument  $X_e$ , având forma:

$$f(X_e) = \left[ f_1(X_e) \quad f_2(X_e) \quad \dots \quad f_{2N_s+1}(X_e) \right]^T \quad (102)$$

cu:

$$f_i(X_e) = e_{i+1} X_e - \left\{ e^{A_i(e_{i+N_s+1} X_e)} (e_i X_e) + A_i^{-1} \left[ e^{A_i(e_{i+N_s+1} X_e)} - I_n \right] B_i V_g \right\} \quad i = \overline{1, N_s} \quad (103)$$

$$f_{N_s+1}(X_e) = (e_{N_s+1} - e_1) X_e \quad (104)$$

$$f_{N_s+2}(X_e) = (e_{N_s+2} + e_{N_s+3} + \dots + e_{2N_s+1}) X_e - T_s \quad (105)$$

unde matricile  $e_i$ ,  $i = \overline{1, N_s + 1}$ , extrag vectorul  $X(T_i)$  din vectorul  $X_e$ , iar matricile  $e_{N_s+1+j}$ ,  $j = \overline{1, N_s}$  extrag pe  $T_j'$  din  $X_e$ . De aceea nu este greu de văzut că matricile  $e_i$ ,  $i = \overline{1, N_s + 1}$  se obțin din matricea unitate  $I_{n(N_s+1)+N_s}$  prin extracția a  $n$  linii (pentru că  $X(T_i)$  e un vector coloană de dimensiune  $n$ ), începând cu linia  $(i-1)n+1$  până la linia  $(i-1)n+n = in$ , iar matricile  $e_{N_s+1+j}$ ,  $j = \overline{1, N_s}$  rezultă tot din matricea unitate  $I_{n(N_s+1)+N_s}$  prin extracția unei singure linii (pentru că  $T_j'$  e un scalar) și anume linia  $n(N_s + 1) + j$ .

Egalitatea (103) reflectă dependența vectorului de stare de la sfârșitul stării topologice  $i$  (egal cu vectorul de stare de la începutul stării topologice  $i+1$ ) de vectorul de stare de la începutul stării topologice  $i$ . Acest lucru este posibil pentru că într-o stare topologică sistemul este LTI, dependența căutată exprimându-se clasic, cu funcția exponențială de matrice [63]. Relația (104) specifică faptul că în stare staționară vectorul de stare la sfârșitul perioadei este egal cu cel de la începutul perioadei, în timp ce (105) evidențiază că suma duratelor stărilor topologice dintr-o perioadă este egală cu perioada de comutație. Aceste relații sunt generale și ele apar obligatoriu în rezolvarea stării staționare a oricărui convertor.

Restul de  $N_s - 1$  componente din  $f$ , adică  $f_{N_s+3} \cdots f_{2N_s+1}$ , sunt specifice tipului de comandă și modului de funcționare al convertorului analizat. Ele se obțin punând în evidență diferite momente în care are loc o comutație specifică sau reflectă diferite intervale de timp prescrise prin comandă. Scrierea lor se face tot cu ajutorul matricilor  $e_i$ ,  $e_j$  sau cu alte matrici extractoare tot printr-o egalitate ce are pe 0 în membrul drept.

De exemplu pentru convertoare PWM CCM cu comandă în factor de umplere avem  $N_s = 2$  și deci pentru completarea vectorului  $f$  mai este necesară o componentă,  $f_5$ , aceasta punând în evidență faptul că durata  $T_1' = T_2 - T_1$  este fixată prin comandă, fiind egală cu  $DT_s$  și scriindu-se în forma:

$$f_5(X_e) = e_4 X_e - DT_s \quad (106)$$

Dacă în studiu este un convertor PWM DICM mai sunt necesare două componente,  $f_6$  și  $f_7$ . Prin  $f_6$  se pune în evidență că  $T_1'$  este fixat prin comanda în factor de umplere (și are aceeași formă ca în CCM), în timp ce prin  $f_7$  se marchează faptul că momentul  $T_3$  este dat de anularea curentului prin bobină. Dacă se admite convertorul cu o singură bobină atunci cele două componente se scriu analitic astfel:

$$f_6(X_e) = e_5 X_e - DT_s \quad (107)$$

$$f_7(X_e) = h(e_3 X_e) \quad (108)$$

cu  $h = [1 \ 0]$  extrăgând pe  $i_L(T_3)$  din vectorul de stare la momentul  $T_3$ , unde vectorul de stare este definit ca  $x = [i_L \ v_C]^T$ .

Revenind la scopul propus și anume rezolvarea exactă și accelerată a stării staționare, notațiile introduse arată că starea staționară se obține rezolvând ecuația vectorială:

$$f(X_e) = 0 \quad (109)$$

unde 0 din membrul drept trebuie înțeles ca matrice zero de ordin adecvat. Ecuația fiind transcendentă, pentru găsirea soluției se pot aplica diferite metode matematice de rezolvare. **Autorul** a propus metoda Newton-Raphson în care se știe că trebuie cunoscută și derivata funcției  $f$ . Acesta se obține prin derivarea matricială formală a componentelor sale (vezi (103), (104) și (105)), obținându-se:

$$f'(X_e) = [f_1'(X_e) \ f_2'(X_e) \ \cdots \ f_{2N_s+1}'(X_e)]^T \quad (110)$$

cu:

$$f_i'(X_e) = e_{i+1} - \left[ A_i e^{A_i(e_{i+N_s+1} X_e)} (e_i X_e) + e^{A_i(e_{i+N_s+1} X_e)} e_i + e^{A_i(e_{i+N_s+1} X_e)} B_i V_g e_{i+N_s+1} \right] \quad (111)$$

$i = \overline{1, N_s}$

$$f_{N_s+1}'(X_e) = e_{N_s+1} - e_1 \quad (112)$$

$$f_{N_s+2}'(X_e) = e_{N_s+2} + e_{N_s+3} + \cdots + e_{2N_s+1} \quad (113)$$

Așa cum s-a mai menționat, componentele  $f_{N_s+1}' \cdots f_{2N_s+1}'$  depind de tehnica de comandă și modul de funcționare al convertorului, având însă în general expresii simple. Reluând exemplul convertorului PWM CCM (relația (81)) se obține:

$$f_5'(X_e) = e_4 \quad (114)$$

Având acum definite toate funcțiile matriciale necesare rezolvării numerice prin metoda Newton-Raphson, algoritmul este cel cunoscut: se pleacă de la o soluție inițială,  $X_{e_0}$ , algoritmul iterativ furnizând soluția de ordinul  $k$ ,  $X_{e_k}$ , din soluția precedentă,  $X_{e_{k-1}}$  conform relației:

$$X_{e_k} = X_{e_{k-1}} - [f'(X_{e_{k-1}})]^{-1} \cdot f(X_{e_{k-1}}) \quad (115)$$



unde ridicarea la puterea -1 semnifică inversarea matricii din paranteze. Procesul iterativ se oprește și soluția se ia egală cu soluția din ultima iterație atunci când fiecare componentă individuală din vectorul diferență  $X_{e_k} - X_{e_{k-1}}$ , luată în modul este mai mică decât eroarea admisă inițial. Definind vectorul eroare de dimensiune  $2N_s + 1$  ca:

$$\varepsilon = \left[ \varepsilon_1 \quad \varepsilon_2 \quad \dots \quad \varepsilon_{2N_s+1} \right] \quad \varepsilon_i > 0, \quad i = \overline{1, 2N_s + 1} \quad (116)$$

analitic acest lucru se scrie în forma:

$$\left| X_{e_k} - X_{e_{k-1}} \right|_i < \varepsilon_i \quad (117)$$

Două aspecte se impun a fi precizate în final. Primul rezultă din examinarea relațiilor (103) unde apare matricea  $A_i^{-1}$ , deci se impune ca toate matricile  $A_i \quad i = \overline{1, N_s}$  să fie nesingulare. Acest lucru se realizează în general dacă întrerupătoarele schemelor se modelează prin rezistențe nenule în conducție și neînfinită în blocare. Al doilea aspect este specific metodei Newton-Raphson și se referă la convergența sau divergența algoritmului, pentru care se știe că decisivă este alegerea soluției inițiale, notată aici cu  $X_{e_0}$ . Încercările autorului de a rezolva starea staționară a diferitor convertoare plecând de la diverse condiții inițiale a dus la concluzia că cel mai bun punct de pornire în algoritmul Newton-Raphson este soluția furnizată de modelul mediat, notată  $\bar{X}_{e_0}$  și dată de:

$$\bar{X}_{e_0} = \left[ \bar{X} \quad \bar{X} \quad \bar{X} \quad \bar{T}'_1 \quad \bar{T}'_2 \quad \dots \quad \bar{T}'_{N_s} \right]^T \quad (118)$$

unde  $\bar{X}$  este soluția pe care o dă modelul mediat pentru vectorul de stare (ea se repetă în  $\bar{X}_{e_0}$  de  $N_s + 1$  ori), iar  $\bar{T}'_i, \quad i = \overline{1, N_s}$  sunt soluțiile modelului mediat pentru duratele stărilor topologice. Fără a garanta convergența algoritmului, toate încercările autorului de a obține soluția exactă a stării staționare pentru diverse convertoare PWM plecând de la această soluție inițială au avut succes (verificarea corectitudinii soluției făcându-se prin simulare), în condițiile unui număr foarte redus de iterații.

Pentru exemplificarea metodei se prezintă aplicația publicată de autor în [86], la care se dorește rezolvarea exactă a stării staționare pentru convertorul SEPIC DIVM din Fig. 6.39, cu următorii parametri:

$$V_g = 12 \text{ V}; \quad L_1 = 500 \text{ } \mu\text{H}; \quad C_1 = 220 \text{ nF}; \quad L_2 = 500 \text{ } \mu\text{H}; \quad C = 100 \text{ } \mu\text{F}; \quad R = 10 \text{ } \Omega;$$

$$f_s = 50 \text{ kHz}; \quad D = 0,6$$

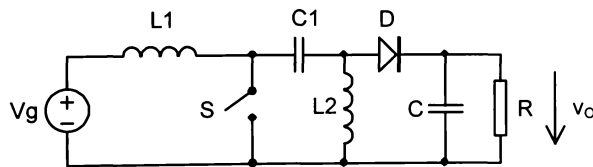


Fig. 6.39. Convertorul SEPIC la care se calculează exact starea staționară.

Pentru a obține matrici nesingulare este suficient să se aleagă rezistența în conducție a întrerupătorului activ nenulă. S-a considerat  $R_{Son} = 0,1 \text{ } \Omega$ . Vectorul de stare a fost definit ca:

$$x = \begin{bmatrix} i_{L_1} \\ i_{L_2} \\ v_C \\ v_{C_1} \end{bmatrix} \quad (119)$$

Convertorul are  $N_s = 3$  stări topologice, cele 12 matrici corespunzătoare fiind:

$$A_1 = \begin{bmatrix} -\frac{R_{Son}}{L_1} & -\frac{R_{Son}}{L_1} & 0 & 0 \\ -\frac{R_{Son}}{L_2} & -\frac{R_{Son}}{L_2} & 0 & \frac{1}{L_2} \\ 0 & 0 & -\frac{1}{RC_2} & 0 \\ 0 & -\frac{1}{C_1} & 0 & 0 \end{bmatrix}; \quad B_1 = \begin{bmatrix} \frac{1}{L_1} \\ 0 \\ 0 \\ 0 \end{bmatrix}; \quad E_1 = [0 \ 0 \ 1 \ 0]; \quad F_1 = [0]; \quad (120)$$

$$A_2 = \begin{bmatrix} 0 & 0 & -\frac{1}{L_1} & -\frac{1}{L_1} \\ 0 & 0 & -\frac{1}{L_2} & 0 \\ \frac{1}{C_2} & \frac{1}{C_2} & -\frac{1}{C_2} \left( \frac{1}{R} + \frac{1}{R_{Son}} \right) & -\frac{1}{C_2 R_{Son}} \\ \frac{1}{C_1} & 0 & -\frac{1}{C_1 R_{Son}} & -\frac{1}{C_2 R_{Son}} \end{bmatrix}; \quad B_2 = \begin{bmatrix} \frac{1}{L_1} \\ 0 \\ 0 \\ 0 \end{bmatrix}; \quad E_2 = [0 \ 0 \ 1 \ 0]; \quad F_2 = [0]; \quad (121)$$

$$A_3 = \begin{bmatrix} 0 & 0 & -\frac{1}{L_1} & -\frac{1}{L_1} \\ 0 & 0 & -\frac{1}{L_2} & 0 \\ \frac{1}{C_2} & \frac{1}{C_2} & -\frac{1}{RC_2} & 0 \\ \frac{1}{C_1} & 0 & 0 & 0 \end{bmatrix}; \quad B_3 = \begin{bmatrix} \frac{1}{L_1} \\ 0 \\ 0 \\ 0 \end{bmatrix}; \quad E_3 = [0 \ 0 \ 1 \ 0]; \quad F_3 = [0]; \quad (122)$$

Pentru reducerea expunerii se vor preciza numai componentele specifice din funcția  $f$ . În DICM sfârșitul primei stări topologice este marcat de anularea tensiunii  $v_{off}$ . Pentru că într-un convertor SEPIC DIVM tensiunea  $v_{off}$  este egală cu suma tensiunilor pe capacități componenta  $f_6$  din funcția  $f$  va fi:

$$f_6(X_e) = h(e_2 X_e) \quad (123)$$

unde matricea extractoare  $h$  a lui  $v_{off}$  (a tensiunii pe diodă) din vectorul de stare este dată de:

$$h = [-R_{Son} \ -R_{Son} \ 1 \ 1] \quad (124)$$

În  $h$  apare și  $R_{Son}$  pentru că o rezistență nenulă în conducție a lui  $S$  face ca în prima stare topologică tensiunea  $v_{off}$  să depindă și de căderea de tensiune produsă de curenții prin bobine pe această rezistență.

A doua componentă specifică din  $f$  este  $f_7$ , ea precizând durata intervalului de conducție a întrerupătorului activ (egală cu  $T_1 + T_2$ ) fixată prin comanda externă la  $DT_s$ :

$$f_7(X_e) = (e_5 + e_6) X_e - DT_s \quad (125)$$

Soluția inițială din algoritmul Newton-Raphson s-a ales ca soluția modelului mediat propus de Maksimović și Čuk în [115]. Componentele vectorului de start  $X_{e_0}$  au rezultat astfel:

– vectorul de stare:

$$\bar{X} = \begin{bmatrix} 1,646 \text{ A} \\ 1,406 \text{ A} \\ 14,056 \text{ V} \\ 12,000 \text{ V} \end{bmatrix}$$

– duratele stărilor topologice:

$$\bar{T}_1' = 9,370 \mu\text{s}; \quad \bar{T}_2' = 2,630 \mu\text{s}; \quad \bar{T}_3' = 8,000 \mu\text{s}$$

Aplicând metoda Newton-Raphson propusă s-au obținut următoarele valori pentru starea staționară exactă:

– valorile exacte ale vectorului de stare în momentele de comutație:

$$X(T_1) = X(T_4) = \begin{bmatrix} 1,351 \text{ A} \\ 1,151 \text{ A} \\ 13,435 \text{ V} \\ 43,908 \text{ V} \end{bmatrix}; \quad X(T_2) = \begin{bmatrix} 1,563 \text{ A} \\ 1,444 \text{ A} \\ 13,314 \text{ V} \\ -13,014 \text{ V} \end{bmatrix}; \quad X(T_3) = \begin{bmatrix} 1,633 \text{ A} \\ 1,365 \text{ A} \\ 13,316 \text{ V} \\ -13,153 \text{ V} \end{bmatrix};$$

– duratele exacte ale stărilor topologice:

$$T_1' = 9,046 \mu\text{s}; \quad T_2' = 2,954 \mu\text{s}; \quad T_3' = 8,000 \mu\text{s}$$

Vectorul eroare ales a fost:

$$\epsilon = 10^{-9} \cdot [1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 1 \ 10^{-3} \ 10^{-3} \ 10^{-3}]$$

Programul de calcul este dat în Anexa A25. Deși formulele sunt destul de lungi, algoritmul propus se implementează extrem de simplu în MATLAB, program care poate calcula funcția exponențială de matrice cerută de algoritm. Rezultatele obținute pentru calculul exact prin metoda propusă au fost comparate cu cele rezultate prin simulare cu programul PETS (Boulder University Colorado) și coincidența s-a dovedit a fi excelentă. În plus convergența și deci rapiditatea cu care se rezolvă exact starea staționară, este foarte bună, calculul necesitând numai 6 iterații.

### 6.8.2.2 Calculul și analiza accelerată a stării staționare prin legarea unui program de calcul matematic și un simulator de circuite de uz general

Așa cum se amintea anterior, o soluție în găsirea stării staționare este simularea circuitului în studiu cu un simulator dedicat de circuite, până ce regimul tranzitoriu dispare și starea staționară este atinsă. Vom numi această metodă primitivă *simulare brută*. Evident această abordare prezintă dezavantaje mari dacă circuitul simulat are constante de timp mult mai mari decât perioada de comutație, adică este un *sistem rigid* ("stiff system"), deoarece vor fi necesari timpi de simulare excesiv de mari. Convertoarele dc-dc intră tocmai în această categorie de sisteme din cauza filtraajului de tip trece jos inerent, necesar pentru atenuarea puternică a pulsațiilor cel puțin la frecvența de comutație (dacă nu cumva este necesară și atenuarea unor armonici ale tensiunii rețelei prezente în sursa de alimentare).

S-au propus o serie de metode pentru o convergență accelerată spre starea staționară [214], [72], dar implementarea lor a fost realizată pe simulatoare dedicate. Simulatoarele de uz general, în care standardul este PSpice (cu diverse variante și derivate ale sale), au avantajul că suportă modele de componente neliniare și deosebit de complexe, dar aplicarea lor în electronica de putere este de multe ori restricționată de problemele de convergență, timpul de simulare lung și lipsa unui accelerator de stare staționară.

Acest paragraf prezintă două metode numerice de calcul computerizat a stării staționare, inspirate din [119], folosind două programe de uz general: MATLAB ca program de calcul matematic pentru implementarea metodelor numerice și PETS sau PSpice pentru a obține răspunsul circuitului. Se va prezenta numai varianta ce utilizează programul PETS, cea care apelează PSpice fiind identică și prezentând numai aspecte formale minore diferite. Totuși câteva considerații pe marginea folosirii programului PSpice se vor face în final de paragraf.

Se consideră un convertor în comutație cu  $n$  variabile de stare descris prin ecuația de stare:

$$\frac{dx}{dt} = G(t, x), \quad x(t_0) = x_0, \quad t \geq t_0 \quad (126)$$

Integrând (126) de la  $t_0$  la  $t_0 + \tau$  se obține soluția la momentul  $t_0 + \tau$ :

$$x(t_0 + \tau) = F_\tau(x_0, t_0) \quad (127)$$

unde  $F_\tau$  reprezintă o mapare neliniară obținută prin integrarea ecuației (126) pe durata  $\tau$ .

În stare staționară (dacă aceasta există)  $x(t)$  este periodic de perioadă  $T_s$  și din (127) rezultă:

$$x(t_0 + T_s) = F_{T_s}(x_0, t_0) = x_0 \quad (128)$$

Periodicitatea temporală a soluției stării staționare permite să facem  $t_0 = 0$  și deci dependența de  $t_0$  în (128) să nu mai fie necesară. Deci rezolvarea stării staționare se reduce la a găsi vectorul de stare  $x$  care este soluția sistemului de  $n$  ecuații neliniare:

$$f(x) = x - F_{T_s}(x) = 0 \quad (129)$$

Metoda propusă obține pe  $F_{T_s}(x)$  numeric, folosind un simulator de circuite, adică efectuând o simulare de la  $t=0$  la  $t=T_s$  cu condițiile inițiale  $x$ . Scrierea condițiilor inițiale  $x$  în fișierul *file.cir* este efectuată de programul de calcul matematic, tot din acest program lansându-se în execuție simulatorul. Simulatorul este setat ca să scrie formele de undă în format ASCII într-un fișier de tip *file.txt* sau echivalent. Programul de calcul matematic citește fișierul *file.txt*, având astfel disponibil pe  $F_{T_s}(x)$ . La nivelul programului de calcul matematic se rezolvă numeric ecuația (129), rezolvare care va cere lansarea de simulări suplimentare pentru calculul numeric al Jacobianului, așa cum va reieși din cele ce urmează. O schemă bloc ce reflectă interacțiunile dintre cele două programe este prezentată în Fig. 6.40.

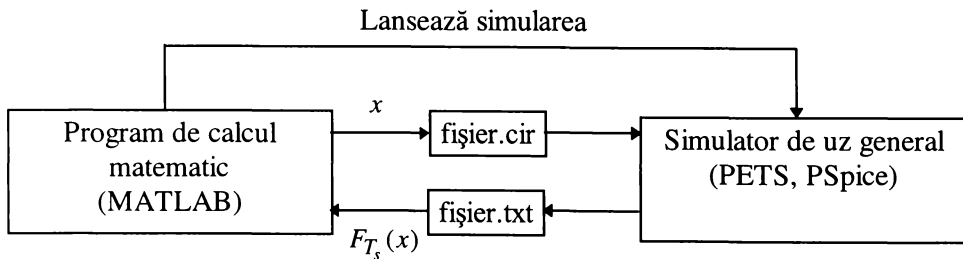


Fig. 6.40. Interacțiunile dintre cele două programe utilizate la obținerea și analiza accelerată a stării staționare.

#### a) Particularități în obținerea lui $F_{T_s}(x)$

Aceste particularități sunt date de faptul că există două clase de convertoare în comutație:

- convertoare *neautonome*, în care perioada de comutație  $T_s$  este determinată de un clock extern.

– convertoare *autonome*, în care perioada de comutație  $T_s$  depinde de parametrii de circuit și condițiile de funcționare.

Pentru sistemele neautonome,  $T_s$  fiind cunoscută,  $F_{T_s}(x)$  se obține simplu, făcând timpul final de simulare în linia *probe* egal cu  $T_s$  sau, dacă suntem interesați de o soluție subarmonică, egal cu  $mT_s$   $m \in N$ . Un exemplu de o astfel de aplicație poate fi găsit în [145].

Pentru sistemele autonome obținerea lui  $F_{T_s}(x)$  este mai complicată pentru că  $T_s$  nu este cunoscută, iar simulatorul cere un timp final de simulare constant, independent de rezultatele simulării. Rezolvarea acestei situații este inspirată din [64], unde se arată că într-un sistem autonom  $T_s$  este determinată de o condiție de zero de forma:

$$C(x(T_s^-)) > 0, \quad C(x(T_s)) = 0, \quad C(x(T_s^+)) > 0 \quad (130)$$

Deoarece condiția de zero corespunde unui moment de comutație, în cele mai multe cazuri condiția se poate evalua prin monitorizarea unui semnal de comandă al unui întrerupător. Pentru a obține pe  $F_{T_s}(x)$  simulatorul este pus să ruleze pe un interval constant, suficient de mare,  $T_x$ . După terminarea simulării și citirea datelor se obține  $T_s$  ca primul moment în care (130) este satisfăcută. Dacă nu s-a găsit nici o condiție de zero atunci simularea este repetată având ca și condiții inițiale pe  $x(T_x)$ .

*b) Rezolvarea numerică a ecuației  $f(x) = x - F_{T_s}(x) = 0$  la nivelul programului de calcul matematic*

În rezolvarea numerică a sistemului (129) se notează cu  $N_{sim}$  numărul total de simulări (evaluări ale lui  $F_{T_s}(x)$ ) și cu  $N_i$  numărul total de iterații indexate prin indicele  $k$ , iar  $x_k$  este vectorul de stare în a  $k$ -a iterație.

*b1) Iterația cu punct fix*

Este cea mai simplă metodă, în care iterația este dată de relația

$$x_{k+1} = F_{T_s}(x_k) \quad (131)$$

În contextul utilizării celor două programe metoda este echivalentă cu simularea brută și a fost utilizată ca și test în verificarea corectitudinii dialogului între programe, a obținerii lui  $F_{T_s}(x)$  și ca o referință cu care să se compare alte metode. Evident, pentru iterația cu punct fix  $N_{sim} = N_i$ .

*b2) Metode de tip Newton*

Este cunoscut faptul că există o mare varietate de metode de tip Newton pentru a rezolva numeric ecuația  $f(x) = 0$ . Ideea de rezolvare comună tuturor acestora este de a aproxima pe  $f(x)$  în a  $k$ -a iterație cu un model afin,  $f^*(x)$ , în forma:

$$f(x) \cong f^*(x) = J_k(x - x_k) + f(x_k) \quad (132)$$

și de a obține pe  $x_{k+1}$  din ecuația  $f^*(x_{k+1}) = 0$ , deci:

$$x_{k+1} = x_k - J_k^{-1} \cdot f(x_k) \quad (133)$$

Diversele metode de tip Newton diferă după modul în care este aleasă matricea  $J_k$ .

Având în vedere că în cazul abordării expuse  $f(x)$  este dată de relația (129), relația (133) devine:

$$x_{k+1} = x_k - J_k^{-1}[x_k - F_{T_s}(x_k)] \quad (134)$$

Se propune pentru rezolvarea numerică a ecuației  $f(x) = 0$  metoda Newton-Raphson la care se știe că matricea  $J_k$  este matricea Jacobian,  $J$ , evaluată în punctul  $x_k$ :

$$J_k = J(x_k) = \frac{df}{dx}(x_k) \quad (135)$$

ceea ce particularizat pentru abordarea prezentată înseamnă:

$$J_k = J(x_k) = I - \frac{dF_{T_s}}{dx}(x_k) \quad (136)$$

$I$  fiind matricea unitate de ordinul  $n$ . Examinând relația (136) rezultă că dificultatea apare la calculul derivatei lui  $F_{T_s}(x)$ , deoarece  $F_{T_s}(x)$  se obține numeric prin simulare. Este de așteptat deci ca și derivata (și deci Jacobianul) în  $x_k$  să se obțină tot numeric și apelând tot la simulare. În literatură au fost raportate diverse metode de a evalua Jacobianul pentru circuite de putere în comutație [214], [72], [64], [24] fie din modelul în spațiul stărilor, fie folosind abordări specifice teoriei circuitelor. Toate însă se bazează pe detalii specifice din descrierea circuitului analizat și de aceea reclamă folosirea unor simulatoare speciale.

Pentru a putea aplica un simulator de uz general sunt deci necesare metode numerice generale de aproximare a Jacobianului, care să nu facă apel la detalii de circuit și care să se aplice identic tuturor circuitelor. Se prezintă mai jos două astfel de metode: metoda diferențelor finite și metoda cu actualizare Broyden, în care relațiile se vor scrie direct pentru forma specifică (129) a funcției  $f(x)$ .

#### Metoda de tip Newton cu diferențe finite

În cazul acestei metode Jacobianul în  $x_k$ ,  $J(x_k)$ , se calculează cu formula:

$$j_i = u_i - \frac{F_{T_s}(x_k + u_i \Delta x_i) - F_{T_s}(x_k)}{\Delta x_i} \quad i = \overline{1, n} \quad (137)$$

unde  $j_i$  este cea de a  $i$ -a coloană a lui  $J(x_k)$ ,  $u_i$  este a  $i$ -a coloană a matricii unitate  $I$ , iar incrementul  $\Delta x_i$  este ales conform [33, pag. 94-99]:

$$\Delta x_i = \sqrt{\eta} \cdot \max(|x_i(t)|, x_m) \quad 0 \leq t \leq T_s \quad (138)$$

cu  $x_i$  cea de a  $i$ -a variabilă scalară din vectorul  $x$ ,  $\eta$  un estimat al erorii relative în calculul lui  $F_{T_s}(x)$  (tipic în gama  $10^{-2}$ - $10^{-5}$  pentru un pas de simulare corect corelat cu perioada de comutație și fenomenele de înaltă frecvență) și  $x_m$  o valoare limită inferioară pentru  $\Delta x_i$ .

#### Metoda de tip Newton cu actualizare Broyden

Este o metodă care, deși necesită mai multe iterații (deci are o convergență mai slabă) față de metoda diferențelor finite, totuși reduce numărul de simulări. Mai mult, se poate arăta că suficient de aproape de soluție, metoda de tip Newton cu actualizare Broyden are o rată de convergență superliniară, apropiată de rata de convergență pătratică a metodei Newton-Raphson [33, pag. 174-186]. Reducerea numărului de simulări se obține calculând Jacobianul prin diferențe finite numai pentru prima iterație, în următoarele iterații Jacobianul fiind actualizat prin efectuarea a numai unei simulări per iterație. Pentru actualizarea Broyden de rangul 1, conform [33, pag. 169-189], formula de actualizare este:

$$J(x_k) = J(x_{k-1}) + \frac{[x_k - F_{T_s}(x_k)](x_k - x_{k-1})^T}{\|x_k - x_{k-1}\|^2} \quad (139)$$

unde  $\| \cdot \|$  semnifică norma euclidiană iar  $^T$  operația de transpunere a matricilor.

### b3) Criteriul de convergență

Pentru toate metodele utilizate criteriul de convergență este dat de condiția ca eroarea relativă maximă  $e$  să fie mai mică decât o limită prescrisă,  $e_0$ , unde  $e_0$  se alege de regulă în gama  $10^{-2}$ - $10^{-5}$ . Relațiile ce definesc pe  $e$  sunt de tipul erorii relative, cu o ușoară corecție pentru a putea estima corect și erori pentru mărimi apropiate de zero:

$$e = \max(e_i), \quad i = \overline{1, n} \quad (140)$$

cu

$$e_i = \frac{|f_i(x)|}{\max(x_i, x_{\min})}, \quad i = \overline{1, n} \quad (141)$$

în care  $f_i(x) = x_i - F_{T_s}(x_i)$  (a  $i$ -a componentă din  $f(x)$ ), iar  $x_{\min}$  o constantă de valoare mică ce suprimă erorile mari asociate variabilelor de stare apropiate de zero.

Din cele expuse până acum rezultă schema bloc din Fig. 6.41 a algoritmului de calcul accelerat a stării staționare.

Se observă că dacă procesul iterativ nu converge după un număr  $N_{sim\max}$  de simulări utilizatorul poate reîncepe calculul accelerat, după ce în prealabil a modificat punctul de pornire  $x_0$  sau unul sau mai mulți dintre parametrii  $e_0, \eta, x_m$ , sau  $x_{\min}$ , care bineînțeles afectează convergența algoritmului. De reținut că eroarea  $e$  este puternic influențată în primul rând de  $\eta$ .

Din schema bloc din Fig. 6.41 se poate deduce că pentru calculul stării staționare în convertoare neautonome, în cazul utilizării metodei de tip Newton cu diferențe finite sunt necesare  $N_{sim} = (n+1)N_i + 1$  simulări, în timp ce dacă se utilizează metoda de tip Newton cu actualizare Broyden numărul de simulări este  $N_{sim} = N_i + n + 1$ . Termenul 1 în formulele de mai sus este dat de simularea necesară pentru evaluarea lui  $F_{T_s}(x_0)$ .

Odată criteriul de convergență satisfăcut și deci starea staționară rezolvată, sunt disponibile în *fișier.txt* și datele ultimei simulări, în care s-a evaluat  $F_{T_s}(x_{k+1})$ , de fapt formele de undă ale mărimilor cerute în linia *.probe*. Pe baza acestora se pot face estimări și extrage unele caracteristici ale stării staționare: diverse caracteristici statice, solicitări de componente, forme de undă parametrizate după o anumită mărime pentru evidențierea de exemplu a diferitelor moduri de funcționare, etc.

Fișierele MATLAB pentru calculul stării staționare sunt următoarele: *fixpoint.m* pentru calculul prin metoda iterației cu punct fix, *finidif.m* pentru calculul stării staționare prin metoda de tip Newton cu diferențe finite și *broyden.m* pentru calculul prin metoda de tip Newton cu actualizare Broyden. Toate aceste fișiere apelează un fișier suplimentar numit *stat.m* necesar pentru corelarea ordinii variabilelor scrise în linia *.probe* cu ordinea în care acestea sunt scrise de simulatorul PETS în fișierul *fișier.txt*. Aceasta pentru că ordinea din linia *.probe* nu este respectată de programul PETS în modul de scriere în *fișier.txt*.

În cazul utilizării programului Pspice apare o problemă suplimentară legată de modelul circuitului și vectorul de stare,  $x$ , utilizat de programul de calcul matematic. Modelele dispozitivelor din Pspice introduc elemente reactive adiționale, astfel încât vectorul de stare real al sistemului,  $x^*$ , este de dimensiune mult mai mare decât  $x$ . În unele cazuri, ca de exemplu macromodelele subcircuitelor pentru amplificatoare operaționale și comparatoare, se pot specifica

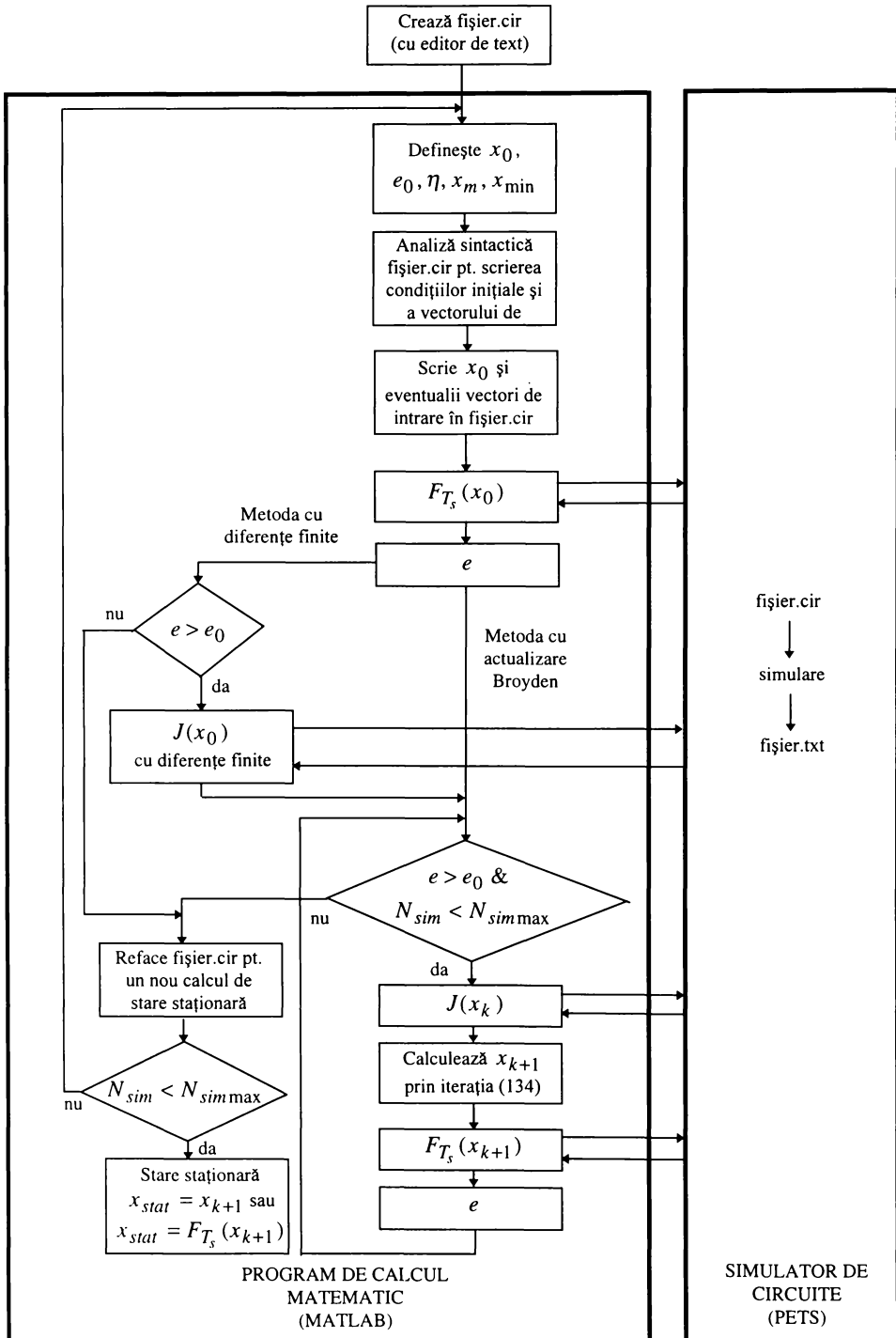


Fig. 6.41. Implementarea celor două metode de tip Newton pentru calculul stării staționare.



condițiile inițiale ale variabilelor de stare adiționale, deși programul de calcul matematic va trebui să acceseze și să modifice bibliotecile unde aceste modele sunt definite. În schimb la variabilele de stare adiționale ale modelelor de dispozitive construite intrinsec în programul Pspice nu există acces. Definim vectorul de stare real  $x^* = [x, x_p]^T$  în care  $x_p$  este vectorul de stare ce cuprinde variabilele de stare “parazite”, în sensul că sunt neglijate de programul de calcul matematic.

Atunci soluția de stare staționară obținută de programul de calcul matematic este  $x_{sm}^* = [x_{sm} \ 0]^T$ , diferită de soluția reală  $x_s^* = [x_s \ x_{sp}]^T$ , în care în general  $x_{sp} \neq 0$ . La o primă evaluare calitativă s-ar părea că erorile introduse neglijând pe  $x_p$  sunt relativ mici deoarece constantele de timp asociate variabilelor de stare “parazite” sunt mici în comparație cu perioada  $T_s$ . Această afirmație nu este însă totdeauna consistentă, putând uneori apare erori destul de mari pentru anumite configurații de circuite.

*Aplicații*

Toate exemplele de mai jos au fost efectuate pe un calculator 486 DX/40MHz, 16 Mb RAM rulând sub Windows 95 programele MATLAB 4.2 și PETS.

*Aplicația 1. Calculul stării staționare pentru un stabilizator în comutație.*

Sursa în comutație în buclă închisă analizată este exemplul tipic propus de Wong în [214], având schema reprezentată în Fig. 6.42 (au fost păstrate notațiile autorului).

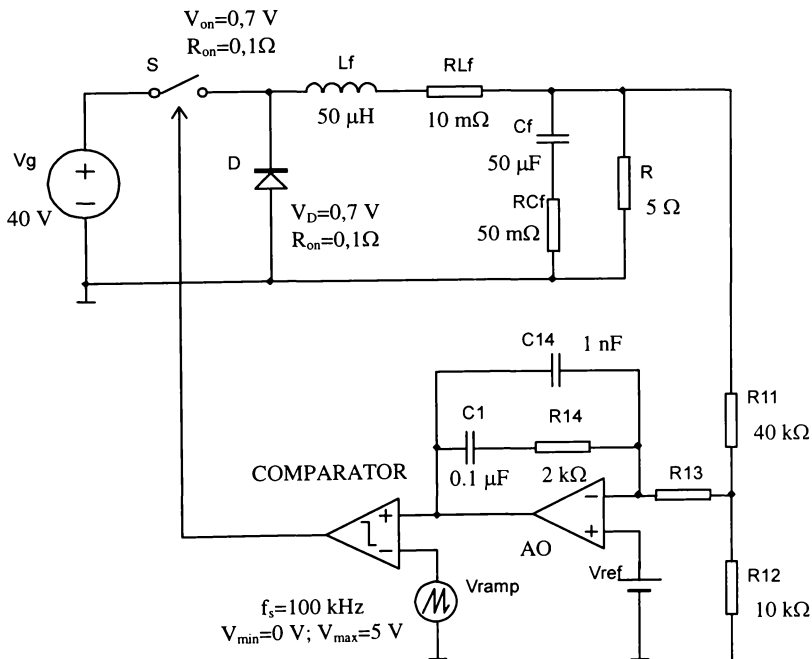


Fig. 6.42. Stabilizatorul în comutație de tip BUCK (după Wong) - obiect de studiu în aplicarea metodelor de calcul accelerat al stării staționare.

Parametrii tuturor algoritmilor de calcul accelerat au fost:

$$x_0 = 0_4; \quad e_0 = 10^{-4}; \quad \eta = 10^{-4}; \quad x_m = 10^{-3}; \quad x_{\max} = 10^{-3}$$

unde  $x = [i_{L_f} \ v_{C_f} \ v_{C_1} \ v_{C_{14}}]^T$ .

Pentru a avea un reper în comportarea reală a circuitului, în Fig. 6.43 s-a simulat regimul tranzitoriu de pornire cu condiții inițiale nule, deci exact din punctul din care începe și aplicarea algoritmilor de calcul accelerat, până ce criteriul de convergență este satisfăcut. Se observă timpul de răspuns lung și caracterul inițial ușor oscilant al răspunsului.

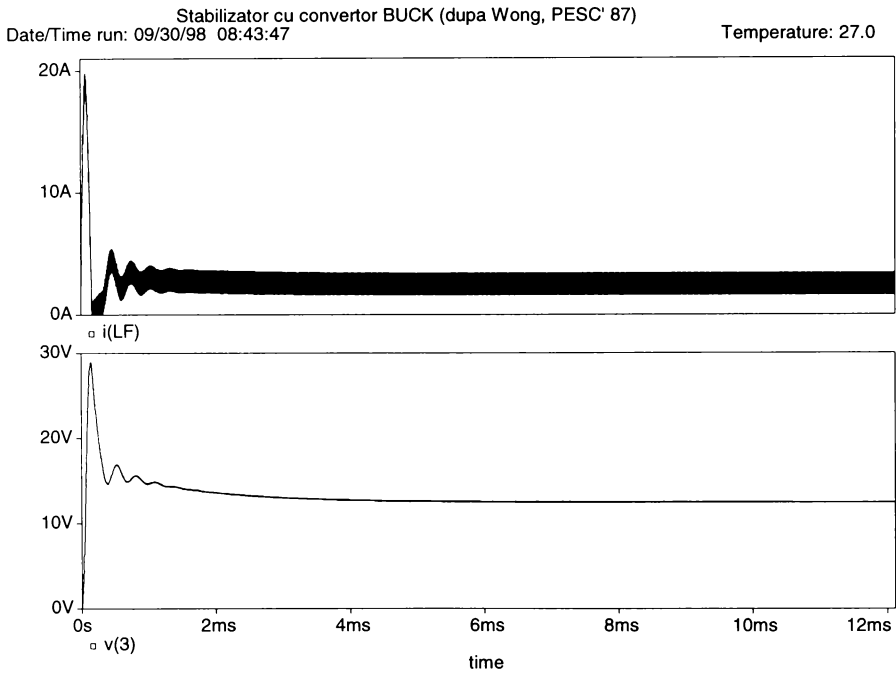


Fig. 6.43. Regimul tranzitoriu de pornire: curentul prin bobină (sus) și tensiunea de ieșire (jos).

Apoi, în urma aplicării algoritmilor de calcul accelerat, s-au obținut rezultatele din Tab. 6.3. Soluția stării staționare caracterizează evident momentul de început al perioadei, fiind:

$$x_s = [1,6076 \text{ A} \quad 12,4901 \text{ V} \quad 0,8224 \text{ V} \quad 0,3 \text{ mV}]^T$$

Tab. 6.3. Rezultatele calculului accelerat al stării staționare pentru convertorul din Fig. 6.42, prin cele 3 metode prezentate.

METODA	$N_i$	$N_{sim}$
Iterație cu punct fix	1212	1212
Newton cu diferențe finite	3	16
Newton cu actualizare Broyden	4	9

Având la dispoziție rezultatele ultimei simulări, din algoritmi de calcul accelerat se pot reprezenta rapid și principalele mărimi în stare staționară, care sunt cele din Fig. 6.44.

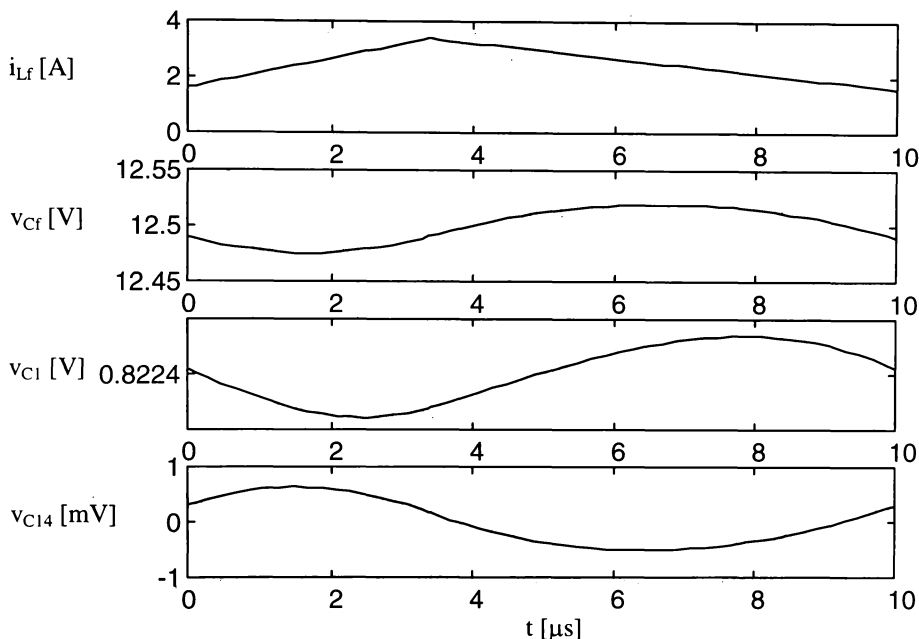


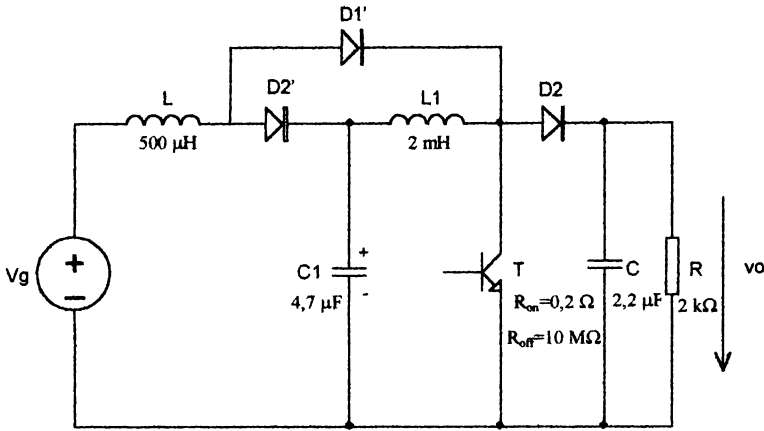
Fig. 6.44. Principalele forme de undă în stare staționară obținute în urma aplicării algoritmilor de calcul accelerat.

Examinând Tab. 6.3 se remarcă în primul rând că aplicând metodele de tip Newton numărul necesar de simulări se reduce cu 2 ordine de mărime comparativ cu iterația cu punct fix (simularea brută). Făcând o comparație între cele două metode de tip Newton se observă că, așa cum era de așteptat, deși din punct de vedere al numărului de iterații metoda cu actualizare Broyden este mai puțin performantă, în schimb din punct de vedere al numărului de simulări ea este net superioară celei cu diferențe finite. Fișierele MATLAB corespunzătoare celor trei algoritmi sunt *fixpoint.m*, *finitdif.m* și *broyden.m*, iar fișierul PETS este *buckw.cir*, toate fiind expuse în Anexa A26.

#### *Aplicația 2. Determinarea caracteristicilor statice reale și a familiilor de forme de undă în stare staționară pentru convertoare dc-dc*

Scopul este de a exemplifica modul în care metodele de calcul accelerat a stării staționare pot fi utilizate în determinarea caracteristicilor statice reale. Pentru studiu s-a ales convertorul  $\delta_1$ -BOOST în buclă deschisă din Fig. 6.45, la care întrerupătoarele nu se mai consideră ideale ci se modelează prin rezistențe de conducție și blocare asociate cu tensiuni în stare de conducție, conform figurii.

Suntem interesați de dependența tensiunii medii de ieșire,  $V_o$ , și a tensiunii pe condensatorul  $C_1$ ,  $V_{C_1}$ , de factorul de umplere  $D$ . Se utilizează metoda cu actualizare Broyden, fișierele fiind *d\_car.m* și *boost2.cir* - Anexa A27. Având disponibile datele ultimei simulări, care



$D'_1, D_2, D'_2: R_{on}=0,2 \Omega, R_{off}=10 \text{ M}\Omega, V_D=0,4 \text{ V}$

Fig. 6.45. Conversorul pătratic  $\delta_1$ -BOOST analizat cu metode de calcul accelerat a stării staționare din punct de vedere al caracteristicilor statice.

este chiar starea staționară, media se calculează cu ușurință. Fig. 6.46 prezintă dependențele dorite, cu  $D$  variat cu pas de 0,05 în domeniul  $0,05 \leq D \leq 0,8$ . Pentru comparație, caracteristicile statice reale sunt prezentate împreună cu cele ideale în CCM, adică  $V_o(D) = \frac{1}{(1-D)^2}$  și

$$V_{C_1}(D) = \frac{1}{(1-D)}$$

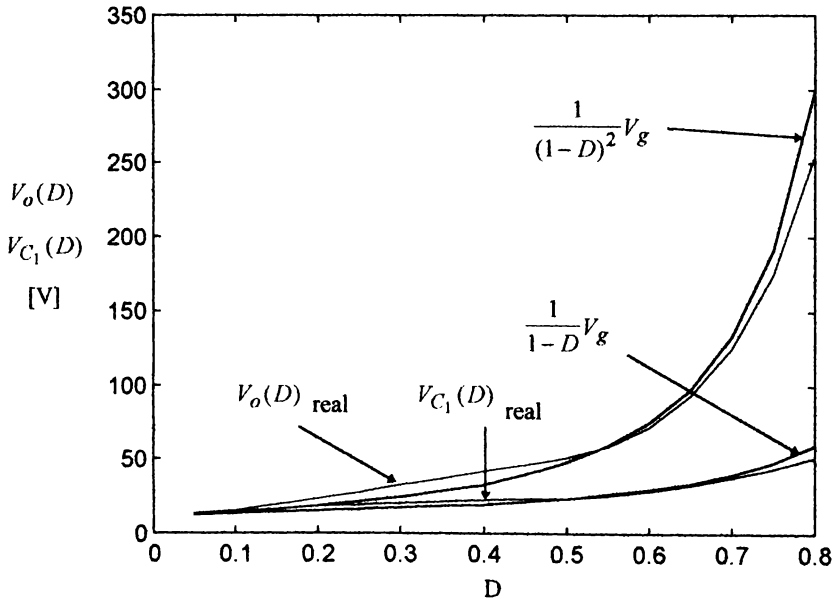


Fig. 6.46. Caracteristicile statice  $V_o = V_o(D)$  (roșu) și  $V_{C_1} = V_{C_1}(D)$  (verde) ale convertorului din Fig. 6.45, obținute folosind metoda de calcul accelerat cu actualizare Broyden.

Faptul că pentru factori de umplere între 0,2 și 0,5 se obțin tensiuni reale mai mari decât cele prognozate de caracteristica ideală CCM se explică prin intrarea convertorului în DICM (vezi Tab. 4.2).

În Fig. 6.47 au fost reprezentate cureții prin cele două bobine în stare staționară, pentru același set de valori ale factorului de umplere, folosind perechea de fișiere *st\_waves.m* și *boost2.cir*.

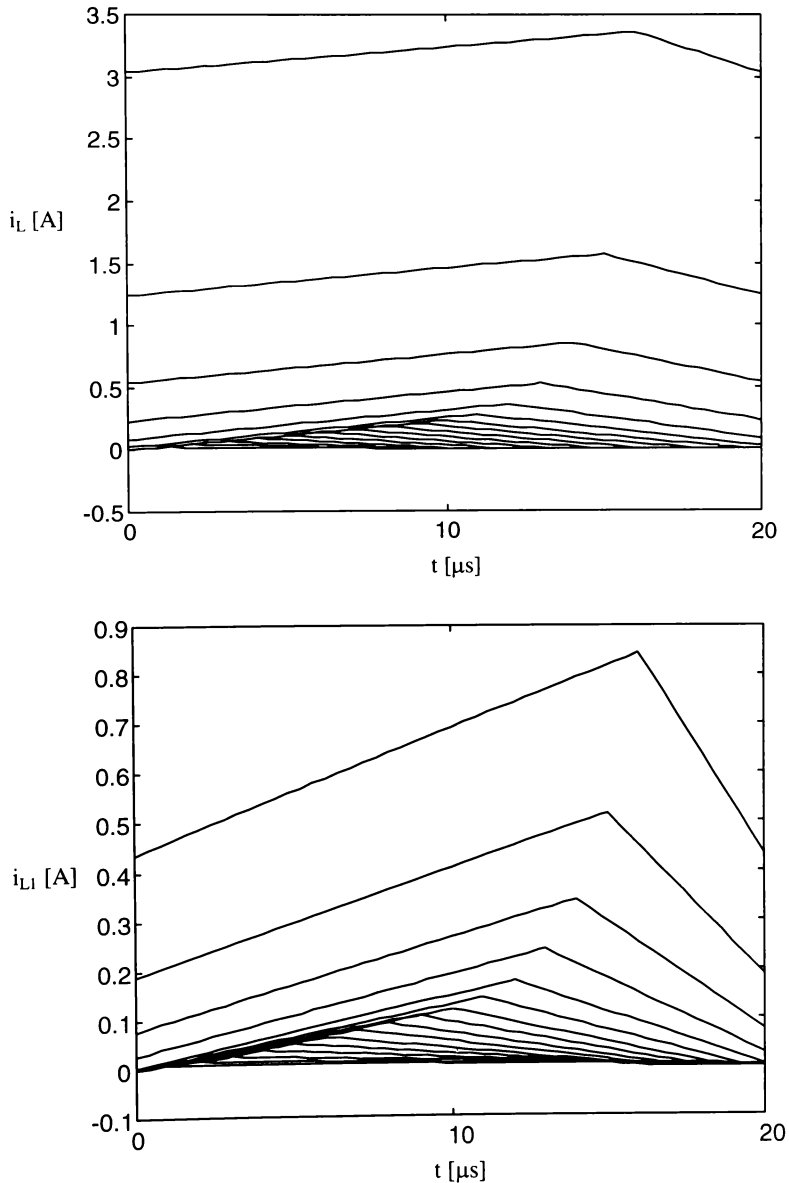


Fig. 6.47. Familiile de forme de undă ale stării staționare pentru cureții prin bobinele convertorului din Fig. 6.45.

Pentru accelerarea simulării valoarea de stare staționară obținută pentru un factorul de umplere este valoare inițială pentru calculul corespunzător următorii valori a factorului de umplere. În acest fel pentru trasarea tuturor caracteristicilor au fost necesare 118 iterații și 198 de simulări, în medie 8 iterații și 13 simulări la o formă de undă, ceea ce spune mult despre eficiența algoritmului.

### Alte aplicații

Legarea unui program de calcul matematic cu un simulator deschide și o altă serie de aplicații, de cu totul altă natură, deosebit de importante. Aceste aplicații sunt legate de cel puțin două aspecte:

- implementarea dificilă, dacă nu imposibilă, a reguletoarelor numerice, a controllerelor fuzzy sau neuronale în simulatoarele de circuite. În schimb operațiile pe care aceste controlleri le execută se pot foarte ușor implementa în programe de calcul matematic, ele reducându-se la memorări de variabile și testări de inegalități asociate cu simple operații algebrice, scalare sau matriciale.
- în regimurile tranzitorii este posibil ca pentru câteva perioade convertorele să funcționeze în alte moduri decât cel din stările staționare. Numeroase modele de semnal mare presupun, chiar dacă nu declarat, funcționarea în același mod, ipoteză care desigur că simplifică modelul, dar de foarte multe ori nu surprinde evoluția reală, corectă, a convertorului. Din acest punct de vedere simularea perioadă cu perioadă cu ajutorul simulatorului nu duce la astfel de erori. La aceasta de adaugă și faptul că se pot lua în considerație toate elementele parazite și modelele de detaliu ale componentelor, care altfel ar fi îngreunat de exemplu scrierea matricilor de stare.

Prima aplicație prezentată este un convertor BUCK-BOOST PWM cu controller digital de tip integrator - Fig. 6.48. Întrerupătoarele se consideră ideale.

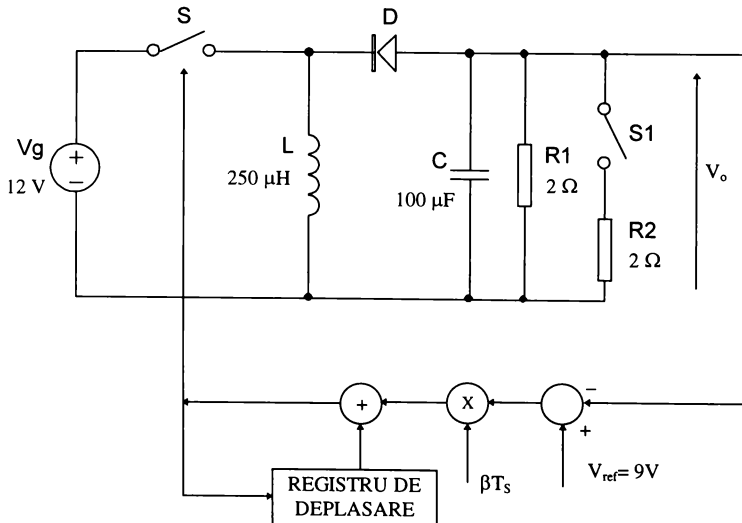


Fig. 6.48. Convertor BUCK-BOOST cu controller digital simulat prin legarea programelor MATLAB și PETS.

Programul MATLAB este cuprins în fișierul *digctrl.m*, iar fișierul pentru PETS este *bb\_fuzzy.cir*, fiind date în Anexa A28. Frecvența de comutație este  $f_s = 20 \text{ kHz}$ , iar controllerul este digital de

tip integrator cu diferență regresivă (“backward difference”) [155], guvernat de următoarea ecuație cu diferențe:

$$d[k] = d[k-1] + T_s \beta (V_{ref} - v_o[k]) \quad (142)$$

sau, echivalent, având funcția de transfer:

$$H_c(z) = \frac{\beta T_s z}{z-1} \quad (143)$$

în care  $d[k]$  este factorul de umplere aferent celei de a  $k$ -a perioade, iar  $v_o[k]$  este tensiunea de ieșire obținută prin eșantionare cu perioada  $T_s$ , la începutul perioadei  $k$ . Valoarea câștigului s-a ales  $\beta = 20$ , iar factorul de umplere a fost limitat (nefigurat în schemă) inferior la 0,05 și superior la 0,95. S-a început simularea direct din starea staționară aferentă conectării numai a lui  $R_1$ . Fig. 6.49 prezintă răspunsul la variații treaptă în sarcină, obținută prin conectarea rezistenței suplimentare  $R_2$  la  $t = 2 \text{ ms}$  și apoi deconectarea ei la  $t = 10 \text{ ms}$ .

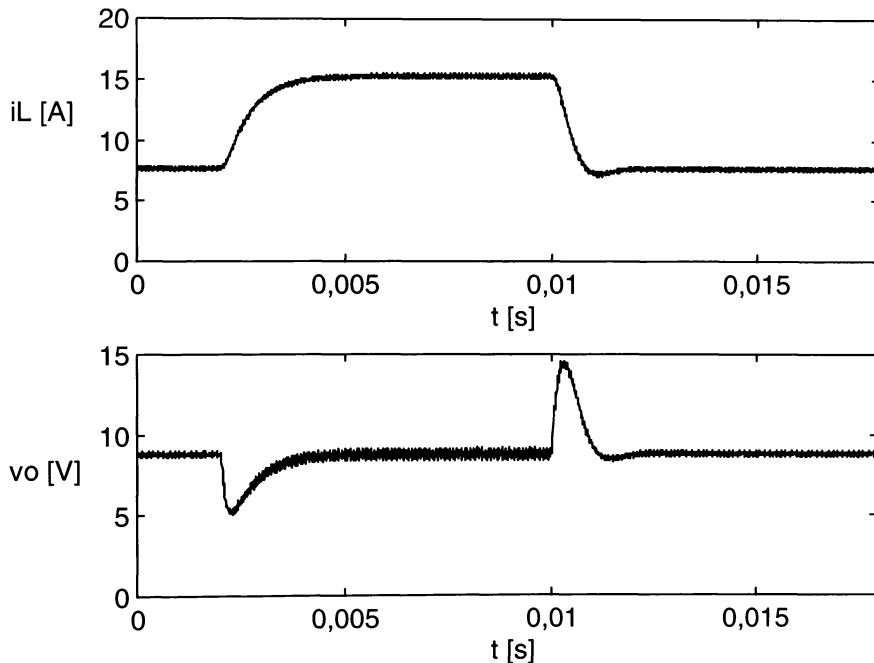


Fig. 6.49. Răspunsul unui convertor BUCK-BOOST cu controller digital la salt treaptă în sarcină, interconectând programul MATLAB pentru simularea controllerului cu programul PETS pentru simularea convertorului.

Deși cu prețul unui timp de simulare mai mare față de situația utilizării unor modele, în acest fel se poate testa în condiții identice cu realitatea, fără a apela la nici un model, stabilitatea și în general dinamica unui astfel de sistem complex, obținând *comportamentul real* al acestuia.

A doua aplicație investigată este un convertor BUCK-BOOST cu controller fuzzy. Schema părții de putere a convertorului este cea din Fig. 6.50, iar controller-ul este inspirat din [174] și este corespondentul unui controller PI și având funcțiile de apartenență simetrice și echidistante. Tabelul de reguli exemplificat în Tab. 6.4, este format din singleton-uri. Parametrii regulatorului sunt (vezi pentru semnificația notațiilor și paragraful 5.3):

$$\eta = 0,01; \beta_e = 5; \beta_{de} = 0,015; V_{ref} = 5V$$

iar factorul de umplere s-a limitat între valorile 0,01 și 0,99.

Atenția se va focaliza nu pe performanțele regulatorului (motiv pentru care simularea nici nu cuprinde tot regimul tranzitoriu) ci pe evoluția convertorului în regimul tranzitoriu cu accent pe modulurile de conducție care apar. Regimul tranzitoriu este cel asociat unui salt treaptă în tensiunea de alimentare, de la 12 V la 20 V, produs la  $t = 0,2 \text{ ms}$ . S-a simulat acest regim dinamic tot interconectând programul MATLAB cu programul PETS. Formele de undă rezultate sunt cele din Fig. 6.51.

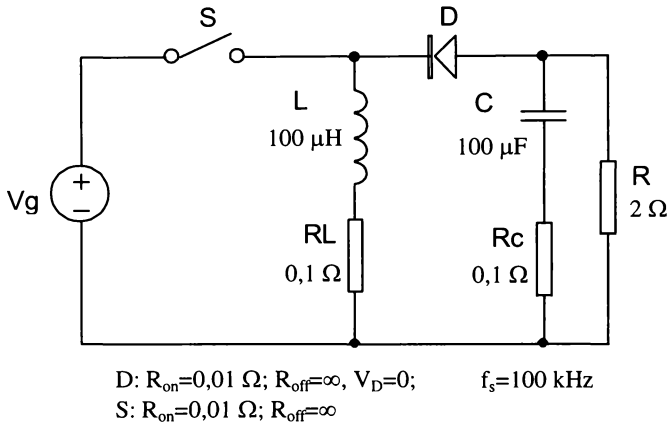


Fig. 6.50. Partea de putere a convertorului BUCK-BOOST cu controller fuzzy.

Tab. 6.4. Tabelul de inferență al regulatorului fuzzy.

de	NB	NS	Z	PS	PB
PB	-0,3	-0,35	-0,45	-0,65	-1
PS	0	-0,1	-0,2	-0,35	-0,5
Z	0,2	0,1	0	-0,1	-0,2
NS	0,5	0,35	0,2	0,1	0
NB	1	0,65	0,45	0,35	0,3

Conform Fig. 6.51 (a), deși inițial convertorul funcționează în CCM, se remarcă modul de conducție DICM instalat pe durata a suficient de multe perioade (o fracțiune semnificativă din regimul tranzitoriu) pentru ca un model simplificat să conducă la erori. Fig. 6.51 (b) care prezintă mărit detaliul încercuit din Fig. 6.51 (a) pune mai bine în evidență funcționarea DICM. Față de modelele simplificate interconectarea programului MATLAB cu programul PETS detectează, așa cum arată Fig. 6.51, trecerea dintr-un mod de conducție într-altul, reușind totodată să simuleze și un controller destul de sofisticat, ca cel fuzzy. Fișierele de simulare sunt *fuzyctrl.m* și *bb\_fuzzy.cir*, prezentate în Anexa A29.

În final se pot trage câteva concluzii. Evident, ca în orice metodă iterativă, punctul de pornire (soluția inițială) joacă un rol decisiv în convergența algoritmului. Autorul tezei propune ca soluție inițială fie starea staționară obținută folosind un model mediat, fie soluția unei stări staționare învecinate (vezi obținerea caracteristicilor statice reale), fie pur și simplu soluția nulă. În



toate experimentele efectuate cu aceste trei soluții inițiale algoritmul au fost convergenți cu un număr redus de iterații, deși, nimeni nu poate garanta pentru ele o convergență necondiționată.

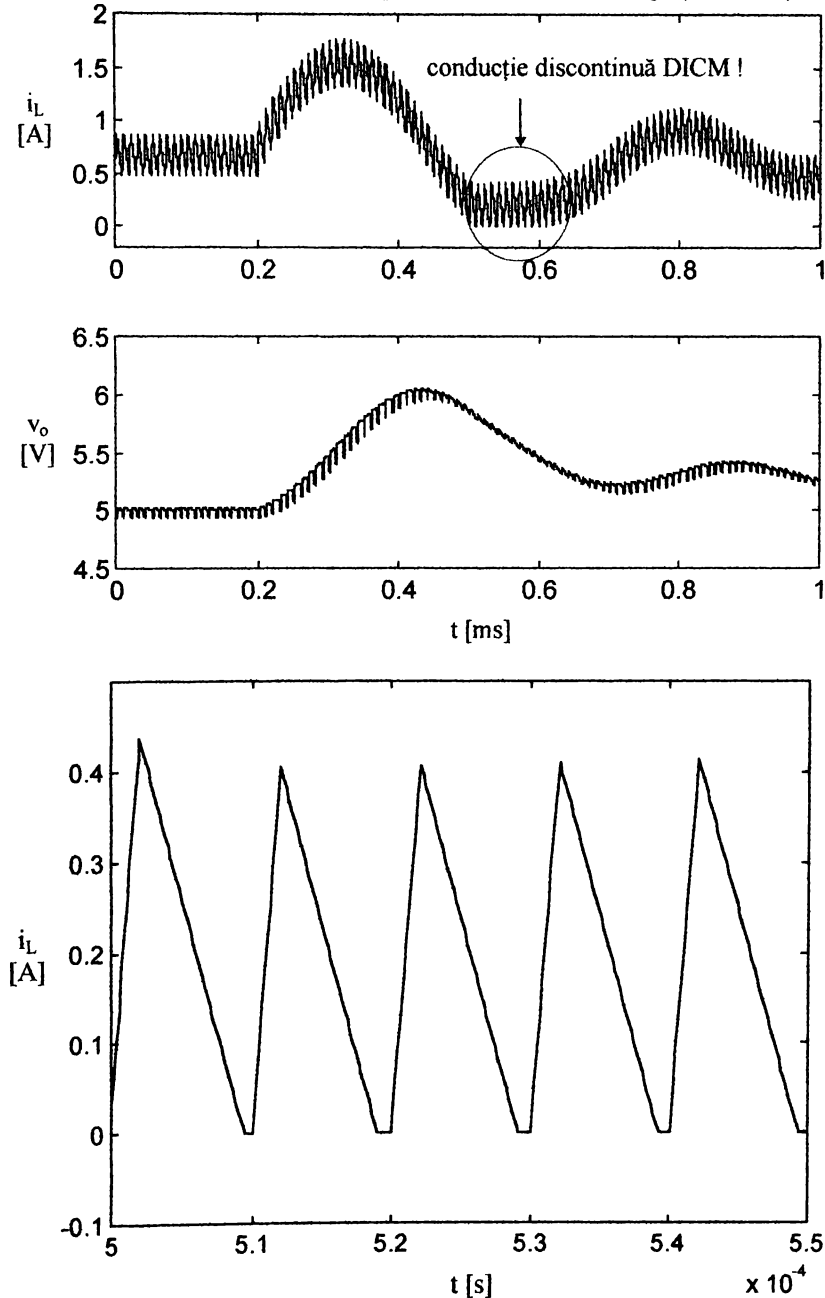


Fig. 6.51. (a) - regimul tranzitoriu. (b) - detaliu în curentul prin bobină pentru evidențierea conducerii discontinue DICM.

Această tehnică de interconectare a unui program de calcul matematic cu un simulator de circuite are însă o serie de avantaje certe și anume:

- pentru rezolvarea stării staționare nu necesită cunoașterea stărilor topologice ale convertorului, nici a succesiunii lor (modul de funcționare) și nici a matricilor de stare. Singurele informații necesare sunt modul de conectare a componentelor, semnalele de comandă care acționează asupra convertorului și care sunt elementele reactive din circuit pentru a putea scrie corect vectorul de stare inițial.
- este aplicabilă nu numai convertoarelor de putere ci oricărui circuit, indiferent de complexitatea sa.
- permite nu numai calculul stării staționare dar și cunoașterea tuturor mărimilor de interes din circuit la orice moment de timp (în particular și în momentele de comutație), reprezentarea formelor de undă și a caracteristicilor statice reale, ținând cont de elementele parazite și comportarea tranzistoarelor și diodelor reale care joacă rol de întrerupătoare.
- este deosebit de performantă pentru simularea convertoarelor cu comandă complexă, de exemplu cu controllere digitale, fuzzy sau neuronale.

Foarte fertilă, această tehnică merită a fi investigată în continuare studiind efectul în convergența metodei pentru alte alegeri ale matricii  $J_k$  din modelul afin (134). Noi aplicații vor viza trasarea curbilor de separare între diferitele moduri de funcționare și metode de extragere a modelelor dinamice ale sistemului. În paralel este de dorit o automatizare mai pronunțată a programului și prezentarea lui sub forma unui mediu integrat, cu interfețe grafice utilizator.

### 6.8.3. Funcțiile de transfer control-ieșire exacte pentru principalele moduri de funcționare și tipuri de comandă în convertoare PWM

Funcțiile de transfer control-ieșire pentru convertoare PWM CCM, PWM DICM, PWM CCM cu comandă în factor de umplere și cu comandă în curent au fost calculate de Tymerski [192], [193].

Folosind tehnica de obținere a funcțiilor de transfer control-ieșire exacte expusă în paragraful 6.8.1, **autorul tezei** a extins acest calcul deducând funcțiile de transfer control-ieșire exacte pentru principalele moduri de funcționare și tipuri de comandă ale convertoarelor PWM. Notațiile sunt cele din paragraful 6.8.1. În dorința concentrării expunerii se va prezenta caracterul specific al fiecărei aplicații în parte, schema convertorului analizat, elementele de circuit, și expresia finală obținută pentru funcția de transfer control-ieșire, tehnica de obținere fiind publicată de autor în referințele bibliografice menționate. Reprezentarea grafică va conține, pentru comparație și caracteristicile de amplitudine și fază furnizate de diverse modele cu mediere.

#### 6.8.3.1. Funcția de transfer control-ieșire exactă pentru convertoare PWM DCVM

Un convertor DCVM se caracterizează prin faptul că tensiunea  $v_{off}$  se anulează pe durata de conducție a întrerupătorului comandat. Prin urmare convertorul are  $N_s = 3$  stări topologice. Notăm cu  $t_1 = kT_S$  momentul de aducere în conducție a întrerupătorului activ, cu  $t_2$  momentul în care dioda intră în conducție deoarece tensiunea  $v_{off}$  se anulează și cu  $t_3 = kT_S + d_k T_S$  momentul de blocare al tranzistorului. Ecuația suplimentară de constrângere este cea care dă pe  $t_2$ , având forma:

$$h_1 x(t_2) = 0 \quad (144)$$

unde  $h_1$  este un vector care extrage pe  $v_{off}$  din vectorul de stare  $x$  și de aceea îl numim vector extractor. Funcția de transfer control-ieșire exactă este [Lascu, 83]:

$$\frac{\hat{y}}{\hat{d}}(j\omega) = (\alpha_3 + \alpha_1\beta_3 + \alpha_2\Gamma_1\beta_1\beta_3 - \zeta_1\mu_1h_1\beta_1\beta_3)(I - \beta_2\Gamma_1\beta_1\beta_3)^{-1}\xi_2 + \zeta_2$$

$$\frac{\omega}{\omega_S} \neq 0.5, 1.5, 1, \dots$$
(145)

cu  $\hat{d} = \frac{\hat{v}_{mp}}{V_{ramp\ pp}}$ , în care  $\hat{v}_{mp}$  reprezintă amplitudinea perturbației armonice care intervine în tensiunea modulatorie, iar  $V_{ramp\ pp}$  amplitudinea vârf la vârf a tensiunii în dinte de fierăstrău cu care se compară tensiunea modulatorie. Celelalte definiții și notații pentru mărimile care intervin în relație sunt cele din paragraful 6.8.1.

Pentru convertorul SEPIC din Fig. 6. 39, a cărui stare staționară a fost rezolvată exact în paragraful 6.8.2.1, vectorul extractor  $h_1$  este:

$$h_1 = [-R_{on} \quad -R_{on} \quad 1 \quad 1]$$
(146)

pentru că  $v_{off}$  este suma tensiunilor pe cele două capacități. Caracteristicile de amplitudine și fază rezultate sunt cele din Fig. 6.52 respectiv Fig. 6.53. Ele conțin și caracteristicile obținute folosind modelul mediat propus de Maksimović în [115]. Fișierul corespunzător, *dsvm.m*, se găsește în Anexa A30.

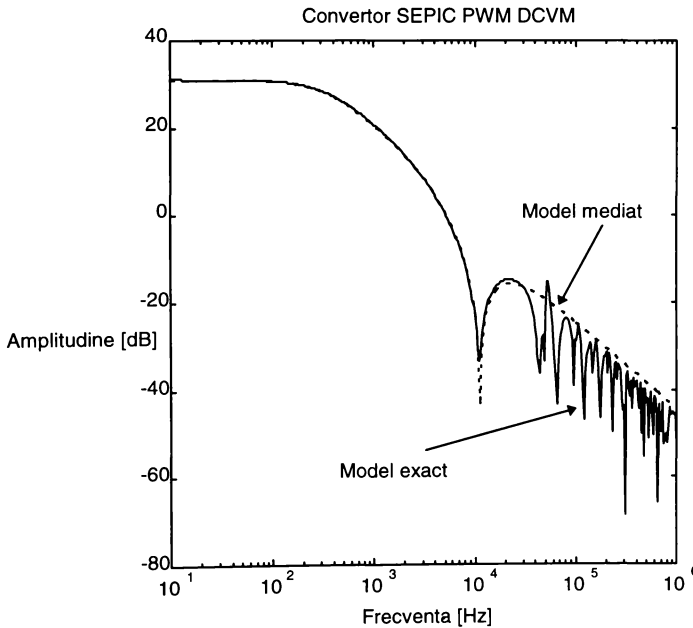


Fig. 6.52. Caracteristica de amplitudine. Linia continuă: modelul exact; linia punctată: modelul mediat după Maksimović.

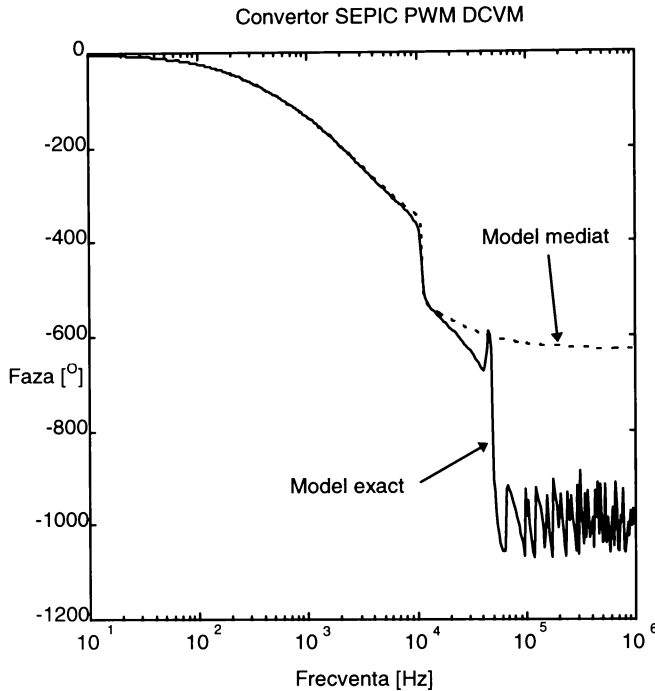


Fig. 6.53. Caracteristica de fază. Linia continuă: modelul exact; linia punctată: modelul mediat după Maksimović.

### 6.8.3.2. Funcția de transfer control-ieșire exactă pentru convertoare PWM DIVM

Un convertor DIVM are o funcționare asemănătoare cu un convertor DCVM cu deosebirea că la aceste convertoare tensiunea  $v_{C_i}$ , de pe capacitatea inserată suplimentar împreună cu o bobină  $L_i$ , se anulează pe durata de conducție a întrerupătorului comandat. Formal ecuațiile specifice rămân aceleași cu cele pentru convertoare DCVM, singura deosebire fiind în alegerea vectorului extractor  $h_1$ , care are o altă expresie.

Se consideră, de exemplu, convertorul BUCK DIVM din Fig. 6.54, identic cu cel din [103], unde este dat și modelul de semnal mic și un set de măsurători, pentru a putea avea o comparație.

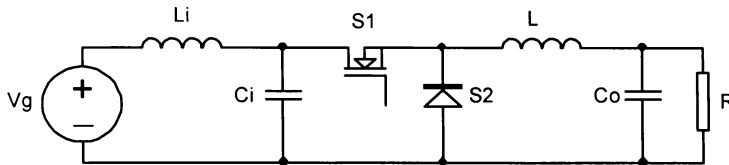


Fig. 6.54. Convertor BUCK DIVM.

Elementele de circuit au valorile:

$$V_g = 15V; \quad L_i = 970 \mu H; \quad L = 212 \mu H; \quad C_o = 2659 \mu F; \quad C_i = 104 nF; \quad R = 5,2 \Omega;$$

$$f_s = 33 kHz; \quad D = 0,345; \quad R_{on} = 0,1 m\Omega$$

Definind vectorul de stare

$$x = [i_L \quad i_{L_i} \quad v_C \quad v_{C_i}]^T \quad (147)$$

atunci vectorul extractor  $h_1$  este:

$$h_1 = [0 \quad 0 \quad 0 \quad 1] \quad (148)$$

Soluția exactă a stării staționare a fost calculată prin metoda Newton-Raphson (dacă nu se specifică mărimile sunt exprimate în Sistemul Internațional de Unități de Măsură) și ea a fost obținută după 9 iterații, fiind:

$$X(T_1) = X(T_4) = \begin{bmatrix} 0.439 \\ 0.089 \\ 3.369 \\ 33.630 \end{bmatrix}; \quad X(T_2) = \begin{bmatrix} 0.830 \\ 0.068 \\ 3.369 \\ 0 \end{bmatrix}; \quad X(T_3) = \begin{bmatrix} 0.751 \\ 0.145 \\ 3.369 \\ 0 \end{bmatrix};$$

$$T_1' = 5.419\mu s; \quad T_1' = 4.930\mu s; \quad T_1' = 19.650\mu s;$$

Caracteristicile de amplitudine și fază sunt date în Fig. 6.55 respectiv Fig. 6.56 - vezi Anexa A30, fișierul *divm.m*.

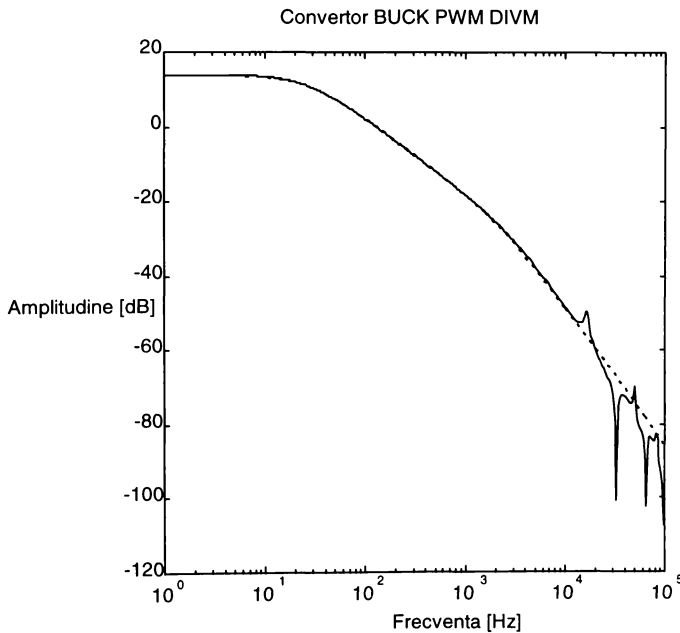


Fig. 6.55. Caracteristica de amplitudine. Linia continuă: modelul exact; linia punctată: modelul mediat.

### 6.8.3.3. Funcția de transfer control-ieșire exactă pentru convertoare PWM CCM cu feedforward

Tehnica de feedforward (reacție anticipativă) reprezintă procedeul prin care, în stare staționară, tensiunea de ieșire a unui convertor PWM,  $V_o$ , este făcută să nu mai depindă de

tensiunea de alimentare,  $V_g$ , fără a se introduce o buclă de reacție după tensiunea de ieșire, ci numai prin modificări la nivelul modulatorului în durată.

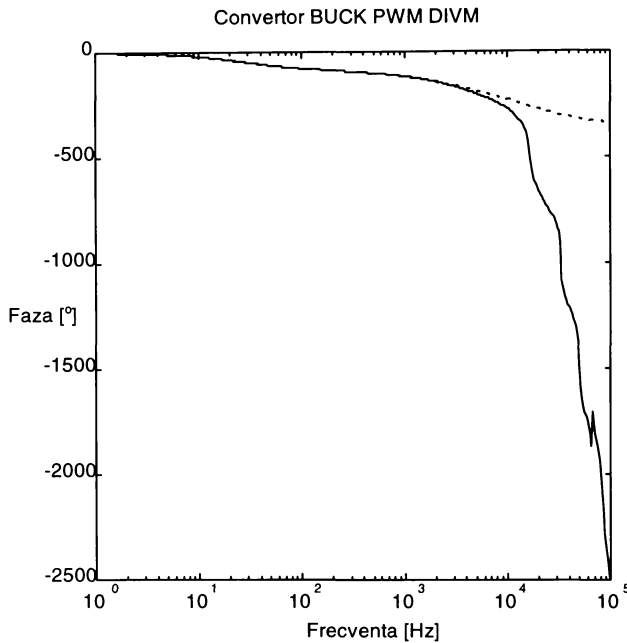


Fig. 6.56. Caracteristica de fază. Linia continuă: modelul exact; linia punctată: modelul mediat.

Numeroase articole au fost publicate în legătură cu diversele procedee de implementare practică a reacției feedforward. Cele mai reprezentative sunt referințele [13],[59] și [5]. Se propune o analiză de semnal mic exactă pentru convertoare având feedforward implementat după tehnica descrisă în [5], apoi pentru convertoare cu reacția feedforward realizată clasic, cu multiplicatoare rapide.

Observația de la care se pleacă în [5] este aceea că în orice convertor PWM CCM cu elemente de circuit ideale raportul static de conversie,  $M$ , este un raport de două polinoame. Nu ne propunem să investigăm și reacția feedforward pentru convertoare pătratice, deși această analiză decurge identic. Ne vom concentra numai asupra convertoarelor BUCK, BOOST, BUCK-BOOST și a celor din familia ZSWJ, la care polinoamele din exprimarea lui  $M$  sunt de gradul I, deci:

$$M = \frac{V_o}{V_g} = \frac{P(D)}{Q(D)} = \frac{p_1 D + p_0}{q_1 D + q_0} \quad p_0, p_1, q_0, q_1 \in \{-2, -1, 0, 1, 2\} \quad (149)$$

Pentru a se obține o tensiune de ieșire independentă de tensiunea de intrare, de forma

$$V_o = A v_m \quad (150)$$

unde  $v_m$  este tensiunea modulatorului (tensiunea de comandă) iar  $A$  o constantă în care intervin parametri ai modulatorului, se arată că funcția de modulație a convertorului trebuie să depindă, pe lângă  $v_m$  și parametrii modulatorului și de  $V_g$ . Din (149) și (150), eliminând pe  $V_o$ , se obține:

$$v_m Q(D) - \frac{1}{A} V_g P(D) = 0 \quad (151)$$

Din relația (151) făcând  $D \rightarrow \frac{t}{T_s}$  se obține funcția de modulație a convertorului în forma:

$$g\left(\frac{t}{T_s}, V_g, v_m\right) = v_m Q\left(\frac{t}{T_s}\right) - \frac{1}{A} V_g P\left(\frac{t}{T_s}\right) \quad (152)$$

Momentul  $t_1 = kT_s$ , de aducere în conducție a întrerupătorului activ, este fixat de un semnal de ceas extern, în timp ce momentul  $t_2$ , în care se produce blocarea, este dat de anularea funcției de modulație:

$$v_m Q\left(\frac{t}{T_s}\right) - \frac{1}{A} V_g P\left(\frac{t}{T_s}\right) = 0 \quad (153)$$

sau, folosind expresiile pentru  $P$  și  $Q$ , se obține pentru  $t_2$  ecuația:

$$\left(q_1 v_m - p_1 \frac{V_g}{A}\right) \frac{t}{T_s} = p_0 \frac{V_g}{A} - q_0 v_m \quad (154)$$

Într-o implementare clasică cu multiplicatoare rapide,  $t_2$  se obține direct din (154). Deși este posibilă, utilizarea multiplicatoarelor rapide constituie un dezavantaj. De aceea în [5] Arbetter propune aproximarea

$$u(t) \frac{t}{T_s} \cong \frac{1}{T_s} \int_0^{T_s} u(\tau) d\tau \quad (155)$$

adevărată dacă  $u(t)$  are o variație lentă pe intervalul de integrare. Aplicarea acestei aproximări în (154) face ca  $t_2$  să rezulte din ecuația:

$$\frac{1}{T_s} \int_{t_1}^{t_2} \left(q_1 v_m - p_1 \frac{V_g}{A}\right) dt = p_0 \frac{V_g}{A} - q_0 v_m(t_2) \quad (156)$$

cu avantajul înlocuirii multiplicatoarelor cu un integrator resetabil. Articolul descrie apoi particularitățile de realizare practică, propunând în final și un model mediat de semnal mic.

Deducerea modelului exact pleacă de la scrierea mărimilor ce intervin în (156) ca sumă de mărimi statice și perturbații de semnal mic:

$$v_m = V_m + \hat{v}_{mp} \cos \omega t \quad (157)$$

$$t_2 = T_2 + \hat{t}_2 \quad (158)$$

care înlocuite în (156), după efectuarea calculelor și a aproximărilor tipice de semnal mic, conduc la expresia lui  $\hat{t}_2$ :

$$\begin{aligned} \hat{t}_2 = & \frac{T_s}{p_1 \frac{V_g}{A} - q_1 V_m} \left[ \frac{1}{2} (q_0 e^{j\omega DT_s} + q_1 D \frac{\sin \frac{\omega DT_s}{2}}{\frac{\omega DT_s}{2}}) e^{j\omega k T_s} + \right. \\ & \left. + \frac{1}{2} (q_0 e^{-j\omega DT_s} + q_1 D \frac{\sin \frac{\omega DT_s}{2}}{\frac{\omega DT_s}{2}}) e^{-j\omega k T_s} \right] \hat{v}_{mp} \end{aligned} \quad (159)$$

În acest fel există toate mărimile necesare calculului funcției de transfer exacte care rezultă:

$$\begin{aligned} \frac{\hat{v}_o}{\hat{v}_{mp}} = & \frac{1}{p_1 \frac{V_g}{A} - q_1 V_m} e^{j\omega DT_s} \left( q_0 + q_1 D \frac{\sin \frac{\omega DT_s}{2}}{\frac{\omega DT_s}{2}} e^{-\frac{j\omega DT_s}{2}} \right) \\ & \cdot \left[ (\alpha_1 + \alpha_2 \beta_1) e^{-j\omega T_s} (I - \beta_2 \beta_1)^{-1} \Phi_2 \xi_1 + (\zeta_1 + \alpha_2 \xi_1) e^{-j\omega DT_s} \right] \end{aligned} \quad (160)$$

unde evident  $D = \frac{T_2 - T_1}{T_s}$ .

În cazul implementării clasice cu multiplicatoare, urmând un raționament similar, se găsește:

$$\hat{v}_2 = \frac{T_s}{V_g - q_1 V_m} (q_0 + q_1 D) \frac{1}{2} \left[ e^{j\omega DT_s} e^{j\omega k T_s} + e^{-j\omega DT_s} e^{-j\omega k T_s} \right] \hat{v}_{mp} \quad (161)$$

și

$$\frac{\hat{v}_o}{\hat{v}_{mp}} = \frac{1}{p_1 \frac{V_g}{A} - q_1 V_m} e^{j\omega DT_s} (q_0 + q_1 D) \cdot \left[ (\alpha_1 + \alpha_2 \beta_1) e^{-j\omega T_s} (I - \beta_2 \beta_1)^{-1} \Phi_2 \xi_1 + (\zeta_1 + \alpha_2 \xi_1) e^{-j\omega DT_s} \right] \quad (162)$$

Unitar, cele două funcții de transfer se scriu în forma:

$$\frac{\hat{v}_o}{\hat{v}_{mp}} = \frac{1}{p_1 \frac{V_g}{A} - q_1 V_m} e^{j\omega DT_s} (q_0 + q_1 D \varepsilon) \cdot \left[ (\alpha_1 + \alpha_2 \beta_1) e^{-j\omega T_s} (I - \beta_2 \beta_1)^{-1} \Phi_2 \xi_1 + (\zeta_1 + \alpha_2 \xi_1) e^{-j\omega DT_s} \right] \quad (163)$$

$$\varepsilon = \begin{cases} 1 & \text{cu multiplicator} \\ \frac{\sin \frac{\omega DT_s}{2}}{\frac{\omega DT_s}{2}} e^{-j \frac{\omega DT_s}{2}} & \text{cu int egrator restabil} \end{cases}$$

Ca și aplicație practică s-a studiat convertorul BOOST examinat și în [5], având următorii parametri:

$$V_g = 335 \text{ V}; \quad V_o = 50 \text{ V}; \quad L = 718 \text{ } \mu\text{H}; \quad R_L = 0,4 \text{ } \Omega; \quad C_o = 10 \text{ } \mu\text{F}; \quad R = 250 \text{ } \Omega;$$

$$V_m = 4 \text{ V}; \quad A = 12,5; \quad f_s = 50 \text{ kHz};$$

Pentru tranzistor rezistența în conducție este  $R_{on} = 0,7 \text{ } \Omega$ . Caracteristicile de amplitudine și de fază obținute sunt reprezentate în Fig. 6.57 și respectiv Fig. 6.58. Pentru reprezentare a se vedea Anexa A 30, fișierul  $ff\_ccm$ . Este interesant de remarcat că toate modelele coincid la frecvențe joase, ceea ce argumentează, de data aceasta prin comparație cu un model exact, superioritatea soluției cu integratoare din [5].

#### 6.8.3.4 Funcția de transfer control-ieșire exactă pentru convertor PWM CCM cu comanda prin sarcină

Schema de principiu a unui convertor cu comandă prin sarcină, preluată din [182] este reluată în Fig. 6.59. Este clar că, pe lângă variabilele de stare ale convertorului de putere cuprinse în vectorul  $x_d$ , sistemul mai conține o variabilă suplimentară și anume tensiunea pe capacitatea  $C_T$ . Funcționarea fiind CCM,  $N_s = 2$ , iar vectorul de stare și matricile de stare se definesc în forma:

$$x = [x_d \quad v_{C_T}]^T \quad (164)$$



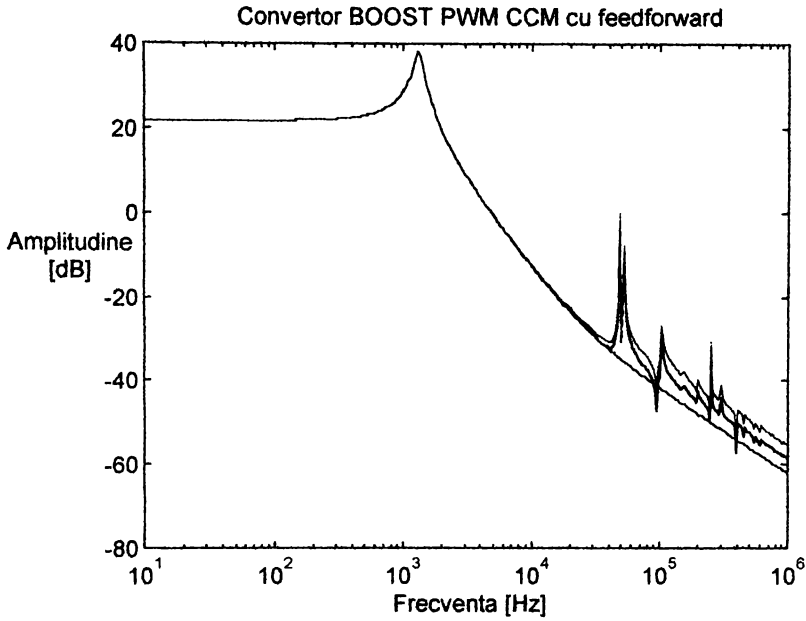


Fig. 6.57 Caracteristica de amplitudine. Roșu - modelul exact pentru varianta cu integrator resetabil; negru - modelul exact pentru varianta cu multiplicator; verde - modelul mediat.

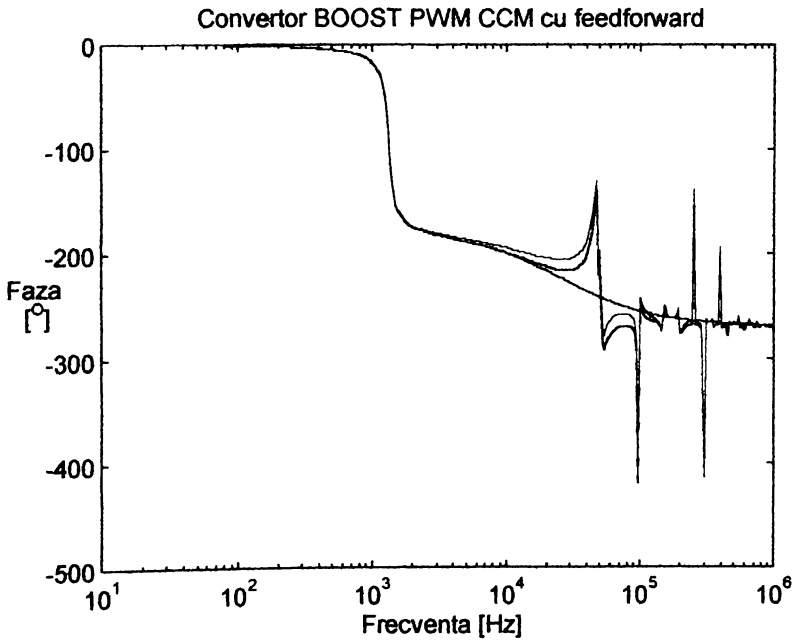


Fig. 6.58. Caracteristica de fază. Roșu - modelul exact pentru varianta cu integrator resetabil; negru - modelul exact pentru varianta cu multiplicator; verde - modelul mediat.

$$\begin{aligned}
 A_1 &= \begin{bmatrix} \dots & 0 \\ [A_{1d}] & 0 \\ \frac{1}{C_T} & 0 & 0 \end{bmatrix} & A_2 &= \begin{bmatrix} \dots & 0 \\ [A_{2d}] & 0 \\ 0 & 0 & -\frac{1}{r_{ON} C_T} \end{bmatrix} & B_1 &= \begin{bmatrix} [B_{1d}] \\ 0 \end{bmatrix} & B_2 &= \begin{bmatrix} [B_{2d}] \\ 0 \end{bmatrix} \\
 E_1 &= \begin{bmatrix} [E_{1d}] \\ 0 \end{bmatrix} & E_2 &= \begin{bmatrix} [E_{2d}] \\ 0 \end{bmatrix} & F_1 &= F_{1d} & F_2 &= F_{2d}
 \end{aligned} \tag{165}$$

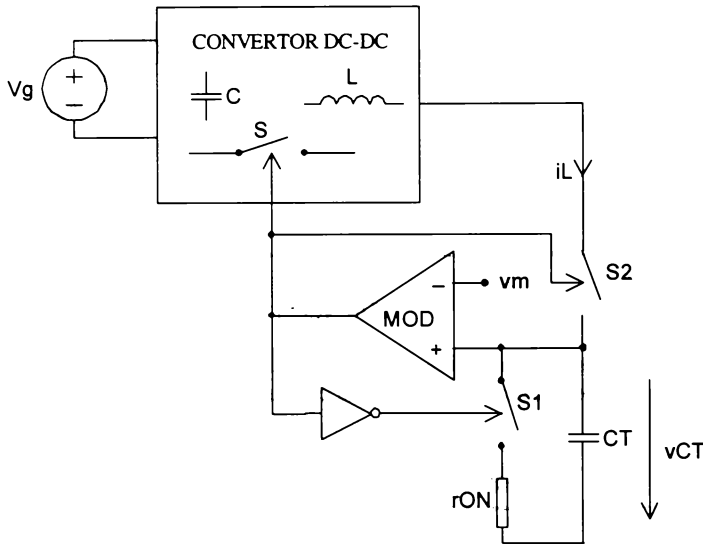


Fig. 6.59. Principiul comenzii prin sarcină.

Momentul  $t_1 = kT_s$  este fixat de un ceas extern, în timp ce momentul  $t_2$ , de blocare a întrerupătoarelor active, este dat de momentul în care  $v_{CT}$  egalează pe  $v_m$ . Matematic aceasta se scrie:

$$h_1 x(t_2) = v_m(t_2) \tag{166}$$

unde  $h_1 = [0 \quad 0 \quad 1]$  extrage pe  $v_{CT}$  din vectorul de stare  $x$ . Pentru calculul exact al stării staționare în CCM este suficientă cunoașterea factorului de umplere  $D$ , în acest fel matricile  $\Phi_i$  și  $\Psi_i$  fiind și ele cunoscute. Soluția stării staționare este [193]:

$$X(T_1) = (I - \Phi_2 \Phi_1)^{-1} (\Phi_2 \Psi_1 + \Psi_2) V_g \tag{167a}$$

$$X(T_2) = (I - \Phi_1 \Phi_2)^{-1} (\Phi_1 \Psi_2 + \Psi_1) V_g \tag{167b}$$

$$T_1 = DT_s; \quad T_2 = (1-D)T_s \tag{167c}$$

Toate elementele necesare calculului funcției de transfer exacte sunt acum cunoscute. Autorul tezei a publicat o deducere detaliată a ei în [82]. Forma finală a rezultatului este:

$$\frac{\hat{y}}{\hat{v}_{mp}}(j\omega) = \frac{\mu_1}{T_S} [(\alpha_1\beta_2 + \alpha_2 - \zeta_1\mu_1h_1\beta_1\beta_2) \cdot (I - \Gamma_1\beta_1\beta_2)^{-1} \zeta_1 + \zeta_1] \quad (168)$$

$$\frac{\omega}{\omega_S} \neq 0,5; 1,5; 1, \dots$$

Aplicarea calculului exact al funcției de transfer control-ieșire s-a făcut pe același convertor BUCK analizat de Tang, Lee, Ridley și Cohen în [182], având parametrii:

$$V_g = 15V; \quad V_O = 5V; \quad L = 37,5 \mu H; \quad C_o = 380 \mu F; \quad R_{C_o} = 20 m\Omega;$$

$$C_T = 733 nF; \quad r_{ON} = 10 m\Omega; \quad f_S = 90 kHz$$

În Fig. 6.60 și Fig. 6.61 au fost reprezentate caracteristicile de amplitudine și de fază, împreună cu cele furnizate de modelul mediat propus sub formă de schemă bloc de autorii menționați mai sus. Fișierul utilizat este *charge.m* din Anexa A30.

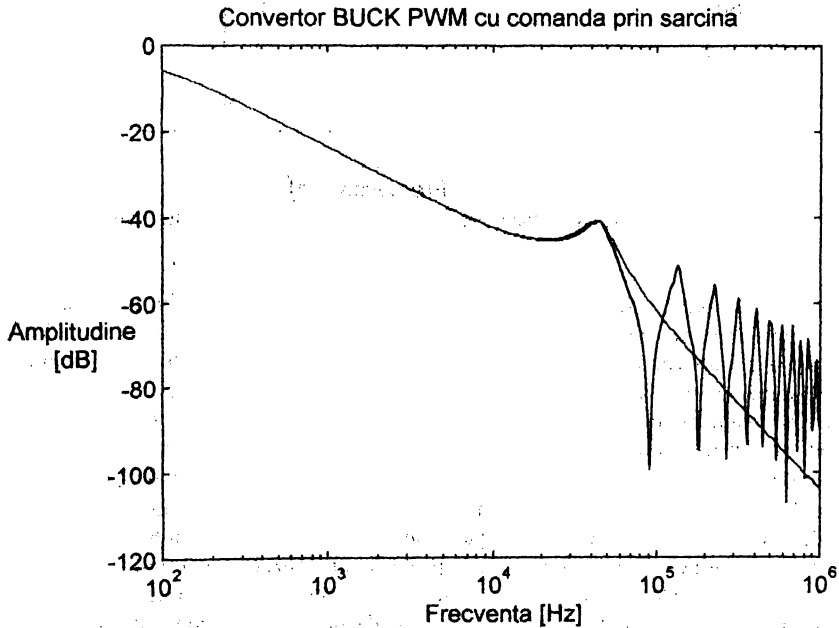


Fig. 6.60. Caracteristica de amplitudine. Negru - modelul exact; roșu - modelul mediat.

### 6.8.3.5 Funcția de transfer control-ieșire exactă pentru convertoare PWM CCM cu comandă integrativă

Pentru comanda integrativă după o variabilă momentul  $t_1 = kT_S$ , de aducere în conducție a întrerupătorului activ, este dat de un ceas extern, iar momentul  $t_2$ , de blocare a acestuia, este dat de egalitatea (vezi și paragraful 5.2.2 relația (3)):

$$\frac{1}{\tau} \int_0^{d_k T_S} u(t) dt = v_m(d_k T_S) \quad (169)$$

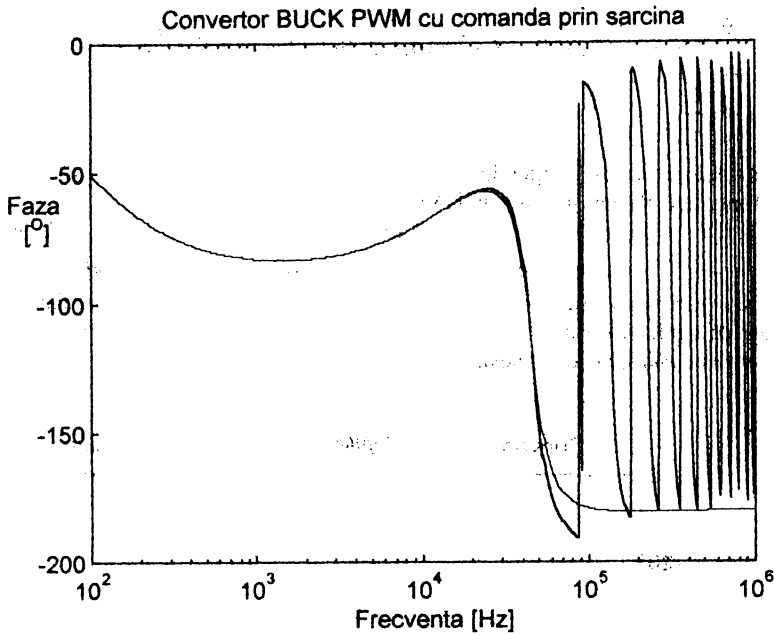


Fig. 6.61. Caracteristica de fază. Negru - modelul exact; roșu - modelul mediat.

în care  $u$  poate fi orice variabilă, care însă de obicei este aleasă astfel încât să fie într-o relație directă cu tensiunea sau curentul de ieșire (de exemplu valorile lor medii să coincidă). În [80] autorul tezei a publicat o metodă generală de calcul a funcției de transfer pentru comanda integrativă după o variabilă. În esență se oferă formule generale de calcul pentru variabila integrată pe durata timpului de conducție și se extinde vectorul de stare prin adăugarea tensiunii pe condensatorul din integrator. Având în vedere aceste observații, pot fi scrise cele două seturi de matrici de stare, plecându-se de la matricile convertorului de putere. Pe de altă parte, momentul  $t_2$  se obține dintr-o ecuație de tipul (166), care perturbată întregeste elementele necesare calculului funcției de transfer control-ieșire exacte după strategia expusă în paragraful 6.8.1.

În cele ce urmează se arată că metoda poate fi generalizată și mai mult, în sensul că poate îngloba orice tehnică integrativă. O primă observație pe care o facem în acest sens este aceea că indiferent de tehnica integrativă considerată (comandă prin sarcină, comandă integrativă după o variabilă sau comandă mixtă) schema conține un integrator (la comanda mixtă se scriu ambele variabile sub aceeași integrală) cu constanta de timp  $\tau = R_{\text{int}} C_{\text{int}}$ , dată de elementele  $R_{\text{int}}$  și  $C_{\text{int}}$ . Notăm cu  $v_{C_{\text{int}}}$  tensiunea de pe capacitatea  $C_{\text{int}}$  a integratorului, cu sensul ca în Fig. 6.62 și definim vectorul de stare, de dimensiune  $n+1$ , al întregului sistem,  $x$ :

$$x = \begin{bmatrix} x_d \\ v_{C_{\text{int}}} \end{bmatrix} \quad (170)$$

unde cu  $x_d$  s-a notat vectorul de stare al convertorului de putere.

În practică cele care se integrează ușor sunt tensiunile, fiind deci necesară o conversie a tuturor mărimilor implicate în relația de definire a momentului de blocare a întrerupătorului activ în tensiuni. Fie vectorul  $p = [p_1 \ p_2 \ \dots \ p_{n+1}]$  și scalarul  $q$  mărimile ce definesc tensiunea de la intrarea în integrator funcție de vectorul  $x$  și tensiunea  $V_g$ , deci:

$$u = px + qV_g \quad (171)$$

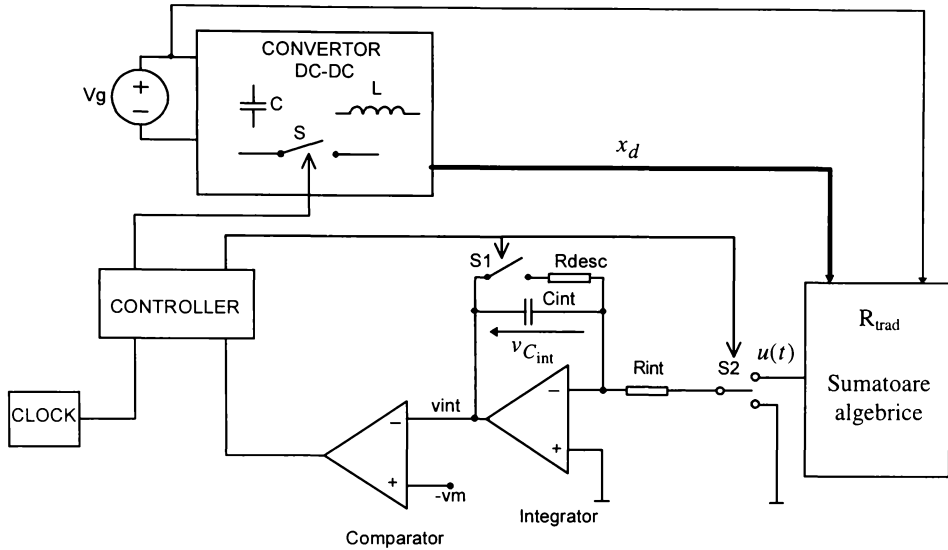


Fig. 6.62. Schema bloc generală a oricărei comenzi integrative.

Elementele vectorului  $p$  sunt tocmai rezistențele de conversie sau rapoartele de divizare însoțite de semnul corespunzător cu care intră în modul de definire al mărimii  $u(t)$ , iar  $q$  eventualul raport de divizare cu care intră  $V_g$  în  $u(t)$ . Comutatorul  $S_2$  este în poziția superioară pe durata de conducție și în cea inferioară pe cea de blocare. În majoritatea schemelor practice el lipsește, conexiunea dintre  $u(t)$  și  $R_{int}$  fiind permanentă. Aceasta nu influențează semnificativ funcționarea dacă rezistența de descărcare a condensatorului  $C_{int}$  pe durata blocării este mult mai mică decât  $R_{int}$ . În continuare se va considera varianta cu  $S_2$  prezent, deoarece modelele mediate publicate în literatură au fost elaborate în acest context.

Cu notațiile de mai sus matricile de stare au forma:

$$A_1 = \begin{bmatrix} & & & 0 \\ & [A_{1d}] & & 0 \\ & & & \vdots \\ & & & 0 \\ \frac{p_1}{\tau} & \frac{p_2}{\tau} & & \frac{p_{n+1}}{\tau} \end{bmatrix}; \quad B_1 = \begin{bmatrix} [B_{1d}] \\ \frac{q}{\tau} \end{bmatrix}; \quad E_1 = [[E_{1d}]0]; \quad F_1 = F_{1d} \quad (172)$$

$$A_2 = \begin{bmatrix} & & & 0 \\ & [A_{2d}] & & 0 \\ & & & \vdots \\ & & & 0 \\ 0 & 0 & 0 & -\frac{1}{R_{desc} C_{int}} \end{bmatrix}; \quad B_2 = \begin{bmatrix} [B_{2d}] \\ 0 \end{bmatrix}; \quad E_2 = [[E_{2d}]0]; \quad F_2 = F_{2d}$$

Din schema bloc se obține ecuația care definește pe  $t_2$  în forma:

$$-v_{C_{int}}(t_2) = -v_m(t_2) \quad (173)$$

sau, echivalenta:

$$h_1 x(t_2) = v_C(t_2) \quad (174)$$

unde vectorul

$$h_1 = [0 \ 0 \ \dots \ 0 \ 1] \quad (175)$$

extrage tensiunea  $v_{C_{int}}$  din vectorul de stare  $x$ .

Având definite toate matricile de stare și toate relațiile de constrângere (de fapt numai una, (174)) aplicând tehnica de calcul a funcției de transfer control-ieșire exacte se obține:

$$\frac{\hat{y}}{\hat{v}_{mp}}(j\omega) = \frac{\mu_1}{T_s} [(\alpha_1 \beta_2 + \alpha_2 - \zeta_1 \mu_1 h_1 \beta_1 \beta_2) \cdot (I - \Gamma_1 \beta_1 \beta_2)^{-1} \xi_1 + \zeta_1] \quad (176)$$

$$\frac{\omega}{\omega_s} \neq 0,5; 1,5; 1, \dots$$

Dacă se compară (176) cu (168) se observă că formal relațiile sunt aceleași, ceea ce era de așteptat.

Pentru comanda prin sarcină, la convertoare cu o bobină și un condensator  $p = [R_T \ 0 \ \dots \ 0]$  și  $q = 0$ , unde  $R_T$  este rezistența de conversie a traductorului curent-tensiune, iar vectorul de stare se presupune că are curentul inductiv pe prima poziție. Dacă în discuție sunt convertoare cu mai multe bobine, atunci corespunzător poziției fiecărui curent inductiv din vectorul de stare care trece și prin întrerupătorul activ în conducție va apare rezistența traductorului respectiv cu semnul minus sau plus.

Pentru comanda integrativă după o variabilă  $q \in \{-1 \ 0 \ 1\}$ , dar forma lui  $p$  poate fi mai complicată, depinzând de expresia variabilei în discuție, pe durata de conducție, în funcție de mărimile de stare și tensiunea de alimentare.

Pentru comanda mixtă, în  $p$  apar rezistențele traductoarelor de curent în pozițiile curenților inductivi care dau pe  $i_{on}$  și +1 sau -1 pe pozițiile tensiunilor capacitive din bucla ce definește pe  $v_{off}$ .

Fie ca exemplu convertorul BUCK CCM studiat în [171] și reluat în [172], la care se aplică comanda mixtă cu următorii parametri:

$$V_g = 15V; \quad L = 0,48 \text{ mH}; \quad C_o = 30 \mu F; \quad R = 25 \Omega; \quad f_s = 30 \text{ kHz}; \quad D = 0,33$$

Pentru o "contribuție" cât mai echilibrată a lui  $i_{on}$  și  $v_{off}$  în tensiunea de la ieșirea integratorului s-a ales un traductor de curent cu transrezistența de  $R_{trad} = 50 \Omega$  și un raport de divizare al tensiunii  $v_{off} = V_g$  de valoare  $q = 1$ . Întrerupătoarele se consideră ideale. Funcția de transfer control ieșire este reprezentată în Fig. 6.63 și Fig. 6.64 - vezi Anexa A30, fișierul *mixt.m*.

Pentru comparație s-au reprezentat și caracteristicile furnizate de modelul mediat publicat de **Lascu** în [81]. Acest model are schema bloc analogă cu cea din [182], parametrării modelului având valorile:

$$F_m = \frac{\tau}{T_s} \frac{1}{R_{trad} I_L (DT_s) + qV_g}; \quad R_i = R_{trad} \frac{DT_s}{\tau};$$

$$k_f = -\frac{D^2 (1-D) T_s^2}{2L\tau} R_{trad} - \frac{DT_s}{\tau}; \quad k_r = \frac{D(1-D) T_s^2}{2L\tau} R_{trad}$$

Se observă că modelul mediat este exact până la aproximativ jumătate din frecvența de comutație.

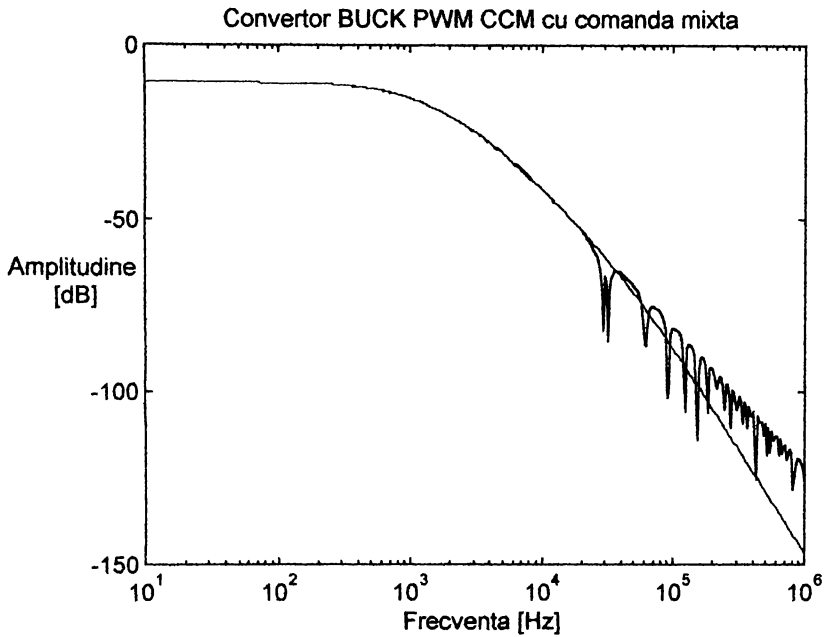


Fig. 6.63. Caracteristica de amplitudine. Negru - modelul exact; roșu - modelul mediat.

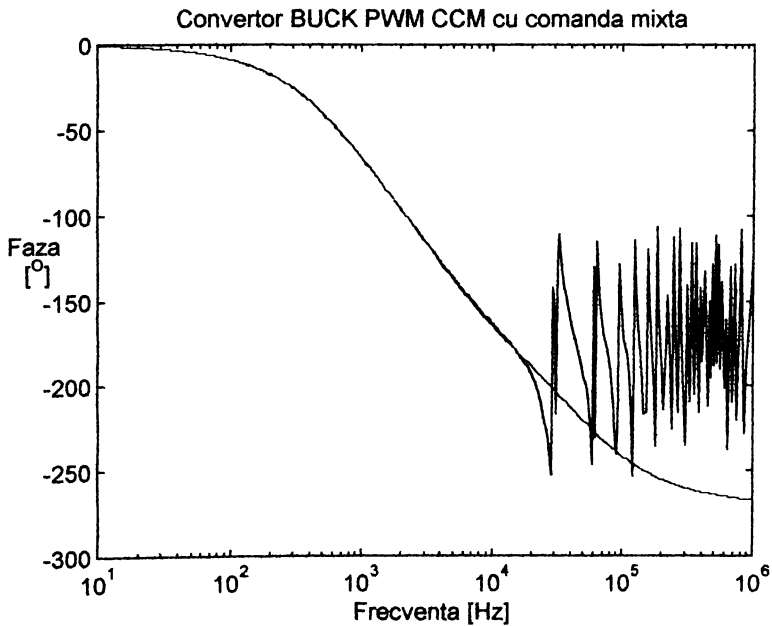


Fig. 6.64. Caracteristica de fază. Negru - modelul exact; roșu - modelul mediat.

### 6.8.3.6 Expresia generală exactă a funcției de transfer control-ieșire pentru orice tip de comandă integrativă în moduri discontinue.

Așa cum se arată în [171] și [172] comenzile integrative se pot aplica și la funcționare în conducție discontinuă, păstrându-se toate avantajele specifice lor. Aceasta justifică necesitatea găsirii expresiilor exacte a funcțiilor de transfer și în aceste moduri.

Păstrând notațiile și definițiile pentru vectorul de stare și matricile de stare formulate în paragraful precedent, o primă remarcă se referă la faptul că în aceste moduri numărul stărilor topologice este  $N_s = 3$ .

La funcționarea DICM a doua comutație are loc la momentul  $t_2$ , când tensiunea pe condensatorul de integrare,  $C_{int}$ , egalează tensiunea modulatorului de comandă, ceea ce se scrie matematic în forma:

$$h_1 x(t_2) = v_m(t_2) \quad (177)$$

unde  $h_1 = \begin{bmatrix} 0 & 0 & 0 & 1 \end{bmatrix}$  este matricea extractoare a tensiunii  $v_{C_{int}}$  din vectorul de stare.

A treia comutație se produce la anularea curentului  $i_{on}$ , momentul  $t_3$  corespunzător fiind dat de ecuația:

$$h_2 x(t_3) = 0 \quad (178)$$

cu  $h_2$  - matricea extractoare a curentului  $i_{on}$  din vectorul de stare, deci având elemente egale cu 0, +1 sau -1, depinzând de expresia concretă a lui  $i_{on}$  în convertorul în studiu.

Examinând (177) și (178) se constată că sub aspect formal situația este identică cu cea a convertoarelor cu comandă în curent, pentru care o expresie exactă a funcției de transfer control-ieșire a fost găsită de Tymerski în [192]. Deci din acest punct deducerea decurge analog cu cea din [192], obținându-se în final:

$$\frac{\hat{v}_o}{\hat{v}_{mp}} = \frac{\mu_1}{T_s} \left\{ [(\alpha_1 \beta_3 + \alpha_3 - \zeta_1 h_1 \mu_1 \beta_1 \beta_3) \Gamma_2 \beta_2 + \alpha_2 - \zeta_2 h_2 \mu_2 \beta_2] (I - \Gamma_1 \beta_1 \beta_3 \Gamma_2 \beta_2)^{-1} \xi_1 \right\} \quad (179)$$

$$\frac{\omega}{\omega_s} \neq 0,5, 1, 1,5, \dots$$

În mod DCVM raționamentul este similar, cu observația că acum  $t_2$  este dat de anularea tensiunii  $v_{off}$ , deci de o relație de forma:

$$h_1 x(t_2) = 0 \quad (180)$$

unde  $h_1$  este acum matricea extractoare a tensiunii  $v_{off}$  din vectorul de stare (cu elemente 0 +1 și -1), iar  $t_3$  este dat de egalitatea tensiunii pe condensatorul de integrare cu tensiunea de comandă:

$$h_2 x(t_3) = v_m(t_3) \quad (181)$$

Pentru că nu există raportată o expresie exactă a funcției de transfer control ieșire pentru un convertor cu constrângeri de tipul (180) și (181), **autorul tezei** a efectuat această deducere, obținând în final următoarea expresie:

$$\frac{\hat{v}_o}{\hat{v}_{mp}} = \frac{\mu_2}{T_s} [(\alpha_3 + \alpha_1 \beta_3 + \alpha_2 \Gamma_1 \beta_1 \beta_3 - \zeta_1 \mu_1 h_1 \beta_1 \beta_3 - \zeta_2 \mu_2 h_2 \beta_2 \Gamma_1 \beta_3) (I - \Gamma_2 \beta_2 \Gamma_1 \beta_1 \beta_3)^{-1} \xi_2 + \zeta_2] \quad (182)$$

$$\frac{\omega}{\omega_s} \neq 0,5, 1, 1,5, \dots$$

De subliniat că formal *aceasta este expresia exactă* a funcției de transfer control-ieșire și pentru convertoare dc-dc cu comandă în curent funcționând în mod DCVM, la care bineînțeles matricile și vectorii extractorii au alte expresii.



### 6.8.3.7 Convertoare la care funcția de transfer control-ieșire mediată are valabilitate la orice frecvență

Ideea care a stat la baza concluziilor ce urmează a plecat de la observația că dacă se reprezintă funcția de transfer control-ieșire exactă sau mediată pentru un convertor BUCK cu comandă în factor de umplere sau cu comandă integrativă după o variabilă, se constată coincidența caracteristicilor reale și mediate. Acest lucru a sugerat **autorului tezei** o investigare analitică a fenomenului, având ca finalitate următoarea teoremă:

**Teorema 1.** Dacă într-un convertor dc-dc PWM CCM, cu întrerupătoare ideale, între matricile de stare există relațiile  $A_1 = A_2$  și  $E_1 = E_2$ , atunci funcția de transfer control-ieșire exactă și mediată coincid pentru orice tip de comandă ce nu depinde de variabilele de stare ale părții de putere din convertor.

**Demonstrație.** Fie  $A$  respectiv  $E$  valoarea comună a matricilor  $A_i$  respectiv  $E_i$ ,  $i = \overline{1,2}$ .

Argumentația teoremei se bazează pe faptul că dacă  $A_1 = A_2$  atunci mărimea  $\xi_1$  din modelul exact (vezi pentru definire relația (93b)) nu depinde de stare, fiind egală cu  $(B_1 - B_2)V_g$ , iar  $\xi_D = (A_1 - A_2)X + (B_1 - B_2)V_g$ , din modelul mediat (vezi relația (11)) nu depinde nici el de stare fiind egal tot cu  $(B_1 - B_2)V_g$ . Deci  $\xi_1 = \xi_D$  și fie  $\xi$  valoarea lor comună. Cu câteva calcule simple se arată că matricea  $A_D = DA_1 + (1 - D)A_2$  devine egală cu  $A$ . Analog, dacă  $E_1 = E_2$  atunci  $\zeta_1 = \zeta_D = (F_1 - F_2)V_g = \zeta$  și nici aceste mărimi nu depind de stare, iar  $E_D = E$ . Se cunoaște [192], [193] că la un convertor PWM CCM funcția de transfer control ieșire exactă pentru comanda în factor de umplere are expresia (notațiile de la începutul capitolului):

$$H_{ex} = (\alpha_1\beta_2 + \alpha_2)(I - \beta_1\beta_2)^{-1}\xi_1 + \zeta_1 \quad (183)$$

iar funcția de transfer control-ieșire din modelul mediat este [63]:

$$H_{med} = E_D(sI - A_D)^{-1}\xi_D + \zeta_D \quad (184)$$

Ținând cont de cele arătate mai sus cele două funcții de transfer devin:

$$H_{ex} = (\alpha_1\beta_2 + \alpha_2)(I - \beta_1\beta_2)^{-1}\xi + \zeta \quad (185)$$

$$H_{med} = E(sI - A)^{-1}\xi + \zeta \quad (186)$$

În continuare, ținând cont de relația demonstrată în [192]

$$\alpha_i = E_i(sI - A_i)^{-1}(I - \beta_i) \quad (187)$$

în care  $E_i = E$  și  $A_i = A$ , relația (185) devine:

$$\begin{aligned} H_{ex} &= [E(sI - A)^{-1}(I - \beta_1)\beta_2 + E(sI - A)^{-1}(I - \beta_1)](I - \beta_1\beta_2)^{-1}\xi + \zeta = \\ &= E(sI - A)^{-1}(\beta_2 - \beta_1\beta_2 + I - \beta_1)(I - \beta_1\beta_2)^{-1}\xi + \zeta = \\ &= E(sI - A)^{-1}(I - \beta_1\beta_2)(I - \beta_1\beta_2)^{-1}\xi + \zeta = E(sI - A)^{-1}\xi + \zeta = H_{med} \end{aligned} \quad (188)$$

ceea ce finalizează demonstrația egalității funcțiilor de transfer control-ieșire exactă și mediată cu comandă în factor de umplere.

Specificându-se că tehnica de comandă nu depinde de stare, rezultă că comanda va depinde numai de tensiunea (tensiunile) de alimentare, care, întrucât în discuție sunt funcțiile de transfer control-ieșire, au perturbații nule, adică  $\hat{v}_g = 0$ . Deci  $\hat{d}$  nu va depinde de perturbații ale mărimilor de stare din convertor.

Din egalitatea funcțiilor de transfer control-ieșire exactă și mediată cu comandă în factor de umplere și din faptul că  $\hat{d}$  nu depinde de perturbații ale mărimilor de stare din convertor rezultă concluzia teoremei.

O consecință importantă a acestei teoreme este că toate convertoarele BUCK CCM și cele derivate din ele (FORWARD, PUNTE, SEMIPUNTE) cu comandă în factor de umplere, cu

comandă integrativă după o variabilă sau cu reacție feedforward au funcția de transfer control-ieșire dată de modelul mediat adevărată pentru orice frecvență, deci fără a mai fi restricționată ca valabilitate la jumătate din frecvența de comutație. Cum modelul mediat furnizează funcția de transfer control-ieșire în formă rațională, studiul stabilității este rezolvat pentru orice frecvență, ceea ce este extrem de folositor. Deși convertoarele reale nu au întrerupătoare ideale, acestea nu modifică substanțial funcția de transfer, deci consecința de mai sus se menține și în cazul practic până la frecvențe la care elementele parazite nu devin importante.

Structura unui sistem luat în discuție în teorema 1 este prezentată în Fig. 6.65. Se observă că toate întrerupătoarele conectează numai tensiuni de alimentare la un circuit liniar. Toate elementele liniare se pot separa de rețeaua de întrerupătoare, iar controllerul furnizează un factor de umplere depinzând numai de tensiunea de comandă (ca la comanda în factor de umplere) și eventual de tensiunile de alimentare (ca la feedforward sau comanda integrativă după o variabilă). O astfel de schemă și teorema 1 constituie o generalizare a celor enunțate de Kassakian ș.a. în [63, pag. 267] unde este prezentat un sistem la care reacția de feedforward asigură și eroare dinamică nulă.

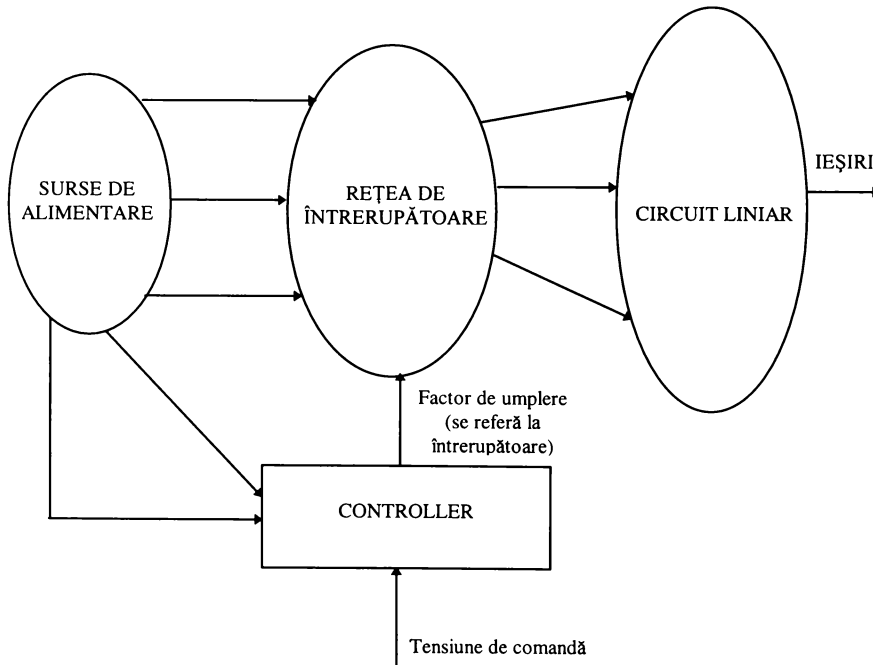


Fig. 6.65. Schema bloc a unui convertor și a comenzii pentru care funcțiile de transfer control-ieșire exactă și mediată coincid

## 6.9 Concluzii

Contribuțiile autorului tezei în domeniul modelării convertoarelor dc-dc se referă la:

- o sistematizare a principalelor tehnici de modelare raportate în literatură, cu sublinierea avantajelor și limitărilor fiecăreia (paragraful 6.1).
- aplicarea modelelor canonice generalizate în rezolvarea de o manieră generală a problemei filtrului de intrare, adică a dimensionării unui filtru trece-jos plasat după sursa de alimentare,

care îmbunătățește proprietățile de EMI ale convertorului dar nu modifică principalele funcții de transfer. Soluția propusă cuprinde convertoarele QRC FW (paragraful 6.2)

- aplicarea metodei generale de mediere a lui Sun și Grotstollen în elaborarea de modele unificate, atât de semnal mare cât și de semnal mic pentru convertoarele rezonante serie (paragrafele 6.3.1 și 6.3.2).
- elaborarea (paragraful 6.3.2) a unui program interactiv de proiectare asistată a convertoarelor rezonante serie,
- dezvoltarea unui model general de semnal mic în spațiul stărilor pentru comanda prin sarcină (paragraful 6.4), cu avantajul unui studiu rapid al stabilității și al determinării caracteristicilor statice.
- un model îmbunătățit pentru comanda integrativă după o variabilă, față de modelul propus de Smedley și Ćuk (paragraful 6.5). Modelul este prezentat sub forma unei scheme bloc ținând cont de efectul de eșantionare. Pe baza lui se poate efectua un studiu mai exact al stabilității acestor convertoare.
- elaborarea de noi modele mediate, generale și unitare, de semnal mic și de semnal mare, pentru convertoare QRC cu aplicații în simularea accelerată și studiul stabilității (paragraful 6.6).
- două noi modele discrete de semnal mic (paragraful 6.7), bazate pe integrarea numerică backward Euler respectiv pe integrarea trapezoidală, mai performante decât cel propus de Kassakian în [63]. Sunt oferite formule generale ce descriu comportarea circuitului, care pot fi ușor asimilate în programe de analiză.
- un algoritm sistematic și general de deducere a funcției de transfer control-ieșire bazat pe procedeul propus de Tymerski în [193] (paragraful 6.8.1).
- dezvoltarea unui algoritm de calcul exact al stării staționare bazat pe metoda Newton-Raphson vectorială și propunerea ca punct de pornire în algoritm a soluției furnizate de modelele mediate (paragraful 6.8.2.1).
- legarea unui program de calcul matematic (MATLAB) cu un simulator de circuite de uz general (PETS sau PSpice) (paragraful 6.8.2.2). Demonstrarea modului de aplicare a acestui tandem în:
  - calculul și analiza accelerată a stării staționare în convertoare autonome și neautonome, exemplificarea făcându-se utilizând trei metode: iterație cu punct fix, metodă de tip Newton cu diferențe finite și metodă de tip Newton cu actualizare Broyden. Este efectuat un studiu comparativ între metode pe un exemplu clasic din literatură propus de Wong [214]. Sunt făcute câteva considerații privind particularitățile în utilizarea programului Pspice împreună cu programul MATLAB.
  - determinarea caracteristicilor statice reale și a familiilor de forme de undă în stare staționară în prezența elementelor parazite și a unor modele mai apropiate de realitate ale dispozitivelor, fără a mai fi necesară o cunoaștere prealabilă a funcționării. Analiza este aplicabilă unui convertor oricât de complex.
  - simularea exactă a convertoarelor complexe cu controllere mai sofisticate: controllere digitale, fuzzy sau neuronale. Utilizatorul nu trebuie să posede nici un fel de informație prealabilă despre convertor, ci trebuie doar să introducă schema de conexiuni a acestuia și modul de funcționare al controller-ului.
- deducerea expresiei exacte (fără valabilitate limitată în frecvență) și în formă închisă a funcției de transfer control-ieșire de semnal mic pentru următoarele tipuri de convertoare (paragraful 6.8.3):
  - convertoare PWM funcționând în modul DCVM cu comandă în factor de umplere (paragraful 6.8.3.1).
  - convertoare DIVM (paragraful 6.8.3.2)
  - convertoare PWM cu feedforward realizat cu integratoare resetabile sau cu multiplicatoare rapide. Deducerea unei expresii unitare a funcției de transfer control-ieșire exacte pentru cele două variante de realizare a reacției feedforward (paragraful 6.8.3.3).

- convertoare PWM CCM cu comanda prin sarcină (paragraful 6.8.3.4).
- deducerea unei expresii generale pentru funcția de transfer control-ieșire exactă pentru orice tip de comandă integrativă la convertoare PWM funcționând în CCM (paragraful 6.8.3.5) cu exemplificare pe un convertor cu comandă mixtă.
- deducerea unei expresii generale pentru funcția de transfer control-ieșire exactă pentru orice tip de comandă integrativă la convertoare PWM funcționând în moduri discontinue (DICM sau DCVM) (paragraful 6.8.3.6).
- enunțarea unei teoreme (paragraful 6.8.3.7) privind structura convertoarelor la care funcția de transfer control-ieșire furnizată de modele mediate este identică cu funcția de transfer control-ieșire exactă și menționarea aplicațiilor acestei teoreme pentru convertoare BUCK și a celor derivate din acesta.

Domeniul modelării convertoarelor este unul din cele mai fertile și dinamice. Orice prognoză de evoluție în această arie are riscul să fie incompletă. Sub această rezervă autorul preconizează totuși că posibilele direcții de cercetare pentru viitor ar putea avea în vedere:

- rezolvarea problemei filtrului de intrare în cazul convertoarelor QRC HW pe baza modelelor canonice generalizate,
- aplicarea metodei generale de mediere a lui Sun și Grotstollen în deducerea de modele mediate de semnal mare și de semnal mic pentru convertoare paralel, convertoare serie pe mod limitat, convertoare cu comandă în fază și în general pentru orice tip de convertor rezonant cu funcționare la frecvență constantă, simultan cu elaborarea de programe de proiectare asistată pentru aceste convertoare,
- dezvoltarea de modele unificate în spațiul stărilor pentru orice tip de comandă integrativă cu luarea în considerație a eventualelor efecte de eșantionare,
- integrarea programelor de calcul matematic cu simulatoarele de circuite. De altfel primii pași în acest sens au și fost făcuți. Un exemplu elocvent în acest sens este blockset-ul Power System scris pentru mediul SIMULINK, odată cu versiunea 5 a programului MATLAB. Deși i s-ar putea imputa destule lacune, totuși ideea este una strălucită și cu siguranță va fi dezvoltată și impusă în scurt timp de firma MathWorks.
- în privința altor posibilități de utilizare a legăturii dintre un program de calcul matematic cu un simulator de circuite se întrevăd aplicații legate de:
  - studierea eficacității sub aspectul numărului de iterații și numărului de simulări a altor modele afine utilizate pentru calculul exact al stării staționare.
  - obținerea curbelor de separație reale între diferitele moduri de funcționare pentru convertoare la care se consideră elementele parazite și se modelează dispozitivele prin modele apropiate de realitate,
  - metode de extragere a modelelor dinamice ale sistemului,
  - automatizarea la maximum a programului, cu prezentare sub forma unui mediu integrat, cu interfețe grafice utilizator,
  - o analiză exactă de semnal mic
- obținerea de expresii exacte pentru funcțiile de transfer control-ieșire ale convertoarelor rezonante și cvasirezonante cu funcționare la frecvență constantă,
- un studiu calitativ, urmat de o generalizare a expresiilor principalelor categorii de funcții de transfer exacte, privite prin prisma numărului de stări topologice și a relațiilor de constrângere ce definesc momentele de comutație,
- autorul crede că ar fi o realizare deosebită aceea ca, plecând de la expresiile în formă închisă ale funcției de transfer control-ieșire exacte, prin metode de dezvoltare în serie să se obțină expresiile funcției de transfer mediate *în formă rațională*. Primele rezultate în acest sens au fost obținute de Tymerski pentru convertoare PWM CCM cu comandă în factor de umplere la care au fost obținute exact formulele rezultate prin aplicarea modelului mediat în spațiul stărilor. Pentru alte tipuri de convertoare însă nu s-au raportat astfel de rezultate Ar fi de dorit un

algoritm sau enunțarea unor reguli universale de obținere a expresiei funcției de transfer control-ieșire mediate din cea exactă. Avantajul ar fi imediat, pentru că s-ar ușura mult studiul stabilității și proiectarea controllerului pentru convertoarele respective.

- autorul întrevide o direcție deosebit de importantă de cercetare aceea a studierii posibilității de obținere a funcției de transfer control-ieșire exacte pentru convertoare cu comandă în frecvență și enunțarea unui algoritm de deducere sistematică a sa.

## CAPITOLUL 7

### Aplicații ale convertoarelor PWM și QRC în circuite de corecție a factorului de putere

Acest capitol își propune investigarea posibilității de utilizare a convertoarelor cvasirezonante ca și circuite pentru corecția factorului de putere la alimentare de la rețeaua de 50 Hz, precum și aplicații ale convertoarelor DIVM introduse în paragraful 4.2.3, în același scop. Ambele studii au ca obiectiv principal miniaturizarea acestor preregulate, datorită utilizării de convertoare cu pierderi de comutație reduse. Creșterea frecvenței de comutație permite astfel utilizarea lor și în magistrale de curent alternativ cu frecvență mai mare (de exemplu 400 Hz, în aeronautică).

#### 7.1 Utilizarea convertoarelor cvasirezonante ca circuite pentru corecția factorului de putere. Analiză și proiectare

Se știe că structurile tradiționale de circuite pentru corecția factorului de putere (“power factor correctors” - PFC) au la bază în primul rând convertorul PWM de tip BOOST, dar și convertoare dc-dc PWM de tip BUCK-BOOST, FLYBACK, SEPIC și ĆUK [37], [218], [219], [220]. Toate aceste convertoare PWM sunt utile până la frecvențe de comutație,  $f_s$ , de aproximativ 100-200 kHz. Dacă însă se dorește creșterea frecvenței peste 200 kHz, în vederea minimizării dimensiunilor și greutății, atunci convertoarele PWM nu mai satisfac condițiile de randament și disipație. Ele se înlocuiesc cu convertoare în clasă E, convertoare complet rezonante (serie sau paralel) sau convertoare cvasirezonante (QRC). Domeniul frecvențelor de comutație până la care se pot utiliza convertoarele ZCS QRC este de până la 1-2 MHz.

În cele ce urmează se prezintă un studiu detaliat, publicat de autor în [95], legat de utilizarea convertoarelor cvasirezonante cu comutare la curent zero (ZCS QRC) ca circuite pentru corecția factorului de putere. Se urmărește modul de variație al frecvenței de comutație în decursul unei semiperioade a tensiunii alternative de alimentare la funcționarea în mod semiundă și undă plină, pentru diverse topologii de convertoare. Apoi, pe baza acestei dependențe, se realizează proiectarea optimală și se justifică alegerea convertorului și a modului său de funcționare. Pentru început se fac două considerații generale [39] privind convertoarele dc-dc (indiferent de tip) folosite ca PFC:

- 1) Raportul de conversie al convertorului, definit ca raportul dintre tensiunea continuă de ieșire și cea alternativă de intrare este:

$$m = \frac{v_o}{v_g} = \frac{V_o}{V_g |\sin \omega t|} = \frac{M_{\min}}{|\sin \omega t|} \quad (1)$$

unde  $V_g$  este amplitudinea tensiunii rețelei,  $M_{\min} = \frac{V_o}{V_g}$ , iar  $\omega$  pulsația tensiunii

alternative de intrare (a rețelei). Se observă că raportul de conversie este variabil, modificându-se între o valoare minimă egală, cu  $M_{\min}$  și infinit.

- 2) Sarcina “văzută” de convertor în ipoteza unei tensiuni constante la ieșire se poate calcula [39] ținând cont de expresia curentului  $i_o$  injectat spre grupul condensatorului de filtrație de la ieșire paralel cu sarcina,  $C_o - R$  - Fig. 7.1 - menținând ipoteza constanței tensiunii de ieșire. Se obține astfel:

$$r = \frac{R}{2 \sin^2 \omega t} = \frac{r_{\min}}{\sin^2 \omega t} \quad (2)$$

Deci, deși sarcina rezistivă conectată la ieșire este constantă, sarcina “văzută” de convertor este rezistivă, dar variabilă în timp și având o valoare minimă  $r_{\min} = \frac{R}{2}$  și o valoare maximă infinită.

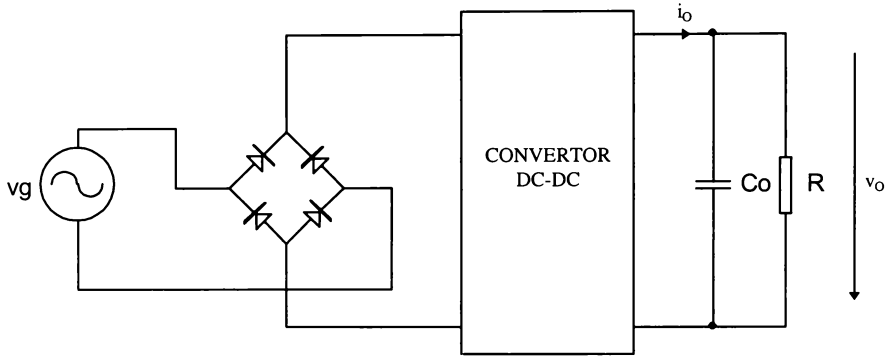


Fig. 7.1. Schema bloc a unui circuit PFC cu convertor dc-dc în comutație.

Analiza care urmează se va face cu *mărimi normalizate* pentru a i se conferi un caracter de *generalitate*. Notațiile sunt cele clasice de la convertoare cvasirezonaante ( $f_N = \frac{f_s}{f_0}$ ,  $Z_0 = \sqrt{\frac{L_r}{C_r}}$ ), singurele notații suplimentare fiind  $n_T$  pentru raportul de transformare presupus  $1:n_T$  (pentru a-l deosebi de  $n$ , al doilea argument al funcției cvasirezonaante) și cea pentru sarcina normalizată văzută de convertorul ZCS QRC:

$$r_N = \frac{r}{Z_0} \quad (3)$$

$r_N$ ,  $f_N$  și  $m$  sunt legate între ele prin relații care depind de topologia folosită (BUCK, BOOST, BUCK-BOOST, ĆUK, SEPIC, FORWARD, FLYBACK, etc.) și de modul de funcționare (HW sau FW) în care funcționează convertorul (deci de tipul de întrerupător utilizat). Aceste relații sunt:

$$f_N = 2\pi \frac{\mu(m, n_T)}{q(\alpha, n)} \quad (4)$$

unde  $q(x, n)$  este funcția cvasirezonaantă, iar  $\alpha$  raportul adimensional (numit de unii autori și “curent normalizat”)

$$\alpha = \frac{mn_T}{r_N} \quad (5)$$

Funcția  $\mu$  depinde de  $m$  și  $n_T$  în funcție de convertorul utilizat conform Tab. 7.1.

De reținut că valoarea lui  $\alpha$  trebuie să fie totdeauna subunitară, pentru că aceasta este condiția de comutare la curent zero a convertorului. De asemenea în modul cu undă plină  $q(\alpha, 2) \equiv 2\pi$ , ceea ce face ca în acest caz

$$f_{N \text{ FW}} \equiv \mu(m, n_T) \quad (6)$$

Tab 7.1.  $\mu(m)$  pentru principalele convertoare ZCS QRC cu un singur tranzistor de putere.

CONVERTORUL	$\mu(m, n_T)$
FORWARD (BUCK, $n_T = 1$ )	$\frac{m}{n_T}$
BOOST	$\frac{m-1}{m}$
FLYBACK, ĆUK, și SEPIC ( $n_T = 1$ dacă nu există transformator)	$\frac{m}{m+n_T}$

Deci în modul FW raportul de conversie  $m$  al tensiunilor este practic independent de sarcina normalizată,  $r_N$ . În schimb modul HW  $q(\alpha, l)$  depinde puternic de  $\alpha$ .

Din (1), (2), (3) și (4) se obține:

$$\alpha = \alpha_{\max} |\sin \omega t| \tag{7}$$

$$f_N(\omega t) = 2\pi \frac{\mu(m(\omega t), n_T)}{q(\alpha, n)} = 2\pi \frac{\mu\left(\frac{M_{\min}}{|\sin \omega t|}, n_T\right)}{q(\alpha_{\max} |\sin \omega t|, n)} \tag{8}$$

unde

$$\alpha_{\max} = \frac{2M_{\min} Z_0 n_T}{R} \tag{9}$$

Literatura [60], [110] precizează că proiectarea optimă a convertoarelor ZCS QRC se face impunând ca  $Z_0$  să fie aleasă astfel încât curentul de vârf prin tranzistorul comutator să fie minim. Această condiție are loc dacă  $\alpha$  are valoarea 1, atunci când curentul comutat de întrerupătorul rezonant este maxim (la tensiune de intrare minimă și putere maximă la ieșire).

În cazul în care convertorul ZCS QRC este folosit într-un PFC, conform relației (6)  $\alpha$  este maxim când  $\omega t = \frac{\pi}{2}$ , deci valoarea lui  $\alpha$  se va impune să fie 1 la  $\omega t = \frac{\pi}{2}$ :

ceea ce implică

$$\alpha_{\max} = 1 \tag{10}$$

Înlocuind (10) în (8) se obține *legea de variație temporală a frecvenței de comutație*:

$$f_N(\omega t) = 2\pi \frac{\mu\left(\frac{M_{\min}}{|\sin \omega t|}, n_T\right)}{q(|\sin \omega t|, n)} \tag{11}$$

Plecând de la relația (11) în Fig. 7.2 s-a reprezentat pentru convertorul BOOST dependența frecvenței normalizate de  $\omega t$ , pentru diferite valori ale mărimii  $M_{\min}$ , atât pentru funcționarea în mod semiundă (reprezentare cu linii continue) cât și la funcționarea cu undă plină (reprezentare cu linii punctate). Aceeași dependență s-a reprezentat în Fig. 7.3 pentru convertoarele BUCK-BOOST, SEPIC și ĆUK. Evident, toate aceste reprezentări admit că rezistența de sarcină este constantă.

De cele mai multe ori însă *sarcina este variabilă*, ea variind între două valori, minimă și maximă,  $R_{\min}$ ,  $R_{\max}$ . Putem scrie acest lucru analitic în forma:

$$R = \lambda R_{\min} \tag{12}$$

unde

$$\lambda \in [1, \lambda_{\max}], \quad \lambda_{\max} \geq 1 \tag{13}$$

Proiectarea optimă ( $\alpha_{\max} = 1$ ) se impune la  $R_{\min}$ , ceea ce face ca:

$$\alpha_{\max} = \frac{1}{\lambda} \tag{14}$$

și prin substituirea relației (14) în (8) se obține:



$$f_N(\omega t) = 2\pi \frac{\mu\left(\frac{M_{\min}}{|\sin \omega t|}, n_T\right)}{q\left(\frac{\lambda}{\lambda}, n\right)} \quad (15)$$

În Fig. 7.4 este reprezentată dependența frecvenței normalizate funcție de  $\omega t$  pentru diferite valori ale mărimii  $\lambda$ , la  $M_{\min}$  fixat, atât pentru funcționarea în mod semiundă cât și la funcționarea cu undă plină. Reprezentările se referă la convertorul BOOST. Aceași dependență s-a reprezentat în figura 7.5 pentru convertoarele BUCK-BOOST, SEPIC și ĆUK.

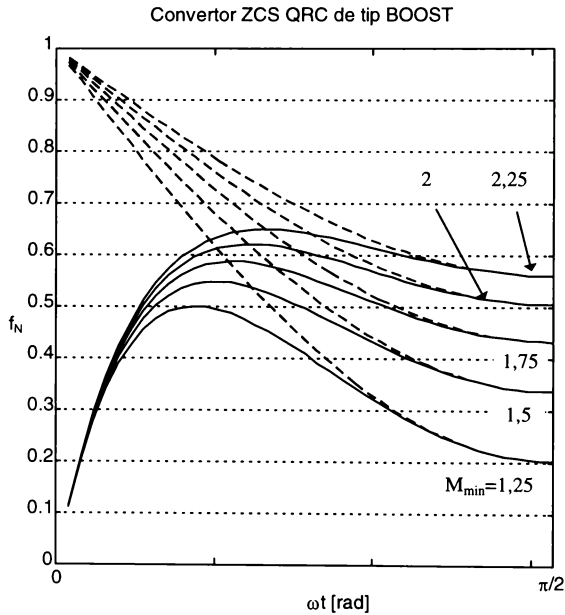


Fig. 7.2. Linia continuă - HW; linia întreruptă - FW.

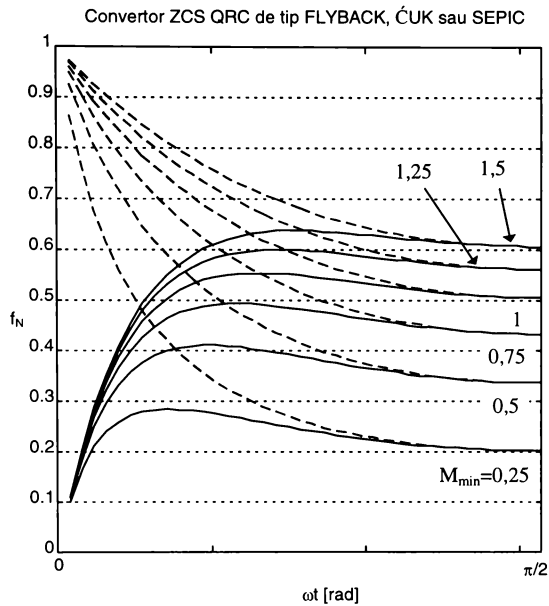


Fig. 7.3. Linia continuă - HW; linia întreruptă - FW.

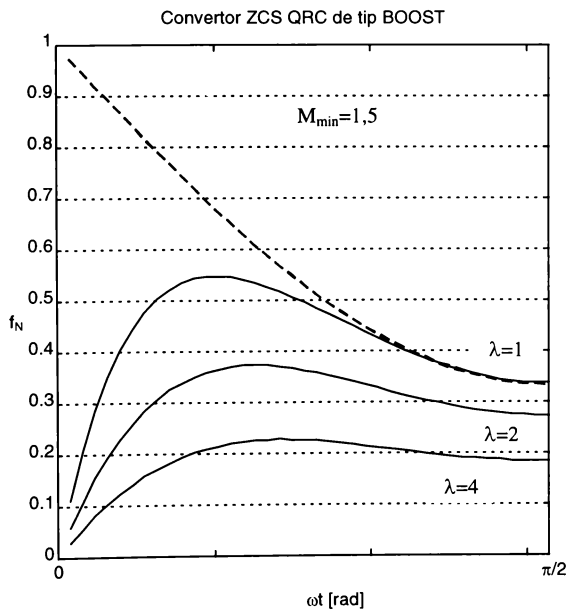
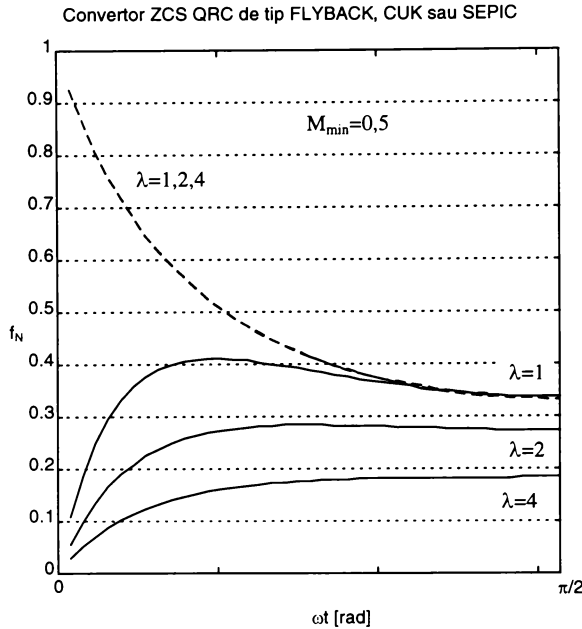


Fig. 7.4. Linia continuă - HW; linia întreruptă - FW.



Se observă că datorită apariției funcției modul în (11) și (15) reprezentarea este suficientă pentru un sfert de perioadă a rețelei, sau, în unități de fază, de la 0 la  $\frac{\pi}{2}$ .

În toate cazurile reprezentate în figurile de mai sus s-au considerat convertoare *fără transformator de izolare* ( $n_T = 1$ ). Analiza acestor reprezentări relevă următoarele concluzii:

- a) Frecvența de comutație normalizată variază pe un domeniu larg atunci când  $\omega t$  se modifică între 0 și  $\frac{\pi}{2}$ .
- b) Dependența frecvenței normalizate de  $\omega t$  pentru modul semiundă și cel cu undă plină este aproape identică în domeniul  $\omega t \in \left[ \frac{\pi}{4}, \frac{\pi}{2} \right]$  și foarte diferită pentru  $\omega t \rightarrow 0$ .
- c) Pentru a evita frecvențe de comutație foarte mici (în cazul funcționării în mod semiundă) sau foarte mari (în cazul funcționării cu undă plină) se poate introduce un mic *time mort* la începutul și la sfârșitul unei semiperioade a tensiunii rețelei, pe ideea dezvoltată în [220]. Acest timp mort face ca și teoretic factorul de putere să nu mai fie unitar.
- d) În cazul în care sarcina este constantă sau se modifică pe un domeniu restrâns este indicat a se utiliza modul semiundă din două rațiuni:
  - frecvența de comutație suferă modificări într-un domeniu mai restrâns în modul semiundă față de modul cu undă plină (vezi Fig. 7.2 și Fig. 7.3).
  - modul semiundă are un randament mai bun decât cel cu undă plină.
- e) Dacă sarcina are o variație pe un domeniu larg ( $\lambda_{\max}$  mult mai mare ca 1) utilizarea modului semiundă nu este cea mai fericită soluție, deoarece frecvența de comutație suferă modificări mari. În această situație se va lucra în modul cu undă plină, alegere justificată de faptul că domeniul de variație al frecvenței de comutație în acest mod nu depinde de  $\lambda$ .

Dacă pentru convertorul ZCS QRC folosit se adoptă *varianta cu transformator de izolare*, atunci prezența acestuia introduce un grad de libertate suplimentar în proiectare prin variabila  $n_T$ . Deci valoarea lui  $M_{\min}$  în punctul  $\frac{\pi}{2}$  nu mai fixează funcționarea convertorului deoarece  $n_T$  poate fi ales arbitrar. Pentru o proiectare univocă este necesară impunerea unei condiții suplimentare. De obicei aceasta este dată prin impunerea valorii frecvenței normalizate,  $f_{NP}$ , pentru tensiune de intrare maximă, deci pentru  $\omega t = \frac{\pi}{2}$ .

Presupunând proiectare optimală ( $\alpha_{\max} = 1$  când  $\lambda = 1$ ) din (15) și folosind definiția frecvenței normalizate pentru  $\omega t = \frac{\pi}{2}$  se obține:

$$f_{NP} = 2\pi \frac{\mu(M_{\min}, n_T)}{q(1, n)} \tag{16}$$

Prin simplă înlocuire se poate arăta că atât în modul semiundă cât și în cel cu undă plină:

$$q(1, n) = \frac{3(\pi + 1)}{2} \quad \forall n \in \{1, 2\} \tag{17}$$

Ecuțiile (16) și (17) conduc la:

$$f_{NP} = \frac{4\pi}{3(\pi + 1)} \mu(M_{\min}, n_T) = 1,01 \mu(M_{\min}, n_T) \equiv \mu(M_{\min}, n_T) \tag{18}$$

Din ecuația (18) se poate exprima  $M_{\min}$  în funcție de  $f_{NP}$  și  $n_T$  și de tipul de convertor, ca în Tab. 7.2, deci:

$$M_{\min} = M_{\min}(f_{NP}, n_T) \tag{19}$$

Înlocuind din (19) pe  $M_{\min}$  în (15) și folosind Tab. 7.1 obținem:

Tab. 7.2. Valoarea lui  $M_{\min}$  funcție de  $f_{NP}$  și  $n_T$  pentru convertoarele cu transformator de izolare și un singur tranzistor de putere

CONVERTORUL cu transformator de izolare	$M_{\min}(f_{NP}, n_T)$
BUCK	$n_T f_{NP}$
FLYBACK, ĆUK și SEPIC	$n_T \frac{f_{NP}}{1 - f_{NP}}$

$$f_N = 2\pi \frac{\mu\left(\frac{M_{\min}(f_{NP}, n_T)}{|\sin \omega t|}, n_T\right)}{q\left(\frac{|\sin \omega t|}{\lambda}, n\right)} = \begin{cases} 2\pi \cdot \frac{f_{NP}}{|\sin \omega t|} \cdot \frac{1}{q\left(\frac{|\sin \omega t|}{\lambda}, n\right)} & \text{pt. FORWARD} \\ 2\pi \cdot \frac{f_{NP}}{f_{NP} + (1 - f_{NP})|\sin \omega t|} \cdot \frac{1}{q\left(\frac{|\sin \omega t|}{\lambda}, n\right)} & \text{pt. FLYBACK,} \\ & \text{ĆUK și SEPIC} \end{cases} \tag{20}$$

Fig. 7.6 reprezintă dependența frecvenței normalizate, dată de (20), în funcție de  $\omega t$  pentru  $\lambda = 1$ , cu  $f_{NP}$  parametru, în cazul convertoarelor FLYBACK, ĆUK și SEPIC. Fig. 7.7 reprezintă aceeași dependență, pentru aceleași convertoare, dar cu  $f_{NP}$  fixat și  $\lambda$  parametru.

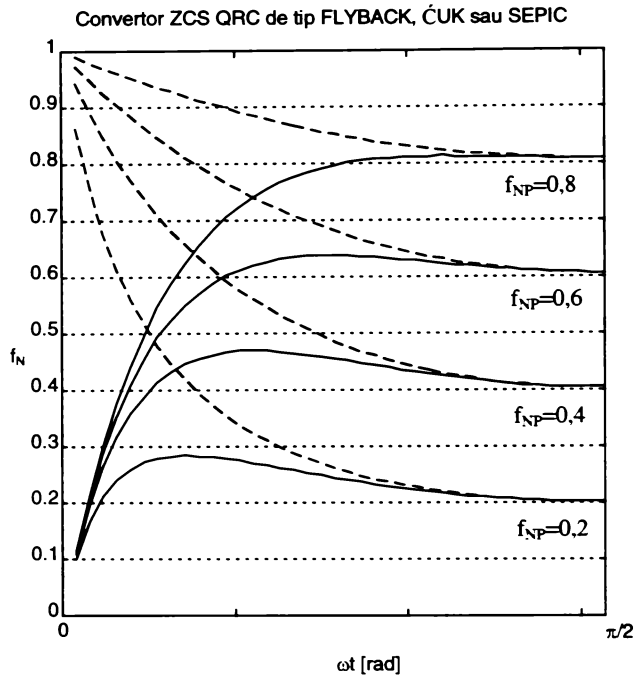


Fig. 7.6. Linia continuă - HW; linia întreruptă - FW.

De reținut că în acest caz  $f_{NP}$  se poate alege independent de  $M_{\min}$ , optând pentru o valoare potrivită pentru  $n_r$ , în vederea optimizării solicitărilor în curent sau tensiune ale tranzistorului, diodei de putere sau a domeniului frecvenței de comutație. Astfel, se știe că în convertoarele ZCS QRC de tip FLYBACK, ĆUK și SEPIC valori ridicate ale frecvenței normalizate determină solicitări importante din punct de vedere al tensiunii pe tranzistor și al curentului prin diodă, în timp ce o valoare mică a frecvenței normalizate duce la solicitări semnificative în curent pentru tranzistorul de putere și în tensiune pentru diodă. Toate aceste valori au maximum la  $\omega t = \frac{\pi}{2}$ .

Aceste aspecte determină următoarele considerații de proiectare:

- a) dacă  $\lambda$  este apropiat de 1, cea mai bună alegere este un convertor funcționând în mod semiundă și o cât mai redusă valoare a lui  $f_{NP}$ , deoarece în acest caz se minimizează domeniul frecvenței de comutație.
- b) dacă  $\lambda$  nu este apropiat de 1 se va alege un convertor funcționând cu undă plină, fiind necesar un compromis între domeniul frecvenței de comutație și energia reactivă vehiculată prin întrerupător.

Toate aceste relații de proiectare au fost implementate în programe de proiectare asistată în MATLAB, cu care de altfel s-au trasat și graficele de mai sus. Aceste programe sunt date în Anexa A 31.

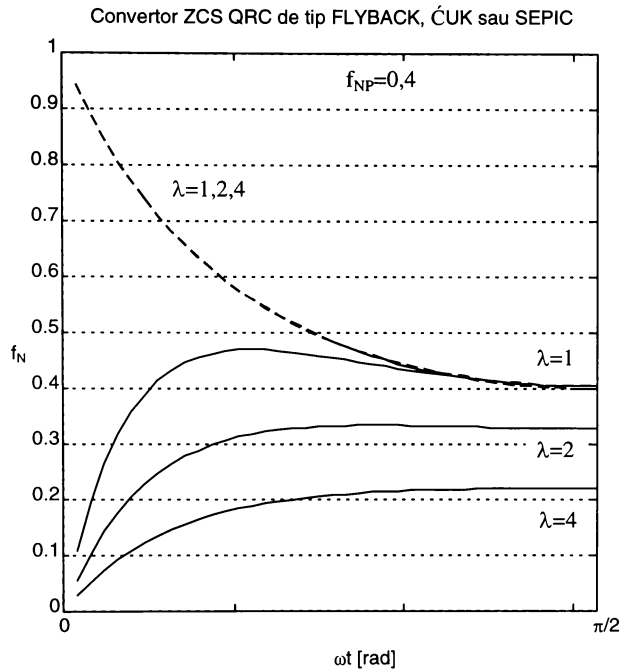


Fig. 7.7. Linia continuă - HW; linia întreruptă - FW.

### 7.2. Funcționarea convertoarelor DIVM ca circuite de corecție a factorului de putere

Pentru convertoarele DIVM introduse în paragraful 4.2.3, ale căror notații vor fi utilizate în continuare, se arată [Lascu, 53] că acestea pot fi utilizate ca și circuite PFC. Un model de semnal mare propus de autorul tezei în [53] este cel din Fig. 7.8. La funcționare cu factor de umplere constant,  $D$ , și cu  $v_g$  lent variabilă, rezistența  $R_i$  "văzută" de tensiunea  $\bar{v}_{ap}$  este:

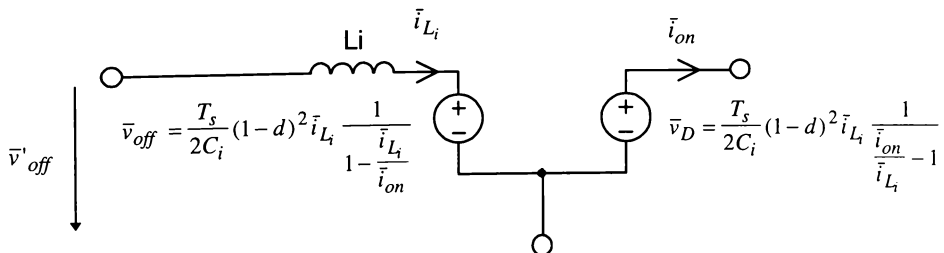


Fig. 7.8. Modelul de semnal mare al celulei DIVM.

$$R_i = \frac{\bar{v}_{off}}{i_{L_i}} = \frac{T_s}{2C_i} (1-D)^2 \frac{1}{1 - \frac{\bar{i}_{L_i}}{\bar{i}_{on}}} \quad (21)$$

Se poate observa că această rezistență este neliniară. Dacă se consideră topologii în care  $v_g$ ,  $L_i$  și  $\bar{v}_{off}$  formează o buclă (de exemplu în convertoarele BUCK și ZETA) dacă:

$$\bar{i}_{L_i} \ll \bar{i}_{on} \quad (22)$$

rezistența  $R_i$  poate fi considerată constantă, valoarea ei fiind:

$$R_i = \frac{T_s}{2C_i} (1-D)^2 \quad (23)$$

Înlocuind pe  $C_i$  din relația (54) (paragraful 4.2.3) se obține:

$$R_i = R \frac{(1-D)^2}{k_v} \quad (24)$$

Se poate arăta că relația (22) se poate scrie în forma echivalentă:

$$k_v \ll M(1-D)^2 \quad (25)$$

Suplimentar, condiția de funcționare DIVM,  $\mu \leq D$ , se poate scrie și în forma:

$$k_v \leq MD(1-D) \quad (26)$$

De obicei (25) este mai puternică decât (26). Tensiunea medie de ieșire,  $V_o$  și pulsațiile ei  $\Delta V_o$  se calculează după regulile din [39] obținându-se valorile:

$$V_o = V_g \max \sqrt{\frac{k_v}{2(1-D)^2}} \cos \varphi \quad (27)$$

$$\Delta V_o = \frac{V_g \max}{4\omega C_o R} \frac{\sqrt{2k}}{1-D} \quad (28)$$

unde

$$\operatorname{tg} \varphi = \frac{\omega L_i}{R_i} = \frac{\omega L_i}{R} \frac{k}{(1-D)^2} \quad (29)$$

Din (26) și (27) se obține:

$$V_o \ll \frac{1}{2} V_g \max \quad (30)$$

Această ultimă relație arată că, atunci când un convertor DIVM este folosit ca circuit PFC, este necesar ca aplicația să necesite o tensiune de ieșire cu mult mai mică decât tensiunea de alimentare. Concluzia este deci că, spre deosebire de circuitele PFC bazate pe convertoare BOOST, circuitele PFC cu convertoare DIVM se vor utiliza pentru tensiuni de ieșire mici.

A fost investigată o aplicație cu circuit PFC construit în jurul convertorului ZETA. Parametrii convertorului sunt:

$$V_g \max = 310 \text{ V}; \quad \omega = 2\pi \times 50 \text{ Hz}; \quad L_i = 15 \text{ mH}; \quad C_i = 7.47 \text{ nF}; \quad L_1 = L_2 = 500 \text{ } \mu\text{H}; \quad C_1 = 100 \text{ } \mu\text{F}; \\ C_o = 30 \text{ mF}; \quad R = 2 \text{ } \Omega; \quad f_s = 100 \text{ kHz}; \quad D = 0,4; \quad V_o = 20 \text{ V}; \quad P_o = 200 \text{ W}$$

Rezultatele simulării sunt prezentate în Fig. 7.9 pentru tensiunea rețelei și curenul absorbit din rețea și în Fig. 7.10 pentru spectrul curenului absorbit din rețea. Figurile evidențiază un factor de defazaj de calitate și puritatea spectrală a curenului. Programul de simulare este dat în Anexa A32.

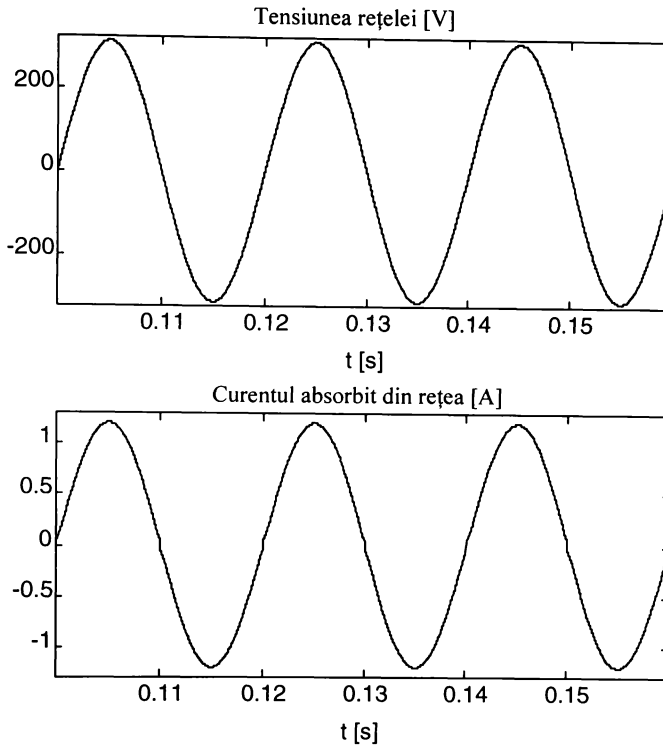


Fig. 7.9. Formele de undă pentru tensiunea rețelei și curentul absorbit din rețea pentru un circuit PFC cu convertor PWM DIVM de tip ZETA .  
(Simulator PSIM - Powersim Technologies Inc.).

### 7.3. Concluzii

Domeniul circuitelor de corecție a factorului de putere este astăzi unul “la modă” în electronica de putere. Constrânși de normativele severe ale ultimilor ani, producătorii continuă să introducă pe piață noi circuite de comandă pentru PFC, din ce în ce mai performante și cu prețuri în continuă scădere, ceea ce spune mult despre nivelul cererii. Este firesc ca și cercetătorii să își canalizeze eforturile în această direcție care oferă o serie de beneficii sub aspectul “poluării armonice” dar și financiare. Nu sunt puține cazurile în care nume de prestigiu în domeniul simulării în electronica de putere au abandonat această arie de preocupări în favoarea celei a circuitelor PFC. Capitolul de mai sus a fost conceput ca să prezinte două aplicații ale convertoarelor investigate în capitolele precedente în domeniul PFC. Acestea sunt de altfel și contribuțiile originale ale autorului în acest capitol:

- aplicarea convertoarelor ZCS QRC ca și circuite de corecție a factorului de putere. Ideea utilizării acestui tip de convertor a fost vehiculată în literatură, autorul preluând-o și efectuând o analiză cantitativă și trăgând câteva concluzii de proiectare.
- demonstrarea posibilității de a utiliza convertoarele DIVM ca și circuite PFC la tensiuni de ieșire mici în raport cu amplitudinea tensiunii alternative de alimentare. Această aplicație este originală.



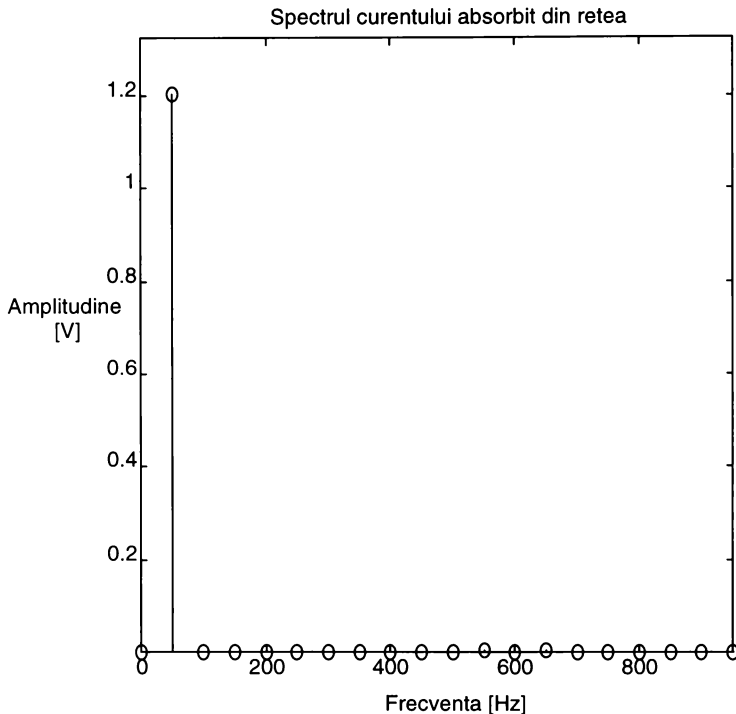


Fig. 7.10. Spectrul curentului absorbit din rețea.

Ca și subiecte de cercetare în viitor, la fel ca în cazul modelării, orice prognoză riscă să fie incompletă. Din literatura parcursă se amintesc totuși următoarele direcții generale:

- modelarea la semnal mare și la semnal mic a circuitelor PFC realizate cu diferite tipuri de convertoare (PWM, QRC sau rezonante), cu avantaje privind solicitările în curent sau tensiune ale dispozitivelor sau cu pierderi de comutație reduse. Se vizează atât domeniul monofazat cât și trifazat, în special la puteri mari;
- îmbunătățirea dinamicii la circuitelor PFC utilizând tehnici de comandă adecvate;
- noi structuri de circuite PFC, bazate pe convertoare cu pulsații de curent de intrare (și eventual și de ieșire) reduse, utilizând tehnici de comandă care să elimine necesitatea măsurării tensiunii și curentului de intrare, și stabile necondiționat [58], [37]. Comanda cu purtătoare neliniară [218], [219] pare o direcție promițătoare în acest sens.

## Concluzii

Teza elaborată își propune să aducă unele contribuții în toate cele șase domenii de bază ale convertoarelor dc-dc: analiza, comanda, modelarea, proiectarea și aplicațiile în alte tipuri de conversie a energiei. Deși obiectivul inițial al tezei viza în primul rând aspectele de comandă și modelare, aspectele de sinteză și analiză statică, la care autorul crede că a reușit ulterior să aducă unele contribuții, prin impactul pe care îl au atât în modelare cât și în proiectare, au făcut să se decidă a fi incluse. A rezultat astfel și un caracter unitar al lucrării.

Pe de altă parte, cristalizarea și sistematizarea problematicii legate de conversia dc-dc a condus la ideea că în înțelegerea “filozofiei” convertoarelor dc-dc este esențial un studiu al convertoarelor PWM, pentru că această familie este cea din care se generează celelalte structuri: convertoare cu izolare galvanică, cvasirezonaante, cu undă cvasidreptunghiulară, etc.

Contribuția originală de bază legată de sinteza convertoarelor dc-dc este elaborarea unei tehnici de sinteză mixte, în sensul că îmbină elemente evoluționiste cu elemente combinaționale, denumită “metoda celei de bază complete”. În deducerea acestei metode autorul a adus o serie de rezultate originale. Dincolo de avantajul cert al obținerii de noi convertoare, metoda celei de bază complete realizează o grupare firească a convertoarelor în familii, toate convertoarele familiei având același ordin. De exemplu poate apărea surprinzător faptul că convertorul ĆUK nu face parte din aceeași familie cu convertoarele SEPIC și ZETA (deși tratările de până acum subînțelegeau acest lucru), ci din familia cu convertoarele BUCK și BOOST cu curenți de intrare și de ieșire cu spectru redus. Autorul crede că se obține astfel cea mai reușită și mai firească clasificare a convertoarelor în familii, deschizându-se totodată posibilitatea unui studiu general pe celule de bază complete și nu pe convertoare concrete. Astfel de abordări prin analiza celei de bază complete sunt realizate în capitolele următoare ale lucrării.

În domeniul convertoarelor cu curenți de intrare/ieșire cu pulsații nule, contribuțiile autorului sunt legate de obținerea de familii de astfel de convertoare prin examinarea și modificarea adecvată a celulelor de bază generatoare. În acest fel rezultatele publicate în articole separate în literatură sunt generalizate.

Se aduc unele îmbunătățiri și clarificări unui algoritm de sinteză a convertoarelor cvasirezonaante publicat în literatură, metoda extinzându-se apoi la celule de bază.

Analizei de curent continuu (statică) a convertoarelor dc-dc i se aduc unele elemente noi. Plecând de la noțiunea de mediu de comutație și factor de umplere generalizat sunt prezentate etapele ce trebuie parcurse pentru a obține caracteristicile statice și se introduc noi tipuri de convertoare cu proprietăți benefice în ceea ce privește puterea disipată în comutație și utilizarea lor ca circuite de corecție a factorului de putere. Toate acestea sunt acompaniate de programe ce înlesnesc semnificativ proiectarea în stare staționară a acestor structuri.

Ca și tehnici de comandă originale autorul tezei introduce tehnica de comandă denumită comandă mixtă. Acest nou tip de comandă păstrează avantajele comenzii prin sarcină și a comenzii integrative după o variabilă, oferind însă în anumite situații un răspuns mai rapid și calitativ mai bun la perturbații față de comanda prin sarcină sau față de comanda integrativă după o variabilă. Tot în domeniul tehnicilor de comandă au fost studiate tehnici de reglare fuzzy și neuronale în comanda convertoarelor dc-dc, care, fiind sisteme neliniare, se pretează la o astfel de abordare. În cazul comenzii fuzzy sistemul a fost construit în jurul modulului EVM30 (Texas Instruments) cu procesor de semnal TMS320C30, rezultatul fiind acela că regulatoarele fuzzy pot înlocui cu succes, în condiții de performanțe dinamice remarcabile, regulatoarele convenționale. Și pentru controlerile neuronale concluzia este că folosind rețele de complexitate medie se poate comanda cu succes un convertor SRC, și că este de așteptat ca în viitor aceste tehnici de comandă să se extindă și la alte familii de convertoare.

Modelarea convertoarelor dc-dc a fost domeniul în care autorul a avut cele mai multe contribuții originale, dată fiind și vastitatea și orizontul de explorare larg pe care problematica modelării îl oferă. Unele dintre aceste contribuții au un caracter punctual, în sensul că se referă la un anumit aspect îngust (de exemplu un model pentru o anume tehnică de comandă), altele au un pronunțat caracter de generalitate (de exemplu calculul exact al stării staționare pentru un convertor neautonom).

Contribuțiile originale ale autorului în domeniul investigat s-au concretizat în:

- o serie de propoziții prin care se extinde conceptul de dualitate pentru tipurile de întrerupătoare introduse de Erickson,
- elaborarea unei condiții necesare și suficiente privitoare la generarea convertoarelor numite “de tip BUCK, BOOST și BUCK-BOOST”, plecând de la o celulă de bază completă cu trei terminale, această condiție aplicându-se spectaculos în generarea convertoarelor pătraticе,
- obținerea unei serii de rezultate care dau condițiile necesare privitoare la aplicabilitatea practică a unor tipuri de celule de bază,
- o propoziție prin care se propune arhitectura unei celule de bază complete ce generează convertoare cu curenți de intrare și de ieșire fără fronturi nule, deci cu foarte bune proprietăți de compatibilitate electromagnetică,
- o nouă celulă de bază completă pentru generarea convertoarelor PWM de bază,
- enunțarea teoremei inversiei bilaterale generalizate, în care inversia bilaterală clasică apare ca un caz particular, teoremă cu aplicație imediată în sinteza convertoarelor cu tensiune de ieșire bipolară,
- o rețea generală de comutație din care rezultă două celule de bază complete: cea din care se obține familia ĆUK și o alta generatoare a familiei numite de autor ZSWJ, celule din care se obțin următoarele structuri originale de convertoare:
  - un convertor BUCK și un convertor BOOST, ambele având atât curenți de intrare cât și curenți de ieșire fără fronturi nule (discontinuități), deci foarte bune proprietăți EMI comparativ cu structurile clasice corespunzătoare, convertoare care, fiind testate și experimental, au confirmat estimările teoretice,
  - un convertor Watkins-Johnson și un convertor Watkins-Johnson invers, fiecare având curenți de intrare și ieșire fără fronturi nule, spre deosebire de convertoarele clasice cu același nume în care acești curenți au un spectru larg, datorat fronturilor nule,
  - o variantă de convertor Watkins-Johnson și o variantă de convertor Watkins-Johnson invers (termenul de “variantă” trebuie înțeles în sensul că topologiile sunt complet diferite față de corespondentele clasice dar raportul static de conversie este același), ambele având posibilitatea de a obține, prin cuplarea bobinelor, curenți de ieșire respectiv de intrare constanți, lucru imposibil în convertoarele clasice cu același nume.
  - aplicarea noii metode de sinteză în obținerea, în funcție de celula generatoare utilizată, a 12 convertoare pătraticе noi, denumite de autor după tipul celulei generatoare și tipul conversiei (ridicător, coborător, atât ridicător cât și coborător ) astfel:  $\alpha_1$ -BOOST,  $\alpha_1$ -BUCK-BOOST,  $\alpha_2$ -BOOST,  $\alpha_2$ -BUCK-BOOST,  $\beta_1$ -BUCK,  $\beta_1$ -BOOST,  $\beta_2$ -BUCK,  $\beta_2$ -BOOST,  $\beta_3$ -BUCK,  $\beta_3$ -BOOST,  $\gamma_1$ -BOOST și  $\gamma_1$ -BUCK-BOOST,
  - propunerea unei celule de bază complete originale denumite  $\delta_1$  și generarea pe baza ei a unei noi familii de convertoare:  $\delta_1$ - BUCK,  $\delta_1$ - BOOST,  $\delta_1$ - BUCK-BOOST, dintre care convertorul  $\delta_1$ - BOOST are o importanță deosebită ca și convertor puternic ridicător de tensiune, dar cu o funcționare a tranzistorului la factori de umplere moderați,

- un convertor la care caracteristica statică are expresia  $M(D) = \frac{1}{(1-D)^n}$  și care se obține plecând de la convertorul  $\delta_1$ - BOOST, conținând însă ca dispozitive de comutație un singur tranzistor iar în rest numai diode,
- argumentarea echivalenței convertorului ZETA cu curent de ieșire fără pulsații și bobine necuplate, citat de Kolar ș.a. în literatură, cu un convertor BUCK-BOOST cu un filtru L-C la ieșire,
- demonstrarea faptului că șase din topologiile convertoarelor cu pulsații nule propuse de autorii citați mai sus sunt de fapt versiuni a numai două topologii,
- obținerea prin cuplarea bobinelor a unui curent de ieșire fără pulsații pentru noua variantă de convertor Watkins-Johnson rezultată prin sinteză,
- demonstrarea faptului că topologia celulei de bază nu permite obținerea de pulsații nule de curent fără cuplarea bobinelor pentru convertoarele din familia ZSWJ, acest lucru fiind însă posibil folosind bobine cuplate,
- elaborarea de convertoare pătratice cu curenți de intrare/ieșire cu pulsații nule,
- sistematizarea unor algoritmi de sinteză propus de Maksimović, demonstrându-se imposibilitatea unor cazuri de inserție a elementelor rezonante, extinzându-se apoi algoritmul de sinteză de la convertoare individuale la celule de bază complete, cu precizarea clară a modului de alegere a întrerupătoarelor,
- obținerea prin modificarea mediului de comutație a unei noi familii de convertoare numită DIVM (cu tensiune de intrare discontinuă - “discontinuous input voltage mode”), a căror funcționare a fost confirmată experimental,
- algoritmi sistematizați de deducere a caracteristicilor statice pentru modul DICM și DCVM,
- stabilirea în mod elegant și unitar a seturilor de condiții necesare funcționării CCM a tuturor convertoarelor pătratice, printr-un studiu pe celule de bază complete,
- dezvoltarea de programe MATLAB interactive pentru trasarea automată a familiilor de caracteristici control-ieșire ale convertoarelor cvasirezonante, atât pentru cele cu funcționare cu frecvență variabilă, cât și pentru cele cu funcționare la frecvență constantă, programe ce se dovedesc instrumente utile în proiectarea acestor structuri,
- o nouă metodă de comandă, numită comandă mixtă, care este o tehnică integrativă care realizează integrarea ponderată a mărimilor lent variabile ce caracterizează mediul de comutație, cu sublinierea avantajelor și dezavantajelor ei.
- investigarea posibilității de aplicare a unor elemente de inteligență artificială în comanda convertoarelor dc-dc, examinându-se următoarele tipuri de comenzi:
  - comanda fuzzy, unde studiul a fost acompaniat de experimente pe un convertor BOOST, folosind un regulator Mamdani cu metoda min de inferență, pentru un tabel de reguli cu singleton-uri și pentru un tabel de inferență cu termeni lingvistici, subliniindu-se:
    - influența medierii eșantioanelor în calitatea tensiunii de ieșire în regim static și dinamic,
    - momentele propice eșantionării pentru a evita zgomotele de comutație,
    - efectul unor anumiți parametri ai regulatorului în răspunsul dinamic al sistemului,
  - comanda cu rețele neuronale, investigată numai prin simulare, având ca obiect de studiu un convertor rezonant serie și la care baza teoretică pentru generarea vectorilor de antrenament a fost comanda pe traiectorie optimă,
- rezolvarea problemei filtrului de intrare pentru convertoare QRC-FW utilizând modelul canonic generalizat al lui Witulski și Erickson,
- aplicarea metodei generale de mediere propuse de Sun și Grotstollen pentru elaborarea de modele unificate în spațiul stărilor (termenul “unificat” se referă la faptul că se oferă o formulă analitică valabilă în CCM atât sub cât și peste frecvența de comutație) pentru convertoare SRC astfel:

- un model unificat de semnal mare care a permis:
  - rezolvarea stării staționare, determinarea și trasarea caracteristicilor statice unificate de tip comandă-ieșire, scriindu-se apoi programe MATLAB de proiectare interactivă asistată, în care utilizatorului i se oferă curbele de solicitare și informații calitative și cantitative privind alegerea unor anumite mărimi specifice,
  - simularea accelerată a comportării dinamice la semnal mare, demonstrată prin aplicarea simplă a modelului în mediul SIMULINK, rezultatele obținute fiind confirmate prin simularea cu simulatorul de circuite CASPOC,
- un model unificat de semnal mic, care, utilizat pentru reprezentarea funcției de transfer control-ieșire și a audiosusceptibilității la unui convertor cunoscut și deja analizat în literatură, a furnizat rezultate identice cu cele ale autorilor, confirmându-se astfel justetea modelului,
- un nou model general (valabil pentru orice convertor) pentru comanda prin sarcină, caracterizat prin aceea că este descris în spațiul stărilor, pretându-se astfel la o serie de analize (implementat în MATLAB el furnizează rezultate identice cu modelele descrise sub formă de schemă bloc de autorii comenzii prin sarcină),
- un model mai exact decât cel al lui Smedley și Čuk pentru comanda integrativă după o variabilă, deoarece se ia în considerație și efectul de eșantionare-memorare prezent în perturbațiile din tensiunile capacitive. Folosind acest model a fost investigată stabilitatea unui convertor ČUK cu comandă integrativă după o variabilă, prognozele modelului arătând o excelentă coincidență cu comportamentul obținut prin simulare.
- noi modele de semnal mare și de semnal mic pentru convertoare cvasirezonante. Se demonstrează că, formal, modelul este identic cu cel al convertorului PWM părinte, cu singura deosebire că factorul de umplere nu este o variabilă independentă ci depinde de un raport al mărimilor de stare ce caracterizează mediul de comutație. Și aceste modele au un caracter general, adică sunt valabile pentru orice tip convertor, cele care se modifică fiind numai 8 matrici de stare ce caracterizează convertorul PWM părinte. Aplicarea modelelor de semnal mare se regăsește în rezolvarea stării staționare și determinarea caracteristicilor statice, dar și în simularea accelerată comparativ cu un simulator de circuite (cel utilizat de autor a fost PETS), timpul de simulare fiind aproximativ cu două ordine de mărime mai mic. Evident, modelul de semnal mic servește pentru determinarea funcțiilor de transfer și verificarea lui s-a făcut pe aceleași convertoare analizate, deja cunoscute în literatură, pentru a avea un element de comparație. Coincidența rezultatelor este revelatoare, validând astfel modelul.
- două noi modele discrete pentru convertoare dc-dc, obținute cu aceeași abordare recursiv-regresivă devenită clasică, dar utilizând o integrare numerică de tip trapezoidală sau backward-Euler. Sunt obținute atât modele de semnal mare cât și modele de semnal mic. Superioritatea modelelor față de modelul propus de Kassakian este evidențiată pe o structură BOOST. În plus, după ce au fost implementate în mediul SIMULINK, modelele de semnal mic au fost verificate experimental, coincidența prognozei cu realitatea fiind perfectă.
- determinarea funcției de transfer control-ieșire exacte pentru convertoare funcționând la frecvență constantă, elementele de nouate aduse de autor fiind:
  - sistematizarea algoritmului Tymerski, de calcul a funcției de transfer control-ieșire exacte, prin precizarea clară a pașilor ce trebuie parcurși și a calculelor care trebuie efectuate,
  - elaborarea unui algoritm de calcul exact al stării staționare folosind o metodă Newton-Raphson vectorială, precizându-se modul de scriere a elementelor funcției vectoriale. Implementată în MATLAB metoda a furnizat rezultate similare cu cele ale unui simulator.
  - o altă tehnică de calcul accelerat a stării staționare realizată prin legarea unui program de calcul matematic de un simulator de circuite. Procedul este deosebit de simplu din

punct de vedere al utilizatorului care trebuie numai să descrie modul de interconectare a componentelor printr-un fișier de tip \*.cir din Pspice și să identifice elementele reactive din circuit. Nu este deci necesară nici un fel de descriere în spațiul stărilor sau de alt gen pentru circuit. Mai mult, tehnica se dovedește a fi deosebit de flexibilă, și de potrivită pentru o gamă de aplicații diverse. Cele investigate de autorul tezei sunt:

- calculul exact al stării staționare utilizând metoda Newton cu diferențe finite și metoda cu actualizare Broyden pentru calculul numeric al Jacobianului. Este efectuată o comparație între cele două metode concluzia fiind că deși metoda cu actualizare Broyden necesită mai multe iterații, ea reclamă mai puține simulări. Cum timpul de rezolvare exactă a stării staționare este cu preponderență dat de numărul de simulări, se poate conchide că metoda cu actualizare Broyden este mai eficientă decât metoda Newton cu diferențe finite.
- determinarea caracteristicilor statice reale și a familiilor de forme de undă în stare staționară. Studiul de caz s-a făcut pe un convertor  $\delta_1$ -BOOST cu elemente de circuit neideale, dovedindu-se posibilitatea trasării caracteristicilor statice și a formelor de undă chiar și când, datorită variației mărimii de pe abscisă, se trece dintr-un mod de funcționare în altul.
- simularea exactă a convertoarelor dc-dc cu regulatoare numerice sau regulatoare fuzzy. Acest lucru este posibil deoarece în regimurile tranzitorii sunt sesizate și eventualele perioade în care convertorul își schimbă modul de lucru, fapt ce nu este posibil de luat în considerație de modelele pe baza cărora se simula evoluția sistemului până acum, deoarece aceste modele erau elaborate pentru un anumit mod de funcționare. Exemplificarea este făcută pe un convertor BUCK-BOOST cu controller numeric și pe același convertor, cu comandă fuzzy.
- expresiile analitice generale ale funcțiilor de transfer control-ieșire exacte și reprezentarea caracteristicile de amplitudine și de fază împreună cu caracteristicile furnizate de modele mediate publicate în literatură (în vederea comparării), pentru următoarele categorii de convertoare:
  - convertoare PWM DCVM,
  - convertoare PWM DIVM,
  - convertoare PWM CCM cu feedforward realizat cu integratoare și feedforward realizat cu multiplicatoare, ambele tipuri de feedforward fiind sintetizate într-o formulă unică.
  - convertoare PWM CCM cu comanda prin sarcină,
  - convertoare PWM CCAM cu comanda integrativă de orice tip - formulă generală,
- deducerea unei expresii generale exacte a funcției de transfer control-ieșire pentru orice tip de comandă integrativă la funcționare în moduri discontinue (DICM, DCVM),
- găsirea unei condiții necesare pentru ca la un convertor funcția de transfer control-ieșire mediată și exactă să coincidă. Se demonstrează că o astfel de clasă de convertoare o constituie aceea, găsită de Kassakian, la care reacția de feedforward asigură și eroare dinamică nulă.
- investigarea posibilității de utilizare a convertoarelor ZCS QRC ca circuite de corecție a factorului de putere, cu efectuarea unei analize cantitative, stabilirea principalelor relații ce guvernează aceste circuite PFC și considerații de proiectare și alegere privind tipul de convertor și modul său de lucru.
- aplicarea convertoarelor DIVM, introduse tot de autorul tezei, ca circuite de corecție a factorului de putere, cu determinarea relațiilor care trebuie îndeplinite de către acestea, demonstrându-se că ele sunt potrivite pentru tensiuni de ieșire mici în raport cu amplitudinea tensiunii de alimentare.

Mai există, desigur, numeroase căi neexplorate în domeniul convertoarelor dc-dc în comutație. De asemenea, autorul este conștient că orice prognoză într-un domeniu atât de vast riscă

din start să fie incompletă, dacă nu cumva chiar hazardată. Sub această rezervă, totuși, autorul îndrăznește să sugereze unele dintre viitoarele provocări legate de tematica abordată în lucrare.

În ceea ce privește sinteza convertoarelor dc-dc în comutație ar fi utilă elaborarea de procedee de sinteză pentru topologii cu sarcina flotantă sau cu surse multiple de alimentare. De asemenea în domeniul convertoarelor pătratică există cu siguranță o serie de celule de bază complete încă nedescoperite și care ar putea, prin relația matematică a caracteristicii statice de transfer și topologia specifică, să-și găsească aplicații practice (de exemplu în circuite de corecție a factorului de putere).

O problemă deosebită pare a fi aceea de a găsi topologii simple de conversoare cu tensiune de ieșire bipolară și curenți de intrare/ieșire cu pulsații nule, folosind bobine necuplate. Investigarea posibilității de realizare a amplificatoarelor de putere în comutație folosind topologii cu pulsații nule de curent ar putea constitui un studiu de viitor, dar poate cea mai importantă direcție de investigație este modul în care noul model de transformator propus de Čuk influențează și își găsește aplicarea în diverse alte topologii.

Sinteza unitară a convertoarelor cvasirezonaante ridică și ea o serie de provocări. Se sugerează aici căutarea unei proceduri de sinteză pentru conversoare cu comutație nedisipativă cu trei sau mai multe întrerupătoare, având în vedere că topologii singulare de acest gen au fost raportate. Apoi ar fi de analizat, dacă este posibil și cum să se obțină conversoare cvasirezonaante având ca părinți conversoare pătratică și de ce proprietăți demne de exploatat se bucură aceste noi structuri.

Desigur că, în privința tehnicilor de comandă, domeniul rămâne deschis multor metode cu caracter de noutate. Odată cu explozia procesoarelor de semnal este de așteptat ca tehnicile fuzzy și neuronale să aibă pe viitor un cuvânt important de spus. Rămân însă numeroase aspecte de aprofundat, legate de exemplu de: alegerea judicioasă a funcțiilor de apartenență, alegerea optimă și generală a coeficienților de normare și a parametrilor regulatorului în ansamblu (pentru a asigura o comportare dinamică de calitate), stabilirea numărului de straturi ascunse și a numărului de neuroni din acestea, etc.

În fine, în partea de modelare, dintre multiplele preocupări de viitor care se întrezăresc, autorul amintește numai pe acelea care au legătură directă cu tematica abordată în prezenta lucrare și anume:

- încercarea de rezolvare a problemei filtrului de intrare în cazul convertoarelor QRC HW pe baza modelelor canonice generalizate,
- deducerea de modele mediate de semnal mare și de semnal mic pentru conversoare paralel, conversoare serie pe mod limitat, conversoare cu comandă în fază și, în general, pentru orice tip de convertor rezonant cu funcționare la frecvență constantă, aplicând metoda generală de mediere a lui Sun și Grotstollen. Elaborarea, în urma analizei, de programe interactive de proiectare asistată și pentru aceste conversoare.
- dezvoltarea de modele unificate în spațiul stărilor pentru comanda integrativă după o variabilă, ca prim pas, apoi pentru orice tip de comandă integrativă, cu luarea în considerație a eventualelor efecte de eșantionare,
- integrarea programelor de calcul matematic cu simulatoarele de circuite. De altfel primii pași în acest sens au și fost făcuți. Un exemplu elocvent în acest sens este blockset-ul Power System scris pentru mediul SIMULINK, împreună cu versiunea 5.0 a programului MATLAB. Deși i s-ar putea imputa destule lacune, totuși ideea este una strălucită și cu siguranță va fi dezvoltată și impusă în scurt timp de firma MathWorks.
- în privința altor posibilități de utilizare a legăturii dintre un program de calcul matematic cu un simulator de circuite se întrevăd noi probleme de elucidat și aplicații legate de:
  - studierea eficacității sub aspectul numărului de iterații și numărului de simulări a altor modele afine utilizate pentru calculul exact al stării staționare,
  - obținerea curbilor de separație reale între diferitele moduri de funcționare pentru conversoare la care se consideră elementele parazite iar elementele de comutație se

- consideră neideale (modelate ce elemente de circuit care să le apropie de elementele de comutație reale),
- metode de extragere a modelelor dinamice ale sistemului,
  - automatizarea la maximum a programului, cu prezentare sub forma unui mediu integrat, cu interfețe grafice utilizator,
  - o analiză exactă de semnal mic,
- obținerea de expresii exacte pentru funcțiile de transfer control-ieșire ale convertoarelor rezonante și cvasirezonante cu funcționare la frecvență constantă,
- un studiu calitativ, urmat de o generalizare a expresiilor principalelor categorii de funcții de transfer exacte, privite prin prisma numărului de stări topologice și a relațiilor de constrângere ce definesc momentele de comutație,
- obținerea prin metode de dezvoltare în serie a expresiei funcției de transfer mediate în formă rațională, plecând de la expresia în formă închisă a funcției de transfer control-ieșire exacte. Primele rezultate în acest sens au fost obținute de Tymerski pentru convertoare PWM CCM cu comandă în factor de umplere, la care au fost obținute, în urma dezvoltării în serie, exact formulele rezultate prin aplicarea modelului mediat în spațiul stărilor. Pentru alte tipuri de convertoare însă nu s-au raportat astfel de rezultate. Ar fi de dorit un algoritm sau enunțarea unor reguli universale de obținere a expresiei funcției de transfer control-ieșire mediate din cea exactă. Avantajul ar fi imediat, pentru că s-ar ușura mult studiul stabilității și proiectarea controllerului pentru convertoarele respective.
- studierea posibilității de obținere a funcției de transfer control-ieșire exacte pentru convertoare cu comandă în frecvență și enunțarea unui algoritm de deducere sistematică a acesteia,
- elaborarea de modele de semnal mare și semnal mic pentru circuitele de corecție a factorului de putere.

Tot în domeniul utilizării convertoarelor dc-dc ca circuite PFC, date fiind și normativele severe în ceea ce privește poluarea armonică a rețelei, autorul consideră că se înscriu ca preocupări de viitor:

– utilizarea convertoarelor cvasirezonante și multirezonante ca și circuite PFC în domeniul monofazat, dar mai ales în domeniul trifazat, cu reducerea numărului de dispozitive de comutație (în special active) necesar. Deja în ultimul an au fost publicate o serie de rezultate promițătoare în acest sens.

– utilizarea de tehnici de comandă care să elimine necesitatea măsurării tensiunii și curentului de intrare, cu o structură cât mai simplă (de exemplu cu eliminarea multiplicatoarelor) și care să se poată integra. Comanda cu purtătoare neliniară elaborată de colectivul de la Boulder University Colorado este un exemplu elocvent în acest sens.

Toate aceste provocări cer pe lângă tenacitate, răbdare și perseverență, o pregătire multidisciplinară exemplară a inginerului specialist în electronica de putere, despre care Rudolf Severns, în prefața unei cărți de referință despre convertoare dc-dc în comutație, spunea atât de frumos că nu trebuie să uite că rămâne “eroul neslăvit al proiectării electronice”



## Bibliografie

- [1] Amran, Y., Hulielhel, F., Ben-Yaakov, S., "A unified SPICE compatible average model of PWM converters", *IEEE Trans. Power Electron.*, vol. 6, no 4, Oct., 1991, pp. 585-594.
- [2] Anunciada, A. V., Silva, M. M., "New constant-frequency current-mode control for power converters, stable for all values of duty ratio, and usable in all four quadrants", *IEEE Trans. Ind. Electron.*, vol. 37, no. 4, August, 1990, pp. 323-325.
- [3] Anunciada, A. V., Silva, M. M., "On the stability and subharmonic susceptibility of current-mode controlled converters", *IEEE PESC Record*, 1992, pp. 345-353.
- [4] Aprille, T. J., Trick, T. N., "Steady state analysis of nonlinear circuits with periodic inputs", *Proc. of IEEE*, vol. 60, no. 1, 1972, pp. 108-114.
- [5] Arbetter, B., Maksimović, D., "Feedforward pulse width modulators for switching power converters", *IEEE Trans. Power Electron.*, vol. 12, no 2, March, 1997, pp. 361-368.
- [6] Bîrcă-Gălăţeanu, Ş., "DC-DC converters with low input-current ripple". Proceedings of the International Symposium on Signal Circuits and Systems SCS'97, 1997, Iaşi, Oct. 2-3, vol. I, pp. 160-163.
- [7] Bedrosian, D. G., Vlach, J., "An accelerated steady-state method for networks with internally controlled switches", *IEEE Trans. on Circuits and Systems I*, vol. 39, no 7, 1992, pp. 520-530.
- [8] Bedrosian, D., Vlach, J., "Time-domain analysis of networks with internally controlled switches", *IEEE Transactions on Circuits and Systems I*, vol. 39, no. 3, March, 1992, pp. 199-212.
- [9] Bird, B. M., King, K. G., Pedder, A. G., An introduction to power electronics, Wiley, 1993.
- [10] Bose, B. K., *Modern power electronics. Evolution, technology, and applications*, IEEE Press, 1992.
- [11] Bose, B. K., *Power electronics and AC drives*, Prentice-Hall, 1986.
- [12] Bose, B. K., *Power electronics and variable frequency drives*, IEEE Press, 1997.
- [13] Calderoni, L., Pinola, L., Varoli, V., "Optimal feedforward compensation for PWM DC/DC converters", *IEEE PESC Record*, 1990, pp. 235-241.
- [14] Capel, A., "A versatile zero Ripple Topology", *IEEE Power Electronics Specialists Conference, PESC*, 1988, pp. 133-141.
- [15] CASPOC - Reference Manual, Simulation Research.
- [16] Chau, K. T., Lee, Y. S., Ioinovici, A., "Computer-aided modeling of quasi-resonant converters in the presence of parasitic losses by using the MISSCO concept", *IEEE Trans. Ind. Electron.*, vol. 38, no. 6, Dec., 1991, 454-461.
- [17] Cheron, Y., Foch, H., "Series resonant converters", *EPE Tutorial resonant converters*, Brighton, Great Britain, 1993, pp. 15-32.
- [18] Chetty, P. R. K., "Current injected equivalent circuit approach (CIECA) to modelling of switching dc-dc converters in continuous inductor conduction mode", *IEEE Trans. Aerosp. Electron. Syst.*, vol. AES-17, no. 6, Nov. 1981.
- [19] Chetty, P. R. K., "Current injected equivalent circuit approach (CIECA) to modelling of switching dc-dc converters in discontinuous inductor conduction mode", *IEEE Trans. Ind. Electron.*, vol. IE-29, no. 3, Aug. 1982.
- [20] Chetty, P. R. K., "Current injected equivalent circuit approach to the modelling and analysis of current programmed switching dc-dc converters (Discontinuous inductor conduction mode)" *IEEE Trans. Ind. Applicat.*, vol. IA-18, no. 3, May/June, 1982.
- [21] Chetty, P. R. K., "Modeling and analysis of Ćuk converter using current injected equivalent circuit approach," *IEEE Trans. Ind. Electron.*, vol. IE-30, no. 1, Feb, 1983.
- [22] Chin, Y., Lee, F. C., "Constant-frequency parallel-resonant converters", *Proceedings of the Industry Applications Society Annual Meeting*, Atlanta, Georgia, Oct. 18-23, 1987, pp. 705-710.
- [23] Chow, M. H. L., Siu, K. W., Tse, C. K., Lee, Y. S., "A novel method for elimination of line-current harmonics in single-stage PFC switching regulators", *IEEE Trans. Power Electron.* vol. 13, no. 1, Jan, 1998, pp. 75-83.
- [24] Chua, L. O., Lin, P. M., *Computer-Aided Analysis of Electronic Circuits. Algorithms and Computational Techniques*, Prentice-Hall, Englewood Cliffs, NJ, 1975.
- [25] Chung, H. S., Ioinovici, A., "Fast computer-aided simulation of switching power regulators based on progressive analysis of the switches' state", *IEEE Trans. Power Electron.*, vol. 9, no 2, March, 1994, pp. 206-212.
- [26] Ćuk, S., "A new zero-ripple switching dc-to-dc converter and integrated magnetics", Proceedings of the *IEEE Power Electronics Specialists Conference*, 1980, Atlanta (GA), Oct. 16-20, pp. 12-23 .
- [27] Ćuk, S., "General topological properties of switching converters", *IEEE Power Electronics Specialists Conference, PESC*, 1979 Record, pp. 109-130, June, 1979.
- [28] Ćuk, S., "Modular telecommunication converters", *Proceedings of the 9<sup>th</sup> International Symposium on Power Electronics, Ee'97*, Novi-Sad, October, 22-24, 1997, Invited paper.
- [29] Ćuk, S., "Switching dc-to-dc converter with zero input or output current ripple", Record of the *IEEE Industry Applications Society Annual Meeting*, 1978, Toronto, Canada, Oct. 1-5, pp. 1131-1146.
- [30] Ćuk, S., Middlebrook, R. D., "A new optimum topology switching dc-to-dc converter", *IEEE Power Electronics Specialists Conference, PESC*, 1977 Record, pp. 160-179, June, 1977.

- [31] Ćuk, S., Middlebrook, R. D., "Coupled-inductor and other extensions of a new optimum topology switching dc-dc converter", Record of the *IEEE Industry Applications Society Annual Meeting*, 1977, Los Angeles (CA), Oct. 2-6, pp. 1110-1126.
- [32] Deisch, C., "Simple switching control method changes power converter into a current source", *IEEE/PESC Conf. Rec.*, 1978, pp. 300-306.
- [33] Dennis, J. E., Schnabel, R. B., *Numerical methods for unconstrained optimisation and nonlinear equations*, SIAM Classics in applied mathematics 16, 1996.
- [34] van Dijk, E., Spruijt, H. J. N., O'Sullivan, D. M., Klaassens, J. B., "PWM-switch modeling of dc-dc converters", *IEEE Trans. Power Electron.*, vol. 10, no. 5, Nov, 1995, pp. 659-665.
- [35] Elbuluk, M. E., Verghese, G. C., Kassakian, L. G., "Sampled data modeling and digital control of resonant converters", *IEEE Trans. Power Electron.*, no. 3, 1988, pp. 121-130.
- [36] Eng S. C., Oruganti, R., Liang, Y. C., "An automated algorithm for small signal analysis of dc-dc power converters", *IEEE Trans. Power Electron.*, vol. 11, no 1, January, 1991, pp. 132-141.
- [37] Erickson, R. W., "Some topologies of high quality rectifiers", *First International Conference on Energy, Power, and Motion Control*, May, 5-6, 1997, Tel Aviv, Israel - Keynote paper.
- [38] Erickson, R. W., "Synthesis of switched-mode converters", *IEEE Power Electron. Specialists Conf. Rec.*, 1983, pp. 9-22.
- [39] Erickson, R. W., *Fundamentals of power electronics*, Chapman & Hall, New-York, 1997.
- [40] Ertl, H., Kolar, J. W., Zach, F. C., "Basic considerations and topologies of switched-mode assisted linear power amplifiers", Proceedings of the 11<sup>th</sup> IEEE Applied Power Electronics Conference, San Jose (CA), 1996, March 3-7, Vol. 1, pp. 207-213.
- [41] Ferrioux, J. P., Forest, F., "Alimentations à découpage. Convertisseurs à résonance", Masson, Paris, 1987.
- [42] Freeland, S. Middlebrook, R. D., "A unified analysis of converters with resonant switches", *IEEE/PESC Rec.*, 1987, pp. 20-30.
- [43] Freitas, L. C., Farias, V. J., Caparelli, P. S., Vieira, J. B., Hey, H. L., Cruz, D. F., "An optimum ZVS-PWM dc-to-dc converter family: analysis, simulation and experimental results", *IEEE/PESC Conf. Rec.*, 1992, pp. 229-235.
- [44] General Dynamic Corporation, "Resonant AC power system: proof of concept", *GDC Interim Report*, vol. I.
- [45] Gil, P., Sebastián, J., Cobos, J. A., Uceda, J., Aldana, F., "Analysis of the zero-current-switched quasiresonant converters used as power factor preregulators", *IEEE PESC Record*, 1992, pp. 1052-1060.
- [46] Girard, M., *Electronique industrielle*, Mc Graw-Hill, 1987.
- [47] Gordon, C. C., Ogborn, L. L., "Computer analysis of thyristor chopper circuits with detailed device modeling", *IEEE/PESC Conf. Rec.*, 1979, pp. 32-39.
- [48] Guinjoan, F., Calvente, J., Poveda, A., Martínez, L., "Large-signal modeling and simulation of switching dc-dc converters", *IEEE Trans. Power Electron.*, vol. 12, no 3, May, 1997, pp. 485-494.
- [49] Harada, K., Gu, W. J., Murata, K. "Controlled resonant converters with switching frequency fixed", *IEEE/PESC Conf. Rec.*, 1987, pp. 431-438.
- [50] Henze, C. P., Martin H. C., Parsley, D. W., "Zero voltage switching in high frequency power converters using pulse width modulation", *Proc. IEEE/APEC*, 1988, pp. 33-40.
- [51] Hissem, M., Chériti, A., Sicard, P., Razzouk, A. B., "Application of the current-injected modeling approach to quasi-resonant converters", *IEEE Trans. Power Electron.*, vol. 12, no. 4, July., 1997, pp. 695-703.
- [52] Hoz A., Cruz, E., "Analysis and design of a zero current switched quasi-resonant converter with synchronous rectification for low output voltage applications", *IEEE/PESC Conf. Rec.*, 1992, pp. 221-228.
- [53] Hua, G., Leu, C. S., Lee, F. C., "Novel zero-voltage-transition PWM converters", *IEEE PESC Record*, 1992, pp. 55-61.
- [54] Hui, S. Y. R., Christopoulos, C., "A discrete approach to the modeling of power electronic switching networks", *IEEE Trans. Power Electron.*, vol. 5, no 4, Oct., 1990, pp. 398-403.
- [55] Hui, S. Y. R., Fung, K. K., "Fast decoupled simulation of large power electronic systems using new two-port companion link models", *IEEE Trans. Power Electron.*, vol. 12, no 3, May, 1997, pp. 462-473.
- [56] Ioinovici, A., "A new computer-aided approach to the analysis of Ćuk converter by using the alternor equations", *IEEE Trans. Power Electron.*, vol. 4, no 3, July, 1989, pp. 319-330.
- [57] Jain, P. K., Tanju, M. C., "A unity power factor resonant AC/DC converter for high-frequency space power distribution system", *IEEE Trans. Power Electron.*, vol. 12, no. 2, March, 1997, pp. 325-331.
- [58] Jayaram, N., Maksimović, D., "Power factor correctors based on coupled-inductor Sepic and Ćuk converters with nonlinear-carrier control", *IEEE APEC*, Anaheim, CA, Feb 15-19, 1998.
- [59] Jin, H., Joos, G., Pande, M., Ziogas, P. D., "Feedforward techniques using integral duty-cycle control", *IEEE PESC Record*, 1990, pp. 370-377.
- [60] Jovanović, M. M., Hopkins, D. C., Lee, F. C., "Design aspects for high-frequency off-line quasi-resonant converters", *High Frequency Power Conversion Conference*, 1987, reprinted in *High frequency resonant, quasi-resonant, and multi-resonant converters*, Virginia Power Electronics Center, vol. I, VPEC Series, 1989, pp. 67-81.
- [61] Jovanović, M. M., Liu, K., H, Oruganti, R. Lee, F. C., "State-plane analysis of quasiresonant converters" *IEEE Trans. Power Electron.* vol. 2, no. 1, Jan., 1987, pp. 56-73.

- [62] Jovanović, M. M.; Lee, F. C.; Chen, D. Y., "A Zero-Current-Switched Off-Line Quasi-Resonant Converter with Reduced Frequency Range", *High Frequency Power Conversion Conference Proceedings*, 1988, pp. 160-170.
- [63] Kassakian, J. G., Schlecht, M. F., Verghese, G. C., *Principles of power electronics*, Addison Wesley, 1991.
- [64] Kato, T., Tachibana, W., "Periodic steady-state analysis of an autonomous power electronic system by a modified shooting method", *Workshop on Computers in Power Electronics*, August 11-14, 1996, Portland, Oregon.
- [65] Khan, I. A., "Synthesis and analysis of transformer-isolated converters", *IEEE Trans. Power Electron.* vol. 10, no. 4, July, 1995, pp. 409-418.
- [66] Kimhi, D., Ben-Yaakov, S., "A Spice model for current mode PWM converters operating under continuous inductor current conditions", *IEEE Trans. Power Electron.*, vol. 6, no 2, April, 1991, pp. 281-286.
- [67] King, R. J., Stuart, T. A., "Small-signal model for the series resonant converter", *IEEE Trans. Aerosp. Electron. Syst.* no. 3, May, 1985, pp. 301-319.
- [68] Kislovski, A. S., "Dynamical analysis of switching dc-dc converters", *EWV Engineering*, No. 1, Bern, Switzerland, 1985.
- [69] Kislovski, A. S., Redl, R., "Generalization of the injected-absorbed-current dynamics analysis method of DC-DC switching power cells", in *IEEE Int. Symp. Circuits and Systems (ISCS)*, 1992, pp. 1895-1898.
- [70] Kolar, J. W., Sree, H., Zach, F. C., "Novel aspects of an application of 'zero'-ripple techniques to basic converter topologies", Proceedings of the *IEEE Power Electronics Specialists Conference PESC'97*, 1997, St. Louis (Mo), June 22-27, pp. 796-803.
- [71] Kolar, J. W., Zach, F. C., "Novel 'zero' ripple ĆUK and SEPIC converter topologies requiring no coupling of input and output inductors", Proceedings of the *Power Conversion and Intelligent Motions Conference*, Nürnberg, 1997, June 6-8, pp. 91-103.
- [72] Kuroe, Y., Maruhashi, T., Kanayama, N., "Computation of sensitivities with respect to conduction time of power semiconductors and quick determination of steady state for closed loop power electronic systems", *IEEE/PESC Conf. Rec.*, 1988, pp. 756-764.
- [73] Lai, Z., Smedley, K. M., "A new extension of one-cycle control and its application to switching power amplifiers" *IEEE Trans. Power Electron.* vol. 11, no. 1, Jan., 1987, pp. 99-105.
- [74] Lascu, D., "A new class of zero-voltage switching PWM dc-dc converters", *Proceedings of the International Conference on Power Conversion and Intelligent Motion, PCIM*, Nürnberg, Germany, 1998, May 26-28, pp. 537-546.
- [75] Lascu, D., "A new, ripple-free output current boost converter topology - part 1", *Buletinul Științific al Universității "Politehnica" din Timișoara, Seria Electrotehnică, Electronică și Telecomunicații (Transactions on Electrotechnics, Electronics and Communications)*, Tom 43 (57), Fascicola 1, 1998, pp. 85-92.
- [76] Lascu, D., "A new, ripple-free output current boost converter topology - part 2", *Buletinul Științific al Universității "Politehnica" din Timișoara, Seria Electrotehnică, Electronică și Telecomunicații (Transactions on Electrotechnics, Electronics and Communications)*, Tom 43 (57), Fascicola 1, 1998, pp. 93-100.
- [77] Lascu, D., "A new, small-signal model for one-cycle controlled Ćuk converters", *Proceedings of the 4<sup>th</sup> International Conference on Engineering of Modern Electric Systems*, Oradea, May 30 - June 1, 1997, pp.131-136.
- [78] Lascu, D., "A steady-state unified model for quasiresonant dc-dc converters", *Lucrările Sesiunii Științifice ale Universității "Aurel Vlaicu" din Arad*, Ediția a IV-a, "25 de ani de învățământ tehnic superior arădean", Arad, 30-31 Octombrie, 1997, vol. VIII, pag. 251-260.
- [79] Lascu, D., "Computer aided analysis and design of a series resonant converter", *Lucrările Sesiunii Științifice ale Universității "Aurel Vlaicu" din Arad*, Ediția a IV-a, "25 de ani de învățământ tehnic superior arădean", Arad, 30-31 Octombrie, 1997, vol VIII, pag. 241-250.
- [80] Lascu, D., "Exact small-signal analysis of one-cycle controlled PWM converters", *Proceedings of the 9<sup>th</sup> International Symposium on Power Electronics, Ee '97*, Novi-Sad, October, 22-24, 1997, pp. 153-160.
- [81] Lascu, D., "Mixed-mode control of PWM converters", *Proceedings of the International Conference on Power Conversion and Intelligent Motion, PCIM*, Nürnberg, Germany, 1995, May 26-28, pp. 110-122.
- [82] Lascu, D., "Small-signal analysis of charge controlled PWM DC-DC converters", *Proceedings of the International Symposium on Signals, Circuits and Systems*, Iași, October 2-3, 1997, pp. 350-353.
- [83] Lascu, D., Lascu, M., "Exact small-signal analysis of PWM dc-dc power converters in discontinuous capacitor voltage mode", *Proceedings of the Symposium on Electronics and Telecommunications "Etc. '98"*, Timișoara, Sept. 17-18, 1998, vol. II, pp 57-62.
- [84] Lascu, D., Lascu, M., "Unified models for quasi-resonant power converters", *Proceedings of the Symposium on Electronics and Telecommunications "Etc. '98"*, Timișoara, Sept. 17-18, 1998, vol II, pp 53-56.
- [85] Lascu, D., Lascu, M., "A ripple-free input current buck-boost converter with discontinuous conduction characteristics", *Proceedings of the International Symposium on System Theory, Robotics, Computers & Process Informatics SINTES 9*, Craiova, June 4-6, 1998, vol. III, pp. 56-63.
- [86] Lascu, D., Lascu, M., "Exact small-signal analysis of discontinuous-input voltage mode PWM dc-dc converters", *Proceedings of the International Symposium on System Theory, Robotics, Computers & Process Informatics SINTES 9*, Craiova, June 4-6, 1998, vol. III, pp. 48-55.

- [87] Lascu, D., Lascu, M., "Unified models for the series resonant converters", *Proceedings of the 6<sup>th</sup> International Conference on Optimization of Electrical and Electronic Equipments, OPTIM'98*, Braşov, May 14-15, 1998, pp.361-364.
- [88] Lascu, D., "Small-signal analysis of charge controlled PWM DC-DC converters", *Proceedings of the 4<sup>th</sup> International Conference on Engineering of Modern Electric Systems*, Oradea, May 30 - June 1, 1997, pp.125-130.
- [89] Lascu, D., Negoitescu, D., "Simulation of a buck-boost converter with a digital controller using the CASPOC package", *Analele Universităţii "Eftimie Murgu" Reşiţa - sesiune jubiliară - "Un sfert de veac de la fondarea învăţământului superior la Reşiţa"*, 1996, fascicola III, pp. 209-214.
- [90] Lascu, D., "Analysis and design of a zero current switching power factor correction circuit", *Lucrările Sesiunii Ştiinţifice ale Universităţii "Aurel Vlaicu" din Arad*, Ediţia a IV-a, "25 de ani de învăţământ tehnic superior arădean", Arad, 30-31 Octombrie, 1997, vol VIII, pag. 261-268.
- [91] Lascu, D., *Realizări şi tendinţe actuale în convertoare rezonante*, referatul no. 1 în cadrul stagiului de pregătire a doctoratului.
- [92] Lascu, D., *Utilizarea tehnicii de calcul în studiul comutaţiei rezonante*, referatul no. 2 în cadrul stagiului de pregătire a doctoratului.
- [93] van Duijsen P. J., Lascu, D., "Simulation of power electronics and drive systems", *Proceedings of the Symposium on Electronics and Telecommunications*, Timişoara, Sept. 29-30, 1994, vol. I, pp. 201-206.
- [94] van Duijsen P. J., Lascu, D., "Small signal analysis using simulation", *Proceedings of the Symposium on Electronics and Telecommunications*, Timişoara, Sept. 29-30, 1994, vol. I, pp. 207-2013.
- [95] Mureşan, T., Popescu, V., Lascu, D., Neag, I., Negoitescu, D., Popovici, A., *Convertoare electronice de energie cu poluare redusă a reţelei şi factor de putere unitar*, Rapoarte de fază şi finale la Granturile de cercetare 4004/95, 5004/96, 7004/97 şi 8004/98 finanţate de Ministerul Învăţământului - Consiliul Naţional al Cercetării Ştiinţifice Universitare.
- [96] Mureşan, T., Popescu, V., Lascu, D., "Optimisation of parameters for welding power supplies", *Proceedings of the 4<sup>th</sup> International Conference on Optimization of Electrical and Electronic Equipments, OPTIM'94*, Braşov, May 12-14, 1994, vol. II, pp.41-46.
- [97] Negoitescu, D., Lascu, D., "Small-signal models and dynamics for buck-boost converter topologies", *Analele Universităţii "Eftimie Murgu" Reşiţa - sesiune jubiliară - "Un sfert de veac de la fondarea învăţământului superior la Reşiţa"*, 1996, fascicola III, pp. 209-214.
- [98] Popescu, V., Lascu, D., "A versatile circuit for power factor improvement of dc converters", *Proceedings of the 4<sup>th</sup> International Conference on Optimization of Electrical and Electronic Equipments, OPTIM'94*, Braşov, May 12-14, 1994, vol. II, pp. 83-89.
- [99] Popescu, V., Lascu, D., "Class E<sup>2</sup> switched mode power supply with zero-voltage switching", *Proceedings of the 4<sup>th</sup> International Conference on Optimization of Electrical and Electronic Equipments, OPTIM'94*, Braşov, May 12-14, 1994, vol. II, pp. 77-82.
- [100] Popescu, V., Lascu, D., "Small-signal models and dynamics for boost converter topologies", *Proceedings of the Symposium on Electronics and Telecommunications*, Timişoara, Sept. 29-30, 1994, vol. I, pp. 214-219.
- [101] Popescu, V., Andreiciuc, D., Lascu, D., *Instalaţie electronică pentru reglarea fluxului luminos la tuburile fluorescente din copiatorul EM 115*, contract de cercetare 180/90 cu Electromureş Târgu-Mureş - raport final.
- [102] Popescu, V., Andreiciuc, D., Lascu, D., "Instalaţie electronică pentru reglarea fluxului luminos la tuburile fluorescente", *Buletinul ştiinţific şi tehnic al Universităţii Tehnice din Timişoara*, Tomul 37 (51), fascicola 1-2, ianuarie-decembrie 1992, pp. 91-94.
- [103] Lee, Y. S., Wang, S. J., Hui S. Y. R., "Modelling analysis and application of buck converters in discontinuous-input-voltage mode operation", *IEEE Trans. Power Electron.*, vol. 12, no 2, March, 1997, pp. 350-360.
- [104] Lehman, B. Bass, R. M., "Switching frequency dependent averaged models for PWM dc-dc converters", *IEEE Trans. Power Electron.* vol. 11, no. 1, Jan., 1996, pp. 89-98.
- [105] Lin, B. T., Lee, Z. S., "A unified approach to modeling, synthesizing and analyzing quasi-resonant converters", *IEEE Trans. Power Electron.* vol. 12, no. 6, Nov., 1997, pp. 983-992.
- [106] Liu, K. H., Lee, F. C., "Resonant switches- A unified approach to improve performances of switching converters", *Proc. IEEE INTELEC*, 1984, pp.344-351.
- [107] Liu, K. H., Lee, F. C., "Zero-voltage switching technique in dc/dc converters", *IEEE/PESC Rec.*, 1986, pp. 58-70.
- [108] Liu, K. H., Oruganti, R., Lee, F. C., "Resonant switches - topologies and characteristics", *IEEE/PESC, 1985 Rec.*, pp. 106-116.
- [109] Liu, R., Lee, C. Q., Batarseh, I., "A unified approach to the design of resonant power factor correction circuits", *IEEE PESC Record*, 1992, pp. 181-188.
- [110] Lotfi, A. W., Vorperian, V., Lee, F. C., "Comparison of stresses in quasi-resonant and pulse-width-modulated converters", *IEEE/PESC Conf. Rec.*, Kyoto, Japan, April 11-14, 1988, pp. 591-598.
- [111] Luciano, A. M., Strollo, A. G. M., "A fast time-domain algorithm for the simulation of switching power converters", *IEEE Trans. Power Electron.*, vol. 5, no 4, July, 1990, pp. 363-370.
- [112] Ma, Y., Smedley, K. M., "Switching flow-graph nonlinear modeling method for multistate switching converters", *IEEE Trans. Power Electron.*, vol. 12, no. 5, Sept., 1997, pp. 854-861.

- [113] Makowski, M. S., Bailly, A., "Switch voltage control versus switch current control method. Principle and realization", *IEEE/PESC Conf. Rec.*, 1992, pp. 749-761.
- [114] Maksimović, D., Čuk, S., "A general approach to synthesis and analysis of quasi-resonant converters", *IEEE Trans. Power Electron.* vol. 6, no. 1, Jan., 1991, pp. 127-140.
- [115] Maksimović, D., Čuk, S., "A unified analysis of PWM converters in discontinuous modes" *IEEE Trans. Power Electron.* vol. 6, no. 3, July., 1991, pp. 476-490.
- [116] Maksimović, D., Čuk, S., "Constant frequency control of quasi-resonant converters", *IEEE Trans. Power Electron.* vol. 6, no. 1, Jan., 1991, pp. 141-150.
- [117] Maksimović, D., Čuk, S., "General properties and synthesis of PWM dc-to-dc converters", *IEEE Power Electronics Specialists Conf. Rec.*, 1989, pp. 515-525.
- [118] Maksimović, D., Čuk, S., "Switching Converters with wide dc conversion range", *IEEE Trans. Power Electron.* vol. 6, no. 1, Jan., 1991, pp. 151-157.
- [119] Maksimović, D., "Automated steady-state analysis of switching power converters using a general-purpose simulation tool", *IEEE PESC*, St. Louis, Missouri, June., 22-27, 1997.
- [120] Maksimović, D., "Automated small-signal analysis of switching converters using a general-purpose time-domain simulator", *IEEE APEC*, Anaheim, CA, Feb., 15-19, 1998.
- [121] Malesani, L., Rosetto, L., Spiazzi, "Small-signal analysis of dc-dc converters with sliding mode control", *IEEE Trans. Power Electron.*, vol. 12, no 1, January, 1997, pp. 96-102.
- [122] Malesani, L., Rosetto, L., Spiazzi, G, Tenti, P., "Performance optimization of Čuk converters by sliding-mode control", *IEEE Trans. Power Electron.*, vol. 10, no. 3, May, 1995, pp. 302-309.
- [123] MATLAB - Reference Manual, MathWorks Inc.
- [124] Matsuo, H., Harada, K., "The cascade connection of switching regulators", *IEEE Trans. Ind. Appl.*, vol. IA-3, no. 2, Mar./Apr. 1976.
- [125] Mattavelli, P., Rosetto, L., Spiazzi, G., Tenti, P., "General-purpose fuzzy controller for dc-dc converters", *IEEE Trans. Power Electron.* vol. 12, no. 1, Jan, 1997, pp. 79-86.
- [126] Middlebrook, R. D., Čuk, S., "A general unified approach to modeling switching-converter power stages", *Int. Journ. of Electron.*, vol. 42, no. 6, June, 1977, pp. 521-550.
- [127] Middlebrook, R. D., Čuk, S., "Modeling and analysis methods for dc-to-dc switching converters", *Proc. of the IEEE International Semiconductor. Power Converter Conference*, 1977 Record, March, 1977, pp. 90-111.
- [128] Middlebrook, R. D., Čuk, S., *Advances in switched-mode power conversion*, vol I-III, Pasadena, CA: TESLAcO, 1983.
- [129] Middlebrook, R. D., "Topics in multiple-loop regulators and current-mode programming", *IEEE Trans. Power Electron.* vol. 2, no. 2, April, 1987, pp. 109-124.
- [130] Mohan, N., Undeland, T. M., Robbins, W. P., *Power electronics: converters, applications and design*, New-York, Wiley, 1989.
- [131] Moszyński, K., *Metode numerice de rezolvare a ecuațiilor diferențiale ordinare*, Editura Tehnică, București, 1973.
- [132] Nakahara, M, Higashi, T., Ninomiya, T., Harada, K., "Dynamic characteristics and stability analysis of resonant converters", *IEEE/PESC Conf. Rec.*, 1989, pp. 752-759.
- [133] Nakahara, M., "A fast computer algorithm for switching converters", *IEEE Trans. Power Electron.*, vol. 12, no 1, January, 1997, pp. 180-186.
- [134] Nakahara, M., Ninomiya, "A general computer algorithm for analysis of switching converters", *IEEE/PESC Conf. Rec.*, 1992, pp. 1181-1188.
- [135] Nakhla, M. S., Branin, F. H., "Determining the periodic response of nonlinear systems by a gradient method", *International Journal of Circuit Theory*, vol. 5, no. 3, 1977, pp. 255-273.
- [136] Ngo, K. D. T., "Generalization of resonant switches and quasi-resonant dc-dc converters ", *IEEE/PESC Rec.*, 1987, pp. 395-403.
- [137] Ngo, K. D. T., "Low frequency characterization of PWM converters", *IEEE Trans. Power Electron.*, vol. PE-1, no. 4, Oct., 1986, pp. 223-230.
- [138] Ninomiya, T., Nakahara, M., Harada, K., "A unified analysis of resonant converters", *IEEE Trans. Power Electron.*, vol. 6, no 2, April, 1991, pp. 260-269.
- [139] Ooi, B. T., "Fast steady state solution for HVDC analysis", *IEEE Transactions on Power Apparatus and Systems*, vol. PAS-99, no. 6, 1980, pp 2453-2459.
- [140] Oruganti, R, Lee, F. C., "Implementation of optimal trajectory control of series resonant converter", *IEEE/PESC Conf. Rec.*, 1987, pp. 451-459.
- [141] Oruganti, R, Lee, F. C., "State plane analysis of parallel resonant converter", *IEEE/PESC Conf. Rec.*, 1985, pp. 56-73.
- [142] Oruganti, R., Lee, F. C., "Resonant power processors. Part 1: State plane analysis", *IEEE Trans. Ind. Appl.* vol. 12, no. 6, Nov/Dec, 1985, pp. 1453-1461.
- [143] Oruganti, R., Lee, F. C., "Resonant power processors. Part 2: Methods of control", *IEEE Trans. Ind. Appl.* vol. 12, no. 6, Nov/Dec, 1985, pp. 1461-1471.

- [144] Patterson, O. D., Divan, D. M., "Pseudo-resonant full bridge dc/dc converter", *IEEE/PESC Conf. Rec.*, 1987, pp. 424-430.
- [145] Pavljašević, S., Maksimović, D., "Subharmonic oscillations in converters with current-mode programming under large parameter variations", *IEEE/PESC Conf. Rec.*, 1997.
- [146] Pedrycz, W., *Fuzzy Control and Fuzzy Systems*, New York: Wiley, 1989.
- [147] Pejović, P., "A method for simulation of power electronic systems using piecewise-linear device models", Ph.D. dissertation, University of Colorado, Boulder, 1995.
- [148] Pejović, P., Maksimović, D., "A method for fast time-domain simulation of networks with switches", *IEEE Trans. Power Electron.*, vol. 9, no 4, July, 1994, pp. 449-456.
- [149] Pejović, P., Maksimović, D., "A new algorithm for simulation of power electronic systems using piecewise-linear device models", *IEEE Trans. Power Electron.*, vol. 10, no 3, May, 1995, pp. 340-348.
- [150] Pejović, P., Maksimović, D., "An algorithm for solving piecewise-linear networks that include elements with discontinuous characteristics", *IEEE Transactions on Circuits and Systems I*, vol. 43, no 6, June, 1996, pp. 453-460.
- [151] Pejović, P., Maksimović, D., "Determination of network state in switching power converters", *IEEE/PESC Conf. Rec.*, Atlanta, GA, June 22-27, 1995, pp. 816-822.
- [152] Pejović, P., Maksimović, D., "PETS - a simulation tool for power electronics", *Workshop on Computers in Power Electronics*, August 11-14, 1996, Portland, Oregon.
- [153] Pejović, P., Maksimović, D., *PETS - Power Electronics Transient Simulator - Reference Manual*.
- [154] Perreault, D. J., Verghese, G. C., "Time-varying effects and averaging issues in models for current-mode control", *IEEE Trans. Power Electron.*, vol. 12, no 3, May, 1997, pp. 453-461.
- [155] Phillips, C. L.; Nagle, H. T., *Digital Control System Analysis and Design*, Prentice Hall, 1984.
- [156] Popescu, V., *Electronică de putere*, Editura de Vest, Timișoara, 1998.
- [157] Popescu, V., *Stabilizatoare de tensiune în comutație*, Editura de Vest, Timișoara, 1991.
- [158] Rajagopalan, V., *Computer-aided analysis of power electronic systems*, Marcel Dekker, New York, 1987.
- [159] Rajagopalan, V., et al., "Computation of almost periodic steady state response of power electronic converter systems", in *Electrical Machines and Converters-Modelling and Simulation*, Elsevier Science Publishers (North-Holland), IMACS, 1984, pp. 133-138.
- [160] Redl, R., Sokal, N. O., "Current-mode control, five different types, used with three basic classes of power converters: small signal AC and large signal DC characterization, stability requirements, and implementation of practical circuits" *IEEE/PESC Conf. Rec.*, 1985, pp. 771-785.
- [161] Ridley, R. B., "A new, continuous time model for current-mode control", *IEEE Trans. Power Electron.*, vol. 6 no 2, April, 1991, pp. 271-280.
- [162] Sabate, J. A., Lee, F. C., "Off-line application of the fixed frequency clamped mode series resonant converter", *Proceedings of the Applied Power Electronics Conference*, Baltimore, Maryland, March 13-17, 1989, pp. 395-402.
- [163] Salvadori, M. G., Baron, M. L., *Numerical methods in engineering*, Prentice-Hall, Englewood Cliffs, New Jersey, 1972.
- [164] Sanders, S. R., Noworolski, J. M., Liu, X. Z., Verghese, G. C., "Generalized averaging method for power conversion circuits", *IEEE Trans. Power Electron.*, vol. 6, no. 2, April, 1991, pp. 254-259.
- [165] Santi, E., Čuk, S., "Comparison and design of three coupled inductor structures", *Proceedings of the 20<sup>th</sup> IEE International Conference on Industrial Electronics, Control and Instrumentation*, Bologna, Italy, 1994, Sept. 5-9, vol. 1, pp. 262-267.
- [166] Sastry, V. V., Prabhakaran, N., "Digital simulation of voltage-controlled AC motor drives", *Electrical Machines and Electromechanics*, 1979, pp. 233-253.
- [167] Severns, R. P., Bloom, G. E., *Modern dc-to-dc switchmode power converter circuits*, Van Nostrand Reinhold, New-York, 1985.
- [168] da Silva, B., April, G. E., Oliver, G., "Real time adaptive controller for an asymmetrical four quadrant power converter", *IEEE-IAS Annual Meeting Conf. Rec.*, 1987, pp. 872-878.
- [169] Skelboe, S., "Computation of the periodic steady-state response of nonlinear networks by extrapolation methods" *IEEE Trans. on Circuits and Systems*, vol. CAS-27, no. 3, March, 1980.
- [170] Smedley, K. M., Čuk, S., "Dynamics of one-cycle control of switching converters", *IEEE Trans. Power Electron.* vol. 10, no. 6, Nov., 1987, pp. 634-639.
- [171] Smedley, K. M., Čuk, S., "One-cycle control of switching converters", *IEEE Trans. Power Electron.* vol. 10, no. 6, Nov, 1995, pp. 625-633.
- [172] Smedley, K. M., Čuk, S., "One-cycle control of switching converters", *IEEE/PESC Conf. Rec.*, 1991, pp. 888-896.
- [173] Smedley, K. M., Čuk, S., "Switching flow-graph nonlinear modelling technique", *IEEE/PESC Conf. Rec.*, 1992, pp. 1173-1180.
- [174] So, W. C., Tse, C. K., Lee, Y. S: "Development of a fuzzy logic controller for dc/dc converters: design, computer simulation and experimental evaluation", *IEEE Trans. Power Electron.*, vol 11, no.1, Jan. 1996, pp. 24-32.

- [175] Spiazzi, G., Mattavelli, P., Rosetto, L., "Power factor preregulators with improved dynamic response", *IEEE Trans. Power Electron.*, vol. 12, no. 2, March, 1997, pp. 343-349.
- [176] SPICE - Reference Manual, Microsim Corp.
- [177] Sudha, S. A., Chandrasekaran, A., Rajagopalan, V., "New approach to switch modelling in the analysis of power electronic systems", *IEE Proceedings - B*, vol. 140, no. 2, March, 1993, pp. 115-123.
- [178] Sun, J. Grotstollen, H., "Averaged modeling of switching power converters: Reformulation and theoretical basis", *IEEE/PESC Conf. Rec.*, 1992, pp. 1165-1172.
- [179] Sun, J. Grotstollen, H., "Symbolic analysis methods for averaged modeling of switching power converters", *IEEE Trans. Power Electron.*, vol. 12, no. 3, May, 1997, pp. 537-546.
- [180] Tabisz, W. A., Lee, F. C., "Zero-voltage switching multi-resonant technique - A novel approach to improve performance of high-frequency quasi-resonant converters", *IEEE/PESC Rec.*, 1988, pp. 9-17.
- [181] Tang, W. Lee, F. C., Ridley, R. B., "Small-signal modeling of average current-mode control", *IEEE Trans. Power Electron.*, vol. 8, no. 2, April, 1993, pp. 112-119.
- [182] Tang, W., Lee, F. C., Ridley R., B., Cohen, I., "Charge Control: modelling, analysis and design", *IEEE Power Electronics Specialists Conf. 1992. Rec.*, pp. 503-511.
- [183] Tang, W., Leu, C. S., Lee, F. C., "Charge control for zero-voltage-switching multiresonant converter", *IEEE Trans. Power Electron.*, vol. 11, no 2, March, 1996, pp. 270-274.
- [184] Tarter, R. E., *Solid-state power conversion handbook*, Wiley, 1993.
- [185] Texas Instruments, "TMS320C30 Evaluation Module", *Tech. Ref., Texas Instruments Inc.*, 1992.
- [186] Texas Instruments, "TMS320C30 User's Guide", *Texas Instruments Inc.*, 1992.
- [187] Thottuvelil, J., "Comparing Spice with other circuit simulation tools for power electronics analysis", *Workshop on Computers in Power Electronics*, August 11-14, 1996, Portland, Oregon.
- [188] Trabert, S. G., Erickson, R. W., "Steady state analysis of the duty cycle controlled series resonant converter" *IEEE/PESC Conf. Rec.*, 1987, pp. 545-556.
- [189] Tsai, F. S., Lee, F. C., "A complete characterization of a constant-frequency, clamped mode, series resonant converter", *Proceedings of the Power Electronics Specialists Conference*, Kyoto, Japan, April 11-14, 1988, pp. 987-996.
- [190] Tsai, F. S., Lee, F. C., "Constant-frequency phase-controlled resonant power processor", *Industry Applications Society Annual Meeting Conference Record*, Denver, Colorado, Sept. 38- Oct. 3, 1986, pp. 617-622.
- [191] Tsai, F. S., Materu, P. Lee, F. C., "Constant-frequency, clamped mode resonant converters", *IEEE Trans. Power Electron.* vol. 3, no. 4, Oct., 1988, pp. 460-473.
- [192] Tymerski, R., "Application of the time-varying transfer function for exact small-signal analysis", *IEEE Trans. Power Electron.*, vol. 9, no 2, March, 1994, pp. 196-205.
- [193] Tymerski, R., "Frequency analysis of time-interval modulated switched networks", *IEEE Trans. Power Electron.*, vol. 6, no 2, April, 1991, pp. 287-295.
- [194] Tymerski, R., Li, D., "State space models for current programmed pulse width modulated converters", *IEEE Trans. Power Electron.*, vol. 8, no 3, July, 1993, pp. 271-278.
- [195] Tymerski, R., Vorperian, V., "Generation, classification and analysis of switched-mode dc-to-dc converters by the use of converter cells", *Proceedings International Telecommunications Energy Conference*, pp. 181-195, October, 1986.
- [196] Tymerski, R., Vorperian, V., Lee, F. C. Y., Baumann, W. T., "Nonlinear modeling of the PWM switch", *IEEE Trans. Power Electron.*, vol.4, no. 2, April, 1989, pp. 225-233.
- [197] Unitrode Corporation, "Off-line power supply controller (UCC3889)", *Unitrode Application Note U-149*, 1994.
- [198] Verghese, G. C., Elbuluk, M. E., Kassakian, J. G., "A general approach to sampled data modeling for power electronic circuits", *IEEE Trans. Power Electron.* no. 4, 1986, pp. 76-87.
- [199] Visser, H. R., van den Bosch, P. P. J., "Modelling of periodically switching networks", *IEEE/PESC Conf. Rec.*, 1991, pp. 1-8.
- [200] Vorperian, V., Čuk, S., "Small signal analysis of resonant converters", *IEEE/PESC Conf. Rec.*, 1985, pp. 269-282.
- [201] Vorperian, V., "Approximate small signal analysis of the series and parallel resonant converters", *IEEE Trans. Power Electron.*, no. 1, Jan., 1989, pp. 15-24.
- [202] Vorperian, V., "High-Q approximate small-signal analysis of resonant converters", *IEEE/PESC Conf. Rec.*, Toulouse, France, June 24-28, 1985, pp. 707-715.
- [203] Vorperian, V., "Quasi-square-wave converters: Topologies and analysis", *Proc. VPEC Conference*, 1987, pp. 72-80.
- [204] Vorperian, V., Tymerski, R., Lee, F. C., "Equivalent circuit models for resonant and PWM switches", *IEEE Trans. Power Electron.*, vol. 4, no. 2, April, 1989, pp. 205-214.
- [205] Vorperian, V.; Tymerski, R.; Lee, F. C., "Generalised Resonant Switches, Analysis and Circuit Models". *IEEE Trans. Power Electron.* vol. 4, no. 2, April, 1989, pp. 205-214.
- [206] Wang, J., Dunford, W., Mauch, K. "Analysis of a ripple-free input current boost converter with discontinuous conduction characteristics", *IEEE Trans. Power Electron.* vol. 12, no. 4, pp. 684-694.

- [207] Wang, J., Dunford, W., Mauch, K. "Design of boost and buck converters with coupled inductor", Proceedings of the *IEEE Power Electronics Specialists Conference PESC'95*, 1995, Atlanta (GA), June 18-22, pp. 273-279 .
- [208] White, J., Leeb, S. B., "An envelope-following approach to switching power converter simulation", *IEEE Trans. Power Electron.*, vol. 6, no 2, April, 1991, pp. 303-313.
- [209] Witulski, A. F., "Buck converter small-signal models and dynamics: comparison of quasi-resonant and pulsewidth modulated switches", *IEEE Trans. Power Electron.*, vol. 6, no 4, Oct, 1991, pp. 727-738.
- [210] Witulski, A. F., Erickson, R. W., "Extension of state space averaging to resonant switching - and beyond", *IEEE Trans. Power Electron.*, vol. 5, no 1, Jan., 1990, pp. 98-109.
- [211] Witulski, A. F., Erickson, R. W., "Small signal AC equivalent circuit modelling of the series resonant converter", *IEEE IAS Annual Meeting*, 1987, pp. 693-704.
- [212] Witulski, A. F., Hernandez, A. F., Erickson, R. W., "Small signal equivalent circuit modelling of resonant converters", *IEEE Trans. Power Electron*, no. 1, Jan., 1991, pp. 11-16.
- [213] Wong, L. K., Leung, F. H., Tam, P. K. S., "A simple large-signal nonlinear modeling approach for fast simulation of zero-current-switch quasi-resonant converters", *IEEE Trans. Power Electron.*, vol. 12, no 3, May, 1997, pp. 437-442.
- [214] Wong, R. C., "Accelerated convergence to steady-state solution of closed-loop regulated switching-mode systems as obtained through simulation", *IEEE/PESC Conf. Rec.*, 1987, pp. 682-692.
- [215] Wong, R. C., Owen Jr., H. A., Wilson, T. G., "An efficient algorithm for the time-domain simulation of regulated energy-storage dc-to-dc converters", *IEEE Trans. Power Electron.*, vol. PE-2, no 2, April, 1987, pp. 154-167.
- [216] Zacharias, Th., Miliias-Argitis, J, Makios, V., "Loop admittance matrix modification for the digital simulation of power converter circuits", *International Journal of Circuit Theory and Applications*, vol. 21, 1993, pp. 41-59.
- [217] Zadeh, L., "Frequency analysis of time-interval-modulated networks", *IRE Proc.*, vol. 38, pp. 291-299, 1950.
- [218] Zane, R., Maksimović, D., "Modeling of high power factor rectifiers based on switching converters with nonlinear-carrier control", *IEEE PESC 96*, pp. 1105-1111.
- [219] Zane, R., Maksimović, D., "Nonlinear-carrier control for high-power-factor rectifiers based on flyback, Ćuk or Sepic converters", *Proceedings of IEEE APEC*, March 3-7, 1996, San Jose, CA, pp. 814-820.
- [220] Zhou, C., Ridley, R. B., Lee, F. C., "Design and analysis of a hysteretic boost power factor correction circuit", *IEEE/PESC Conf. Rec.*, 1990, pp. 800-807.



## ANEXA

**A1 Fișierul PETS pentru simularea noului convertor BUCK cu curent de intrare cu pulsații reduse**

fișierul buck\_nou.cir

\*Noua topologie BUCK cu curent de intrare cu pulsatii reduse (Dan Lascu)

```

* convertor
Vg 1 0 dc 30V
L1 1 2 828u
RL1 2 3 0.51
C1 3 4 102u
L2 5 4 521u
RL2 5 0 0.19
k L1 L2 0.1
#vcs 3 6 100 0 vcs
#d 4 6 d
Co 6 0 12.8u
R 6 0 10.07

* comanda
v2 100 0 pulse -5 10V 0 10u 25u

.tran 0.2us 25m 23m 0.2u

* modele
.model vcs #vcs
.model d #d

* relaxeaza tolerantele
.options reltol=0.01 vntol=1e-4 abstol=1e-9
.probe i(L1) i(L2) v(4) v(5) v(6)
.end

```

**A2 Fișierul PETS pentru simularea noului convertor BOOST cu curent de intrare cu pulsații reduse**

fișierul boos\_nou.cir

\*Noua topologie BOOST cu curent de iesire cu pulsatii reduse (Dan Lascu)

```

* convertor
Vg 1 0 dc 10V
#vcs 1 2 100 0 vcs
#d 1 4 d
L1 2 3 600u
RL1 3 0 0.68
C1 4 2 199.7u
L2 4 5 625u
RL2 5 6 0.35
Co 6 0 21.04u
R 6 0 68.02

* comanda
v2 100 0 pulse -5V 10V 0u 15u 25u

.tran 0.2u 20m 18m 0.2u

* modele
.model vcs #vcs
.model d #d

* relaxeaza tolerantele
.options reltol=0.01 vntol=1e-4 abstol=1e-9
.probe i(L1) i(L2) v(2) v(3) v(6)
.end

```

### A3 Fișierul PETS pentru simularea noului convertor BOOST patratic

fișierul boost2.cir

```

Convertor boost patratic (boost2.cir)

* convertor
vg      1 0 DC 10V
L       1 2 100u
#d1prime 2 4 d
#d2prime 2 3 d
C1      3 0 4.7u
L1      3 4 2m
#vcs1   4 0 100 0 vcs
#d2     4 5 d
C       5 0 2.2u
R       5 0 2k

* comanda intrerupatorului
vc      100 0 pulse 0 10V 0 15u 20u

.tran 0.1us 50ms 49.5m 0.1u

* modele
.model vcs #vcs (Vton=4.9V Vtoff=5.1V Ron=0.01 Goff=100n)
.model d #d (Ron=0.01 Goff=100n)

* relaxeaza tolerantele
.options reltol=0.01 vntol=1e-4 abstol=1e-9

.probe i(L1) i(L) v(3) v(5) i(D1prime) i(D2prime) i(D2)

.end

```

### A4 Programul de trasare a caracteristicilor de ieșire normate ale lui $\mu$ în CCM și DICM

```

% Caracteristicile statice normalizate pentru convertoare DIVM, miu=f(ION)
clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

ION=0:0.01:1; D=0.1:0.2:0.9;
for j=1:length(D)
    for i=1:length(ION)
        if D(j)^2/(D(j)^2+ION(i))>=D(j)
            miu(i)=D(j)^2/(D(j)^2+ION(i));
        else
            miu(i)=D(j);
        end
    end
end

plot(ION,miu,'k'); hold on; xlabel('ION'); ylabel('miu'); axis([0 1 0 1]);
end

```

### A5 Programul de trasare a caracteristicilor de ieșire normate ale lui $\mu$ în CCM și DIVM

```

% Caracteristicile statice normalizare pentru convertoare DIVM, miu=f(ION)
clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

ION=0:0.01:10; D=0.2:0.2:1;
for j=1:length(D)
    for i=1:length(ION)
        if 1/(1+ION(i))*(1-D(j))^2<=D(j)
            miu(i)=1/(1+ION(i)*(1-D(j))^2);
        else
            miu(i)=D(j);
        end
    end
end

plot(ION,miu,'k'); hold on; xlabel('ION'); ylabel('u0');

```

end

## A6 Programul de simulare CASPOC al unui convertor BOOST DIVM

Fișierul *boostdiv.cir*

```
* Convertor BOOST functionand in mod DIVM

Vg 1 0 10
L 1 2 3.41m
L1 2 3 11m
S 3 0
C1 3 4 48.9n
D 4 0 did
C 2 4 4.7u
RL 2 4 50
.model did DSWITCH vtho=0 vthof=0 ron=1u roff=10meg
.tran 50n 10 uic
.options Tscreen=5m method=euler
.draw 1 v(3,4)
.draw 2 v(3)
.draw 3 v(0,4)
.draw 4 i(L)
.draw 5 i(L1)
.end
```

Fișierul *boostdiv.stm*

```
* Circuit de corectie a factorului de putere utilizand un
* convertor BUCK functionand in conductie discontinua dpdv al
* tensiunii de intrare.

t time
gate signal t 7.5 7.5 50k 0 0.4 0 0 1
S switch gate 7.5 7.5 S
.end
```

## A7 Programul de trasare a caracteristicilor control-ieșire normalizate pentru convertoare QRC

Fișierul *qrcstat.m*

% Caracteristicile statice ale convertoarelor QRC .

```
disp('Alegeti tipul de convertor dorit a se analiza');
disp(' 1 - BUCK. ');
disp(' 2 - BOOST. ');
disp(' 3 - BUCK-BOOST. ');
disp(' 0 - Abandonati analiza ');
type=input('Selectati un număr și apoi apasati tasta "Enter": ');
fprintf('\n');
while (type==0)&(type==1)&(type==2)&(type==3);
    disp('Numar incorect (numarul trebuie sa fie 1,2,3 sau 0)');
    type=input('Selectati un număr și apoi apasati tasta "Enter": ');
fprintf('\n');
end;

if(type==0)
    disp('Alegeti tipul comutarii. ');
    disp(' 1 - comutare la curent nul (ZCS). ');
    disp(' 2 - comutare la tensiune nula (ZVS). ');
    disp(' 0 - Abandonati analiza ');
    zs=input(' Selectati un număr și apoi apasati tasta "Enter": ');
    fprintf('\n');
    while (zs==0)&(zs==1)&(zs==2);
        disp('Numar incorect (numarul trebuie sa fie 1,2 sau 0)');
        n=input('Selectati un număr și apoi apasati tasta "Enter": ');
    end;
    fprintf('\n');
    if (zs==0)
        disp('Alegeti modul dorit a se analiza. ');
        disp(' 1 - semiunda (HW). ');
        disp(' 2 - unda plina (FW). ');
        disp(' 0 - Abandonati analiza. ');
        n=input(' Selectati un număr și apoi apasati tasta "Enter": ');
    end;
end;
```

```

fprintf('\n');
while (n~=0)&(n~=1)&(n~=2);
    disp('Numar incorect (numarul trebuie sa fie 1,2 sau 0)');
    n=input(' Selectati un număr și apoi apasati tasta "Enter" : ');
end;
fprintf('\n');
if (n~=0)
    if type==1
        buck
    elseif type==2
        boost
    else
        bucbboost
    end
else
    disp('Parasire program');
end

else
    disp('Parasire program');
end

disp('Parasire program');
end

```

Fișierul buck.m

```

Rn=input('Introduceti valorile lui Rn intre "[ ]" si separate de blankuri sau virgule: ');
clear xextend;
clear yextend;
clear abscisa;
clear fnorm;
clear fN;
clear y;
clear M;
clear step;
clg;
m=0.01;
j=1;
delta=0.00001;
for r=1:length(Rn)
    if zs==1
        a=1E-8;
        b=Rn(r);
        xmax1=(2.*pi.*Rn(r))./quasi(1,n);
    else
        a=Rn(r);
        b=1;
        xmax1=(1-Rn(r)).*(2.*pi)./quasi(1,n);
    end
% xmax se alege pt. y=Rn deoarece functia x=f(y) este strict crescatoare și deci la fel va fi si
% inversa ei y=f-1(x)
% deci cea mai mare valoare permisa pt. xmax se obtine cand y=Rn (pt ZCS e necesar y<=Rn)
% si o mica valoare, delta, este scazuta pt. a asigura puncte de pornire corecte in aloritmul numeric.
    xmax(r)=xmax1-delta;
% Intervalul (0,xmax(r)) este impartit in pasi de aprox. m=0.01;
    step=round(xmax(r)/m);
    abscisa=xmax(r)/step:xmax(r):xmax(r);
    i=1;
    q=1;
    for k=1:step,
        x=abscisa(k);
        s=sol('buckzs',x,Rn(r),n,zs,a,b);
% Testeaza daca capacitatea rezonanta este descarcata la 0 intr-o perioada
% si retine numai acele puncte care satisfac conditia (MODUL I de functionare).
% fN contine (pt. fiecare Rn) valoarea maxima permisa pt. fN care inca asigura descarcarea la 0.
% Matricea M valorile pt toti parametrii, in timp ce vectorul y este sters la fiecare
% Rn pt. a obtine un grafic corect (altfel unele coloane din y ar putea fi umplute cu zerouri),
% cum se intampla in M.
        if zs==1
            if x<=(2.*pi)./(quasi(s./Rn(r),n)+s./(2.*Rn(r)))
                y(q)=s;
                fnorm(q)=abscisa(k);
            end
        end
    end
end

```

```

    fN(r)=abscisa(k);
    M(i,j)=s;
    i=i+1;
    q=q+1;
    else
    i=i+1;
    end
else
    if x<=(2.*pi)./(quasi(Rn(r)./s,n)+Rn(r)./(2.*s))
    y(q)=s;
    fnorm(q)=abscisa(k);
    fN(r)=abscisa(k);
    M(i,j)=s;
    i=i+1;
    q=q+1;
    else
    i=i+1;
    end
end
end

% Graficul se va extinde la valoarea maxima permisa pt x si la fel pt. y.
yextend(r)=max(y);
xextend(r)=max(fnorm);
plot(fnorm,y);
l=length(y);
text(fnorm(l),y(l),num2str(Rn(r)));
grid;
axis([0 max(xextend) 0 max(yextend)]);
if (zs==1)&(n==1)
title('BUCK ZCS QRC HW - Caracteristici statice normalizate');
elseif (zs==1)&(n==2)
title('BUCK ZCS QRC FW - Caracteristici statice normalizate');
elseif (zs==2)&(n==1)
title('BUCK ZVS QRC FW - Caracteristici statice normalizate');
elseif (zs==2)&(n==2)
title('BUCK ZVS QRC FW - Caracteristici statice normalizate');
else
end
end
xlabel('f/f0');
ylabel('Vo/Vl');
j=j+1;
clear fnorm;
clear y;
hold on;
end

```

Fișierul boost.m

```

Rn=input('Introduceti valorile lui Rn intru "[ ]" si separate de blancuri sau virgule: ');
clear xextend;
clear yextend;
clear abscisa;
clear fnorm;
clear fN;
clear y;
clear M;
clear step;
clg;
m=0.01;
j=1;
delta=0.00001;
for r=1:length(Rn)
    if zs==1
    a=1E-8;
    b=Rn(r);
    xmax1=(2.*pi)/quasi(1,n).*(1-(Rn(r)).^(-1));
    else
    a=Rn(r);
    b=500;
    xmax1=(2.*pi)/(quasi(1,n).*Rn(r));
    end
% xmax se alege pt. y=Rn deoarece functia x=f(y) este strict crescatoare și deci la fel va fi si
% inversa ei y=f-1(x)
% deci cea mai mare valoare permisa pt. xmax se obtine cand y=Rn (pt ZCS e necesar y<=Rn)

```

```

% si o mica valoare, delta, este scazuta pt. a asigura puncte de pornire corecte in aloritmul numeric.
xmax(r)=xmax1-delta;
% Intervalul (0,xmax(r)) este impartit in pasi de aprox. m=0.01;
step=round(xmax(r)/m);
abscisa=xmax(r)/step:xmax(r)/step:xmax(r);
i=1;
q=1;
for k=1:step,
    x=abscisa(k);
    s=sol(x,Rn(r),n,zs,a,b);
% Testeaza daca capacitatea rezonanta este descarcata la 0 intr-o perioada
% si retine numai acele puncte care satisfac conditia (MODUL I de functionare).
% fN contine (pt. fiecare Rn) valoarea maxima permisa pt. fN care inca asigura descarcarea la 0.
% Matricea M valorile pt toti parametrii, in timp ce vectorul y este sters la fiecare
% Rn pt. a obtine un grafic corect (altfel unele coloane din y ar putea fi umplute cu zerouri),
% cum se intampla in M.
if zs==1
    if x<=(2.*pi)/(quasi(s./Rn(r),n)+s./(2.*Rn(r)))
        y(q)=s;
        fnorm(q)=abscisa(k);
        fN(r)=abscisa(k);
        M(i,j)=s;
        i=i+1;
        q=q+1;
    else
        i=i+1;
    end
else
    if x<=(2.*pi)/(quasi(Rn(r)./s,n)+Rn(r)/(2.*s))
        y(q)=s;
        fnorm(q)=abscisa(k);
        fN(r)=abscisa(k);
        M(i,j)=s;
        i=i+1;
        q=q+1;
    else
        i=i+1;
    end
end
end
% Graficul se va extinde la valoarea maxima permisa pt x si la fel pt. y.
yextend(r)=max(y);
xextend(r)=max(fnorm);
plot(fnorm,y);
l=length(y);
text(fnorm(l),y(l),num2str(Rn(r)));
grid;
axis([0 max(xextend) 0 max(yextend)]);
if (zs==1)&(n==1)
    title('BOOST ZCS QRC HW - Caracteristici statice normalizate');
elseif (zs==1)&(n==2)
    title('BOOST ZCS QRC FW - Caracteristici statice normalizate');
elseif (zs==2)&(n==1)
    title('BOOST ZVS QRC FW - Caracteristici statice normalizate');
elseif (zs==2)&(n==2)
    title('BOOST ZVS QRC FW - Caracteristici statice normalizate');
else
end
xlabel('f/f0');
ylabel('Vo/Vl');
j=j+1;
clear fnorm;
clear y;
hold on;
end

```

Fişierul *bucboost.m*

```

Rn=input('Introduceti valorile lui Rn intre "[ ]" si separate de blancuri sau virgule: ');
clear xextend;
clear yextend;
clear abscisa;
clear fnorm;
clear fN;
clear y;

```

```

clear M;
clear step;
clg;
m=0.01;
j=1;
delta=0.00001;
for r=1:length(Rn)
    if zs==1
        a=1E-8;
        b=Rn(r);
        xmax1=(Rn(r)/(Rn(r)+1)).*(2.*pi)/quasi(1,n);
    else
        a=Rn(r);
        b=500;
        xmax1=(Rn(r)+1).^(-1).*(2.*pi)/quasi(1,n);
    end
    % xmax se alege pt. y=Rn deoarece functia x=f(y) este strict crescatoare și deci la fel va fi și
    % inversa ei y=f-l(x)
    % deci cea mai mare valoare permisă pt. xmax se obține când y=Rn (pt ZCS e necesar y<=Rn)
    % și o mica valoare, delta, este scăzută pt. a asigura puncte de pornire corecte în algoritmul numeric.
    xmax(r)=xmax1-delta;
    % Intervalul (0,xmax(r)) este împărțit în pași de aprox. m=0.01;
    step=round(xmax(r)/m);
    abscisa=xmax(r)/step:xmax(r)/step:xmax(r);
    i=1;
    q=1;
    for k=1:step,
        x=abscisa(k);
        s=sol(x,Rn(r),n,zs,a,b,);
    % Testează dacă capacitatea rezonantă este descărcată la 0 într-o perioadă
    % și reține numai acele puncte care satisfac condiția (MODUL I de funcționare).
    % fN conține (pt. fiecare Rn) valoarea maximă permisă pt. fN care încă asigură descărcarea la 0.
    % Matricea M valorilor pt. toți parametrii, în timp ce vectorul y este sters la fiecare
    % Rn pt. a obține un grafic corect (altfel unele coloane din y ar putea fi umplute cu zerouri),
    % cum se întâmplă în M.
    if zs==1
        if x<=(2.*pi)/(quasi(s./Rn(r),n)+s./(2.*Rn(r)))
            y(q)=s;
            fnorm(q)=abscisa(k);
            fN(r)=abscisa(k);
            M(i,j)=s;
            i=i+1;
            q=q+1;
        else
            i=i+1;
        end
    else
        if x<=(2.*pi)/(quasi(Rn(r)/s,n)+Rn(r)/(2.*s))
            y(q)=s;
            fnorm(q)=abscisa(k);
            fN(r)=abscisa(k);
            M(i,j)=s;
            i=i+1;
            q=q+1;
        else
            i=i+1;
        end
    end
end
end
% Graficul se va extinde la valoarea maximă permisă pt x și la fel pt. y.
yextend(r)=max(y);
xextend(r)=max(fnorm);
plot(fnorm,y);
l=length(y);
text(fnorm(1),y(1),num2str(Rn(r)));
grid;
axis([0 max(xextend) 0 max(yextend)]);
if (zs==1)&(n==1)
    title('BUCK-BOOST ZCS QRC HW - Caracteristici statice normalizate');
elseif (zs==1)&(n==2)
    title('BUCK-BOOST ZCS QRC FW - Caracteristici statice normalizate');
elseif (zs==2)&(n==1)
    title('BUCK-BOOST ZVS QRC FW - Caracteristici statice normalizate');
elseif (zs==2)&(n==2)
    title('BUCK-BOOST ZVS QRC FW - Caracteristici statice normalizate');

```

```

else
end
xlabel('f/f0');
ylabel('Vo/Vf');
j=j+1;
clear fnorm;
clear y
hold on;
end

```

*Fișierul buckzs.m*

```

% Functia multivariabila ce definește ecuația transcendentă ce furnizează
% caracteristicile statice pt convertoare BUCK QRC.
function v=buckzs(x,y,Rn,n,zs);
if zs==1
v=y-1/(2.*pi).*x.*quasi(y./Rn,n);
else
v=y+x.*(quasi(Rn./y,n))/(2.*pi)-1;
end

```

*Fișierul boostzs.m*

```

% Functia multivariabila ce definește ecuația transcendentă ce furnizează
% caracteristicile statice pt convertoare BOOST QRC.
function v=boostzs(x,y,Rn,n,zs);
if zs==1
v=y.*(1-x)/(2.*pi).*quasi(y./Rn,n)-1;
else
v=y.*x/(2.*pi).*quasi(Rn./y,n)-1;
end

```

*Fișierul bucboozs.m*

```

% Functia multivariabila ce definește ecuația transcendentă ce furnizează
% caracteristicile statice pt convertoare BUCK-BOOST QRC.
function v=bucboozs(x,y,Rn,n,zs);
if zs==1
v=y.*(1-x)/(2.*pi).*quasi(y./Rn,n)-x/(2.*pi).*quasi(y./Rn,n);
else
v=(y+1).*x/(2.*pi).*quasi(Rn./y,n)-1;
end

```

*Fișierul quasi.m*

```

% Definește funcția cvasirezonanță.
function z=quasi(x,n)
z=x/2+n.*pi+(-1).^(n-1).*asin(x)+(x.^(-1)).*(1+(-1).^(n-1)).*sqrt(1-x.^2));

```

*Fișierul sol.m*

```

% Rezolvă ecuația transcendentă prin metoda injumatirii intervalului
function u=sol(FunFcn,x,Rn,n,zs,a,b);
if (feval(FunFcn,x,a,Rn,n,zs).*feval(FunFcn,x,b,Rn,n,zs))<=0
epsilon=0.0001;
yl=min(a,b);
yh=max(a,b);
yn=min(a,b);
cn=max(a,b);
if (feval(FunFcn,x,b,Rn,n,zs)-feval(FunFcn,x,a,Rn,n,zs))/(b-a)>0
while abs(yn-cn)>=epsilon
if feval(FunFcn,x,yn,Rn,n,zs)<0
cn=yn;
yl=yn;
yn=(c+yh)/2;
elseif feval(FunFcn,x,yn,Rn,n,zs)>0
cn=yn;
yh=yn;
yn=(c+yl)/2;
else
break
end
end
end
u=yn;

```



```

else
while abs(yn-c)>=epsilon
if feval(FunFcn,x,yn,Rn,n,zs)<0
c=yn;
yh=yn;
yn=(c+y)/2;
elseif feval(FunFcn,x,yn,Rn,n,zs)>0
c=yn;
yl=yn;
yn=(c+yh)/2;
else
break
end
end
u=yn;
end
else
error('Functia trebuie sa aiba semne opuse in punctele de plecare.')
end

```

## A8 Programele de trasare a caracteristicilor control-iesire și de iesire normalizate pentru convertoare CF-ZVS în modul II de funcționare

Fișierul *cf\_zvs1.m*

```

% Caracteristicile de iesire la convertoare CF-ZVS în modul II
clear all
figure
fN=0.6;
D=[0.2 0.1 0];

alpha=0:0.01:3.5;
for j=1:length(D)
for i=1:length(alpha)
miu(i)=1-fN/pi*(pi-atan(alpha(i)+2*pi*D(j)/fN)+alpha(i)+2*pi*D(j)/fN);
if miu(i)<0
break
else
end
end
miu=miu(1:length(miu)-1);
alphatr=alpha(1:length(miu));
plot(alphatr,miu,'k'); hold on; xlabel('alpha'); ylabel('u'); axis([0 3.5 0 1]); title('CF-ZVS. Caracteristici de iesire')
end

```

Fișierul *cf\_zvs2.m*

```

% Caracteristicile control-iesire la convertoare CF-ZVS în modul II
clear all
figure
fN=0.6;
alpha=[2 1 0];

D=0:0.001:1;
for j=1:length(alpha)
for i=1:length(D)
miu(i)=1-fN/pi*(pi-atan(alpha(j)+2*pi*D(i)/fN)+alpha(j)+2*pi*D(i)/fN);
if miu(i)<0
break
else
end
end
miu=miu(1:length(miu)-1);
Dtr=D(1:length(miu));
plot(Dtr,miu,'k'); hold on; xlabel('alpha'); ylabel('u'); axis([0 1 0 1]); title('CF-ZVS. Caracteristici control-iesire')
end

```

## A9 Programele CASPOC de comparare a diverselor comenzi integrative

Fișierul mmdyn1.cir

```
* MMC - dinamica de semnal mare relativ la tensiunea de intrare *
Bg 1 0 Vg
rsw 1 a lu
S a 2
D 0 2 did
L 2 3 70u
C 3 4 390u
Rc 4 0 20m
*R 3 0 5
Io 0 3 5
*****
.model did DSWITCH vth=0 vthoff=0
.option method=Euler Tscreen=1.85m
.tran 100n 5
*.draw 1 b(Vc)
*.draw 2 b(integ)
*.draw 3 v(3)
*.draw 4 b(reset)
*.draw 5 b(clock)
*.draw 6 b(bist)
*.draw 7 i(L)
*.draw 8 v(2,3)
*.draw 9 v(1)
.end
```

Fișierul mmdyn1.stm

```
* MMC - dinamica de semnal mare relativ la tensiunea de intrare *
t time
sensei current rsw
sensev voltage 2
***** 1/CT 1/TAU
norm sum sensei sensev 0 100000 41667 0
clock signal t 0.5 0.5 50k 0 0.02 0 0 1
integ intreset norm cond 0
bist fflt reset clock 0.5u 0.5u 1 1
Vc con 8.301
dif sub integ vc
reset bng dif 0 1 0
S switch bist 0 0.5 S
nbist not bist
cond add nbist -0.5
*****
Vg signal t 12 5 10k 0 0.1 12 30.2m 4
.end
```

Fișierul mmdyn2.cir

```
* Comanda prin sarcina - dinamica de semnal mare relativ la tensiunea de intrare
Bg 1 0 Vg
rsw 1 a lu
S a 2
D 0 2 did
L 2 3 70u
C 3 4 390u
Rc 4 0 20m
*R 3 0 5
Io 0 3 5
*****
.model did DSWITCH vth=0 vthoff=0
.option method=Euler Tscreen=1.85m
.tran 100n 5
*.draw 1 b(Vc)
*.draw 2 b(integ)
*.draw 3 v(3)
*.draw 4 b(reset)
*.draw 5 b(clock)
*.draw 6 b(bist)
*.draw 7 i(L)
*.draw 8 v(2,3)
*.draw 9 v(1)
.end
```

Fișierul mmdyn2.stm

```
* Comanda prin sarcina - dinamica de semnal mare relativ la tensiunea de intrare
t time
```

```

sensei current rsw
sensev voltage 2
***** 1/CT 1/TAU
norm sum sensei sensev 0 100000 1u 0
clock signal t 0.5 0.5 50k 0 0.02 0 0 1
integ intreset norm cond 0
bist fflt reset clock 0.5u 0.5u 1 1
Vc con 4.151
dif sub integ vc
reset bng dif 0 1 0
S switch bist 0 0.5 S
nbist not bist
cond add nbist -0.5
*****
Vg signal t 12 5 10k 0 0.1 12 30.2m 4
.end

```

Fişierul mmdyn3.cir

```

* MMC - dinamica de semnal mare relativ la tensiunea de comanda *
Bg 1 0 Vg
rsw 1 a lu
S a 2
D 0 2 did
L 2 3 70u
C 3 4 390u
Rc 4 0 20m
*R 3 0 5
Io 0 3 5
*****
.model did DSWITCH vthn=0 vthoff=0
.option method=Euler Tscreen=2m
.tran 100n 5
.draw 1 b(Vc)
*.draw 2 b(integ)
.draw 3 v(3)
*.draw 4 b(reset)
*.draw 5 b(clock)
*.draw 6 b(bist)
.draw 7 i(L)
*.draw 8 v(2,3)
*.draw 9 v(1)
.end

```

Fişierul mmdyn3.stm

```

* MMC - dinamica de semnal mare relativ la tensiunea de comanda *
t time
sensei current rsw
sensev voltage 2
***** 1/CT 1/TAU
norm sum sensei sensev 0 100000 41667 0
clock signal t 0.5 0.5 50k 0 0.02 0 0 1
integ intreset norm cond 0
bist fflt reset clock 0.5u 0.5u 1 1
Vg con 12
dif sub integ vc
reset bng dif 0 1 0
S switch bist 0 0.5 S
nbist not bist
cond add nbist -0.5
*****
Vc signal t 0 8.301 10k 0 0.1 3.305 14.2m 5
.end

```

Fişierul mmdyn4.cir

```

* Comanda integrativa dupa o variabila - dinamica de semnal mare relativ la *
* tensiunea de comanda *
Bg 1 0 Vg
rsw 1 a lu
S a 2
D 0 2 did
L 2 3 70u
C 3 4 390u
Rc 4 0 20m
*R 3 0 5
Io 0 3 5
*****
.model did DSWITCH vthn=0 vthoff=0
.option method=Euler Tscreen=2m
.tran 100n 5

```

```
.draw 1      b(Vc)
*.draw 2     b(integ)
.draw 3     v(3)
*.draw 4     b(reset)
*.draw 5 b(clock)
*.draw 6 b(bist)
.draw 7 i(L)
*.draw 8 v(2,3)
*.draw 9 v(1)
.end
```

Fișierul mmcdyn4.stm

```
* Comanda integrativa dupa o variabila - dinamica de semnal mare relativ la *
* tensiunea de comanda *
t      time
sensei current rsw
sensev voltage 2
***** 1/CT      1/TAU
norm  sum  sensei  sensev  0  1u  41667  0
clock signal  t  0.5  0.5  50k  0  0.02  0  0  1
integ intreset  norm  cond  0
bist  fflt  reset  clock  0.5u  0.5u  1  1
Vg    con    12
dif   sub    integ   vc
reset bng     dif     0      1  0
S     switch  bist    0  0.5  S
nbist not     bist
cond  add     nbist   -0.5
*****
Vc    signal  t  0  4.151  10k  0  0.1  1.652  14.2m  5
.end
```

## A 10 Convertor BOOST cu comanda fuzzy - suprafata de reglare - program MATLAB

Fișierul regrsf.m

```
clear all

% Defineste FC
emax=1;
demax=1;

NB1 = mf_tri_p([-emax -emax -0.5.*emax]);
NS1 = mf_tri_p([-emax -0.5.*emax 0]);
ZE1 = mf_tri_p([-0.5.*emax 0 0.5.*emax]);
PS1 = mf_tri_p([0 0.5.*emax emax]);
PB1 = mf_tri_p([0.5.*emax emax emax]);

NB2 = mf_tri_p([-demax -demax -0.5.*demax]);
NS2 = mf_tri_p([-demax -0.5.*demax 0]);
ZE2 = mf_tri_p([-0.5.*demax 0 0.5.*demax]);
PS2 = mf_tri_p([0 0.5.*demax demax]);
PB2 = mf_tri_p([0.5.*demax demax demax]);

NB = mf_tri_p([-1 -1 -0.5]);
NS = mf_tri_p([-1 -0.5 0]);
ZE = mf_tri_p([-0.5 0 0.5]);
PS = mf_tri_p([0 0.5 1]);
PB = mf_tri_p([0.5 1 1]);

A1 = [NB1; NS1; ZE1; PS1; PB1];
A2 = [NB2; NS2; ZE2; PS2; PB2];
Q = [PB; PS; ZE; NS; NB];

RB = [5 4 3 3 2
      4 4 3 3 2
      4 4 3 2 2
      4 3 3 2 2
      4 3 3 2 1];
d_def=0;

% Genereaza suprafata de reglare

e = -emax.emax./20;emax;
de = -demax.demax./20;demax;
```

```

for i = 1:length(e)
    for j = 1:length(de)
        y_p(i,j) = fc_mam(e(i), de(j), A1, A2, Q, RB, d_def);
    end
end

% Ploteaza suprafata
mesh(e,de,y_p);
view(-30,10);

```

## A 11 Convertor SRC cu comanda neuronală - setul de fișiere MATLAB

*Fișierul src\_neur.m*

```

% Simularea unui SRC cu controller neuronal
% la o variatie treapta a razei in comanda pe traiectorie optimala
clear all
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

tic
% Parametrii convertorului si domeniul de variatie al
% marimilor ce intra in retea neuronală
Vg=25; L0=81.8e-6; C0=0.125e-6; Cf=20e-6; R=12;
Z0=sqrt(L0/C0);
vCmin=100; vCmax=100; iLmin=-4; iLmax=4; lomin=1; lomax=2.5; % universul variabilelor antrenate
Vo=12;

% Construirea controlerului neuronal din fisierul de ponderi "srcclass.wei"
% din care, dupa crearea de catre programul Neural Graphics, s-au sters primele
% 2 linii pentru a avea forma de matrice
load srcclass.wei;
wei=srcclass;
in=3; nh1=7; nh2=2; out=2; % numarul intrarilor, al neuronilor din primul respectiv al doilea strat ascuns si numarul iesirilor

for i=1:nh1
    for k=1:in
        w1(i,k)=wei((i-1)*in+k); % genereaza matricea ponderilor din primul strat ascuns
    end
end

for j=1:nh1
    ph1(j)=wei(in*nh1+j);% genereaza matricea pragurilor din primul strat ascuns
end

for i=1:nh2
    for k=1:nh1
        w2(i,k)=wei(in*nh1+nh1+(i-1)*nh1+k); % genereaza matricea ponderilor din al doilea strat ascuns
    end
end

for j=1:nh2
    ph2(j)=wei(in*nh1+nh1+nh1*nh2+j);% genereaza matricea pragurilor din al doilea strat ascuns
end

for i=1:out
    for k=1:nh2
        wo(i,k)=wei(in*nh1+nh1+nh1*nh2+nh2+(i-1)*nh2+k); % genereaza matricea ponderilor din al doilea strat ascuns
    end
end

for j=1:nh2
    po(j)=wei(in*nh1+nh1+nh1*nh2+nh2+out*nh2+j);% genereaza matricea pragurilor din al doilea strat ascuns
end

% Simularea functionarii convertorului de putere in CCM
% Conditii initiale (calculul punctului de plecare de pe traiectorie)
% Calculeaza raza de referinta curenta si marimile de stare initiale
Io=Vo/R; VoN=Vo/Vg; RN=R/Z0;
err=1e-5;

```

```

Ra=1+VoN+err; n=0; % contorul iteratiilor
Xa=0;Xc=2*err;
while ~(Xa==Xc)
    Rc=Ra-f(Ra,VoN,RN)/fprim(Ra,VoN,RN); % procesul iterativ Newton
    dif=abs(Rc-Ra);
    if dif>err
        Ra=Rc; n=n+1;
    else
        Xa=Xc; % forteaza parasirea buclei
    end
end
xa(1,1)=1e-6; xa(2,1)=-Vg*(Rc+1-VoN); xa(3,1)=Vo; % pentru convertor x=[iL0; vC0; vC]
t(1)=0; iL0(1)=xa(1,1); vC0(1)=xa(2,1); vo(1)=xa(3,1);
xc=xa; xsign=[-1e-9; xa(2,1)]; bant=-1; b=-1;
l=eye(3);
%l=eye(2);
dT=50e-9; i=2;
% Testeaza daca curentul iL isi schimba semnul si in caz afirmativ pozitioneaza corespunzator
% flagul a, verifica daca e necesar ca tranzistorul corespunzator sa comute,
% iar in caz negativ continua simularea cu aceleasi matrici
while (i-1)<1e-3
    if t(i-1)<300e-6
        R=10;
    else
        R=4;
    end
    if sign(xc(1,1)*xsign(1,1))==-1
        if xc(1,1)>=0
            a=1; % conduc diodele 1 si 3 din puntea redresoare
        else
            a=-1; % conduc diodele 2 si 4 din puntea redresoare
        end
        f=0; % marcheaza ca se monitorizeaza eventuala comutare
        xa(1,1)=0; xsign=xc; % corecteaza pe iL la 0 si actualizeaza xsign
        % Actualizeaza matricile
        A=[ 0 -1/L0 -a/L0
            1/C0 0 0 % pentru sarcina rezistiva
            a/Cf 0 -1/(R*CF)];
        B=[b/L0; 0; 0]; E=[0 0 1]; F=[0];
        %A=[ 0 -1/L0
            % 1/C0 0 ]; % pt sarcina tensiune constanta egala cu Vo
        % B=[1/L0; 0];
    else
        xc=expm(A*dT)*xa+inv(A)*(expm(A*dT)-I)*B*Vg;
        %xc=expm(A*dT)*xa+inv(A)*(expm(A*dT)-I)*B*(b*Vg-a*Vo);
        iL0(i)=xc(1,1); vC0(i)=xc(2,1); vo(i)=E*xc+F*Vg;
        t(i)=(i-1)+dT;
        xa=xc;
        i=i+1;
    if f==0
        % Efectueaza testul cu retea neuronală pt a vedea dacă e necesară comutarea tranzistorului
        % intrările în rețea sunt Io,vC și iL normalize cu maximele domeniilor respective
        x1=Vo/R/max(abs(Iomin),abs(Iomax)); % intrările în rețea
        x2=abs(xc(2,1))/max(abs(vCmin),abs(vCmax)); % reduce la primul cadran
        x3=abs(xc(1,1))/max(abs(iLmin),abs(iLmax)); % și normalize în universul propriu
        x=[x1;x2;x3];
        for k=1:nh1
            h1(k)=sigmoid(w1(k,:)*x+ph1(k)); % calculează valorile neuronilor din primul strat ascuns
        end
        for k=1:nh2
            h2(k)=sigmoid(w2(k,:)*h1'+ph2(k)); % calculează valorile neuronilor din al doilea strat ascuns
        end
        for k=1:out
            o(k)=sigmoid(wo(k,:)*h2'+po(k)); % calculează valorile neuronilor de ieșire
        end

        if (o(1)<o(2))
            else, b=a;
            %elseif xc(1,1)<0
            % b=a;
            %elseif xc(1,1)>=0
            % b=a;
        end

        if b==bant
            else f=1; bant=b;

```

```

end
else
end
A=[ 0 -1/L0 -a/L0
    1/C0 0 0
    a/Cf 0 -1/(R*CF)]; %pentru sarcina rezistiva
B=[b/L0; 0; 0]; E=[0 0 1]; F=[0];
% A=[ 0 -1/L0
    % 1/C0 0 ]; % pentru sarcina tensiune constanta egala cu Vo
% B=[1/L0; 0];
end
end

% Ploteaza caracteristicile;
plot(vC0/Vg,iL0*Z0/Vg,'k'); grid; xlabel('vCN'); ylabel('iLN');
figure;
plot(t,iL0,'k'); xlabel('t [s]'); ylabel('iL0 [A]');
figure;
plot(t,vC0,'k'); xlabel('t [s]'); ylabel('vC [V]');
figure;
plot(t,v0,'k');
toc

break

% Secventa pentru comanda pe traiectorie optima
if (xc(2,1)/Vg-a*(1-VoN))^2+(xc(1,1)*Z0/Vg)^2<Rc^2
b=a;
else
end

Fişierul train.m

% Generarea vectorilor de antrenament pt controlerul neuronal al unui SRC
% Se genereaza un fisierul "src.dat" pentru programul Neural Graphics,
% pe tipicul cerut de acesta.
clear all
tic
format short
% Parametrii convertorului si domeniul de variatie al
% marimilor ce intra in retea neuronală
L0=81.8e-6; C0=0.125e-6; Cf=50e-6; Z0=sqrt(L0/C0);
Vg=25;
vCmin=-100; vCmax=100; iLmin=-4; iLmax=4; Iomin=1; Iomax=2.5; Vo=12;

i=1;
for Io=1:1.5:2.5
for vC=0:20:vCmax
for iL=0:0.8:iLmax
F=sign(iL);

% Calculeaza raza de referinta curenta
VoN=Vo/Vg; RN=Vo/Io/Z0;
err=1e-5;
Ra=1+VoN+err; n=0; % contorul iteratiilor
Xa=0;Xc=2*err;
while ~(Xa==Xc)
Rc=Ra-f(Ra,VoN,RN)/fprim(Ra,VoN,RN); % procesul iterativ Newton
dif=abs(Rc-Ra);
if dif>err
Ra=Rc; n=n+1;
else
Xa=Xc; % forteaza parasirea buclei
end
end

% Testeaza cum este raza curenta fata de raza de referinta curenta
% si comuta tranzistoarele corepunzator. Marimile sunt convertite
% la un singur cadran, cel corespunzator comutatiei lui Q2, deci la
% |vcN| si -|iLN|
vCN=vC/Vg; iLN=iL/(Vg/Z0);
if (vCN-VoN+1)^2+iLN^2<Rc^2
%Q1=0.8*((1+F)/2*abs(F))+0.1; Q2=0.8*((1-F)/2*abs(F))+0.1;
Q=1; % tranzistorul trebuie adus in conductie

```

```

else
    %Q1=0.1;      Q2=0.1;
    Q=0; % tranzistorul ramane blocat
end
%A(1,i)=i; A(2,i)=Io/2.5; A(3,i)=vC/100; A(4,i)=iL/4; A(5,i)=Q1; A(6,i)=Q2;
A(1,i)=i; A(2,i)=Io/max(abs(Iomin), abs(Iomax)); A(3,i)=vC/max(abs(vCmin), abs(vCmax)); A(4,i)=iL/max(abs(iLmin), abs(iLmax));
A(5,i)=Q;
i=i+1;
end
end
end

```

```

% Salveaza matricea A intr-un fisier ASCII cu numele "src.dat"
fid = fopen('srcclass.dat','w');
fprintf(fid,'%i %6.2f %6.2f %6.2f %i\n',A);
toc

```

#### Fișierul neur\_net.m

```

% Functia raspuns a unei retele neuronale
function y=neur_net(x1,x2,x3)

```

```

% Construirea controlerului neuronal din fisierul de ponderi "srcclass.wei"
% din care, dupa crearea de catre programul Neural Graphics, s-au sters primele
% 2 linii pentru a avea forma de matrice
load srcclass.wei;
wei=srcclass;
in=3; nh1=7; nh2=2; out=2; % numarul intrarilor, al neuronilor din primul respectiv al doilea strat ascuns si numarul iesirilor

```

```

for i=1:nh1
    for k=1:in
        w1(i,k)=wei((i-1)*in+k); % genereaza matricea ponderilor din primul strat ascuns
    end
end

```

```

for j=1:nh1
    ph1(j)=wei(in*nh1+j);% genereaza matricea pragurilor din primul strat ascuns
end

```

```

for i=1:nh2
    for k=1:nh1
        w2(i,k)=wei(in*nh1+nh1+(i-1)*nh1+k); % genereaza matricea ponderilor din al doilea strat ascuns
    end
end

```

```

for j=1:nh2
    ph2(j)=wei(in*nh1+nh1+nh1*nh2+j);% genereaza matricea pragurilor din al doilea strat ascuns
end

```

```

for i=1:out
    for k=1:nh2
        wo(i,k)=wei(in*nh1+nh1+nh1*nh2+nh2+(i-1)*nh2+k); % genereaza matricea ponderilor din al doilea strat ascuns
    end
end

```

```

for j=1:nh2
    po(j)=wei(in*nh1+nh1+nh1*nh2+nh2+out*nh2+j);% genereaza matricea pragurilor din al doilea strat ascuns
end

```

```

% Apelarea retelei neuronale

```

```

x=[x1;x2;x3];
for k=1:nh1
    h1(k)=sigmoid(w1(k,:)*x+ph1(k)); % calculeaza valorile neuronilor din primul strat ascuns
end
for k=1:nh2
    h2(k)=sigmoid(w2(k,:)*h1+ph2(k)); % calculeaza valorile neuronilor din al doilea strat ascuns
end
for k=1:out
    o(k)=sigmoid(wo(k,:)*h2+po(k)); % calculeaza valorile neuronilor de iesire
end
y=o;

```



## A12 Caracteristicile statice unificate ale unui convertor SRC - setul de fişiere CASPOC

*Fisierul c\_o\_unif.m*

```
% Caracteristicile unificate normate control iesire la un convertor SRC.
% Rezistenta de sarcina normalizata este parametru.
clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

fN=0.5:0.001:1.5;
param=[0.7 0.5 0.05];
for i=1:3
    RN=param(i);
    VON=(sqrt(2.*(pi./(2.*fN.*RN)).^2.*(1+cos(pi./fN))+1-cos(pi./fN)).^2)-sign(fN-1)).*(pi./(2.*fN.*RN)).*(1+cos(pi./fN))./(1-
    cos(pi./fN))+pi./(2.*fN.*RN)).^2.*(1+cos(pi./fN)));
    plot(fN,VON,'k'); hold on
end
xlabel('fN'); ylabel('VON'); title('Caracteristicile unificate control-iesire');
```

*Fisierul out\_uni.m*

```
% Caracteristicile unificate normate de iesire la un convertor SRC.
% Rezistenta de sarcina normalizata este parametru.
clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

par=[0.8 0.9 1.2 1.5];
for i=1:length(par)
    fN=par(i);
    IONmax=2.*fN./pi*(sqrt(2/(1+cos(pi/fN)))-sign(fN-1));
    if fN<1
        IONmin=4.*fN./pi;
    else
        IONmin=0;
    end
    delta=1e-3;
    ION=IONmin:delta:IONmax;
    VON=sqrt(2./(1-cos(pi./fN))-(1+cos(pi./fN))./(1-cos(pi./fN))).*(pi./(2.*fN)*ION+sign(fN-1)).^2);
    %VON=sqrt(1-pi.^2/(4.*fN.^2)*(1+cos(pi./fN))./(1-cos(pi./fN)).*ION.^2-sign(fN-1).*pi./fN.*(1+cos(pi./fN))./(1-cos(pi./fN)).*ION);
    plot(ION,VON,'k'); hold on
end
xlabel('ION'); ylabel('VON'); title('Caracteristicile de iesire unificate');
```

## A13 Proiectarea automata a convertoarelor SRC - setul de fişiere MATLAB

*Fişierul idealsrc.m*

```
% Convertor rezonant serie (SRC) - proiectare (CCM) sub rezonanta.

VINom=input('Introduceti valoarea nominala a tensiunii de alimentare: ');
VImIn=input('Introduceti valoarea minima a tensiunii de alimentare: ');
VImax=input('Introduceti valoarea maxima a tensiunii de alimentare: ');

while (VImIn>VINom)|(VImax<VINom)|(VImIn>VImax)
    disp('Date incorecte ! Valorile trebuie sa fie a.i VImIn<=VINom<=VImax');
    fprintf('\n');
    VINom=input('Introduceti valoarea nominala a tensiunii de alimentare: ');
    VImIn=input('Introduceti valoarea minima a tensiunii de alimentare: ');
    VImax=input('Introduceti valoarea maxima a tensiunii de alimentare: ');
end
fprintf('\n');
end

Iomin=input('Introduceti valoarea minima a curentului de sarcina: ');
Iomax=input('Introduceti valoarea maxima a curentului de sarcina: ');
```

```

while (lomin>lomax)
    disp('Date incorecte ! Valorile trebuie sa fie ai lomin=<lomax');
    fprintf('\n');
    lomin=input(' Introduceți valoarea minima a curentului de sarcina:');
    lomax=input(' Introduceți valoarea maxima a curentului de sarcina:');
    fprintf('\n');
end

Vo=input('Introduceți valoarea tensiunii de iesire: ');

nrec=0.5.*(VINom./Vo);
nmax=Vlmin./Vo;
fprintf('Valoarea recomandata a raportului de transformare (pt. transfer maxim de putere) este nrec=%6.3g\n',nrec);
fprintf(' Valoarea maxima a raportului de transformare(pt. care VoNmax=n.Vo./Vlmin=1) este nmax=%6.3g\n',nmax);
disp('Doriti modificarea raportului de transformare "n=n1/n2" ? ');
disp(' 1 - Da');
disp(' 0 - Nu');
answer=input('Selectati numarul dorit si apasati tasta "Enter" : ');
fprintf('\n');
while (answer~=0)&(answer~=1);
    disp('Numar incorect (Numarul trebuie sa fie 1 sau 0)');
    answer=input('Selectati numarul dorit si apasati tasta "Enter": ');
fprintf('\n');
end;

if answer==1
    disp('Doriti afisarea curbelor normalizate pt a va ajuta in luarea deciziei ? ');
    disp(' 1 - DA');
    disp(' 0 - NU');

    curve=input('Selectati numarul dorit si apasati tasta "Enter": ');
    fprintf('\n');
    while (curve~=0)&(curve~=1);
        disp('Numar incorect (Numarul trebuie sa fie 1 sau 0)');
        curve=input('Selectati numarul dorit si apasati tasta "Enter": ');
    fprintf('\n');
    end;
    if curve==1
        normgraf
    else
        end
end

n=input('Introduceți valoarea raportului de transformare n = ');
else
n=nrec;
end

while n>nmax
fprintf('Valoare incorecta pt. n ! Valoarea trebuie sa fie mai mica decat nmax=%6.3g\n',nmax);
disp('Doriti modificarea raportului de transformare "n=n1/n2" ? ');
disp(' 1 - DA');
disp(' 0 - NU');
answer=input('Selectati numarul dorit si apasati tasta "Enter": ');
fprintf('\n');
while (answer~=0)&(answer~=1);
    disp('Numar incorect (numarul trebuie sa fie 1 or 0)');
    answer=input('Selectati numarul dorit si apasati tasta "Enter": ');
fprintf('\n');
end;

if answer==1
    disp('Doriti afisarea curbelor normalizate pt a va ajuta in luarea deciziei ? ');
    disp(' 1 - DA');
    disp(' 0 - NU');

    curve=input('Selectati numarul dorit si apasati tasta "Enter": ');
    fprintf('\n');
    while (curve~=0)&(curve~=1);
        disp('Numar incorect (numarul trebuie sa fie 1 or 0)');
        curve=input('Selectati numarul dorit si apasati tasta "Enter": ');
    fprintf('\n');
    end;
    if curve==1
        normgraf
    else
        end
end

n=input('Introduceți valoarea raportului de transformare = ');

```

```

else
n=nrec;
end
end

```

```

f0kHz=input('Introduceti valoarea frecventei de rezonanta in kilohertz f0 = ');
f0=f0kHz.*1000;

```

```

disp('Doriti curbe pt alegerea uneia din limitele frecventei ? ');
disp(' 1 - DA');
disp(' 0 - NU');
answer=input('Selectati numarul dorit si apasati tasta "Enter": ');
fprintf('\n');
while (answer~=0)&(answer~=1);
disp('Numar incorect (numarul trebuie sa fie 1 or 0)');
answer=input('Selectati numarul dorit si apasati tasta "Enter": ');
fprintf('\n');
end;

```

```

if answer==1
normgraf
else
end
disp('Ce frecventa ati ales, fmin or fmax ? ');
disp(' 1 - fmin');
disp(' 2 - fmax');
bound=input('Selectati numarul dorit si apasati tasta "Enter": ');
fprintf('\n');
while (bound~=1)&(bound~=2);
disp('Numar incorect (numarul trebuie sa fie 1 or 2)');
bound=input('Selectati numarul dorit si apasati tasta "Enter": ');
fprintf('\n');
end;
if bound==1
fminkHz=input('Introduceti frecventa de comutatie minima in kilohertz, fmin = ');
fprintf('\n');
while (fminkHz>=f0kHz)|(fminkHz<=(f0kHz./2))
if (fminkHz>=f0kHz)
disp('Decizie incorecta ! Ati selectat functionarea peste frecventa de rezonanta');
elseif (fminkHz<f0kHz./2)
disp('Decizie incorecta ! Ati selectat functionarea DCM');
end
fminkHz=input('Alegeti o frecventa sub f0 si peste f0/2, apoi apasati tasta "Enter" : ');
end
fmin=fminkHz.*1000;
fNmin=fmin./f0;
else
fmaxkHz=input('Introduceti valoarea maxima a frecventei in kilohertz, fmax = ');
fprintf('\n');
while (fmaxkHz>=f0kHz)|(fmaxkHz<=(f0kHz./2))
if (fmaxkHz>=f0kHz)
disp('Decizie incorecta ! Ati selectat functionarea peste frecventa de rezonanta');
else
disp('Decizie incorecta ! Ati selectat functionarea in mod DCM');
end
fmaxkHz=input('Alegeti o frecventa sub f0 si apasati tasta "Enter": ');
end
fmax=fmaxkHz.*1000;
fNmax=fmax./f0;
end

```

% Efectueaza analiza de cc:

```

Vl=[Vlmin Vlmin Vlmax Vlmax];
lo=[lomin lomax lomin lomax];
Vout=[Vo Vo Vo Vo];
VoN=(n.*Vo)./Vl;
clear fN f_kHz ILAVN ILAV ILPN ILP VCPN VCP ISWN ISW VC0N VC0 Fr ISWr VC0r ILPr VCPr DC RANGE;

```

```

if bound==1
VoNmin=n.*(Vo./Vlmax);
Z0=(n.*Vlmax)./lomin.*(2.*fNmin./pi).*(1+sqrt(((n.*Vo)./Vlmax).^2+(1-((n.*Vo)./Vlmax).^2).*2./(1+cos(pi./fNmin)))));
L=Z0./(2.*pi.*f0);

```

```

C=1./(2.*pi.*f0.*Z0);
else
VoNmax=n.*(Vo./Vlmin);
Z0=(n.*Vlmin)/Iomax.*(2.*fnmax./pi).*(1+sqrt(((n.*Vo)/Vlmin).^2+(1-(n.*Vo)/Vlmin).^2).*(1+cos(pi./fnmax)));
L=Z0./(2.*pi.*f0);
C=1./(2.*pi.*f0.*Z0);

while src(0.5,Vlmax,Iomin,Vo,Z0,n)>0
Z0m=(n.*Vlmax)/Iomin.*(2.*0.5./pi).*(1+sqrt(((n.*Vo)/Vlmax).^2+(1-(n.*Vo)/Vlmax).^2).*(1+cos(pi./0.5)));
fmax_min=f0kHz.*solsrc('src',Vlmin,Iomax,Vo,Z0m,n,0.5,0.9999);
fprintf('Valoare prea mica pt. fmax. Cu acesata valoarea pt Vlmax si Iomin functionarea va fi DCM.\n Valoarea minima pt fmax este %5.2f
kHz.\n',fmax_min);
fmaxkHz=input('Introduceti valoarea frecventei maxime in kilohertz, fmax = ');
fprintf('\n');
while (fmaxkHz>=f0kHz)|(fmaxkHz<=(f0kHz./2))
if (fmaxkHz>=f0kHz)
disp('Decizie incorecta ! Ati ales functionarea peste frecventa de rezonanta');
else
disp('Decizie incorecta ! Ati ales functionarea DCM');
end
fmaxkHz=input('Alegeti a frecventa sub f0 si apasati tasta "Enter": ');
end
fmax=fmaxkHz.*1000;
fnmax=fmax./f0;
VoNmax=n.*(Vo./Vlmin);
Z0=(n.*Vlmin)/Iomax.*(2.*fnmax./pi).*(1+sqrt(((n.*Vo)/Vlmin).^2+(1-(n.*Vo)/Vlmin).^2).*(1+cos(pi./fnmax)));
L=Z0./(2.*pi.*f0);
C=1./(2.*pi.*f0.*Z0);
end
end

LuH=L.*1e6;
CnF=C.*1e9;
tank=[LuH CnF];
disp(' L[uH] C[nF]');
fprintf('\n');
disp(tank);

for r=1:length(VI)
fn(r)=solsrc('src',VI(r),Io(r),Vo,Z0,n,0.5,0.9999);
end
fnmin=min(fn);
fnmax=max(fn);
R=VoN+sqrt(VoN.^2+(1-VoN.^2).*(1+cos(pi./fn)));
Rdiode=R-2.*VoN;
alpha=pi-acos((Rdiode.^2+4-R.^2)/(4.*Rdiode));
beta=pi-acos((R.^2+4-Rdiode.^2)/(4.*R));
VCPN=R+1-VoN;
ILAVN=2./pi.*fn.*VCPN;
ILPN=R;
ISWN=R.*sin(beta);
VC0N=VoN.*(R+1-VoN);
IQRMSN=sqrt(1./(2.*pi).*(0.5.*ISWN.^2.*(beta+0.5.*sin(2.*beta))-0.5.*((VC0N-1+VoN)/Z0).^2.*(beta-0.5.*sin(2.*beta))+0.5.*ISWN.*(VC0N-1+VoN)/Z0).*(1-cos(2.*beta)));
IQRMS=IQRMSN.*VI./Z0;
IDRMSN=sqrt(1./(4.*pi).*((VCPN+1+VoN)/Z0).^2.*(alpha-0.5.*sin(2.*alpha)));
IDRMS=IDRMSN.*VI./Z0;
ITRMS=sqrt(IQRMS.^2+IDRMS.^2);

f_kHz=f0kHz.*fn;
ILAV=ILAVN.*VI./Z0;
ILP=ILPN.*VI./Z0;
VCP=VCPN.*VI;
ISW=ISWN.*VI./Z0;
VC0=VC0N.*VI;

% Afisaza rezultatele analizei de cc

DC=[VI' Io' f_kHz' ILAV' ILP' VCP'];
disp('Rezultatele analizei de cc pt combinatii extreme intre VI and Io: ');
fprintf('\n');
disp(' VI(V) Io(A) F(kHz) ILAV(A) ILP(A) VCP(V)');
fprintf('\n');
disp(DC);

% Calculeaza domeniile de variatie ale principalilor parametri ca si consecinta a analizei de cc

```

```

Fr=[min(f_kHz) max(f_kHz)];
ISWr=[min(ISW) max(ISW)];
VC0r=[max(VC0) min(VC0)];
ILPr=[min(ILP) max(ILP)];
VCPPr=[min(VCP) max(VCP)];

RANGE=[Fr' ISWr' VC0r' ILPr' VCPPr];
disp(' Domeniile de variatie ale principalilor parametri ca si consecinta a analizei de cc:');
fprintf('\n');
disp(' F(kHz) ISW(A) VC0(V) ILP(A) VCP(Vy);');
fprintf('\n');
disp(RANGE);

% Calculeaza domeniile principalilor parametri considerand dinamica.

VC0m=n.*Vo.*(1+sqrt(((n.*Vo)/Vlmin).^2+(1-((n.*Vo)/Vlmin).^2).*2./(1+cos(pi./fNmin))));
VC0M=n.*Vo.*(1+sqrt(((n.*Vo)/Vlmax).^2+(1-((n.*Vo)/Vlmax).^2).*2./(1+cos(pi./fNmax))));

ILPm=n.*Vo./Z0+1./Z0.*sqrt((n.*Vo).^2+(Vlmin.^2-(n.*Vo).^2).*2./(1+cos(pi./fNmin)));
ILPM=n.*Vo./Z0+1./Z0.*sqrt((n.*Vo).^2+(Vlmax.^2-(n.*Vo).^2).*2./(1+cos(pi./fNmax)));

VCPm=Vlmin.*(1+sqrt(((n.*Vo)/Vlmin).^2+(1-((n.*Vo)/Vlmin).^2).*2./(1+cos(pi./fNmin))));
VCPM=Vlmax.*(1+sqrt(((n.*Vo)/Vlmax).^2+(1-((n.*Vo)/Vlmax).^2).*2./(1+cos(pi./fNmax))));

VC0mM=[VC0m VC0M];
ILPmM=[ILPm ILPM];
VCPmM=[VCPm VCPM];
mMRANGE=[VC0mM' ILPmM' VCPmM'];
disp(' Domeniile principalilor parametri considerand analiza de cc si dinamica:');
fprintf('\n');
disp(' VC0(V) ILP(A) VCP(Vy);');
fprintf('\n');
disp(mMRANGE);

disp('Doriti afisarea curbelor curentului inductiv de varf si a tensiunii maxime pe capacitate');
disp(' 1 - DA');
disp(' 0 - NU');
option=input('Selectati numarul dorit si apoi apasati tasta "Enter": ');
fprintf('\n');
while (option~=0)&(option~=1);
disp('Numar incorect (numarul trebuie sa fie 1 or 0)');
option=input('Selectati numarul dorit si apoi apasati tasta "Enter": ');
fprintf('\n');
end;

if option==1
peak
else
end

disp('Rezultatele sunt nesatisfacatoare si se doreste reiterarea proiectarii ? ');
disp(' 1 - YES, se doreste reiterarea');
disp(' 0 - NO, se accepta rezultatele si se paraseste programul');
answer=input('Selectati numarul dorit si apoi apasati tasta "Enter": ');
fprintf('\n');
while (answer~=0)&(answer~=1);
disp('Numar incorect (numarul trebuie sa fie 1 or 0)');
answer=input('Selectati numarul dorit si apoi apasati tasta "Enter": ');
fprintf('\n');
end;

if answer==1
iteratio
else
end

Fi șierul normgraf.m

clg;
VoN=0.5:0.1:0.9;
fN=0.5:0.01:0.99;
for r=1:length(VoN)
R=VoN(r)+sqrt(VoN(r).^2+(1-VoN(r).^2).*2./(1+cos(pi./fN)));
ILPN=R;

```

```

VoN1=VoN(r);
VCPN=R+1-VoN1;

subplot(211),plot(fN,ILPN,'k');
%axis([0.5 1 -1 27]);
text(0.9,2.5,'VoN=0.9','sc');
text(0.85,8,'VoN=0.5','sc');
grid on;
xlabel('fN');
ylabel('ILPN');
title('Curbele normalizate pt curentul maxim prin bobina - sub rezonanta')
clear ILPN;
hold on;

subplot(212),plot(fN,VCPN,'k');
%axis([0.5 1 -1 27]);
text(0.9,2.5,'VoN=0.9','sc');
text(0.83,8,'VoN=0.5','sc');
grid on;
xlabel('fN');
ylabel('VCPN');
title('Curbele normalizate pt tensiunea maxima pe capacitate - sub rezonanta')
clear VCPN;
hold on;
end

```

*Fi șierul src.m*

```

function y=src(fN,VI,Io,Vo,Z0,n);
y=(2.*fN./pi).*(1+sqrt(((n.*Vo)./VI).^2+(1-((n.*Vo)./VI).^2).*(1+cos(pi./fN))))-(Io.*Z0)/(n.*VI);

```

*Fi șierul solsrc.m*

```

function u=solsrc(FunFcn,VI,Io,Vo,Z0,n,a,b);

if (feval(FunFcn,a,VI,Io,Vo,Z0,n).*feval(FunFcn,b,VI,Io,Vo,Z0,n))<=0
    epsilon=0.00001;
    yl=min(a,b);
    yh=max(a,b);
    yn=min(a,b);
    c=max(a,b);
    if (feval(FunFcn,b,VI,Io,Vo,Z0,n)-feval(FunFcn,a,VI,Io,Vo,Z0,n))./(b-a)>0
        while abs(yn-c)>=epsilon
            if feval(FunFcn,yn,VI,Io,Vo,Z0,n)<0
                c=yn;
                yl=yn;
                yn=(c+yh)/2;
            elseif feval(FunFcn,yn,VI,Io,Vo,Z0,n)>0
                c=yn;
                yh=yn;
                yn=(c+yl)/2;
            else
                break
            end
        end
        u=yn;
    else
        while abs(yn-c)>=epsilon
            if feval(FunFcn,yn,VI,Io,Vo,Z0,n)<0
                c=yn;
                yh=yn;
                yn=(c+yl)/2;
            elseif feval(FunFcn,yn,VI,Io,Vo,Z0,n)>0
                c=yn;
                yl=yn;
                yn=(c+yh)/2;
            else
                break
            end
        end
        u=yn;
    end
else
end
end
end

```

```
error('Funcția trebuie să aibă semne opuse în punctele de pornire')
end
```

*Fișierul peak.m*

```
clg;
clear ILpeak VCpeak
Vlplot=[Vlmin Vlmax];
step=100;
f=Fr(1):(Fr(2)-Fr(1))/step:Fr(2);
for k=1:length(Vlplot)
ILpeak=n.*Vo./Z0+(1./Z0).*sqrt((n.*Vo).^2+(Vlplot(k).^2-(n.*Vo).^2).*(1+cos(pi.*(f./f0k*Hz))));
VCpeak=Vlplot(k).*(1+sqrt(((n.*Vo)./Vlplot(k).^2+(1-((n.*Vo)./Vlplot(k).^2).*(1+cos(pi.*(f./f0k*Hz))))));
subplot(211),plot(f,ILpeak);
grid on;
xlabel('Frequency [kHz]');
ylabel('ILpeak [A]');
title('Curentul de varf prin bobina funcție de frecvența de comutație - VI este parametru');
text(f(step+1),ILpeak(step+1),num2str(Vlplot(k)));
hold on;
subplot(212),plot(f,VCpeak);
grid on;
xlabel('Frequency [kHz]');
ylabel('VCpeak [V]');
title('Tensiunea maximă pe capacitate funcție de frecvența de comutație - VI este parametru');
text(f(step+1),VCpeak(step+1),num2str(Vlplot(k)));
hold on;
end
```

*Fișierul iteratio.m*

Este fișierul idealsrc cu câteva modificări minore în partea de date cerute a se introduce

## A14 Comportarea la semnal mare a unui convertor SRC - setul de fișiere MATLAB/SIMULINK care implementează modelul de semnal mare

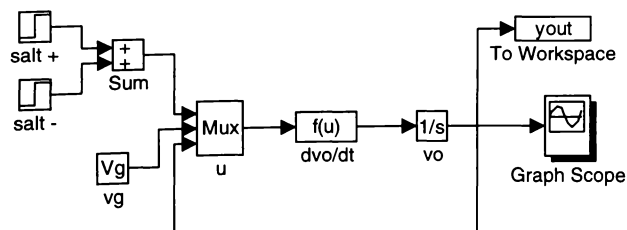
% Modelul unitar (pt  $f, f_0$  și  $f > f_0$ ) al convertorului rezonant serie.  
 % Simularea comportării la semnal mare la un salt treapta în frecvența de comandă

```
clear all % șterge variabile
Fer=4;
for i=1:Fer
close % șterge ultimele 'Fer' ferestre grafice
end
whitebg
```

```
% Valorile elementelor de circuit
Vg=50; L=0.197e-3; C=0.051e-6; Cf=32e-6; R=60;
fN1=0.6; fN2=0.85; % valorile de salt ale frecvenței
```

```
% Se calculează alte elemente necesare simulării
f0=1/(2*pi*sqrt(L*C)); Z0=sqrt(L/C); f1=fN1*f0; f2=fN2*f0;
% Se calculează condiția inițială pt tensiunea de ieșire
theta1=pi*f0/f1; Q1=theta1*Z0/(2*R); theta2=pi*f0/f2; Q2=theta2*Z0/(2*R);
Vo1=Vg*(sqrt(2*Q1^2*(1+cos(theta1)))+(1-cos(theta1))^2)-sign(f1-f0)*Q1*(1+cos(theta1)))/(1-cos(theta1)+Q1^2*(1+cos(theta1)));
Vo2=Vg*(sqrt(2*Q2^2*(1+cos(theta2)))+(1-cos(theta2))^2)-sign(f2-f0)*Q2*(1+cos(theta2)))/(1-cos(theta2)+Q2^2*(1+cos(theta2)));
simul
```

*Fișierul simul.m* (fișier SIMULINK)



## A15 Comportarea la semnal mare a unui convertor SRC - setul de fișiere CASPOC

Fișierul src.cir

```
* Convertor rezonant serie. Simulare la salturi ale frecvenței de comutație.
Vg1 2 1 50
Vg2 1 0 50
Dgtol 2 3 uni sgate=gatel
D1 3 2 did
Dgtol2 3 0 uni sgate=gate2
D2 0 3 did
C 3 4 51n
L 4 5 197u
Dr1 5 6 did
Dr2 1 6 did
Dr3 7 1 did
Dr4 7 5 did
Cf 6 7 32u
R 6 7 60

.model uni GTO vthn=0 vthoff=0 vthgate=1u ron=1u roff=10meg
.model did Dswitch vthn=0 vthoff=0 ron=1u roff=10meg
.draw 1 i(L)
.draw 2 v(6,7)
.draw 3 v(3,5)
.draw 4 b(gatel)
.draw 5 b(gate2)
.tran 2n 1 uic
.options Tscreen=10m method=euler
.end
```

Fișierul src.stm

```
t time
f1 con 30.127k
f2 con 42.68k
df sub f2 f1
f signal t f1 df 200 0 0.2 30.127k 15m 4
gatel signal t 0 1 f 0 0.5 0 0 1
gate2 mul gatel -1
.end
```

## A16 Funcțiile de transfer control-ieșire și audiosusceptibilitate pentru un convertor SRC - setul de fișiere MATLAB

% Modelul unitar (pt.  $f < f_0$  și  $f > f_0$ ) de semnal mic al convertorului rezonant serie.

```
clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end
```

```
% Valorile elementelor de circuit
L=197e-6; C=51e-9; Cf=32e-6;
Qs=6; % pt a lucra cu datele lui Vorperian [201]
fN=0.6;
```

```
% Se calculeaza elementele necesare
f0=1/(2*pi*sqrt(L*C)); F=fN*f0; Z0=sqrt(L/C); R=2*pi*f0*L/Qs; theta0=pi*f0/F; Q=theta0*Z0/(2*R);
M=(sqrt(2*Q^2*(1+cos(theta0)+(1-cos(theta0))^2))+Q*(1+cos(theta0)))/(1-cos(theta0)+Q^2*(1+cos(theta0)));
Vg=10*(8/20)*F*((M*Q*(Q*M+sign(F-f0))*(1+cos(theta0))+((M)^2-(Q*M+sign(F-f0))^2)*theta0/2*sin(theta0))/(Q*(Q*M+sign(F-f0))*(1+cos(theta0))+M*(1-cos(theta0))));
```

```
Vo=Vg*(sqrt(2*Q^2*(1+cos(theta0)+(1-cos(theta0))^2))+Q*(1+cos(theta0)))/(1-cos(theta0)+Q^2*(1+cos(theta0)));
wc=1/(R*C*f)*(1+(Vo/Vg*(1-cos(theta0)))/(Q*(Q*Vo/Vg+sign(F-f0))*(1+cos(theta0))));
Ac=Vg/F*(Vo/Vg*Q*(Q*Vo/Vg+sign(F-f0))*(1+cos(theta0))+((Vo/Vg)^2-(Q*Vo/Vg+sign(F-f0))^2)*theta0/2*sin(theta0))/(Q*(Q*Vo/Vg+sign(F-f0))*(1+cos(theta0))+Vo/Vg*(1-cos(theta0)));
Ag=(1-cos(theta0)-sign(F-f0)*Q*Vo/Vg*(1+cos(theta0)))/(Q*(Q*Vo/Vg+sign(F-f0))*(1+cos(theta0))+Vo/Vg*(1-cos(theta0)));
```

```
k=200; % numarul de puncte de pe axa frecvenței in care se reprezinta caracteristicile
f=logspace(1,4,k); % axa frecvenței de la 10 la 10^4, logaritmică
w=2*pi*f; % definire omega
j=sqrt(-1);
```

```
% Bucla principala pentru calculul funcției de transfer control-iesire si audiosusceptibilitate
for n=1:k
```



```

s=j*w(n);
hc(n)=Ac*1/(1+s/wc);
hg(n)=Ag*1/(1+s/wc);
end
magc=20*log10(abs(hc));
phasec=180/pi*unwrap(angle(hc));
magg=20*log10(abs(hg));
phaseg=180/pi*unwrap(angle(hg));

% Plotarea caracteristicilor
semilogx(f,magc,'k');hold on;
title('Funcția de transfer control-iesire')
xlabel('Frecvența [Hz]'); ylabel('Amplitudine [dB]');

figure;
semilogx(f,phasec,'k'); hold on
title('Funcția de transfer control-iesire')
xlabel('Frecvența [Hz]'); ylabel('Faza [°]');

figure
semilogx(f,magg,'k');hold on;
title('Audiosusceptibilitatea')
xlabel('Frecvența [Hz]'); ylabel('Amplitudine [dB]');

figure;
semilogx(f,phaseg,'k'); hold on
title('Audiosusceptibilitatea')
xlabel('Frecvența [Hz]'); ylabel('Faza [°]');

```

## A 17 Implementarea modelului în spațiul stărilor pentru convertoare cu comandă prin sarcină - setul de fișiere MATLAB

*Fisierul chrgstsp.m*

% Modelul in spatiul starilor pt un convertor BUCK. Functiile de transfer control-iesire comparativ cu modelul Ridley

```

clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

k=401; % numarul de puncte de pe axa frecventei in care se reprezinta caracteristicile
f=logspace(2,5,k); % axa frecventei de la 10^2 la 10^5, logaritmica
w=2*pi*f; % definire omega

% Marimile de circuit:
Vg=12; D=0.42; Dprime=1-D; fs=90e3; Ts=1/fs; % comanda
L=37.5e-6; R=20; C=380e-6; Rc=20e-3; CT=733e-9; % elementele de circuit

% Functia de transfer control iesire folosind modelul mediat Ridley
Fm=CT/(Vg*(D/R+D*Dprime/(2*L*fs))*Ts);
Ri=D*Ts/CT; wn=pi/Ts; Qz=-2/pi; Kr=(D*Dprime*Ts^2)/(2*L*CT); % parametrii functiei
A1=-1/L*(R*Rc/(R+Rc)) -R/(L*(R+Rc)); 1/C*(R/(R+Rc)) -1/(C*(R+Rc)); % matricile numai
A2=[-1/L*(R*Rc/(R+Rc)) -R/(L*(R+Rc)); 1/C*(R/(R+Rc)) -1/(C*(R+Rc));] % pt. convertor (2X2)
B1=[1/L; 0; ];
B2=[0; 0; ];
E1=[R*Rc/(R+Rc) R/(R+Rc)];
E2=[R*Rc/(R+Rc) R/(R+Rc)];
F1=[0];
F2=[0];
I2=eye(2);
AD=D*A1+Dprime*A2;
BD=D*B1+Dprime*B2;
ED=D*E1+Dprime*E2;
FD=D*F1+Dprime*F2;
X=-inv(AD)*B2*Vg;
csid=(A1-A2)*X+(B1-B2)*Vg;
zetad=(E1-E2)*X+(F1-F2)*Vg;
e1=[1 0]; % matricea extractoare a lui iL din x al convertorului de putere.

for m=1:k
He(m)=1+j*w(m)/(wn*Qz)+(j*w(m))^2/(wn^2);
Hd(m)=ED*(inv(j*w(m)*I2-AD))*csid+zetad; % functia de transfer a convertorului de putere de la d la vo.

```

```
havegen(m)=(Fm*Hd(m))/(1+Fm*Ri*He(m)*e1/(j*w(m)*l2-AD)*csid-Fm*Kr*Hd(m)); % functia de transfer de la vc la vo (pt intreg sistemul).
end
```

```
magavegen=20*log10(abs(havegen));
phaseavegen=180/pi*unwrap(angle(havegen));
```

```
% Exprimarea cu modelul in spatiul starilor pt. intreg sistemul (inclusiv He(s))- model de ordinul 4 pt BUCK.
```

```
K1=(-D^2*Dprime*Ts^2)/(2*CT); K2=(D*Dprime^2*Ts^2)/(2*CT);
hx=e1*(K1*A1-K2*A2); hg=e1*(K1*B1-K2*B2);
Ap=[-wn/Qz 1; -wn^2 0];
Bp=[0; wn^2];
Ep=[1 0];
Fp=[0];
n=size(A,1); % se determina ordinul modelului in spatiul starilor pt.etajul de putere.
In=eye(n);
l2=eye(2);
In_2=eye(n+2);% se defineste matricea In+2 (de ordin cu 2 mai mare decat al etajului de putere)
A=(In-1/(Ri*e1*csid)*csid*Ri*e1)*AD 1/(Ri*e1*csid)*csid*Ep*Ap;
Bp*(hx+1/(Fm*Ri*e1*csid)*Ri*e1*AD) (l2-1/(Fm*Ri*e1*csid)*Bp*Ep)*Ap];
B=[(In-1/(Ri*e1*csid)*csid*Ri*e1)*BD; Bp*(hg+1/(Fm*Ri*e1*csid)*Ri*e1*BD)];
J=[zeros(n,1);Bp];
E=[ED-1/(Ri*e1*csid)*zetad*Ri*e1*AD 1/(Ri*e1*csid)*zetad*Ep*Ap];
F=[FD-1/(Ri*e1*csid)*zetad*Ri*e1*BD];
H=[0];
```

```
for l=1:k
Hstsp(l)=E/(j*w(l)*In_2-A)*J+H;
Audiostsp(l)=E/(j*w(l)*In_2-A)*B+F;
end
```

```
magstsp=20*log10(abs(Hstsp));
phastsp=180/pi*unwrap(angle(Hstsp));
```

```
% Ploteaza caracteristicile de amplitudine si faza
```

```
semilogx(f,magavegen,'k-');hold on;
semilogx(f,magstsp,'k'); hold on;
title('Convertor BUCK PWM cu comanda prin sarcina')
xlabel('Frecventa [Hz]'); ylabel('Amplitudine [dB]');
```

```
figure;
```

```
semilogx(f,phaseavegen,'k-'); hold on;
semilogx(f,phastsp,'k-');
title('Convertor BUCK PWM cu comanda prin sarcina')
xlabel('Frecventa [Hz]'); ylabel('Faza [°]');
```

```
% Gasirea polilor si zerourilor functiei de transfer control iesire.
```

```
format long
[P,Z]=pzmap(A,J,E,H);
disp('Polii si zerourile sistemului: ');
fprintf('\n');
disp(' P= ');
disp(P/(2*pi));
fprintf('\n');
disp(' Z= ');
disp(Z/(2*pi));
```

## A18 Un nou model de semnal mic pentru convertoare cu comanda integrativa, ce tine cont de efectul de esantionare. Trasarea functiei de transfer control-iesire si studiul stabilitatii

```
% Functia de transfer control iesire exacta, mediata cf. Smedley
% si mediata cf. Lascu pentru un convertor CUK cu comanda integrativa dupa o variabila
% Calculul polilor si a zerourilor.
```

```
clear % sterge variabile
Fer=4;
for i=1:Fer
close % sterge ultimele Fer ferestre grafice
end
```

```
k=201; % numarul de puncte de pe axa frecventei in care se reprezinta caracteristicile
f=logspace(2,5,k); % axa frecventei de la 10^2 la 10^5, logaritmica
w=2*pi*f; % definire omega
```

% Marimile de circuit:

Vg=12; D=0.8; Dprime=1-D; fs=45e3; Ts=1/fs; %comanda

L1=0.62e-3; RL1=30e-3; L2=0.2e-3; RL2=10e-3; R=10; C1=4.7e-6; C2=2.2e-6; RC2=80e-3; CT=1e-9; Ron=1e-3; %elementele de circuit

flag=0;

t1=ceil(2\*f(length(f))/fs)+1;

for i=1:length(f)

  for l=1:t1

    if f(i)==l\*fs/2

      flag=1;

    else

      end

  end

end

if flag==1

  error('Frequency points not properly chosen - a frequency point is equal to p\*fs/2, p natural number')

else

end

% Matricile aferente celor doua stari topologice ale convertorului:

```
A1=[ -RL1/L1      0      0      0      0;
      0  -1/L2*(RL2+R*RC2/(R+RC2))  1/L2  -1/L2*R/(R+RC2)  0;
      0      -1/C1      0      0      0;
      0  1/C2*R/(R+RC2)      0  -1/(C2*(R+RC2))  0;
      0      0      1/Ts      0      0];
```

B1=[1/L1; 0; 0; 0; 0];

E1=[0 R\*RC2/(R+RC2) 0 R/(R+RC2) 0];

F1=[0];

```
A2=[ -RL1/L1      0      -1/L1      0      0;
      0  -1/L2*(RL2+R*RC2/(R+RC2))  0  -1/L2*R/(R+RC2)  0;
      1/C1      0      0      0      0;
      0  1/C2*R/(R+RC2)      0  -1/(C2*(R+RC2))  0;
      0      0      0      0  -1/(Ron*CT)];
```

B2=[1/L1; 0; 0; 0; 0];

E2=[0 R\*RC2/(R+RC2) 0 R/(R+RC2) 0];

F2=[0];

I=eye(5);

% Matricile si toate marimile ce nu depind de w, necesare in calcul (dupa prof. Tymerski)

T1=0; T2=D\*Ts; T3=Ts; % calculul timpilor stationari

[phi1, psi1]=c2d(A1,B1,(T2-T1)); % modelul analogic in spatiul starilor se

[phi2, psi2]=c2d(A2,B2,(T3-T2)); % converteste in model discret

XT1=(I-phi2\*phi1)\(phi2\*psi1+psi2)\*Vg; % valoarea din stare stationara la T1

XT2=(I-phi1\*phi2)\(phi1\*psi2+psi1)\*Vg; % valoarea din stare stationara la T2

XT3=XT1; % valoarea din stare stationara la T3

gamma1=A1\*XT2+B1\*Vg; %matricile necesare pt. perturbatii

gamma2=A2\*XT3+B2\*Vg;

csi1=(A1-A2)\*XT2+(B1-B2)\*Vg;

zeta1=(E1-E2)\*XT2+(F1-F2)\*Vg;

h1=[0 0 0 1]; % matricea extractoare pt tensiunea pe CT

miu1=inv(h1\*gamma1);

Gamma1=I-miu1\*csi1\*h1;

% Bucla principala pentru calculul functiei de transfer cu formula exacta

for n=1:k

  beta1=expm(-j\*w(n)\*(T2-T1))\*phi1;

  beta2=expm(-j\*w(n)\*(T3-T2))\*phi2; % Aici se definesc toti coeficientii care depind de w

  alpha1=E1\*(j\*w(n)\*I-A1)\(I-beta1);

  alpha2=E2\*(j\*w(n)\*I-A2)\(I-beta2);

  hex(n)=(miu1/Ts)\*((alpha1\*beta2+alpha2-zeta1\*miu1\*h1\*beta1\*beta2)/(I-Gamma1\*beta1\*beta2)\*csi1+zeta1);

end

magex=20\*log10(abs(hex));

phaseex=180/pi\*unwrap(angle(hex));

% Functia de transfer control iesire folosind modelul mediat Smedley [171], [172]

```
Aq1=[ -RL1/L1      0      0      0      ;
      0  -1/L2*(RL2+R*RC2/(R+RC2))  1/L2  -1/L2*R/(R+RC2)  ;
      0      -1/C1      0      0      ;
      0  1/C2*R/(R+RC2)      0  -1/(C2*(R+RC2))  ];
```

Bq1=[1/L1; 0; 0; 0];

```
Eq1=[0 R*RC2/(R+RC2) 0 R/(R+RC2)];
Fq1=[0];
```

```
Aq2=[-RL1/L1 0 -1/L1 0 ;
0 -1/L2*(RL2+R*RC2/(R+RC2)) 0 -1/L2*R/(R+RC2)
1/C1 0 0 0 ;
0 1/C2*R/(R+RC2) 0 -1/(C2*(R+RC2)) ];
```

```
Bq2=[1/L1; 0; 0; 0];
Eq2=[0 R*RC2/(R+RC2) 0 R/(R+RC2)];
Fq2=[0];
I4=eye(4);
```

```
AD=D*Aq1+Dprime*Aq2;
BD=D*Bq1+Dprime*Bq2;
ED=D*Eq1+Dprime*Eq2;
FD=D*Fq1+Dprime*Fq2;
X=-inv(AD)*BD*Vg;
csid=(Aq1-Aq2)*X+(Bq1-Bq2)*Vg;
zetad=(Eq1-Eq2)*X+(Fq1-Fq2)*Vg;
h2=[0 0 1 0]; % matricea extractoare a lui vC1 din x pt convertor
```

```
for m=1:k
havegen(m)=((ED-zetad*D/(h2*X)*h2)/(j*w(m)*I4-AD+csid*D/(h2*X)*h2)*csid+zetad)*1/(h2*X); % calculul la modul general, care nu tine cont
de % convertor
end
magavegen=20*log10(abs(havegen));
phaseavegen=180/pi*unwrap(angle(havegen));
```

```
% Formula din articolul Smedley.
%for m=1:k
%have(m)=D/Dprime*(1-(D*L1)/(R*Dprime^2)*(j*w(m)))/(1+L1*C1/(Dprime^2)*(j*w(m))^2);
%end
```

```
% Formula din modelul mediat propus de Lascu.
%e1=[1 0 0 0]; e2=[0 1 0 0]; e3=[0 0 1 0]; e4=[0 0 0 1];
%Vc1DTS=e3*X-1/2*(e2*X)/C1*D*Ts; Fm=1/Vc1DTS; Av=D;
%Kon=D^2*(1-D)*Ts/(2*C1); Koff=-D*(1-D)^2*Ts/(2*C1);
%wn=pi/Ts; Qz=-2/pi;
%for i=1:k
%He(i)=1+1/(wn*Qz)*j*w(i)+(j*w(i))^2/wn^2;
%Hmine(i)=(ED+zetad*(Fm*Kon*e2+Fm*Koff*e1-Fm*Av*He(i)*e3))/(j*w(i)*I4-AD-csid*(Fm*Kon*e2+Fm*Koff*e1-
Fm*Av*He(i)*e3))*csid+Fm+zetad*Fm;
%end
%magmine=20*log10(abs(Hmine));
%phasemine=180/pi*unwrap(angle(Hmine));
```

```
% Exprimarea cu modelul in spatiul starilor propus de Lascu si dedus similar cu cel
% de la comanda prin sarcina pt. intreg sistemul (inclusiv He(s)) - model de ordinul 6.
```

```
e1=[1 0 0 0]; e2=[0 1 0 0]; e3=[0 0 1 0]; e4=[0 0 0 1];
Kon=D^2*(1-D)*Ts/(2*C1); Koff=-D*(1-D)^2*Ts/(2*C1);
wn=pi/Ts; Qz=-2/pi;
Vc1DTS=e3*X-1/2*(e2*X)/C1*D*Ts; Fm=1/Vc1DTS; Av=D;
hx=Kon*e2+Koff*e1; hg=[0];
Ap=[-wn/Qz 1; -wn^2 0];
Bp=[0; wn^2];
Ep=[1 0];
Fp=[0];
I4=eye(4);
I2=eye(2);
I6=eye(6);
A=[(I4-1/(Av*e3*csid))*csid*Av*e3*AD 1/(Av*e3*csid)*csid*Ep*Ap;
Bp*(hx+1/(Fm*Av*e3*csid))*Av*e3*AD (I2-1/(Fm*Av*e3*csid))*Bp*Ep*Ap];
B=[(I4-1/(Av*e3*csid))*csid*Av*e3*BD; Bp*(hg+1/(Fm*Av*e3*csid))*Av*e3*BD];
cs=[zeros(4,1);Bp];
E=[(I4-1/(Av*e3*csid))*zetad*Av*e3*AD 1/(Av*e3*csid)*zetad*Ep*Ap];
F=[FD-1/(Av*e3*csid)*zetad*Av*e3*BD];
zeta=[0];
```

```
for l=1:k
Hstsp(l)=E/(j*w(l)*I6-A)*csi+zeta;
Audiostsp(l)=E/(j*w(l)*I6-A)*B+F;
end
```

```
magstsp=20*log10(abs(Hstsp));
phaststsp=180/pi*unwrap(angle(Hstsp));
```

```

% plot the magnitude (amplitude)and phase
%semilogx(f,magex,'k-'); hold on;
semilogx(f,magavegen,'r-');hold on;
%semilogx(f,magmine,'g-'); hold on;
semilogx(f,magstsp,'b'); hold on;
title('Convertor CUK cu comanda integrativa dupa o variabila')
xlabel('Frecventa [Hz]'); ylabel('Amplitudine [dB]');

figure;
%semilogx(f,phaseex,'k-'); hold on;
semilogx(f,phasevegen,'r-'); hold on;
%semilogx(f,phasemine,'g-'); hold on;
semilogx(f,phasestsp,'b-');
title(' Convertor CUK cu comanda integrativa dupa o variabila ')
xlabel('Frecventa [Hz]'); ylabel('Faza [°]');

% Gasirea polilor si zerourilor functiei de transfer control iesire.
format long
[P,Z]=pzmap(A,csi,E,zeta);
disp('Polii si zerourile sistemului: ');
fprintf('\n');
disp(' P= ');
disp(P/(2*pi));
fprintf('\n');
disp(' Z= ');
disp(Z/(2*pi));

```

## A 19 Fişierele CASPOC de simulare a convertorului CUK cu comanda integrativa după o variabila studiat la punctul precedent

### Fişierul onecycle.cir

```

* Controlul integrativ după o variabilă ("one cycle control")
* al convertoarelor PWM (aici CUK)
Vg 1 0 12
L1 1 2 0.62m
RL1 2 3 30m
S 3 0
C1 3 4 4.7u ic=12
D 4 0 did
RL2 5 4 10m
L2 6 5 0.2m
C2 6 7 2.2u
RC2 7 0 80m
R 0 6 10
*****
.model did DSWITCH vthon=0 vthoff=0
.option method=Euler Tscreen=4m
.tran 100n 5 uic
.draw 1 b(Vc)
.draw 2 b(integ)
.draw 3 v(0,6)
.draw 4 b(reset)
.draw 5 b(clock)
.draw 6 b(bist)
.draw 7 i(L2)
*.draw 8 v(3,2)
.end

```

### Fişierul onecycle.stm

```

t time
vap VOLTAGE 0 4
*****1/tau****
***** i1 i2 i3 p1 p2 p3
norm sum vap 0 0 40k 0 0 0
clock signal t 0.5 0.5 40k 0 0.02 0 0 1
integ intreset norm cond 0
bist fflt reset clock 0.5u 0.5u 1 1
Vc con 4
dif sub integ Vc
reset bng dif 0 1 0
S switch bist 0 0.5 S
nbist not bist
cond add nbist -0.5
*****
.end

```

## A 20 Modelul de semnal mare pentru convertoare QRC implementat în MATLAB/SIMULINK. Comportarea la semnal treapta in freventa de comanda

Fișierul *qrc\_ord2.m*

```
% Simularea unitara a convertoarelor QRC cu un model neliniar de semnal mare.
clear all

% Convertorul BOOST ZVS QRC HW
Vg=15; Lr=1.6e-7; Cr=0.064e-6; L=400e-6; C=100e-6; Rn=0.8; zc=2; n=1;
f0=1/(2*pi*sqrt(Lr*Cr)); Z0=sqrt(Lr/Cr); R=Rn*Z0;
% matricile aferente starii "on"
A1=[0 0;
    0 -1/(R*C)];
B1=[1/L; 0];
E1=[0 1];
F1=[0];
% matricile aferente starii "off"
A2=[0 -1/L;
    1/C -1/(R*C)];
B2=[1/L; 0];
E2=[0 1];
F2=[0];

fs1=0.2*f0; fs2=0.4*f0;
fs=fs1;
% Calculeaza starea initiala (corespunzatoare lui fs1)
% Se gaseste raportul de conversie M prin
% rezolvarea ecuatiei transcendente cu metoda Newton-Raphson scalara
% Aceasta este necesara pt a da valorile initiale pt. variabilele de stare corespunzatoare
% starii stationare initiale
ma=1; mc=0;
flag=1;
Ma=ma; Mc=mc; err=1e-5; iter=0;
while ~(Ma==Mc)
% mc=ma-(ma-ma*1/(2*pi)*fs/f0*quasi(ma/Rn,n)-1)/(1-1/(2*pi)*fs/f0*quasi(ma/Rn,n)-ma/Rn*1/(2*pi)*fs/f0*quasider(ma/Rn,n));
% procesul iterativ Newton pt BOOST ZVS
% mc=ma-(ma-1+1/(2*pi)*fs/f0*quasi(Rn/ma,n))/(1+1/(2*pi)*fs/f0*quasider(Rn/ma,n)*(-Rn/ma^2)); % iteratia pt BUCK ZVS
mc=ma-(ma*1/(2*pi)*fs/f0*quasi(Rn/ma,n)-1)/(1/(2*pi)*fs/f0*(quasi(Rn/ma,n)-quasider(Rn/ma,n)*Rn/ma)); % iteratia pt BOOST ZVS
iter=iter+1;
dif=abs(mc-ma);
if dif>err flag=1;
else flag=0;
end
if ~(flag==0), ma=mc;
else Ma=Mc;
end
end
M=mc; iter

vc1=M*Vg; iL1=(M*Vg)^2/(R*Vg);
if zc==1
qrc_zcs
else
qrc_zvs
end
```

Fișierul *quasider.m*

```
% Functia cvasirezonaanta derivata
function z=quasider(x,n)
z=1/2-(x.^(-2)).*(1+(-1).^(n-1)).*sqrt(1-x.^2));
```

Fișierul *quasi.m*

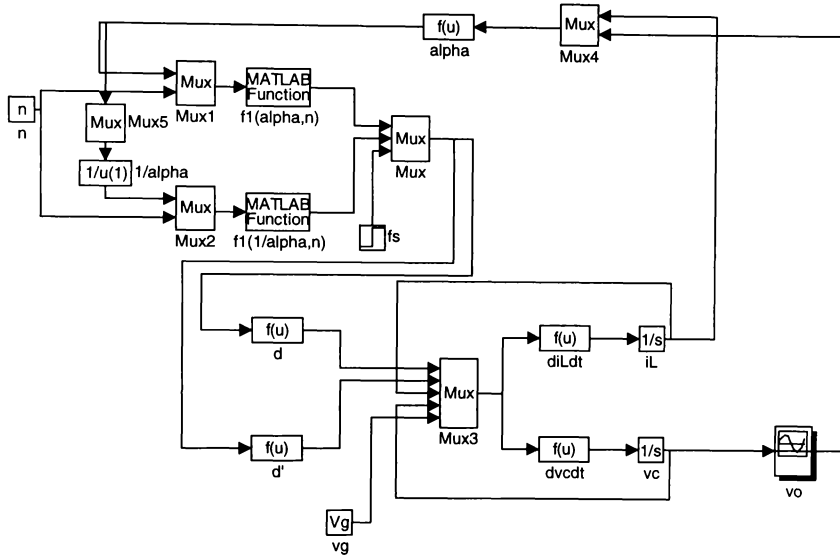
```
% Definese functia cvasirezonaanta
function z=quasi(x,n)
z=x./2+n.*pi+(-1).^(n-1).*asin(x)+(x.^(-1)).*(1+(-1).^(n-1)).*sqrt(1-x.^2));
```

Fișierul *fl.m*

```
% Definirea functiei fl pt calculul lui ton si toff;
% Functia fl este functia cvasirezonaanta, deci identica cu functia quasi
```

% doar ca nu are 2 variabile scalare ci una singura, dar de tip vector cu 2 componente  
function y=f1(x)  
y=x(1)/2+x(2)\*pi+(-1)^(x(2)-1)\*asin(x(1))+1/x(1)\*(1+(-1)^(x(2)-1)\*sqrt(1-x(1)^2));  
%y=2\*pi; pentru FW

Fișierul qrc\_zvs.m (fișier SIMULINK)



## A 21 Modelul de semnal mare pentru convertoare QRC implementat în PETS. Comportarea la semnal treapta in frecventa de comanda

\* Convertor ZVS HW BOOST la salt treapta in frecventa de comanda

```
Vg 1 0 DC 15
L 1 2 400u ic=101.522
L0 2 4 160n ic=101.522
C0 4 0 64n ic=0
#VCS 4 0 30 0 switch
.model switch #vcs vton=7.5 vtoff=7.5
*Danti 0 4 did
#D 2 5 did
.model did #D
C 5 0 100u ic=43.419
R 5 0 1.2649
```

```
#HC 10 4 11 comp
.model comp #HC
Vcomp 10 0 DC 5m
```

```
vser1 11 12 pulse 0 15 0 20n 3.179u
* pe frecventa fs1 de perioada 3.179us
#vcser1 12 14 100 0 switch
vser2 11 13 pulse 0 15 0 20n 1.5895u
* pe frecventa fs2 de perioada 1.5895us
#vcser2 13 14 200 0 switch
```

```
#FFbist 14 15 30 bist off
.model bist #PF
#vcsres1 15 16 100 0 switch
vreset1 16 0 pulse 0 15 0 20n 3.179u
* pe frecventa fs1
#vcsres2 15 17 200 0 switch
vreset2 17 0 pulse 0 15 0 20n 1.5895u
* pe frecventa fs2
```

```
* control
vcom1 100 0 pulse -15 15 0 4m 10
vcom2 200 0 pulse 15 -15 0 4m 10

.tran 1600n 12m uic
.probe v(5)
.end
```

## A 22 Modelul de semnal mic pentru convertoare QRC implementat în MATLAB. Funcțiile de transfer control-ieșire

Fișierul smsgboos.m

```
% Functia de transfer control-iesire la un BOOST ZCS FW

clear all % sterge variabile
Fer=2;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

k=401; % numarul de puncte de pe axa frecventei in care se reprezinta caracteristicile
f=logspace(1,5,k); % axa frecventei de la 10^-1 la 10^5, logaritmica
w=2*pi*f; %definire omega

% Elementele de circuit:
Vg=15; Fs=30e3; L0=3.47e-6; C0=430e-9; L=500e-6; C=100e-6; R=10;
f0=1/(2*pi*sqrt(L0*C0)); Z0=sqrt(L0/C0);

A=1/f0*Vg/(1-Fs/f0)^2;
for n=1:k
    s=j*w(n);
    num(n)=1-L/(R*(1-Fs/f0)^2)*s;
    den(n)=1+L/(R*(1-Fs/f0)^2)*s+L*C/(1-Fs/f0)^2*s^2;
    Hc(n)=A*num(n)/den(n);
end
mag=20*log10(abs(Hc));
phase=180/pi*unwrap(angle(Hc));

semilogx(f,mag,'k'); xlabel('Frequency [Hz]'); ylabel('Magnitude [dB]'); title('ZCS QRC FW BOOST
converter'); hold on
figure
semilogx(f,phase,'k'); xlabel('Frequency [Hz]'); ylabel('Phase [Deg]');
title('ZCS QRC FW BOOST converter');
```

## A 23 Fișierul MATLAB de comparatie a modelelor discrete bazate pe diferite metode de integrare numerica. Calculul polilor și zerourilor furnizate de aceste modele și răspunsul la semnal treapta în mărimea de comanda

Fișierul compar.m

```
% Functiile de transfer (in z) pt diverse metode de integrare numerica
% pt un convertor BOOST CCM.
```

```
Fer=7;
for i=1:Fer
    close % delete the last 'Fer' graphic windows
end
clear all

L=3.704e-3; RL=0; C=32.1e-6; Rc=0; R=180;
Vg=10; D=0.6; fs=6.25e3; Ts=1/fs;

A1=[-RL/L 0;
     0 -1/((R+Rc)*C)];
B1=[1/L; 0];
E1=[0 R/(R+Rc)]; F1=[0];

A2=[-1/L*(RL+R*Rc/(R+Rc)) -1/L*R/(R+Rc);
     1/C*R/(R+Rc) -1/(C*(R+Rc))];
B2=[1/L; 0];
E2=[R*Rc/(R+Rc) R/(R+Rc)]; F2=[0];
```



```
l2=eye(2);
AD=D*A1+(1-D)*A2; BD=D*B1+(1-D)*B2; ED=D*E1+(1-D)*E2; FD=D*F1+(1-D)*F2;
```

```
% Matricile pentru modelul mediat continuu (in s)
% Matricile pt calculul PSF (ss=steady-state)
```

```
Assc=AD;
Bssc=BD;
Essc=ED;
Fssc=FD;
```

```
Xc=-inv(Assc)*Bssc*Vg;
Voc=(-Essc*inv(Assc)*Bssc+Fssc)*Vg;
```

```
% Matricile modelului continuu liniarizat
```

```
Ac=AD;
Bc=BD;
Ec=ED;
Fc=FD;
COM1c=(A1-A2)*Xc+(B1-B2)*Vg; % notata uneori si cu csi
COM2c=(E1-E2)*Xc+(F1-F2)*Vg; % notata uneori si cu zeta
```

```
% Functia de transfer control-iesire. Coeficientii polinoamelor sunt in ordinea descrescatoare a puterilor lui s
```

```
[NUMc DENc]=ss2tf(Ac,COM1c,Ec,COM2c);
%figure; sgrid('new');
%{Pc,Zc} = pzmap(NUMc, DENc);
%pzmap(NUMc, DENc);
```

```
% conversie s -> z
```

```
[NUMconv,DENconv]=c2dm(NUMc,DENc,Ts,'zoh');
%figure; zgrid('new');
[Pconv,Zconv] = pzmap(NUMconv, DENconv);
```

```
% Matricile pentru modelul discret cu "forward Euler"
```

```
% Matricile pt calculul PSF (ss=steady-state)
%Assf=l2+Ts*AD+Ts^2*D*(1-D)*(A2*A1); % pentru provenientia din modelul comutat
%Bssf=Ts*BD+Ts^2*D*(1-D)*(A2*B1); % pentru provenientia din modelul comutat
%Essf=E1; % pentru provenientia din modelul comutat
%Fssf=F1; % pentru provenientia din modelul comutat
Assf=l2+Ts*AD; % pentru provenientia din modelul mediat
Bssf=Ts*BD; % pentru provenientia din modelul mediat
Essf=ED; % pentru provenientia din modelul mediat
Fssf=FD; % pentru provenientia din modelul mediat
```

```
Xf=inv(l2-Assf)*Bssf*Vg;
Vof=(Essf*inv(l2-Assf)*Bssf+Fssf)*Vg;
```

```
% Matricile modelului liniarizat
```

```
Af=Assf;
Bf=Bssf;
Ef=Essf;
Ff=Fssf;
%COM1f=(Ts*(A1-A2)+Ts^2*(1-2*D)*(A2*A1))*Xf+(Ts*(B1-B2)+Ts^2*(1-2*D)*(A2*B1))*Vg; % prov. model comutat
%COM2f=0 % prov. model comutat
COM1f=Ts*(A1-A2)*Xf+Ts*(B1-B2)*Vg; % prov. model mediat
COM2f=(E1-E2)*Xf+(F1-F2)*Vg; % prov. model mediat
```

```
% Functia de transfer control-iesire. Coeficientii polinoamelor sunt in ordinea descrescatoare a puterilor lui z
```

```
[NUMf DENf]=ss2tf(Af,COM1f,Ef,COM2f);
[Pf,Zf] = pzmap(NUMf, DENf);
```

```
% Matricile pentru modelul discret cu "backward Euler"
```

```
% Matricile pt calculul PSF (ss=steady-state)
%Assb=inv(l2-Ts*AD+Ts^2*D*(1-D)*(A1*A2)); % pentru provenientia din modelul comutat
%Bssb=Assb*(Ts*BD-Ts^2*D*(1-D)*(A1*B2)); % pentru provenientia din modelul comutat
%Essb=E1; % pentru provenientia din modelul comutat
%Fssb=F1; % pentru provenientia din modelul comutat
Assb=inv(l2-Ts*AD); % pentru provenientia din modelul mediat
Bssb=Assb*(Ts*BD); % pentru provenientia din modelul mediat
Essb=ED; % pentru provenientia din modelul mediat
Fssb=FD; % pentru provenientia din modelul mediat
```

```
Xb=inv(l2-Assb)*Bssb*Vg;
Vob=(Essb*inv(l2-Assb)*Bssb+Fssb)*Vg;
```

```
Ab=Assb;
```

```

Bb=Bssb;
Eb=Essb;
Fb=Fssb;
%COM1b=Ab*(Ts*(A1-A2)+Ts^2*(1-2*D)*(A1*A2))*Xb+(Ts*(B1-B2)-Ts^2*(1-2*D)*(A1*B2))*Vg; % prov. model comutat
%COM2b=0; % prov. model comutat
COM1b=Ab*(Ts*(A1-A2))*Xb+Ts*(B1-B2)*Vg; % prov. model mediat
COM2b=(E1-E2)*Xb+(F1-F2)*Vg; % prov. model mediat

```

```

% Functia de transfer control-iesire. Coeficientii polinoamelor sunt in ordinea descrescatoare a puterilor lui z
[NUMb DENb]=ss2tf(Ab,COM1b,Eb,COM2b);
[Pb,Zb] = pzmap(NUMb, DENb);

```

```

% Matricile pentru modelul discret cu "trapezoidal"

```

```

% Matricile pt calculul PSF (ss=steady-state)
%Asst=inv(I2-Ts*AD+Ts^2*D*(1-D)*(A1*A2)); % pentru provenienta din modelul comutat
%Bsst=Assb*(Ts*BD-Ts^2*D*(1-D)*(A1*B2)); % pentru provenienta din modelul comutat
%Esst=E1; % pentru provenienta din modelul comutat
%Fsst=F1; % pentru provenienta din modelul comutat
Asst=inv(I2-1/2*Ts*AD)*(I2+1/2*Ts*AD); % pentru provenienta din modelul mediat
Bsst=inv(I2-1/2*Ts*AD)*Ts*BD; % pentru provenienta din modelul mediat
Esst=ED; % pentru provenienta din modelul mediat
Fsst=FD; % pentru provenienta din modelul mediat

```

```

Xt=inv(I2-Asst)*Bsst*Vg;
Vot=(Esst*inv(I2-Asst)*Bsst+Fsst)*Vg;

```

```

At=Asst;
Bt=Bsst;
Et=Esst;
Ft=Fsst;

```

```

%COM1t=... % pentru provenienta din modelul comutat
%COM2t=0; % pentru provenienta din modelul comutat
COM1t=inv(I2-1/2*Ts*AD)*Ts*(A1-A2)*Xt+(B1-B2)*Vg; % pentru provenienta din modelul mediat
COM2t=(E1-E2)*Xt+(F1-F2)*Vg; % pentru provenienta din modelul mediat

```

```

% Functia de transfer control-iesire. Coeficientii polinoamelor sunt in ordinea descrescatoare a puterilor lui z
[NUMt DENt]=ss2tf(At,COM1t,Et,COM2t);
[Pt,Zt] = pzmap(NUMt, DENt);

```

```

plot(real(Pconv),imag(Pconv),'kx'); hold on;
plot(real(Pf),imag(Pf),'rx'); hold on;
plot(real(Pb),imag(Pb),'gx'); hold on;
plot(real(Pt),imag(Pt),'bx');
zgrid; axis([-1.1 1.1 -1.1 1.1]); xlabel('Real axis'); ylabel('Imaginary axis');

```

```

figure;
plot(real(Zconv),imag(Zconv),'kx'); hold on;
plot(real(Zf),imag(Zf),'rx'); hold on;
plot(real(Zb),imag(Zb),'gx'); hold on;
plot(real(Zt),imag(Zt),'bx'); hold on; xlabel('Real axis'); ylabel('Imaginary axis');

```

```

disp(' Zconv Zf Zb Zt');
disp([Zconv Zf Zb Zt]);
disp(' Pconv Pf Pb Pt');
disp([Pconv Pf Pb Pt]);
disp(' |Pconv| |Pf| |Pb| |Pt|');
disp(abs([Pconv Pf Pb Pt]));

```

```

NN=300; TT=0:Ts:(NN-1)*Ts;
[Yconv,Xconv]=dstep(NUMconv,DENconv,NN);
[Yf,Xf]=dstep(NUMf,DENf,NN);
[Yb,Xb]=dstep(NUMb,DENb,NN);
[Yt,Xt]=dstep(NUMt,DENt,NN);
[Yc,Xc]=step(NUMc,DENc,TT);

```

```

figure; plot(TT,Yconv,'k-',TT,Yf,'r'); axis([0 NN*Ts 0 120]); title('Forward Euler (rosu) comparativ cu modelul mediat din care provine (negru)');
figure; plot(TT,Yconv,'k-',TT,Yb,'r'); axis([0 NN*Ts 0 120]); title('Backward Euler (rosu) comparativ cu modelul mediat din care provine (negru)');
figure; plot(TT,Yconv,'k-',TT,Yt,'r'); axis([0 NN*Ts 0 120]); title('Trapezoidal (rosu) comparativ cu modelul mediat din care provine (negru)');

```

```

figure; plot(TT,Yconv,'k',TT,Yf,'r',TT,Yb,'g',TT,Yt,'b',TT,Yc,'y'); axis([0 NN*Ts 0 120]);

```

```

figure
subplot(4,1,1);dstep(NUMf,DENf,NN); %axis([0 NN 0 120]); %title('Raspunsul cu "forward Euler"');
subplot(4,1,2);dstep(NUMb,DENb,NN); axis([0 NN 0 120]); %title('Raspunsul cu "backward Euler"');
subplot(4,1,3);dstep(NUMt,DENt,NN); axis([0 NN 0 120]); %title('Raspunsul cu "trapezoidal"');

```

```
subplot(4,1,4);step(NUMc,DENc,TT); axis([0 NN*Ts 0 120]); %title('Raspunsul cu modelul continuu');
%subplot(5,1,5);dstep(NUMconv,DENconv,NN); axis([0 NN 0 120]); title('Raspunsul cu modelul continuu convertit in z');
```

## A 24 Studiul dinamicii de semnal mic a convertoarelor PWM folosind modelul bazat pe aproximația trapezoidală. Comportarea la variații în mărimea de comandă

Fișierul *trapez.m*

```
% Simularea comportarii unui convertor BOOST la semnal mic
% folosind modelul discret de semnal mic obtinut prin metoda trapezoidala.
clear all
Fer=4;
for i=1:Fer
    close
end
clear all

L=3.704e-3; RL=0.47; C=32.1e-6; Rc=0.356; R=67.7;
Vg=10; D=0.56; fs=6.25e3; Ts=1/fs;

A1=[-RL/L    0;
     0    -1/(R*C)];
B1=[1/L; 0];
E1=[0 R/(R+Rc)]; F1=[0];

A2=[-1/L*(RL+R*Rc/(R+Rc))    -1/L*R/(R+Rc);
     1/C*R/(R+Rc)            -1/(C*(R+Rc))];
B2=[1/L; 0];
E2=[R*Rc/(R+Rc) R/(R+Rc)]; F2=[0];

I2=eye(2);
AD=D*A1+(1-D)*A2; BD=D*B1+(1-D)*B2; ED=D*E1+(1-D)*E2; FD=D*F1+(1-D)*F2;

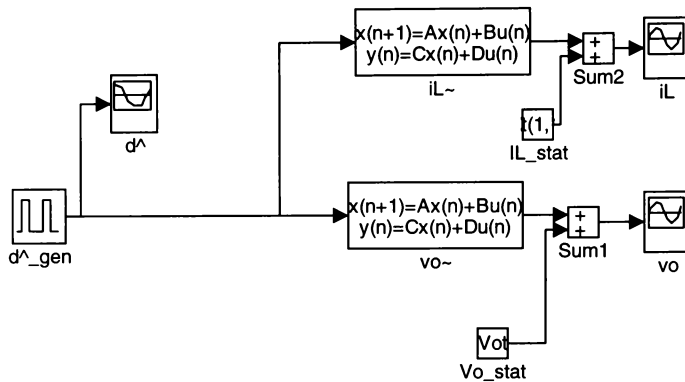
% Matricile pentru modelul discret cu "trapezoidal"
% Matricile pt calculul PSF (ss=steady-state)
% Asst=inv(I2-Ts*AD+Ts^2*D*(1-D)*(A1*A2)); % pentru provenientia din modelul comutat
% Bsst=Assb*(Ts*BD-Ts^2*D*(1-D)*(A1*B2)); % pentru provenientia din modelul comutat
% Esst=E1; % pentru provenientia din modelul comutat
% Fsst=F1; % pentru provenientia din modelul comutat
Asst=inv(I2-1/2*Ts*AD)*(I2+1/2*Ts*AD); % pentru provenientia din modelul mediat
Bsst=inv(I2-1/2*Ts*AD)*Ts*BD; % pentru provenientia din modelul mediat
Esst=ED; % pentru provenientia din modelul mediat
Fsst=FD; % pentru provenientia din modelul mediat

Xt=inv(I2-Asst)*Bsst*Vg
Vot=(Esst*inv(I2-Asst)*Bsst+Fsst)*Vg
%Vot1=Esst*Xt+Fsst*Vg

At=Asst;
Bt=Bsst;
Et=Esst;
Ft=Fsst;
%COM1t=... % pentru provenientia din modelul comutat
%COM2t=0; % pentru provenientia din modelul comutat
COM1t=inv(I2-1/2*Ts*AD)*Ts*(A1-A2)*Xt+(B1-B2)*Vg; % pentru provenientia din modelul mediat
COM2t=(E1-E2)*Xt+(F1-F2)*Vg; % pentru provenientia din modelul mediat

zt_model
```

Fișierul *zt\_model.m* (fișier SIMULINK)



## A 25 Calculul exact a stării staționare prin metoda Newton-Raphson vectorială - fișier MATLAB

Fișierul *n\_r\_vec.m*

% Aplicarea metodei Newton-Raphson la calculul exact al stării staționare  
% la un convertor dc-dc de tip DCVM SEPIC și funcția de transfer control-iesire exacta.

```
clear % șterge variabile
Fer=4;
for i=1:Fer
    close % șterge ultimele Fer ferestre grafice
end
format long;
```

```
D=0.6; Ts=20e-6; Dprime=1-D; fs=1/Ts; %comanda
Vg=12; L1=500e-6; L2=500e-6; C=100e-6; C1=220e-9; R=10; %elementele de circuit
Ron=1e-4; % se considera ca D e ideala iar S are Roff infinita și Ron nenula
```

% Matricile aferente celor trei stări topologice ale convertorului:

```
A1=[ -Ron/L1 -Ron/L1 0 0;
     -Ron/L2 -Ron/L2 0 1/L2;
     0 0 -1/(R*C) 0;
     0 -1/C1 0 0 ];
B1=[1/L1; 0; 0; 0]; E1=[0 0 1 0]; F1=[0];
```

```
A2=[ 0 0 -1/L1 -1/L1;
     0 0 -1/L2 0;
     1/C 1/C -1/C*(1/R+1/Ron) -1/(C*Ron);
     1/C1 0 -1/(C1*Ron) -1/(C1*Ron) ];
B2=[1/L1; 0; 0; 0]; E2=[0 0 1 0]; F2=[0];
```

```
A3=[ 0 0 -1/L1 -1/L1;
     0 0 -1/L2 0;
     1/C 1/C -1/(R*C) 0;
     1/C1 0 0 0 ];
B3=[1/L1; 0; 0; 0]; E3=[0 0 1 0]; F3=[0];
```

```

Ns=3; % numarul starii topologice in stare stationara
n=4; % numarul variabilelor de stare
In=eye(n);
I=eye(n*(Ns+1)+Ns);

e1=l(1:4,:); % extrage pe X(T1)
e2=l(5:8,:); % extrage pe X(T2) % cele Ns+1 matrici extractoare ale
e3=l(9:12,:); % extrage pe X(T3) vectorilor de stare X(Ti), i=1,(Ns+1)
e4=l(13:16,:); % extrage pe X(T4)

e5=l(17,:); % extrage pe T1p
e6=l(18,:); % extrage pe T2p % cele Ns matrici extractoare
e7=l(19,:); % extrage pe T3p timpilor Tip=Ti+1-Ti
h=[-Ron -Ron 1 1]; % matricile extractoare ale componentelor din vectori ce satisfac relatia de stare (anulare)

% Se estimeaza cu modelul de semnal mare starea stationara aproximativa
kv=2*(C*C1/(C+C1))*R*fs;
Vo=Vg/(1-D)*sqrt(kv); VC=Vo; % marimile de stare
VC1=Vg; IL1=Vg/(1-D)^2*2*(C*C1/(C+C1))*fs; IL2=Vg/(1-D)*sqrt(2*(C*C1/(C+C1))*fs/R);

D1=sqrt(kv) ;
T1p=D1*Ts; T2p=(D-D1)*Ts; T3p=(1-D)*Ts; % timpii

Xmed=[IL1; IL2; VC; VC1]; T=[T1p; T2p; T3p]
xstart=[Xmed; Xmed; Xmed; Xmed; T];
xa=xstart; xc=zeros(n*(Ns+1)+Ns,1); flag=ones(1,n*(Ns+1)+Ns);
Xa=xstart; Xc=xc; err=[1e-9*ones((Ns+1)*n,1); 1e-12*ones(Ns,1)];
iter=0;
while ~(Xa==Xc)
    f1=(e2*xa)-(expm(A1*(e5*xa))*(e1*xa)+inv(A1)*(expm(A1*(e5*xa))-In)*B1*Vg);
    f2=(e3*xa)-(expm(A2*(e6*xa))*(e2*xa)+inv(A2)*(expm(A2*(e6*xa))-In)*B2*Vg); % cele Ns relatii de recurenta
    f3=(e4*xa)-(expm(A3*(e7*xa))*(e3*xa)+inv(A3)*(expm(A3*(e7*xa))-In)*B3*Vg);

    f4=(e4-e1)*xa; % relatia care egaleaza vectorul de stare de la inceputul si sfarsitul perioadei

    f5=h*(e2*xa);
    f6=(e5+e6+e7)*xa-Ts; % relatiile ce se refera la timpii
    f7=e7*xa-(1-D)*Ts;
    fxa=[f1;f2;f3;f4;f5;f6;f7];

    f1prim=e2-(A1*expm(A1*(e5*xa))*(e1*xa)*e5+expm(A1*(e5*xa))*e1+expm(A1*(e5*xa))*B1*Vg*e5);
    f2prim=e3-(A2*expm(A2*(e6*xa))*(e2*xa)*e6+expm(A2*(e6*xa))*e2+expm(A2*(e6*xa))*B2*Vg*e6);
    f3prim=e4-(A3*expm(A3*(e7*xa))*(e3*xa)*e7+expm(A3*(e7*xa))*e3+expm(A3*(e7*xa))*B3*Vg*e7);

    f4prim=e4-e1;

    f5prim=h*e2;
    f6prim=e5+e6+e7;
    f7prim=e7;
    fprimxa=[f1prim;f2prim;f3prim;f4prim;f5prim;f6prim;f7prim];

    xc=xa-inv(fprimxa)*fxa; iter=iter+1; % procesul iterativ Newton

    dif=abs(xc-xa);
    for i=1:n*(Ns+1)+Ns,
        if dif(i)>err(i) flag(i)=1;
        else flag(i)=0;
        end
    end
    m=sum(flag);
    if ~(m==0), xa=xc;
    else Xa=Xc;
    end
end
X=xc, iter.

```

## A 26 Calculul accelerat al stării staționare legând programul MATLAB de simulatorul PETS

Fișierul *fixpoint.m*

```

% Iteratia cu punct fix ("fixed point iteration") = simulare bruta pana ce croarea
% devine mai mica decat e0 sau pana se depaseste numarul maxim de simulari admise, Nsimmax.

% Tot ce este intre liniile cu diezi este specific convertorului analizat

```

```
% (nr. variabilelor de stare si dependenta tens. pe capacitati numai de
% tensiunile raportate fata de masa) si se va actualiza pt fiecare convertor
% analizat in parte
```

```
% Matricea F1 contine datele in binar ale fisierului analizat
% Matricea 'newline' contine indicii din F unde incep liniile
% Matricea 'new' contine indicii din F unde incep liniile cu caractere diferite de blank sau TAB
% Matricea 'state' contine numarul liniilor si indicii din F unde sunt definite variabilele de stare (L, I, C, sau c)
% Matricea 'location' contine indicii din F unde incep definirile conditiilor initiale (ic=anynumber)
% Aceasta matrice se actualizeaza in urma fiecarei simulari deoarece noile valori pt variabilele de
% stare nu sunt scrise cu acelasi numar de cifre zecimale.
```

```
% Declaratia 'ic=anynumber' trebuie scrisa cu litere mici, fara spatiu intre litere si
% fara comentariu anterior (*) deoarece programul nu face aceste verificari.
% In linia 'probe' se vor scrie variabilele de stare in ordinea in care au fost enunate
% elementele reactive in fisier.
```

```
clear all;
fclose('all');
clear % sterge variabile
Fer=1;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end
```

```
tic
```

```
% Deschide fisierul specificat, citeste datele binare in matricea F, apoi inchide fisierul
% si transforma in string datele binare citite.
file=input('Introduceti numele fisierului dorit a se analiza cu cale si nume (pana la 8 caractere), fara extensie: ','s');
A1=['FID1=fopen(""); A2=['.cir","r"];']; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
F1=fread(FID1);
fclose(FID1);
S1=setstr(F1);
```

```
% Gaseste indicii tuturor inceputurilor de linii
j=2; newline(1)=1;
for i=1:length(F1)
    if F1(i)==10; % 10 este codul binar pt carriage return
        newline(j)=i+1; j=j+1;
    else
        end
end
```

```
% Variabila NL contine numarul de linii din fisier
NL=length(newline);
```

```
% Gaseste indicii din F1 ai primelor elemente diferite de blank sau TAB
% din toate liniile, cu exceptia ultimei
for j=1:length(newline)-1
    k=0;
    while k<=(newline(j+1)-newline(j)-1)
        if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9) % 32 este codul binar pt. blank iar 9 pt TAB
            k=k+1;
        else
            break
        end
    end
    new(j)=newline(j)+k;
end
```

```
% Gaseste indexul din F al primului element diferit de blank sau TAB din ultima linie
j=length(newline);
k=0;
while k<=(length(F1)-newline(j)-1)
    if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9)
        k=k+1; % gaseste primele elemente diferite de blank
    else % sau TAB din ultima linie
        break
    end
    new(j)=newline(j)+k;
```

```
% Daca ultimul caracter din fisier este carriage return atunci ultima linie nu are sens a
% se analiza pt. a gasi primul caracter diferit de blank sau TAB
if F1(length(F1))==10 l=length(new)-1;
else l=length(new);
```

```

end

% Scrie in vectorul 'state' indicii din F1 unde apar: L,I,C,c
% si care apar ca prime elemente pe linii
k=1;
for j=1:l
    if (F1(new(j))==76)|(F1(new(j))==108)|(F1(new(j))==67)|(F1(new(j))==99) % 76 este codul pt L,
        state(k,1)=j; state(k,2)=new(j); k=k+1; % 108 pt I,
    else % 67 pt C % 99 pt c
    end
end
Nstates=length(state);

% Matricea de verificare
for k=1:Nstates
    S1(k)=setstr(F1(state(k,2)));
end
S1';

% Gaseste indeciii din F1 de unde incepe valoarea conditiei initiale
% (cauta numai in liniile care contin L,I,C sau c) si avertizeaza daca nu a gasit 'ic=anynumber' pe o linie cu variabila de stare
target1=['ic=anynumber']; % vectorul de comparare
for k=1:length(state(:,1))
    flag=0;
    for j=0:newline(state(k,1)+1)-state(k,2)-length(target1)

        for t=1:length(target1)
            block1(1,t)=S1(state(k,2)+j+(t-1)); % defineste stringul de comparare
        end

        comp1=(block1==target1);
        if sum(comp1)==12 location(k)=state(k,2)+j; flag=1;
        else
            j=j+1;
        end
    end
    if flag==0
        fprintf('Eroare ! Conditia initiala in forma "ic=anynumber" nu este definita in linia %i\n',state(k,1));
    else
    end
end

if flag==0 break
else
end

% Se efectueaza maxim Nsimmax simulari sau pana ce eroarea dintre vectorul de intrare
% in simulare si vectorul de stare de iesire din simulare, e, devine mai mica decat e0.
% Extrage din fisierul F1 indicii corespunzatori conditiilor initiale vechi,
% insereaza pe cei corespunzatori conditiilor initiale noi (deoarece valorile de actualizare
% a conditiilor initiale nu au acelasi numar de cifre)
% Conditii initiale au 9 cifre semnificative dupa virgula (vezi linia comentata cu ***)

% Defineste parametrii algoritmului gasirii starii stationare:
e0=1e-4; xmin=10*e0; % Sqreta si xm nu sunt necesari in iteratia cu punct fix

#####
x0=[0,0,0,0]; % !!! dimensiunea depinde de numarul variabilelor de stare din convertor
#####

shifta=9*ones(1,Nstates); % initializeaza matricea de shiftarea anterioara care are dimensiunea 9
% pt. fiecare element reactiv (stringul 'anynumber' are 9 caractere
Nsimmax=3000; % numarul maxim de perioade simulate
countsim=0; % contorul numarului de simulari
%Ts=50e-6; % necesara numai pt a actualiza timpul curent de simulare de care e nevoie in plotarea evolutiei regimului tranzitoriu

for k=1:length(x0)
    b=abs(sprintf('% 9f,x0(k))); % ***
    shift(k)=length(b);
    F1prim=F1(1:location(k)+2,1); % scrie valorile lui x0 in fisier la conditiile initiale
    F1second=F1(location(k)+3+shifta(k):length(F1),1);
    F1=[F1prim;b';F1second];
    for i=k+1:length(location)
        location(i)=location(i)+shift(k)-shifta(k);
    end
end

```

```

    shifta(k)=shift(k);
end

A1=['FID1=fopen(""); A2=['.cir","w");]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);

% Lanseaza o simulare pt a obtine matricea 'index' in care sunt corelate
% ordinea din linia 'probe' cu cea din fisierul *.txt (se face o singura data)
% si Ftx0, necesara la calculul erorii care se calculeaza si ea
A1=['dos("pets "); A2=[""];]; B=[A1,file,A2]; eval(B); %dos('pets buck');
countsim=countsim+1;
statdet; % m-file (pt. PETS) pentru detectarea ordinii de scriere in fisierul *.txt
    % a variabilelor si corelarea lor cu cele definite in comanda 'probe'.

A1=['load '); A2=['.txt;]; B=[A1,file,A2]; eval(B); % load buck.txt;
A1=['varpets=']; A2=['(:,2:length("); A3=['(1,:);)]; B=[A1,file,A2,file,A3]; eval(B); % varpets=buck(:,2:length(buck(1,:));
for k=1:length(varpets(1,:))
    var(:,k)=varpets(:,index(k));
end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie difera de la schema la schema. Daca capacitatile au o borna la masa linia nu mai e necesara

#####
var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)];
#####

Ftx0=var(length(var,:));
%A1=['time=']; A2=['(:,1);]; B=[A1,file,A2]; eval(B); %time=file(:,1)
%IL=var(:,1); vo=var(:,2); % variabilele ce se vor plota pt tot regimul tranzitoriu
clear var

for i=1:length(x0)
    er(i)=abs(x0(i)-Ftx0(i))/max(x0(i),xmin); % calculul erorilor partiale
end
e=max(er);

if e>e0
    x0=Ftx0; % reactualizeaza vectorul de intrare in bucla cu simularile
else
end

while (countsim<Nsimmax)&(e>e0)
    for k=1:length(x0)
        b=abs(sprintf('%9f,x0(k))); % ***
        shifta(k)=length(b);
        F1prim=F1(1+location(k)+2,1);
        F1secund=F1(location(k)+3+shifta(k):length(F1),1);
        F1=[F1prim;b';F1secund];
        for i=k+1:length(location)
            location(i)=location(i)+shift(k)-shifta(k);
        end
        shifta(k)=shift(k);
    end
    A1=['FID1=fopen(""); A2=['.cir","w");]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
    fwrite(FID1,F1);
    fclose(FID1);

% Lanseaza simularea din bucla while
A1=['dos("pets "); A2=[""];]; B=[A1,file,A2]; eval(B); %dos('pets buck');
countsim=countsim+1;
A1=['load '); A2=['.txt;]; B=[A1,file,A2]; eval(B); %load file.txt;
A1=['varpets=']; A2=['(:,2:length("); A3=['(1,:);)]; B=[A1,file,A2,file,A3]; eval(B);
% varpets=buck(:,2:length(buck(1,:))

    for k=1:length(varpets(1,:))
        var(:,k)=varpets(:,index(k));
    end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie difera de la schema la schema

#####

```



```

var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)];
%#####

Ftx0=var(length(var),:);
% A1=['t=']; A2=['(:,1)']; B=[A1,file,A2]; eval(B); %t=file(:,1)
% for i=1:length(t);
% time1(i,1)=(countsim-1)*Ts+t(i); % se actualizeaza vectorul de timp curent de simulare
% iL1(i,1)=var(i,1); vo1(i,1)=var(i,2); % si variabilele
% end
% time=[time;time1]; iL=[iL;,iL1]; vo=[vo; vo1];
clear var; % pt ca de la o simulare a alta nr. de elemente pe coloana difera, PETS-ul variind nr de puncte in jurul comutarii

for i=1:length(x0)
er(i)=abs(x0(i)-Ftx0(i))/max(x0(i),xmin); % calculul erorilor pariale
end
e=max(er);
x0=Ftx0;

end

% Scrie din nou 'anynumber' in conditiile initiale pentru ca fisierul sa fie pregatit
% pentru o noua rulare
b=['anynumber'];
for k=1:length(x0)
shift(k)=length(b);
F1prim=F1(1:location(k)+2,1);
F1secund=F1(location(k)+3+shifta(k):length(F1),1);
F1=[F1prim;b',F1secund];
for i=k+1:length(location)
location(i)=location(i)+shift(k)-shifta(k);
end
shifta(k)=shift(k);
end
A1=['FID1=fopen(""); A2=['.cir","w"];]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);

if countsim>=Nsimmax
fprintf('Numarul maxim de simulari depasit !\n');
break
else
end

Xstat=Ftx0', % afisaza vectorul de stare aferent starii stationare obtinute

% Extrage undele marimilor de stare pe o perioada pt starea stationara
A1=['load ']; A2=['.txt']; B=[A1,file,A2]; eval(B); %load file.txt;
A1=['t=']; A2=['(:,1)']; B=[A1,file,A2]; eval(B); % t=file(:,1); timpul
A1=['varpets=']; A2=['(:,2:length(')); A3=['(1,:)');]; B=[A1,file,A2,file,A3]; eval(B);
% varpets=buck(:,2:length(buck(1,:)))
for k=1:length(varpets(1,:))
var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie va diferi de la schema la schema

%#####
xsteady={var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)}; % corelate din schema cu ordinea din .probe
%#####

fprintf('Starea stationara atinsa dupa %i simulari',countsim);

% Ploteaza starea stationara
l=length(xsteady(1,:));
for i=1:l
subplot(1,1,i); plot(t,xsteady(:,i),'k');
end

toc

```

## Fișierul finitdif.m

```

% Iteratia prin metoda de tip Newton cu diferente finite ("finite-difference quasi-Newton method")
% se itereaza pana ce eroarea e devine mai mica decat e0
% sau pana se depaseste numarul maxim de simulari admise, Nsimmax.

% Tot ce este intre liniile cu diezi este specific convertorului analizat
% (nr. variabilelor de stare si dependenta tens. pe capacitati numai de
% tensiunile raportate fata de masa) si se va actualiaza pt fiecare convertor
% analizat in parte

% Matricea F1 contine datele in binar ale fisierului analizat
% Matricea 'newline' contine indicii din F unde incep liniile
% Matricea 'new' contine indicii din F unde incep liniile cu caractere diferite de blank sau TAB
% Matricea 'state' contine numarul liniilor si indicii din F unde sunt definite variabilele de stare (L, I, C, sau c)
% Matricea 'location' contine indicii din F unde incep definirile conditiilor initiale (ic=anynumber)
% Aceasta matrice se actualizeaza in urma fiecarei simulari deoarece noile valori pt variabilele de
% stare nu sunt scrise cu acelasi numar de cifre zecimale.

% Declaratia 'ic=anynumber' trebuie scrisa cu litere mici, fara spatiu intre litere si
% fara comentariu anterior (*) deoarece programul nu face aceste verificari.
% In linia 'probe' se vor scrie variabilele de stare in ordinea in care au fost enunate
% elementele reactive in fisier.

clear all;
fclose('all');
clear % sterge variabile
Fer=1;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

tic

% Deschide fisierul specificat, citeste datele binare in matricea F, apoi inchide fisierul
% si transforma in string datele binare citite.
file=input(' Introduceti numele fisierului dorit a se analiza cu cale si nume (pana la 8 caractere), fara extensie: ','s');
A1=['FID1=fopen("]; A2=['.cir','.m'); B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
F1=fread(FID1);
fclose(FID1);
S1=setstr(F1);

% Gaseste indicii tuturor inceputurilor de linii
j=2; newline(1)=1;
for i=1:length(F1)
    if F1(i)==10; % 10 este codul binar pt carriage return
        newline(j)=i+1; j=j+1;
    else
        end
end
% Variabila NL contine numarul de linii din fisier
NL=length(newline);

% Gaseste indicii din F1 ai primelor elemente diferite de blank sau TAB
% din toate liniile, cu exceptia ultimei
for j=1:length(newline)-1
    k=0;
    while k<=(newline(j+1)-newline(j)-1)
        if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9) % 32 este codul binar pt. blank iar 9 pt TAB
            k=k+1;
        else
            break
        end
    end
    new(j)=newline(j)+k;
end

% Gaseste indexul din F al primului element diferit de blank sau TAB din ultima linie
j=length(newline);
k=0;
while k<=(length(F1)-newline(j)-1)
    if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9)
        k=k+1; % gaseste primele elemente diferite de blank
    else % sau TAB din ultima linie
        break
    end
end

```

```

end
new(j)=newline(j)+k;

% Daca ultimul caracter din fisier este carriage return atunci ultima linie nu are sens a
% se analiza pt. a gasi primul caracter diferit de blanc sau TAB
if F1(length(F1))==10 l=length(new)-1;
else l=length(new);
end

% Scrie in vectorul 'state' indicii din F1 unde apar: L,I,C,c
% si care apar ca prime elemente pe linii
k=1;
for j=1:l
    if (F1(new(j))==76)|(F1(new(j))==108)|(F1(new(j))==67)|(F1(new(j))==99) % 76 este codul pt L,
        state(k,1)=j; state(k,2)=new(j); k=k+1; % 108 pt l,
        else % 67 pt C % 99 pt c
        end
    end
Nstates=length(state);

% Matricea de verificare
for k=1:Nstates
S1(k)=setstr(F1(state(k,2)));
end
S1';

% Gaseste indeciii din F1 de unde incepe valoarea conditiei initiale
% (cauta numai in liniile care contin L,I,C sau c) si avertizeaza daca nu a gasit 'ic=anynumber' pe o linie cu variabila de stare
target1=['ic=anynumber']; % vectorul de comparare
for k=1:length(state(:,1))
flag=0;
for j=0:newline(state(k,1)+1)-state(k,2)-length(target1)

    for t=1:length(target1)
        block1(1,t)=S1(state(k,2)+j+(t-1)); % defineste stringul de comparare
    end

    comp1=(block1==target1);
    if sum(comp1)==12 location(k)=state(k,2)+j; flag=1;
    else
        j=j+1;
    end
end
if flag==0
fprintf('Eroare ! Conditia initiala in forma "ic=anynumber" nu este definita in linia %i\n',state(k,1));
else
end

end

if flag==0 break
else
end

% Se efectueaza maxim Nsimmax simulari sau pana ce eroarea dintre vectorul de intrare
% in simulare si vectorul de stare de iesire din simulare, e, devine mai mica decat e0.
% Extrage din fisierul F1 indicii corespunzatori conditiilor initiale vechi,
% insereaza pe cei corespunzatori conditiilor initiale noi (deoarece valorile de actualizare
% a conditiilor initiale nu au acelasi numar de cifre)
% Conditiiile initiale au 9 cifre semnificative dupa virgula (vezi linia comentata cu ***)

% Defineste parametrii algoritmului gasirii accelerate a starii stationare:
e0=1e-4; Sqreta=sqrt(e0); xmin=10*e0; xm=10*e0;

#####
x0=[0;0;0;0]; % !!! dimensiunea depinde de numarul variabilelor de stare din convertor
#####

I=eye(length(x0));
shifta=9*ones(Nstates,1); % initializeaza matricea de shiftare anterioara care are dimensiunea 9
% pt. fiecare element reactiv (stringul 'anynumber' are 9 caractere
%eps=1e-3*ones(1,Nstates); % initializeaza vectorul eroare
Nsimmax=3000; % numarul maxim de perioade simulate
countsim=0; % contorul numarului de simulari
iter=0; % contorul de iteratii

```

```

for k=1:length(x0)
    b=abs(sprintf('%9f,x0(k)); % ***
    shift(k)=length(b);
    F1prim=F1(1:location(k)+2,1); % scrie valorile lui x0 in fisier la conditiile initiale
    F1secund=F1(location(k)+3+shifta(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(location)
        location(i)=location(i)+shift(k)-shifta(k);
    end
    shifta(k)=shift(k);
end

A1=['FID1=fopen(""); A2=['.cir","w");]; B={A1,file,A2}; eval(B); %FID1=fopen('buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);

% Lanseaza o simulare pt a obtine matricea 'index' in care sunt corelate
% ordinea din linia 'probe' cu cea din fisierul *.txt (se face o singura data)
% si Ftx0, necesara la calculul erorii care se calculeaza si ea
A1=['dos\"pets ']; A2=[""]; B={A1,file,A2}; eval(B); %dos('pets buck');
countsim=counts+1;
statdet; % m-file (pt. PETS) pentru detectarea ordinii de scriere in fisierul *.txt
    % a variabilelor si corelarea lor cu cele definite in comanda 'probe'.

A1=['load ']; A2=['.txt']; B={A1,file,A2}; eval(B); % load buck.txt;
A1=['varpets=']; A2=['(:,2:length()); A3=['(1:);']; B={A1,file,A2,file,A3}; eval(B); % varpets=buck(:,2:length(buck(1,:)));
for k=1:length(varpets(1,:))
    var(:,k)=varpets(:,index(k));
end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie difera de la schema la schema
%#####
var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)];
%#####

Ftx0=var(length(var,:)); % se va inlocui cu linia '*.tran' din *.cir pt alte variabile de stare
absvar=abs(var); absmax=max(absvar); % determina maximele variabilelor din ultima simulare
clear var

for i=1:length(x0)
    er(i)=abs(x0(i)-Ftx0(i))/max(x0(i),xmin); % calculul erorilor partiale
end
e=max(er);

while (counts+1<Nsimmax)&(e>c0)
    %#####
    % intre aceste linii cu stelute se calculeaza Jacobianul in x0 prin metoda diferentelor finite

for i=1:length(x0)
    dx(i,1)=Sqrta*max(absmax(i),xm);
    x1=x0+1(:,i)*dx(i);

    % simuleaza pt a obtine Ft(x0+ui*dxi)=Ft(x1)
    for k=1:length(x1)
        b=abs(sprintf('%9f,x1(k)); % ***
        shift(k)=length(b);
        F1prim=F1(1:location(k)+2,1);
        F1secund=F1(location(k)+3+shifta(k):length(F1),1);
        F1=[F1prim;b';F1secund];
        for p=k+1:length(location)
            location(p)=location(p)+shift(k)-shifta(k);
        end
        shifta(k)=shift(k);
    end
end

A1=['FID1=fopen(""); A2=['.cir","w");]; B={A1,file,A2}; eval(B); %FID1=fopen('buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);
A1=['dos\"pets ']; A2=[""]; B={A1,file,A2}; eval(B); %dos('pets buck');
counts+1=counts+1;
A1=['load ']; A2=['.txt']; B={A1,file,A2}; eval(B); %load file.txt;
A1=['varpets=']; A2=['(:,2:length()); A3=['(1:);']; B={A1,file,A2,file,A3}; eval(B);
% varpets=buck(:,2:length(buck(1,:)))
for k=1:length(varpets(1,:))

```

```

var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie va diferi de la schema la schema
#####
var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)]; % corelate din schema cu ordinea din .probe
#####

Ftx1=var(length(var,:));
clear var
j=[(i-1)/dx(i).*(Ftx1-Ftx0); % calculeaza coloana curenta din J(x0)
Jx0=[Jx0 j];
end

%*****

% Se efectueaza iteratia
x1=x0-inv(Jx0)*(x0-Ftx0);
iter=iter+1;
Jx0=[]; % sterge Jacobianul pentru ca acesta sa nu tot creasca de la o iteratie la alta
x0=x1; % actualizeaza vectorul cu care se intra in simulare si dupa care eventual se va
% calcula viitorul Jacobian

for k=1:length(x0)
b=abs(sprintf('%9f,x0(k))); % ***
shift(k)=length(b);
F1prim=F1(1:location(k)+2,1);
F1secund=F1(location(k)+3+shifta(k):length(F1),1);
F1=[F1prim;b';F1secund];
for i=k+1:length(location)
location(i)=location(i)+shift(k)-shifta(k);
end
shifta(k)=shift(k);
end
A1=[FID1=fopen(""); A2=['.cir","w");]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);

% Lanseaza simularea pentru noua solutie gasita prin iteratie
A1=['dos("pets '); A2=["");]; B=[A1,file,A2]; eval(B); %dos('pets buck');
countsim=countsim+1;
A1=['load '); A2=['.txt.']; B=[A1,file,A2]; eval(B); %load buck.txt;
A1=["varpets="]; A2=[(';',2:length('); A3=[('1.,:));]; B=[A1,file,A2,file,A3]; eval(B);
% varpets=buck(:,2:length(buck(1,:)))

for k=1:length(varpets(1,:))
var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie va diferi de la schema la schema
#####
var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)];
#####

Ftx0=var(length(var,:));
absvar=abs(var); absmax=max(absvar); % determina maximele variabilelor din ultima simulare
clear var; % pt ca de la o simulare a alta nr. de elemente pe coloana difera,
%PETS-ul variind nr de puncte in jurul comutarii

for i=1:length(x0)
er(i)=abs(x0(i)-Ftx0(i))/max(x0(i),xmin); % calculul erorilor partiale
end
e=max(er);

end

% Scrie din nou 'anynumber' in conditiile initiale pentru ca fisierul sa fie pregatit
% pentru o noua rulare
b=['anynumber'];
for k=1:length(x0)
shift(k)=length(b);
F1prim=F1(1:location(k)+2,1);

```

```

F1secund=F1(location(k)+3+shifta(k):length(F1),1);
F1=[F1prim;b';F1secund];
for i=k+1:length(location)
    location(i)=location(i)+shift(k)-shifta(k);
end
shifta(k)=shift(k);
end
A1=['FID1=fopen(""); A2=['.cir',"w"];]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);

if countsim>=Nsimmax
fprintf('Numarul maxim de simulari depasit !\n');
break
else
end

Xstat=Ftx0, % afisaza vectorul de stare aferent starii stationare obtinute

% Extrage unelele marimilor de star pe o perioada pt starea stationara
A1=['load ']; A2=['.txt']; B=[A1,file,A2]; eval(B); %load file.txt;
A1=['t=']; A2=['(:,1)']; B=[A1,file,A2]; eval(B); % t=file(:,1); timpul
A1=['varpets=']; A2=['(:,2:length(')); A3=['(1,:)');]; B=[A1,file,A2,file,A3]; eval(B);
% varpets=buck(:,2:length(buck(1,:)))
for k=1:length(varpets(1,:))
    var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie va diferi de la schema la schema
#####
xsteady=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)]; % corelate din schema cu ordinea din .probe
#####

fprintf('Starea stationara atinsa dupa %i iteratii si %i simulari',iter,countsim);

% Ploteaza starea stationara
l=length(xsteady(1,:));
for i=1:l
subplot(l,1,i); plot(t,xsteady(:,i),'k');
end

toc

Fișierul broyden.m

% Iteratia prin metoda de tip Newton cu actualizare Broyden ("Broyden-update quasi-Newton method")
% se itereaza pana ce eroarea e devine mai mica decat e0
% sau pana se depaseste numarul maxim de simulari admise, Nsimmax.

% Tot ce este intre liniile cu diezi este specific convertorului analizat
% (nr. variabilelor de stare și dependenta tens. pe capacitati numai de
% tensiunile raportate fata de masa) si se va actualizeaza pt fiecare convertor
% analizat in parte

% Matricea F1 contine datele in binar ale fisierului analizat
% Matricea 'newline' contine indicii din F unde incep liniile
% Matricea 'new' contine indicii din F unde incep liniile cu caractere diferite de blank sau TAB
% Matricea 'state' contine numarul liniilor si indicii din F unde sunt definite variabilele de stare (L,I, C, sau c)
% Matricea 'location' contine indicii din F unde incep definirile conditiilor initiale (ic=anynumber)
% Aceasta matrice se actualizeaza in urma fiecarei simulari deoarece noile valori pt variabilele de
% stare nu sunt scrise cu acelasi numar de cifre zecimale.

% Declaratia 'ic=anynumber' trebuie scrisa cu litere mici, fara spatiu intre litere si
% fara comentariu anterior (*) deoarece programul nu face aceste verificari.
% In linia 'probe' se vor scrie variabilele de stare in ordinea in care au fost enunate
% elementele reactive in fisier.

clear all;
fclose('all');
clear % sterge variabile
Fer=1;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

```

tic

```

% Deschide fisierul specificat, citeste datele binare in matricea F, apoi inchide fisierul
% si transforma in string datele binare citite.
file=input(' Introduceti numele fisierului dorit a se analiza cu care si nume (pana la 8 caractere), fara extensie: ','s');
A1=['FID1=fopen("]; A2=['.cir","r");]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
F1=fread(FID1);
fclose(FID1);
S1=setstr(F1);

% Gaseste indicii tuturor inceputurilor de linii
j=2; newline(1)=1;
for i=1:length(F1)
    if F1(i)=='\0'; % 10 este codul binar pt carriage return
        newline(j)=i+1; j=j+1;
    else
        end
end
% Variabila NL contine numarul de linii din fisier
NL=length(newline);

% Gaseste indicii din F1 ai primelor elemente diferite de blank sau TAB
% din toate liniile, cu exceptia ultimei
for j=1:length(newline)-1
    k=0;
    while k<=(newline(j+1)-newline(j))-1
        if (F1(newline(j)+k)=='32')|(F1(newline(j)+k)=='9') % 32 este codul binar pt. blank iar 9 pt TAB
            k=k+1;
        else
            break
        end
    end
    new(j)=newline(j)+k;
end

% Gaseste indexul din F al primului element diferit de blank sau TAB din ultima linie
j=length(newline);
k=0;
while k<=(length(F1)-newline(j))-1
    if (F1(newline(j)+k)=='32')|(F1(newline(j)+k)=='9')
        k=k+1; % gaseste primele elemente diferite de blank
    else % sau TAB din ultima linie
        break
    end
end
new(j)=newline(j)+k;

% Daca ultimul caracter din fisier este carriage return atunci ultima linie nu are sens a
% se analiza pt. a gasi primul caracter diferit de blank sau TAB
if F1(length(F1))=='\0' l=length(new)-1;
else l=length(new);
end

% Scrie in vectorul 'state' indicii din F1 unde apar: L,I,C,c
% si care apar ca prime elemente pe linii
k=1;
for j=1:l
    if (F1(new(j))=='76')|(F1(new(j))=='108')|(F1(new(j))=='67')|(F1(new(j))=='99') % 76 este codul pt L,
        state(k,1)=j; state(k,2)=new(j); k=k+1; % 108 pt I,
    else % 67 pt C
        % 99 pt c
    end
end
Nstates=length(state);

% Matricea de verificare
for k=1:Nstates
    S1(k)=setstr(F1(state(k,2)));
end
S1';

% Gaseste indecsii din F1 de unde incepe valoarea conditiei initiale
% (cauta numai in liniile care contin L,I,C sau c) si avertizeaza daca nu a gasit 'ic=anynumber' pe o linie cu variabila de stare
target1='ic=anynumber'; % vectorul de comparare
for k=1:length(state(:,1))
    flag=0;

    for j=0:newline(state(k,1)+1)-state(k,2)-length(target1)

```

```

for t=1:length(target1)
    block1(1,t)=S1(state(k,2)+j+(t-1)); % defineste stringul de comparare
end

comp1=(block1==target1);
if sum(comp1)==12 location(k)=state(k,2)+j; flag=1;
else
    j=j+1;
end
end
if flag==0
fprintf('Eroare ! Conditia initiala in forma "ic=anynumber" nu este definita in linia %i\n',state(k,1));
else
end

end

if flag==0 break
else
end

% Se efectueaza maxim Nsimmax simulari sau pana ce eroarea dintre vectorul de intrare
% in simulare si vectorul de stare de iesire din simulare, e, devine mai mica decat e0.
% Extrage din fisierul F1 indicii corespunzatori conditiilor initiale vechi,
% insereaza pe cei corespunzatori conditiilor initiale noi (deoarece valorile de actualizare
% a conditiilor initiale nu au acelasi numar de cifre)
% Conditiiile initiale au 9 cifre semnificative dupa virgula (vezi linia comentata cu ***)

% Defineste parametrii algoritmului gasirii starii stationare:
e0=1e-4; Sqreta=sqrt(e0); xmin=10*e0; xm=10*e0;

#####
x0=[0;0;0;0]; % !!! dimensiunea depinde de numarul variabilelor de stare din convertor
#####

I=eye(length(x0));
shifta=9*ones(Nstates,1); % initializeaza matricea de shiftare anterioara care are dimensiunea 9
% pt. fiecare element reactiv (stringul 'anynumber' are 9 caractere
%eps=1e-3*ones(1,Nstates); % initializeaza vectorul eroare
Nsimmax=3000; % numarul maxim de perioade simulate
countsim=0; % contorul numarului de simulari
iter=0; % contorul de iteratii

for k=1:length(x0)
    b=abs(sprintf('%9f',x0(k))); % ***
    shift(k)=length(b);
    F1prim=F1(1:location(k)+2,1); % scrie valorile lui x0 in fisier la conditiile initiale
    F1secund=F1(location(k)+3+shifta(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(location)
        location(i)=location(i)+shift(k)-shifta(k);
    end
    shifta(k)=shift(k);
end

A1=['FID1=fopen(''); A2=['.cir','w');]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);

% Lanseaza o simulare pt a obtine matricea 'index' in care sunt corelate
% ordinea din linia 'probe' cu cea din fisierul *.txt (se face o singura data)
% si si Ftx0, necesara la calculul erorii care se calculeaza si ea
A1=['dos("pets "); A2=[""];]; B=[A1,file,A2]; eval(B); %dos('pets buck');
countsim=countsim+1;
statdet; % m-file (pt. PETS) pentru detectarea ordinii de scriere in fisierul *.txt
% a variabilelor si corelarea lor cu cele definite in comanda 'probe'.

A1=['load ']; A2=['.txt;']; B=[A1,file,A2]; eval(B); % load buck.txt;
A1=['varpets=']; A2=['(:,2:length('); A3=['(1,:));']; B=[A1,file,A2,file,A3]; eval(B); % varpets=buck(:,2:length(buck(1,:)));
for k=1:length(varpets(1,:))
    var(:,k)=varpets(:,index(k));
end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe

```



```

% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie difera de la schema la schema

#####
var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)];
#####

Ftx0=var(length(var,:)); % se va inlocui cu linia '*.tran' din *.cir pt alte variabile de stare
absvar=abs(var); absmax=max(absvar); % determina maximele variabilelor din ultima simulare
clear var

for i=1:length(x0)
    er(i)=abs(x0(i)-Ftx0(i))/max(x0(i),xmin); % calculul erorilor pariale
end
e=max(er);

% Se testeaza conductia de stare stationara si daca nu e stare stationara se calculeaza
% Jacobianul in x0 prin metoda diferentelor finite, apoi x1.
if e>e0
    %*****
    % intre aceste linii cu stelute se calculeaza o singura data Jacobianul in
    % punctul solutie initiala prin metoda diferentelor finite

    for i=1:length(x0)
        dx(i,1)=Sqrta*max(absmax(i),xm);
        x1=x0+I(:,i).*dx(i);

        % simuleaza pt a obtine Ft(x0+ui*dxi)=Ft(x1)
        for k=1:length(x1)
            b=abs(sprintf('%9f',x1(k))); % ***
            shift(k)=length(b);
            F1prim=F1(1:location(k)+2,1);
            F1secund=F1(location(k)+3+shifta(k):length(F1),1);
            F1=[F1prim;b';F1secund];
            for p=k+1:length(location)
                location(p)=location(p)+shift(k)-shifta(k);
            end
            shifta(k)=shift(k);
        end

        A1=['FID1=fopen(""); A2=['.cir',"w"];']; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
        fwrite(FID1,F1);
        fclose(FID1);
        A1=['dos("pets '); A2=[""];']; B=[A1,file,A2]; eval(B); %dos('pets buck');
        countsim=counts+1;
        A1=['load '); A2=['.txt;']; B=[A1,file,A2]; eval(B); %load file.txt;
        A1=['varpets=']; A2=['(:,2:length('); A3=['(1,:));']; B=[A1,file,A2,file,A3]; eval(B);
        % varpets=buck(:,2:length(buck(1,:)))
        for k=1:length(varpets(1,:))
            var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
        end

        % Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
        % (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
        % aceasta linie va diferi de la schema la schema

        #####
        var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)]; % corelate din schema cu ordinea din .probe
        #####

        Ftx1=var(length(var,:));
        clear var
        jcol=(i-1)/dx(i).*(Ftx1-Ftx0); % calculeaza coloana curenta din J(x0)
        Jx0=[Jx0 jcol];
        end

        %*****
        x1=x0-inv(Jx0)*(x0-Ftx0);
        iter=iter+1;
    else
    end

while (countsim<Nsimmax)&(e>e0)
    %*****
    % intre aceste linii cu stelute se calculeaza Jacobianul prin actualizarea Broyden

```

```

% simuleaza pt a obtine Ft(x1)
for k=1:length(x1)
    b=abs(sprintf('%9f',x1(k))); % ***
    shift(k)=length(b);
    F1prim=F1(1:location(k)+2,1);
    F1second=F1(location(k)+3+shifta(k):length(F1),1);
    F1=[F1prim;b';F1second];
    for p=k+1:length(location)
        location(p)=location(p)+shift(k)-shifta(k);
    end
    shifta(k)=shift(k);
end

A1=['FID1=fopen(""); A2=['.cir","w");]; B=[A1,file,A2]; eval(B); %FID1=fopen("buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);
A1=['dos("pets '); A2=[""];]; B=[A1,file,A2]; eval(B); %dos("pets buck");
countsim=countsim+1;
A1=["load "; A2=[".txt;"]; B=[A1,file,A2]; eval(B); %load file.txt;
A1=["varpets="]; A2=['(:,2:length(', A3=['(1,:));');]; B=[A1,file,A2,file,A3]; eval(B);
% varpets=buck(:,2:length(buck(1,:)))
for k=1:length(varpets(1,:))
    var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie va diferi de la schema la schema

%#####
var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)]; % corelate din schema cu ordinea din .probe
%#####

Ftx1=var(length(var),:);
clear var

for i=1:length(x1)
    er(i)=abs(x1(i)-Ftx1(i))/max(x1(i),xmin); % calculul erorilor partiale
end
e=max(er); % eroarea totala

if e>e0
    Jx1=Jx0+(x1-Ftx1)*(x1-x0)/sum((x1-x0).^2); % actualizarea Broyden
    x0=x1; Jx0=Jx1; % actualizeaza pt viitorul calcul de Jacobian
else
    break % e pus pt a sarge bucla while si a nu mai contoriza iteratia de la
end % sfarsitul buclei care se face chiar daca conditia de eroare s-a indeplinit.
% Aceasta pt ca testul in bucla while se face la inceputul buclei
%*****

% Se efectueaza iteratia
x1=x1-inv(Jx1)*(x1-Ftx1);
iter=iter+1;

end
Jx0=[];

% Scrie din nou 'anynumber' in conditiile initiale pentru ca fisierul sa fie pregatit
% pentru o noua rulare
b=['anynumber'];
for k=1:length(x0)
    shift(k)=length(b);
    F1prim=F1(1:location(k)+2,1);
    F1second=F1(location(k)+3+shifta(k):length(F1),1);
    F1=[F1prim;b';F1second];
    for i=k+1:length(location)
        location(i)=location(i)+shift(k)-shifta(k);
    end
    shifta(k)=shift(k);
end
A1=['FID1=fopen(""); A2=['.cir","w");]; B=[A1,file,A2]; eval(B); %FID1=fopen("buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);

if countsim>=Nsimmax

```

```
fprintf('Numarul maxim de similari depasit \n');
break
else
end
```

```
Xstat=Ftx1, % afisaza vectorul de stare aferent starii stationare obtinute
```

```
% Extrage undele marimilor de stare pe o perioada pt starea stationara
A1='load '; A2=['.txt;']; B=[A1,file,A2]; eval(B); %load file.txt;
A1=['t=']; A2=['(:,1);']; B=[A1,file,A2]; eval(B); % t=file(:,1); timpul
A1=['varpets=']; A2=['(:,2:length(')); A3=['(1,:);']; B=[A1,file,A2,file,A3]; eval(B);
% varpets=buck(:,2:length(buck(1,:)))
for k=1:length(varpets(1,:))
    var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
end
```

```
% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie va diferi de la schema la schema
```

```
#####
xsteady=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)]; % corelate din schema cu ordinea din .probe
#####
```

```
fprintf('Starea stationara atinsa dupa %i iteratii si %i similari',iter,countsim);
```

```
% Ploteaza starea stationara
l=length(xsteady(1,:));
for i=1:l
    subplot(l,1,i); plot(t,xsteady(:,i),'k');
end
```

```
toc
```

#### Fişierul statdet.m

```
% Fisier pentru detectarea ordinii de scriere in fisierul *.txt a variabilelor (pt. PETS)
% si corelarea lor cu cele definite in comanda 'probe'.
```

```
clear index
% Deschide fisierul *.nam
A1=['FID2=fopen(""); A2=['.nam","r");']; B=[A1,file,A2]; eval(B); %FID2=fopen('buck.nam','r');
G2=fread(FID2); fclose(FID2); S2=setstr(G2);
```

```
% Gaseste locatia de inceput (poznam) de unde sunt enumerate variabilele in fisierul *.nam'.
```

```
target2=['output variables:'];
flag=0;
for j=1:length(S2)-length(target2)
    for t=1:length(target2)
        b2(t)=S2(j+t-1); % defineste stringul curent de comparare
    end
    comp2=(b2==target2);
    if sum(comp2)==length(target2) poznam=j+length(target2)+4; flag=1; break
    else
        j=j+1;
    end
end
```

```
if flag==0
    fprintf('Eroare ! Stringul "output variables:" nu a fost gasit!');
else
end
% Verificare
%S2(poznam), keyboard
```

```
% Gaseste locatia de inceput (pozprobe) de unde sunt enumerate variabilele din comanda 'probe' in fisierul *.cir
A1=['FID3=fopen(""); A2=['.cir","r");']; B=[A1,file,A2]; eval(B); %FID3=fopen('buck.cir','r');
G3=fread(FID3); fclose(FID3); S3=setstr(G3);
```

```
target3=['.probe'];
flag=0;
for j=1:length(S3)-length(target3)
    for t=1:length(target3)
        b3(t)=S3(j+t-1); % defineste stringul curent de comparare
    end
```

```

comp3=(b3==target3);
if sum(comp3)==length(target3) pozprobe=j+length(target3); flag=1;
else
j=j+1;
end
end

if flag==0
fprintf('Eroare ! Stringul ".probe" nu a fost gasit');
else
end

%Verificare
%S3(pozprobe), keyboard

% In matricea varpets se retin din fisierul *.txt numai coloanele cu variabile, nu si coloana cu timpul
A1=['load']; A2=['.txt']; B=[A1,file,A2]; eval(B); %load buck.txt;
A1=['varpets=']; A2=[(';',2:length('); A3=['(1,);)]; B=[A1,file,A2,file,A3]; eval(B);
%varpets=buck(:,2:length(buck(1,)));

k=1; pvar=poznam; % initializeaza variabila curenta de cautare (pvar) in *.nam
while ~(pvar>=length(S2))

clear b2; i=1;
while ~(G2(pvar)==13) % determina stringurile ce dau variabilele scrise in
b2(i)=S2(pvar); % fisierul *.txt (dar in alta ordine decat cea din
i=i+1; pvar=pvar+1; % linia probe)
end

j=1; poz=pozprobe; clear b3; % initializeaza variabila curenta de cautare in *.cir pe linia probe
while ~(G3(poz)==13)&~(G3(poz)==10)
m=1;
while ~(G3(poz)==32)&~(G3(poz)==9)&~(G3(poz)==13)&~(G3(poz)==10)
b3(m)=S3(poz); poz=poz+1; m=m+1; % citeste stringurile din linia
end
% probe dintre 2 blankuri sau 2
% TAB-uri succesive
if length(b2)==length(b3)
comp23=(b2==b3); % compara stringul curent din *.nam cu stringul curent din
if (sum(comp23)==length(b2)) % linia de probe. In caz de egalitate formeaza vectorul
index(j)=k; % index care contine ca si elemente numerele de ordine ale
clear b3; % variabilelor enumerate in *.txt corespunzatoare ordinii definirii
j=j+1; poz=poz+1; % variabilelor in *. in linia 'probe'. Adica daca de exemplu
else % index(3)=7 aceasta inseamna ca a treia variabila definita
clear b3; % in linia 'probe' corespunde celei de a 7-a variabila enuntata
poz=poz+1; j=j+1; % *.nam, adica regasita in coloana a 8-a (7+1, caci prima este timpul)
end
% din fisierul *.txt.
else
clear b3;
j=j+1; poz=poz+1;
end
end
k=k+1; pvar=pvar+2;
end

```

*Fisierul buckw.cir* - fisier PETS utilizat de programele MATLAB pentru calculul accelerat al stării staționare în cazul stabilizatorului cu convertor BUCK (după Wong)

Stabilizator cu convertor BUCK (dupa Wong, PESC' 87)

```

* convertor
Vg 1 0 dc 40V
#vcsq1 1 2 10 0 s
#d1 0 2 d
LF 2 n1 50uH ic=anynumber
rf n1 3 10mohm
rc 3 nc 50mohm
Cf nc 0 50u ic=anynumber
R1 3 0 5

* controller
R11 3 4 40kohm
R12 4 0 10kohm
R13 4 5 10kohm
R14 5 7 2kohm

```

```

C1 7 8 100nF ic=0
C14 7 5 1nF ic=0

#oa 6 5 8
Vref 6 0 dc 2.5V

* modulator
#hc1 8 9 10 hc
Vramp 9 0 ramp -0.01V 5V 0u 10u

* models
.model s #vcs von=0.7 ron=0.1 vton=5V vtoff=5V
.model d #d vd=0.7 ron=0.1
.model hc #hc

.tran 0.5u 12.12m
*.options dispmode=ascii
.options reltol=0.001 loctime=10ns minstep=10ns

.probe i(LF) v(NC) v(7) v(8) v(5) v(3)

.end

```

*Fișierul buckw.cir* - fișier PETS pentru simularea regimului tranzitoriu de pornire în cazul stabilizatorului cu convertor BUCK (după Wong)

Stabilizator cu convertor BUCK (dupa Wong, PESC' 87)

```

* convertor

Vg 1 0 dc 40V
#vcsq1 1 2 10 0 s
#d1 0 2 d
LF 2 n1 50uH ic=0
rf n1 3 10mohm
rc 3 nc 50mohm
cf nc 0 50u ic=0
Rl 3 0 5

* controller

R11 3 4 40kohm
R12 4 0 10kohm
R13 4 5 10kohm
R14 5 7 2kohm
C1 7 8 100nF ic=0
C14 7 5 1nF ic=0

#oa 6 5 8
Vref 6 0 dc 2.5V

* modulator
#hc1 8 9 10 hc
Vramp 9 0 ramp -0.01V 5V 0u 10u

* models
.model s #vcs von=0.7 ron=0.1 vton=5V vtoff=5V
.model d #d vd=0.7 ron=0.1
.model hc #hc

.tran 0.5u 12.12m
*.options dispmode=ascii
.options reltol=0.001 loctime=10ns minstep=10ns

.probe i(LF) v(NC) v(7) v(8) v(5) v(3)

.end

```

## A 27 Determinarea caracteristicilor statice reale și a familiilor de forme de undă în stare staționară.

*Fișierul d\_car.m*

% Ridicarea caracteristicilor statice ale convertoarelor prin metoda de tip Newton cu  
% actualizare Broyden ("Broyden-update quasi-Newton method").

% se itereaza pana ce eroarea e devine mai mica decat e0  
% sau pana se depaseste numarul maxim de simulari admise, Nsimmax.

```

% Tot ce este intre liniile cu diezi este specific convertorului analizat
% (nr. variabilelor de stare si dependenta tens. pe capacitati numai de
% tensiunile raportate fata de masa) si se va actualiza pt fiecare convertor
% analizat in parte

% Matricea F1 contine datele in binar ale fisierului analizat
% Matricea 'newline' contine indicii din F unde incep liniile
% Matricea 'new' contine indicii din F unde incep liniile cu caractere diferite de blank sau TAB
% Matricea 'state' contine numarul liniilor si indicii din F unde sunt definite variabilele de stare (L,I, C, sau c)
% Matricea 'location' contine indicii din F unde incep definirile conditiilor initiale (ic=anynumber)
% Aceasta matrice se actualizeaza in urma fiecarei simulari deoarece noile valori pt variabilele de
% stare nu sunt scrise cu acelasi numar de cifre zecimale.

% Declaratia 'ic=anynumber' trebuie scrisa cu litere mici, fara spatiu intre litere si
% fara comentariu anterior (*) deoarece programul nu face aceste verificari.
% In linia 'probe' se vor scrie variabilele de stare in ordinea in care au fost enunate
% elementele reactive in fisier.

clear all;
fclose('all');
clear % sterge variabile
Fer=1;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

tic

% Deschide fisierul specificat, citeste datele binare in matricea F, apoi inchide fisierul
% si transforma in string datele binare citite.
file=input(' Introduceti numele fisierului dorit a se analiza cu cale si nume (pana la 8 caractere), fara extensie: ','s');
A1=[FID1=fopen(""); A2=['.cir','.r'];]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
F1=fread(FID1);
fclose(FID1);
S1=setstr(F1);

% Gaseste indicii tuturor inceputurilor de linii
j=2; newline(1)=1;
for i=1:length(F1)
    if F1(i)==10; % 10 este codul binar pt carriage return
        newline(j)=i+1; j=j+1;
    else
        end
end
% Variabila NL contine numarul de linii din fisier
NL=length(newline);

% Gaseste indicii din F1 ai primelor elemente diferite de blank sau TAB
% din toate liniile, cu exceptia ultimei
for j=1:length(newline)-1
    k=0;
    while k<=(newline(j+1)-newline(j)-1)
        if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9) % 32 este codul binar pt. blank iar 9 pt TAB
            k=k+1;
        else
            break
        end
    end
    new(j)=newline(j)+k;
end

% Gaseste indexul din F al primului element diferit de blank sau TAB din ultima linie
j=length(newline);
k=0;
while k<=(length(F1)-newline(j)-1)
    if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9)
        k=k+1; % gaseste primele elemente diferite de blank
    else % sau TAB din ultima linie
        break
    end
end
new(j)=newline(j)+k;

% Daca ultimul caracter din fisier este carriage return atunci ultima linie nu are sens a
% se analiza pt. a gasi primul caracter diferit de blank sau TAB
if F1(length(F1))==10 l=length(new)-1;

```

```

else l=length(new);
end

% Scrie in vectorul 'state' indicii din F1 unde apar: L,I,C,c
% si care apar ca prime elemente pe linii
k=1;
for j=1:l
    if (F1(new(j))==76)|(F1(new(j))==108)|(F1(new(j))==67)|(F1(new(j))==99) % 76 este codul pt L,
        state(k,1)=j; state(k,2)=new(j); k=k+1; % 108 pt I,
    else % 67 pt C
    end % 99 pt c
end
Nstates=length(state);

% Matricea de verificare
for k=1:Nstates
S1(k)=setstr(F1(state(k,2)));
end
S1';

% Gaseste indecsii din F1 de unde incepe valoarea conditiei initiale
% (cauta numai in liniile care contin L,I,C sau c) si avertizeaza daca nu a gasit 'ic=anynumber' pe o linie cu variabila de stare
target1=['ic=anynumber']; % vectorul de comparare
for k=1:length(state(:,1))
flag=0;
for j=0:newline(state(k,1)+1)-state(k,2)-length(target1)

    for t=1:length(target1)
        block1(1,t)=S1(state(k,2)+j+(t-1)); % defineste stringul de comparare
    end

    comp1=(block1==target1);
    if sum(comp1)==12 location(k)=state(k,2)+j; flag=1;
    else
        j=j+1;
    end
end
if flag==0
fprintf('Eroare ! Conditia initiala in forma "ic=anynumber" nu este definita in linia %i\n',state(k,1));
else
end

end

if flag==0 break
else
end

% Gaseste indexul din F1 de unde incepe valoarea conditiei initiale
target2=['width']; flag=0; % vectorul de comparare si semnalizarea gasirii stringului
j=1;
for k=1:length(S1)-length(target2)
    for t=1:length(target2)
        block2(1,t)=S1(k+t-1); % defineste stringul curent de comparare
    end

    comp2=(block2==target2);
    if sum(comp2)==5 locwidth(j)=k; j=j+1; flag=1;
    else
        k=k+1;
    end
end
if flag==0
fprintf('Eroare: Conditia initiala in forma "width" nu a fost gasita'); break;
else
end

% Se efectueaza maxim Nsimmax simulari sau pana ce eroarea dintre vectorul de intrare
% in simulare si vectorul de stare de iesire din simulare, e, devine mai mica decat e0.
% Extrage din fisierul F1 indicii corespunzatori conditiilor initiale vechi.
% insereaza pe cei corespunzatori conditiilor initiale noi (deoarece valorile de actualizare
% a conditiilor initiale nu au acelasi numar de cifre)
% Conditile initiale au 9 cifre semnificative dupa virgula (vezi linia comentata cu ***)

% Defineste parametrii algoritmului gasirii starii stationare:
e0=1e-4; Sqreta=sqrt(e0); xmin=10*e0; xm=10*e0;

```

```

#####
x0=[0;0;0;0]; % !!! dimensiunea depinde de numarul variabilelor de stare din convertor
#####
Ts=20e-6; D=0.8;-0.05:0.05; %width=D*Ts;
I=eye(length(x0));
shiftax=9*ones(Nstates,1); % initializeaza matricea de shiftare anterioara care are dimensiunea 9
% pt. fiecare element reactiv (stringul 'anynumber' are 9 caractere
shiftaw=5*ones(1,Nstates); % pt. ca stringul 'width' are 5 caractere
Nsimmax=3000; % numarul maxim de perioade simulate
countsim=0; % contorul numarului de similari
iter=0; % contorul de iteratii

dcont=1;
while dcont<=length(D)
width=D(dcont)*Ts;

for k=1:length(x0)
b=abs(sprintf('% 9f',x0(k))); % ***
shiftx(k)=length(b);
F1prim=F1(1:location(k)+2,1); % scrie valorile lui xisim in fisier la conditiile initiale
F1secund=F1(location(k)+3+shiftax(k):length(F1),1);
F1=[F1prim;b';F1secund];
for i=k+1:length(location)
location(i)=location(i)+shiftx(k)-shiftax(k);
end

for j=1:length(locwidth)
if location(k)<=locwidth(j)
locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
else
end
end
shiftax(k)=shiftx(k);
end

for k=1:length(width)
b=abs(sprintf('% 9f',width(k))); % *** valoarea pulsului se scrie cu 9 cifre dupa virgula
shiftw(k)=length(b);
F1prim=F1(1:locwidth(k)-1,1); % scrie valorile lui xisim in fisier la conditiile initiale
F1secund=F1(locwidth(k)+shiftaw(k):length(F1),1);
F1=[F1prim;b';F1secund];
for i=k+1:length(width)
locwidth(i)=locwidth(i)+shiftw(k)-shiftaw(k);
end

for j=1:length(location)
if locwidth(k)<=location(j)
location(j)=location(j)+shiftw(k)-shiftaw(k);
else
end
end
shiftaw(k)=shiftw(k);
end
A1=['FID1=fopen(""); A2=['.cir","w");]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);

% Lanseaza o simulare pt a obtine matricea 'index' in care sunt corelate
% ordinea din linia 'probe' cu cea din fisierul *.txt (se face o singura data)
% si Ftx0, necesara la calculul erorii care se calculeaza si ea
A1=['dos("pets "); A2=[""];]; B=[A1,file,A2]; eval(B); %dos('pets buck');
countsim=countsim+1;
statdet; % m-file (pt. PETS) pentru detectarea ordinii de scriere in fisierul *.txt
% a variabilelor si corelarea lor cu cele definite in comanda 'probe'.

A1=['load '); A2=['.txt;']; B=[A1,file,A2]; eval(B); % load buck.txt;
A1=['varpets=']; A2=['(:,2:length('); A3=['(1,:));]; B=[A1,file,A2,file,A3]; eval(B); % varpets=buck(:,2:length(buck(1,:)));
for k=1:length(varpets(1,:))
var(:,k)=varpets(:,index(k));
end

```

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe  
 % (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa  
 % aceasta linie difera de la schema la schema



```

#####
%var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)];
#####

Ftx0=var(length(var,:)); % se va inlocui cu linia '*.tran' din *.cir pt alte variabile de stare
absvar=abs(var); absmax=max(absvar); % determina maximele variabilelor din ultima simulare
clear var

for i=1:length(x0)
    er(i)=abs(x0(i)-Ftx0(i))/max(x0(i),xmin); % calculul erorilor partiale
end
e=max(er);
%end

% Se testeaza conditia de stare stationara si daca nu e stare stationara se calculeaza
% Jacobianul in x0 prin metoda diferentelor finite, apoi x1.
if e>e0
    %*****
    % intre aceste linii cu stelute se calculeaza o singura data Jacobianul in
    % punctul solutie initiala prin metoda diferentelor finite

    for i=1:length(x0)
        dx(i,1)=Sqreta*max(absmax(i),xm);
        x1=x0+I(:,i).*dx(i);

        % simuleaza pt a obtine Ft(x0+ui.*dx(i))=Ft(x1)
        for k=1:length(x1)
            b=abs(sprintf('%9f',x1(k))); % ***
            shiftx(k)=length(b);
            F1prim=F1(1:location(k)+2,1); % scrie valorile lui xisim in fisier la conditiile initiale
            F1secund=F1(location(k)+3+shiftx(k):length(F1),1);
            F1=[F1prim;b';F1secund];
            for m=k+1:length(location)
                location(m)=location(m)+shiftx(k)-shiftax(k);
            end

            for j=1:length(locwidth)
                if location(k)<=locwidth(j)
                    locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
                else
                    end
            end
            shiftax(k)=shiftx(k);
        end

        A1=[FID1=fopen(''); A2=['.cir', 'w']; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
        fwrite(FID1,F1);
        fclose(FID1);
        A1=['dos("pets '); A2=['']; B=[A1,file,A2]; eval(B); %dos('pets buck');
        countsim=countsim+1;
        A1=['load '); A2=['.txt']; B=[A1,file,A2]; eval(B); %load file.txt;
        A1=['varpets=']; A2=[';',2:length(''); A3=['(1,:)'); B=[A1,file,A2,file,A3]; eval(B);
        varpets=buck(:,2:length(buck(1,:)))
        for k=1:length(varpets(1,:))
            var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
        end

        % Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
        % (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
        % aceasta linie va diferi de la schema la schema

        %*****
        %var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)]; % corelate din schema cu ordinea din .probe
        %*****

        Ftx1=var(length(var,:));
        clear var
        jcol=I(:,i)-1/dx(i).*(Ftx1-Ftx0); % calculeaza coloana curenta din J(x0)
        Jx0=[Jx0 jcol];
    end

    %*****
    x1=x0-inv(Jx0)*(x0-Ftx0);
    iter=iter+1;
else
end

```

```

while (countsim<Nsimmax)&(e>e0)
%*****
% intre aceste linii cu stelute se calculeaza Jacobianul prin actualizarea Broyden

% simuleaza pt a obtine Ft(x1)
for k=1:length(x1)
    b=abs(sprintf('%9f',x1(k))); % ***
    shiftx(k)=length(b);
    F1prim=F1(1:location(k)+2,1); % scrie valorile lui xisim in fisier la conditiile initiale
    F1secund=F1(location(k)+3+shiftax(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(location)
        location(i)=location(i)+shiftx(k)-shiftax(k);
    end

    for j=1:length(locwidth)
        if location(k)<=locwidth(j)
            locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
        else
            end
        end
    shiftax(k)=shiftx(k);
end

A1=['FID1=fopen(""); A2=['.cir',"w"];]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);
A1=['dos("pets "); A2=[""];]; B=[A1,file,A2]; eval(B); %dos('pets buck');
countsim=countsim+1;
A1=['load '); A2=['.txt']; B=[A1,file,A2]; eval(B); %load file.txt;
A1=['varpets=']; A2=['(:,2:length('); A3=['(1,:));]; B=[A1,file,A2,file,A3]; eval(B);
% varpets=buck(:,2:length(buck(1,:)))
for k=1:length(varpets(1,:))
    var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie va diferi de la schema la schema

%#####
%var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)]; % corelate din schema cu ordinea din .probe
%#####

Ftx1=var(length(var,:));
clear var

for i=1:length(x1)
    er(i)=abs(x1(i)-Ftx1(i))/max(x1(i),xmin); % calculul erorilor partiale
end
e=max(er); % eroarea totala

if e>e0
Jx1=Jx0+(x1-Ftx1)*(x1-x0)/sum((x1-x0).^2); % actualizarea Broyden
x0=x1; Jx0=Jx1; % actualizeaza pt viitorul calcul de Jacobian
else
break % e pus pt a sparge bucla while internă si a nu mai contoriza iteratia de la
end % sfarsitul buclei care se face chiar daca conditia de eroare s-a indeplinit.
% Aceasta pt ca testul in bucla while se face la inceputul buclei
%*****

% Se efectueaza iteratia
x1=x1-inv(Jx1)*(x1-Ftx1);
iter=iter+1;
end % sfarsitul buclei interne while

if countsim>=Nsimmax
fprintf('Numarul maxim de simulari depasit !\n');
break
else
end
end

A1=['load '); A2=['.txt']; B=[A1,file,A2]; eval(B); %load file.txt;
A1=['varpets=']; A2=['(:,2:length('); A3=['(1,:));]; B=[A1,file,A2,file,A3]; eval(B);

```

```

% varpets=buck(:,2:length(buck(1,:)))
for k=1:length(varpets(1,:))
var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
end
xsteady=var;
clear var
% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat variabili fata de masa
% aceasta linie va diferi de la schema la schema

%#####
%var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)]; % corelate din schema cu ordinea din .probe
Vo(dcont)=1/length(xsteady(:,4)).*sum(xsteady(:,4)); % Vo e corelat cu ordinea marimilor
VC1(dcont)=1/length(xsteady(:,2)).*sum(xsteady(:,2)); % din linia probe
%#####

dcont=dcont+1; Jx0=[];
x0=Ftx1; % pleaca pt urmatorul D de la valoarea gasita la precedentul, pt a reduce nr de iteratii si simulari
end % de la bucla externa de while

% Scrie din nou 'anynumber' in conditiile initiale pentru ca fisierul sa fie pregatit
% pentru o noua rulare
b=['anynumber'];
for k=1:length(x0)
shiftx(k)=length(b);
F1prim=F1(1:location(k)+2,1);
F1secund=F1(location(k)+3+shiftx(k):length(F1),1);
F1=[F1prim;b';F1secund];
for i=k+1:length(location)
location(i)=location(i)+shiftx(k)-shiftax(k);
end

for j=1:length(locwidth)
if location(k)<=locwidth(j)
locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
else
end
end
shiftax(k)=shiftx(k);
end

b=['width'];
for k=1:length(width)
shiftw(k)=length(b);
F1prim=F1(1:locwidth(k)-1,1); % scrie valorile lui xisim in fisier la conditiile initiale
F1secund=F1(locwidth(k)+shiftaw(k):length(F1),1);
F1=[F1prim;b';F1secund];
for i=k+1:length(width)
locwidth(i)=locwidth(i)+shiftw(k)-shiftaw(k);
end

for j=1:length(location)
if locwidth(k)<=location(j)
location(j)=location(j)+shiftw(k)-shiftaw(k);
else
end
end
shiftaw(k)=shiftw(k);
end
A1=['FID1=fopen(""); A2=['.cir","w"];]; B=[A1,file,A2]; eval(B); %FID1=fopen('file.cir','w');
fwrite(FID1,F1); fclose(FID1);

fprintf('Caracteristica a necesitat %i iteratii si %i simulari',iter,countsim);

% Ploteaza caracteristica se comanda ideala si reala
%#####
Vg=12;
%#####
for i=1:length(D)
Void(i)=Vg/(1-D(i))^2;
VC1id(i)=Vg/(1-D(i));
end
plot(D,Void,'k'); hold on; plot(D,VC1id,'k'); hold on; plot(D,Vo,'r'); plot(D,VC1,'g');

```

toc

*Fișierul st\_waves.m*

```
% Ridicarea formelor de unda in stare stationara pt convertoare prin metoda de tip Newton cu
% actualizare Broyden ("Broyden-update quasi-Newton method").

% se itereaza pana ce eroarea e devine mai mica decat e0
% sau pana se depaseste numarul maxim de simulari admise, Nsimmax.

% Tot ce este intre liniile cu diezi este specific convertorului analizat
% (nr. variabilelor de stare si dependenta tens. pe capacitati numai de
% tensiunile raportate fata de masa) si se va actualiaza pt fiecare convertor
% analizat in parte

% Matricea F1 contine datele in binar ale fisierului analizat
% Matricea 'newline' contine indicii din F unde incep liniile
% Matricea 'new' contine indicii din F unde incep liniile cu caractere diferite de blank sau TAB
% Matricea 'state' contine numarul liniilor si indicii din F unde sunt definite variabilele de stare (L, I, C, sau e)
% Matricea 'location' contine indicii din F unde incep definirile conditiilor initiale (ic=anynumber)
% Aceasta matrice se actualizeaza in urma fiecarei simulari deoarece noile valori pt variabilele de
% stare nu sunt scrise cu acelasi numar de cifre zecimale.

% Declaratia 'ic=anynumber' trebuie scrisa cu litere mici, fara spatiu intre litere si
% fara comentariu anterior (*) deoarece programul nu face aceste verificari.
% In linia 'probe' se vor scrie variabilele de stare in ordinea in care au fost enunate
% elementele reactive in fisier.

clear all;
fclose('all');
clear % sterge variabile
Fer=2;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end
figure; figure;

tic

% Deschide fisierul specificat, citeste datele binare in matricea F, apoi inchide fisierul
% si transforma in string datele binare citite.
file=input('Enter the filename to be analysed with full path and filename (up to 8 characters),without extension: ','s');
A1=['FID1=fopen("'); A2=['.cir","r");]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
F1=fread(FID1);
fclose(FID1);
S1=setstr(F1);

% Gaseste indicii tuturor inceputurilor de linii
j=2; newline(1)=1;
for i=1:length(F1)
    if F1(i)==10; % 10 este codul binar pt carriage return
        newline(j)=i+1; j=j+1;
    else
        end
end
% Variabila NL contine numarul de linii din fisier
NL=length(newline);

% Gaseste indicii din F1 ai primelor elemente diferite de blank sau TAB
% din toate liniile, cu exceptia ultimei
for j=1:length(newline)-1
    k=0;
    while k<=(newline(j+1)-newline(j))-1
        if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9) % 32 este codul binar pt. blank iar 9 pt TAB
            k=k+1;
        else
            break
        end
    end
    new(j)=newline(j)+k;
end

% Gaseste indexul din F al primului element diferit de blank sau TAB din ultima linie
j=length(newline);
k=0;
while k<=(length(F1)-newline(j))-1
    if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9)
        k=k+1;
        % gaseste primele elemente diferite de blank
    end
end
```

```

else                % sau TAB din ultima linie
break
end
end
new(j)=newline(j)+k;

% Daca ultimul caracter din fisier este carriage return atunci ultima linie nu are sens a
% se analiza pt. a gasi primul caracter diferit de blank sau TAB
if F1(length(F1))==10 l=length(new)-1;
else l=length(new);
end

% Scrie in vectorul 'state' indicii din F1 unde apar: L,I,C,c
% si care apar ca prime elemente pe linii
k=1;
for j=1:l
    if (F1(new(j))==76)|(F1(new(j))==108)|(F1(new(j))==67)|(F1(new(j))==99) % 76 este codul pt L,
        state(k,1)=j; state(k,2)=new(j); k=k+1;                % 108 pt I,
    else
        % 67 pt C
    end
    % 99 pt c
end
Nstates=length(state);

% Matricea de verificare
for k=1:Nstates
S1(k)=setstr(F1(state(k,2)));
end
S1';

% Gaseste indeciii din F1 de unde incepe valoarea conditiei initiale
% (cauta numai in liniile care contin L,I,C sau c) si avertizeaza daca nu a gasit 'ic=anynumber' pe o linie cu variabila de stare
target1=['ic=anynumber']; % vectorul de comparare
for k=1:length(state(:,1))
flag=0;
for j=0:newline(state(k,1)+1)-state(k,2)-length(target1)

    for t=1:length(target1)
        block1(1,t)=S1(state(k,2)+j+(t-1)); % defineste stringul de comparare
    end

    comp1=(block1==target1);
    if sum(comp1)==12 location(k)=state(k,2)+j; flag=1;
    else
        j=j+1;
    end
end
if flag==0
fprintf('Error: Initial condition in the form "ic=anynumber" not defined on line %i\n',state(k,1));
else
end

end

if flag==0 break
else
end

% Gaseste indexul din F1 de unde incepe valoarea conditiei initiale
target2=['width']; flag=0; % vectorul de comparare si semnalizarea gasirii stringului
j=1;
for k=1:length(S1)-length(target2)
    for t=1:length(target2)
        block2(1,t)=S1(k+t-1); % defineste stringul curent de comparare
    end

    comp2=(block2==target2);
    if sum(comp2)==5 locwidth(j)=k; j=j+1; flag=1;
    else
        k=k+1;
    end
end
if flag==0
fprintf('Error: Initial condition in the form "width" not found'); break;
else
end
end

```

```

% Se efectueaza maxim Nsimmax simulari sau pana ce eroarea dintre vectorul de intrare
% in simulare si vectorul de stare de iesire din simulare, e, devine mai mica decat e0.
% Extrage din fisierul F1 indicii corespunzatori conditiilor initiale vechi,
% insereaza pe cei corespunzatori conditiilor initiale noi (deoarece valorile de actualizare
% a conditiilor initiale nu au acelasi numar de cifre)
% Conditiiile initiale au 9 cifre semnificative dupa virgula (vezi linia comentata cu ***)

% Definese parametrii algoritmului gasirii starii stationare:
e0=1e-4; Sqreta=sqrt(e0); xmin=10*e0; xm=10*e0;

%#####
x0=[0;0;0;0]; % !!! dimensiunea depinde de numarul variabilelor de stare din convertor
%#####
Ts=20e-6; D=0.8;-0.05:0.05; %width=D*Ts;
l=eye(length(x0));
shiftax=9*ones(Nstates,1); % initializeaza matricea de shiftare anterioara care are dimensiunea 9
% pt. fiecare element reactiv (stringul 'anynumber' are 9 caractere
shiftaw=5*ones(1,Nstates); % pt. ca stringul 'width' are 5 caractere
Nsimmax=3000; % numarul maxim de perioade simulate
countsim=0; % contorul numarului de simulari
iter=0; % contorul de iteratii

dcont=1;
while dcont<=length(D)
width=D(dcont)*Ts;

for k=1:length(x0)
b=abs(sprintf('%9f,x0(k))); % ***
shiftx(k)=length(b);
F1prim=F1(1:location(k)+2,1); % scrie valorile lui xisim in fisier la conditiile initiale
F1secund=F1(location(k)+3+shiftax(k):length(F1),1);
F1=[F1prim;b';F1secund];
for i=k+1:length(location)
location(i)=location(i)+shiftx(k)-shiftax(k);
end

for j=1:length(locwidth)
if location(k)<=locwidth(j)
locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
else
end
end
shiftax(k)=shiftx(k);
end

for k=1:length(width)
b=abs(sprintf('%9f,width(k))); % *** valoarea pulsului se scrie cu 9 cifre dupa virgula
shiftw(k)=length(b);
F1prim=F1(1:locwidth(k)-1,1); % scrie valorile lui xisim in fisier la conditiile initiale
F1secund=F1(locwidth(k)+shiftw(k):length(F1),1);
F1=[F1prim;b';F1secund];
for i=k+1:length(width)
locwidth(i)=locwidth(i)+shiftw(k)-shiftw(k);
end

for j=1:length(location)
if locwidth(k)<=location(j)
location(j)=location(j)+shiftw(k)-shiftw(k);
else
end
end
shiftw(k)=shiftw(k);
end
A1=['FID1=fopen(''); A2=['.cir","w");]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);

% Lanseaza o simulare pt a obtine matricea 'index' in care sunt corelate
% ordinea din linia 'probe' cu cea din fisierul *.txt (se face o singura data)
% si Ftx0, necesara la calculul erorii care se calculeaza si ea
A1=['dos("pets "); A2=[""];]; B=[A1,file,A2]; eval(B); %dos('pets buck');
countsim=countsim+1;
statdet; % m-file (pt. PETS) pentru detectarea ordinii de scriere in fisierul *.txt
% a variabilelor si corelarea lor cu cele definite in comanda 'probe'.

A1=['load ']; A2=['.txt;']; B=[A1,file,A2]; eval(B); % load buck.txt;

```

```

A1=['varpets=']; A2=['(:,2:length(')); A3=['(1,:));']; B=[A1,file,A2,file,A3]; eval(B); % varpets=buck(:,2:length(buck(1,:)));
for k=1:length(varpets(1,:))
    var(:,k)=varpets(:,index(k));
end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie difera de la schema la schema

#####
%var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)];
#####

Ftx0=var(length(var,:)); % se va inlocui cu linia '*.tran' din *.cir pt alte variabile de stare
absvar=abs(var); absmax=max(absvar); % determina maximele variabilelor din ultima simulare
clear var

for i=1:length(x0)
    er(i)=(abs(x0(i)-Ftx0(i))/max(x0(i),xmin)); % calculul erorilor partiale
end
e=max(er);
%end

% Se testeaza conditia de stare stationara si daca nu e stare stationara se calculeaza
% Jacobianul in x0 prin metoda diferentelor finite, apoi x1.
if e>e0
    %*****
    % intre aceste linii cu stelute se calculeaza o singura data Jacobianul in
    % punctul solutie initiala prin metoda diferentelor finite

    for i=1:length(x0)
        dx(i,1)=Sqrta*max(absmax(i),xm);
        x1=x0+I(:,i).*dx(i);

        % simuleaza pt a obtine Ft(x0+ui*dxi)=Ft(x1)
        for k=1:length(x1)
            b=abs(sprintf('%9f',x1(k))); % ***
            shiftx(k)=length(b);
            F1prim=F1(1:location(k)+2,1); % scrie valorile lui xisim in fisier la conditiile initiale
            F1secund=F1(location(k)+3+shiftax(k):length(F1),1);
            F1=[F1prim;b';F1secund];
            for m=k+1:length(location)
                location(m)=location(m)+shiftx(k)-shiftax(k);
            end

            for j=1:length(locwidth)
                if location(k)<=locwidth(j)
                    locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
                else
                    end
            end
            shiftax(k)=shiftx(k);
        end

        A1=['FID1=fopen(""); A2=['.cir","w");']; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
        fwrite(FID1,F1);
        fclose(FID1);
        A1=['dos("pets "); A2=[""];']; B=[A1,file,A2]; eval(B); %dos('pets buck');
        countsim=counts+1;
        A1=['load ]; A2=['.txt;']; B=[A1,file,A2]; eval(B); %load file.txt;
        A1=['varpets=']; A2=['(:,2:length(')); A3=['(1,:));']; B=[A1,file,A2,file,A3]; eval(B);
        % varpets=buck(:,2:length(buck(1,:)))
        for k=1:length(varpets(1,:))
            var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
        end

        % Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
        % (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
        % aceasta linie va difera de la schema la schema

        #####
        %var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)]; % corelate din schema cu ordinea din .probe
        #####

        Ftx1=var(length(var,:));
        clear var

```

```

jcol=l(:,i)-1/dx(i).*(Ftx1-Ftx0); % calculeaza coloana curenta din J(x0)
Jx0=[Jx0 jcol];
end

%*****
x1=x0-inv(Jx0)*(x0-Ftx0);
iter=iter+1;
else
end

while (countsim<Nsimmax)&(e>e0)
%*****
% intre aceste linii cu stelute se calculeaza Jacobianul prin actualizarea Broyden

% simuleaza pt a obtine Ft(x1)
for k=1:length(x1)
b=abs(sprintf('%f',x1(k))); % ***
shiftx(k)=length(b);
F1prim=F1(1:location(k)+2,1); % scrie valorile lui xisim in fisier la conditiile initiale
F1secund=F1(location(k)+3+shiftx(k):length(F1),1);
F1=[F1prim;b';F1secund];
for i=k+1:length(location)
location(i)=location(i)+shiftx(k)-shiftax(k);
end

for j=1:length(locwidth)
if location(k)<=locwidth(j)
locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
else
end
end
shiftax(k)=shiftx(k);
end

A1=['FID1=fopen(""); A2=['.cir","w");]; B=[A1,file,A2]; eval(B); %FID1=fopen('buck.cir','w');
fwrite(FID1,F1);
fclose(FID1);
A1=['dos("pets "); A2=[""];]; B=[A1,file,A2]; eval(B); %dos('pets buck');
countsim=countsim+1;
A1=['load '); A2=['.txt;]; B=[A1,file,A2]; eval(B); %load file.txt;
A1=['varpets=']; A2=['(:,2:length('); A3=['(1,:));]; B=[A1,file,A2,file,A3]; eval(B);
% varpets=buck(:,2:length(buck(1,:)))
for k=1:length(varpets(1,:))
var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
end

% Linia dintre diezi care urmeaza da dependenta variabilelor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie va diferi de la schema la schema

%#####
%var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)]; % corelate din schema cu ordinea din .probe
%#####

Ftx1=var(length(var,:));
clear var

for i=1:length(x1)
er(i)=abs(x1(i)-Ftx1(i))/max(x1(i),xmin); % calculul erorilor pariale
end
e=max(er); % eroarea totala

if e>e0
Jx1=Jx0+(x1-Ftx1)*(x1-x0)/sum((x1-x0).^2); % actualizarea Broyden
x0=x1; Jx0=Jx1; % actualizeaza pt viitorul calcul de Jacobian
else
break % e pus pt a sparge bucla while internă si a nu mai contoriza iteratia de la
end % sfarsitul buclei care se face chiar daca conditia de eroare s-a indeplinit.
% Aceasta pt ca testul in bucla while se face la inceputul buclei
%*****

% Se efectueaza iteratia
x1=x1-inv(Jx1)*(x1-Ftx1);
iter=iter+1;
end % sfarsitul buclei interne while

```



```

if countsim>=Nsimmax
fprintf('Numarul maxim de simulari depasit !\n');
break
else
end

A1=['load ']; A2=['.txt;']; B=[A1,file,A2]; eval(B); %load file.txt;
A1=['varpets=']; A2=['(:,2:length(')); A3=['(1,:));']; B=[A1,file,A2,file,A3]; eval(B);
% varpets=buck(:,2:length(buck(1,:)))
for k=1:length(varpets(1,:))
var(:,k)=varpets(:,index(k)); % in variabila var sunt rezultatele simularii pt marimile de stare
end
A1=['t=']; A2=['(:,1);']; B=[A1,file,A2]; eval(B); % t=file(:,1; timpul de afisat pt unde
xsteady=var;
clear var
% Linia dintre diezi care urmeaza da dependenta variabileor de stare functie de marimile din linia .probe
% (ordinea se ia din linia .probe) pentru ca PETS nu stie decat tensiuni fata de masa
% aceasta linie va diferi de la schema la schema

#####
%var=[var(:,1) var(:,2) var(:,3)-var(:,4) var(:,3)-var(:,5)]; % corelate din schema cu ordinea din .probe
iL=xsteady(:,1); % iL, iL1 sunt corelate cu ordinea marimilor
figure(1); plot(t,iL,'k'); hold on

iL1=xsteady(:,3); % din linia probe
figure(2); plot(t,iL1,'k'); hold on
#####

dcont=dcont+1; Jx0=[];
x0=F1x1; % pleaca pt urmatorul D de la valoarea gasita la precedentul, pt a reduce nr de iteratii si simulari
end % de la bucla externa de while

% Scrie din nou 'anynumber' in conditiile initiale pentru ca fisierul sa fie pregatit
% pentru o noua rulare
b=['anynumber'];
for k=1:length(x0)
shiftx(k)=length(b);
F1prim=F1(1:location(k)+2,1);
F1secund=F1(location(k)+3+shiftx(k):length(F1),1);
F1=[F1prim;b';F1secund];
for i=k+1:length(location)
location(i)=location(i)+shiftx(k)-shiftax(k);
end

for j=1:length(locwidth)
if location(k)<=locwidth(j)
locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
else
end
end
shiftax(k)=shiftx(k);
end

b=['width'];
for k=1:length(width)
shifw(k)=length(b);
F1prim=F1(1:locwidth(k)-1,1); % scrie valorile lui xisim in fisier la conditiile initiale
F1secund=F1(locwidth(k)+shifw(k):length(F1),1);
F1=[F1prim;b';F1secund];
for i=k+1:length(width)
locwidth(i)=locwidth(i)+shifw(k)-shifaw(k);
end

for j=1:length(location)
if locwidth(k)<=location(j)
location(j)=location(j)+shifw(k)-shifaw(k);
else
end
end
shifaw(k)=shifw(k);
end
A1=['FID1=fopen(''); A2=['.cir','w');']; B=[A1,file,A2]; eval(B); %FID1=fopen('file.cir','w');
fwrite(FID1,F1); fclose(FID1);

```

```
fprintf('Caracteristicile au necesitat %i iteratii si %i simulari', iter, countsim);
```

```
toc
```

*Fișierul boost2.cir*

```
Convertor boost patratric (boost2.cir)

* converter
vg      1 0 DC 12V
L       1 2 500u ic=anynumber
#d1prime 2 4 d
#d2prime 2 3 d
C1      3 0 4.7u ic=anynumber
L1      3 4 2m ic=anynumber
#vcs1   4 0 100 0 vcs
#d2     4 5 d
C       5 0 2.2u ic=anynumber
R       5 0 2k

* switch control
vc      100 0 pulse 0 10V 0 width 20u

.tran 0.1us 20u

* models
.model vcs #vcs (Vton=4.9V Vtoff=5.1V Ron=0.2 Goff=100n)
.model d #d (Ron=0.2 Goff=100n VD=0.4)

* relax tolerances
.options reltol=0.01 vntol=1e-4 abstol=1e-9
.options dispmode=ascii

.probe i(L) v(3) i(L1) v(5)

.end
```

## A 28 Simularea unui convertor BUCK-BOOST cu controller digital prin legarea programelor MATLAB și PETS

*Fișierul digctrl.m*

```
% Simularea unui controller digital de tip I legand programele MATLAB si PETS.
% Comportarea unui convertor BUCK-BOOST cu controler digital la salt treapta in sarcina
% in conditiile in care tensiunea de referinta este constanta si egala cu 9V
% Se pleaca direct din starea stationara aferenta starii cu R1 ca sarcina (vezi vectorul xisim)
% In PETS se simuleaza o perioada, iar in MATLAB se calculeaza noul factor de umplere pt.
% perioada urmatoare, d(k+1), si furnizeaza fisierului 'cir' durata impulsului de
% comanda, egala evident cu d(k+1)*Ts, precum si valorile variabilelor de stare cu care
% sa inceapa simularea in perioada urmatoare.
% Se considera ca vo la sfarsitul perioadei este egala cu cea de la inceputul perioadei
% urmatoare, chiar daca nu e marime de stare.

% Cea care se modifica este durata impulsului din tensiunea de comanda.
% Se detecteaza pozitia in care apare acest parametru in comanda 'pulse' si apoi permanent
% el se actualizeaza.
% Pt. o detectie usoara in fisierul initial se va scrie la durata impulsului stringul 'width'

% Matricea F1 contine datele in binar ale fisierului analizat (de tip 'cir').
% Matricea 'newline' contine indicii din F1 unde incep liniile
% Matricea 'new' contine indicii din F1 unde incep liniile cu caractere diferite de blank sau TAB
% Matricea 'state' contine numarul liniilor si indicii din F1 unde sunt definite variabilele de stare (L, C, sau c)
% Matricea 'location' contine indicii din F1 unde incep definirile conditiilor initiale (ic=anynumber)
% Aceasta matrice ('location') se actualizeaza in urma fiecarei simulari deoarece noile valori
% pt variabilele de stare nu sunt scrise cu acelasi numar de cifre zecimale.
% Matricea 'locwidth' contine indicii din F1 unde incep definirile duratei impulsurilor (width)
% Si aceasta matrice se actualizeaza dupa fiecare simulare a unei perioade.
% Declaratiile 'anynumber' si 'width' trebuie scrise cu litere mici, fara spatiu intre
% litere si fara comentariu anterior (*) deoarece programul nu face aceste verificari.
% In linia 'probe' se vor scrie variabilele de stare IN ORDINEA IN CARE AU FOST ENUNTATE
% ELEMENTELE REACTIVE in fisier.

clear all;
fclose('all');
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
```

end

tic

```
% Deschide fisierul specificat, citeste datele binare in matricea F1, apoi inchide fisierul si transforma
% in string datele binare citite
```

```
file=input('Introduceti numele fisierului PETS cu cale si nume (pana la 8 caractere), fara extensie: ','s');
A1=[FID1=fopen(""); A2=['.cir','.r']; B=[A1,file,A2]; eval(B); %FID1=fopen('file.cir','w');
F1=fread(FID1); fclose(FID1); S1=setstr(F1);
```

```
% Gaseste indicii tuturor inceputurilor de linii (vectorul in care se scriu este 'newline')
```

```
j=2; newline(1)=1;
```

```
for i=1:length(F1)
```

```
    if F1(i)==10; % 10 este codul binar pt carriage return
        newline(j)=i+1; j=j+1;
```

```
    else
```

```
    end
```

```
end
```

```
% Variabila NL contine numarul de linii din fisier
```

```
NL=length(newline);
```

```
% Gaseste indicii din F1 ai primelor elemente diferite de blank sau TAB
```

```
% din toate liniile, cu exceptia ultimei (vectorul in care se scriu este 'new')
```

```
for j=1:length(newline)-1
```

```
    k=0;
```

```
    while k<=(newline(j+1)-newline(j))-1
```

```
        if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9) % 32 este codul binar pt. blank iar 9 pt TAB
```

```
            k=k+1;
```

```
        else
```

```
        break
```

```
        end
```

```
    end
```

```
    new(j)=newline(j)+k;
```

```
end
```

```
% Gaseste indexul din F1 al primului element diferit de blank sau TAB din ultima linie
```

```
j=length(newline);
```

```
k=0;
```

```
while k<=(length(F1)-newline(j))-1
```

```
    if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9)
```

```
        k=k+1;
```

```
        % gaseste primele elemente diferite de blank
```

```
        % sau TAB din ultima linie
```

```
    else
```

```
    break
```

```
    end
```

```
end
```

```
new(j)=newline(j)+k;
```

```
% Daca ultimul caracter din fisier este carriage return atunci ultima linie nu are sens a
```

```
% se analiza pt. a gasi primul caracter diferit de blank sau TAB
```

```
if F1(length(F1))==10 l=length(new)-1;
```

```
else l=length(new);
```

```
end
```

```
% Scrie in vectorul 'state' indicii din F1 unde apar: L,I,C,c
```

```
% si care apar ca prime elemente pe linii
```

```
k=1;
```

```
for j=1:l
```

```
    if (F1(new(j))==76)|(F1(new(j))==108)|(F1(new(j))==67)|(F1(new(j))==99) % 76 este codul pt L,
```

```
        state(k,1)=j; state(k,2)=new(j); k=k+1; % 108 pt I,
```

```
        else % 67 pt C
```

```
        end % 99 pt c
```

```
end
```

```
Nstates=length(state);
```

```
% Matricea de verificare V1
```

```
for k=1:Nstates
```

```
V1(k)=setstr(F1(state(k,2)));
```

```
end
```

```
V1';
```

```
% Gaseste indexul din F1 de unde incepe valoarea conditiei initiale
```

```
target1=['ic=anynumber']; % vectorul de comparare
```

```
for k=1:length(state(:,1))
```

```
    flag=0;
```

```
    for j=0:newline(state(k,1)+1)-state(k,2)-length(target1)
```

```

for t=1:length(target1)
    block1(1,t)=S1(state(k,2)+j+(t-1)); % defineste stringul de comparare
end

comp1=(block1==target1);
if sum(comp1)==12 location(k)=state(k,2)+j; flag=1;
else
    j=j+1;
end
end
if flag==0
    fprintf('Eroare ! Conditia initiala in forma "ic=anynumber" nu este definita in linia %i\n',state(k,1));
else
end

end

if flag==0 break
else
end

% Gaseste indexul din F1 de unde incepe valoarea conditiei initiale
target2=['width']; flag=0; % vectorul de comparare si semnalizarea gasirii stringului
j=1;
for k=1:length(S1)-length(target2)
    for t=1:length(target2)
        block2(1,t)=S1(k+t-1); % defineste stringul curent de comparare
    end

    comp2=(block2==target2);
    if sum(comp2)==5 locwidth(j)=k; j=j+1; flag=1;
    else
        k=k+1;
    end
end
if flag==0
    fprintf('Eroare ! Conditia initiala in forma "width" nu a fost gasita'); break;
else
end

% Defineste:
% - starea initiala din care pleaca sistemul
% - aici parametri ai intrarilor care nu pot fi definiti in fisierul *.cir (aici dk_1,
% - parametrii simularii
% - initializari (timpul curent de simulare, marimi dorite a se afisa, etc)
xisim=[7.4426; 8.9998]; % starea initiala
% Intrari (Ts, factor de umplere anterior si actual, regulator)
Ts=20e-6; dk_1=0.4236; width=[dk_1*Ts 0]; tsalt1=2e-3; tsalt2=10e-3;

% parametrii regulatorului
Vref=9; beta=20;
dmax=0.95; dmin=0.05; % limitarile in factor de umplere

% Parametrii simularii
tmax=18e-3; %eps=[1e-5; 1e-5]; eps=1e-3*ones(Nstates,1);
% Initializari
tcurrent=0; % initializeaza vectorul de timp curent de simulare
N=100*(tmax/Ts);
iL=zeros(1,N); vc=zeros(1,N); vo=zeros(1,N); time=zeros(1,N);
p=1; % contorul pt. variabilele de interes
countper=0; % contorul de perioade

Xisim=xisim; Xfsim=xisim+eps+ones(Nstates,1); % pentru intrare sigura in bucla while de simulare

% Lanseaza o simulare numai pentru a obtine matricea 'index',
% in care sunt corelate ordinea enuntarii variabilelor de stare din linia probe cu cea
% facuta de PETS in fisierul *.nam.

% Extrage din fisierul F1 indicii corespunzatori conditiilor initiale vechi,
% insereaza pe cei corespunzatori conditiilor initiale noi (deoarece valorile de actualizare
% ale conditiilor initiale nu au acelasi numar de cifre si translateaza astfel locatiile)
% Conditieile initiale au 9 cifre semnificative dupa virgula (vezi linia comentata cu ***)

shiftax=9*ones(1,Nstates); %matricea de shiftare anterioara care aici are dimensiunea 9
% pt fiecare element reactiv (stringul 'anynumber' are 9 caractere)

```

```

%xisim=zeros(1,Nstates); % valoarea initiala a vectorului de stare de unde incepe simularea
for k=1:length(xisim)
    b=abs(sprintf('%9f',xisim(k))); % ***
    shiftx(k)=length(b);
    F1prim=F1(1:location(k)+2,1); % scrie valorile lui xisim in fisier la conditiile initiale
    F1secund=F1(location(k)+3+shiftax(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(location)
        location(i)=location(i)+shiftx(k)-shiftax(k);
    end

    for j=1:length(locwidth)
        if location(k)<=locwidth(j)
            locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
        else
            end
        end
    shiftax(k)=shiftx(k);
end

% Efectueaza acelasi lucru pt valorile de actualizare ale latimii impulsurilor
shiftaw=5*ones(1,Nstates);
% pt. ca stringul 'width' are 5 caractere
for k=1:length(width)
    b=abs(sprintf('%9f',width(k))); % *** valoarea pulsului se scrie cu 9 cifre dupa virgula
    shiftw(k)=length(b);
    F1prim=F1(1:locwidth(k)-1,1); % scrie valorile lui xisim in fisier la conditiile initiale
    F1secund=F1(locwidth(k)+shiftaw(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(width)
        locwidth(i)=locwidth(i)+shiftw(k)-shiftaw(k);
    end

    for j=1:length(location)
        if locwidth(k)<=location(j)
            location(j)=location(j)+shiftw(k)-shiftaw(k);
        else
            end
        end
    shiftaw(k)=shiftw(k);
end

A1=[FID1=fopen(''); A2=['.cir','w');]; B=[A1,file,A2]; eval(B); %FID1=fopen('file.cir','w');
fwrite(FID1,F1); fclose(FID1);

% Lanseaza o simulare pt a obtine matricea 'index',
% in care sunt corelate ordinea enuntarii varabilelor de stare din linia probe cu cea
% facuta de PETS in fisierul * nam
A1=['dos("pets '); A2=[""];]; B=[A1,file,A2]; eval(B); %dos('pets file');
statdet;
pack;

% Starteaza simularea
while tcurrent<tmax
%while (countper<Nper)&any(abs(Xfsim-Xisim)>eps)

    if tcurrent<tsalt1
        width(2,1)=2e-3; % mai mare decat timpul de simulare scris la .tran (o perioada)
    elseif (tcurrent>=tsalt1)&(tcurrent<tsalt2)
        width(2,1)=0;
    else
        width(2,1)=2e-3;
    end

    % scrie ca valori initiale ale variabilelor de stare pe cele rezultate din simularea precedenta
    for k=1:length(xisim)
        b=abs(sprintf('%9f',xisim(k))); % ***
        shiftx(k)=length(b);
        F1prim=F1(1:location(k)+2,1); % scrie valorile lui xisim in fisier la conditiile initiale
        F1secund=F1(location(k)+3+shiftax(k):length(F1),1);
        F1=[F1prim;b';F1secund];
        for i=k+1:length(location)
            location(i)=location(i)+shiftx(k)-shiftax(k);
        end

        for j=1:length(locwidth)

```

```

    if location(k)<=locwidth(j)
        locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
    else
        end
    end
    shiftax(k)=shiftx(k);
end

for k=1:length(width)
    b=abs(sprintf('%9f,width(k)); % *** valoarea pulsului se scrie cu 9 cifre dupa virgula
    shiftw(k)=length(b);
    F1prim=F1(1:locwidth(k)-1,1); % scrie valorile lui ton in fisier la definirea tensiunii(lor) de comanda
    F1second=F1(locwidth(k)+shiftw(k):length(F1),1);
    F1=[F1prim;b';F1second];
    for i=k+1:length(width)
        locwidth(i)=locwidth(i)+shiftw(k)-shiftaw(k);
    end

    for j=1:length(location)
        if locwidth(k)<=location(j)
            location(j)=location(j)+shiftw(k)-shiftaw(k);
        else
            end
        end
        shiftaw(k)=shiftw(k);
    end

A1=['FID1=fopen(""); A2=['.cir','w'];]; B=[A1,file,A2]; eval(B); %FID1=fopen('file.cir','w');
fwrite(FID1,F1); fclose(FID1);

% Lanseaza simularea unei perioade din bucla while
A1=['dos("pets '); A2=[""];]; B=[A1,file,A2]; eval(B); %dos('pets file');
A1=['load '); A2=['.txt;']; B=[A1,file,A2]; eval(B); %load file.txt;
% Se defineste timpul curent pt o perioada de simulare
A1=['t='; A2=['(:,1);']; B=[A1,file,A2]; eval(B); %t=file(:,1)
A1=['varpets=']; A2=['(:,2:length('); A3=['(1,:));']; B=[A1,file,A2,file,A3]; eval(B); %varpets=file(:,2:length(file(1,:)));

    for k=1:length(varpets(1,:))
        var(:,k)=varpets(:,index(k)); % obtine matricea de variabile conforma cu linia '.probe'
    end

    for i=1:length(t);
        time(p)=countper*Ts+t(i); % se actualizeaza vectorul de timp curent de simulare
        Vg(p)=-var(i,1); iL(p)=var(i,1); vc(p)=-var(i,2); vo(p)=-var(i,2); % variabilele curente
        % linia de mai sus se va inlocui cu linia *.tran' din *.cir pt alte expresii
        % ale variabilelor de stare, linie prin care se calculeaza
        % variabilelor de stare, importata din fisierul *.cir
        % cu *.transform. Sau pur si simplu se stie expresia variabilei
        % de stare respective si se scrie direct in fisierul MATLAB
        p=p+1;
    end

xfsim=[iL(p-1); vc(p-1)]; tcurrent=time(p-1);

    % Se implementeaza controlerul. Aici controler numeric cu beta=20
    d_k=dk_1+ beta*Ts*(Vref-vo(p-1)); % pt ca tensiunea de iesire e de polaritate opusa

    if d_k>=dmax
        d_k=dmax;
    elseif d_k<=dmin % se limiteaza factorul de umplere superior si inferior
        d_k=dmin;
    else
        end

        width(1,1)=d_k*Ts;
        xisim=xfsim;
    %else
        %Xisim=Xfsim;
    %end

    countper=countper+1;
    clear var; % pt ca de la o simulare a alta nr. de elemente pe coloana difera, PETS-ul variind nr de puncte in jurul comutarii
end

% Scrie din nou 'ic=anynumber' in conditiile initiale si 'width' in sursele de comanda

```

```

% pentru ca fisierul sa fie pregatit pentru o noua rulare
b=['anynumber'];
for k=1:length(xfsim)
    shiftx(k)=length(b);
    F1prim=F1(1:location(k)+2,1);
    F1secund=F1(location(k)+3+shiftx(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(location)
        location(i)=location(i)+shiftx(k)-shiftax(k);
    end

    for j=1:length(locwidth)
        if location(k)<=locwidth(j)
            locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
        else
            end
        end
    end
    shiftax(k)=shiftx(k);
end

b=['width'];
for k=1:length(width)
    shiftw(k)=length(b);
    F1prim=F1(1:locwidth(k)-1,1); % scrie valorile lui xisim in fisier la conditiile initiale
    F1secund=F1(locwidth(k)+shiftaw(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(width)
        locwidth(i)=locwidth(i)+shiftw(k)-shiftaw(k);
    end

    for j=1:length(location)
        if locwidth(k)<=location(j)
            location(j)=location(j)+shiftw(k)-shiftaw(k);
        else
            end
        end
    end
    shiftaw(k)=shiftw(k);
end

A1=['FID1=fopen(""); A2=['.cir","w");]; B=[A1,file,A2]; eval(B); %FID1=fopen('file.cir','w');
fclose(FID1,F1); fclose(FID1);

% Ploteaza variabilele de stare pt o perioada din starea stationara
A1=['t=']; A2=['(:,1);']; B=[A1,file,A2]; eval(B); % t=file(:,1);

for k=1:length(index)
A1=['steady(:,k)=']; A2=['(:,index(k)+1);']; B=[A1,file,A2]; eval(B); % steady(:,k)=file(:,index(k)+1);
end
iL_st=steady(:,1); vc_st=steady(:,2); vo_st=steady(:,2);

M=0; j=2;
while ~(time(j)==0)&(j<=N)
    j=j+1;
end
M=j-1;
if M==0
    fprintf('Error: Dimensiunea initiala a variabilelor de interes este prea mica'); break;
else
end

time=time(1:M); iL=iL(1:M); vc=vc(1:M); vo=vo(1:M);
save digit t iL_st vo_st time iL vo;

subplot(2,1,1); plot(t,iL_st,'k'); title('Curentul prin bobina'); xlabel('t [s]'); ylabel('iL [A]');
subplot(2,1,2); plot(t,vc_st,'k'); title('Tensiunea pe capacitate (vo)'); xlabel('t [s]'); ylabel('vc [V]');

figure
subplot(3,1,1); plot(time,iL,'k'); title('Curentul prin bobina'); xlabel('t [s]'); ylabel('iL [A]');
subplot(3,1,2); plot(time,vc,'k'); title('Tensiunea pe capacitate'); xlabel('t [s]'); ylabel('vc [V]');
subplot(3,1,3); plot(time,vo,'k'); title('Tensiunea de iesire'); xlabel('t [s]'); ylabel('vo [V]');
toc

```

Fișierul *bb\_dig.cir* - fișier PETS

\* Convertor BUCK-BOOST cu controler numeric implementat in MATLAB

```

* Comportarea la salt in sarcina
* Controlerul este de tip I cu beta=20

Vg 1 0 DC 12
#vcs1 1 2 100 0 switch
#D 3 2 dioda

L 2 0 250u ic=anynumber
* switch control
vcom 100 0 pulse -2 10 0 width 20u

C 0 3 100u ic=anynumber
R1 3 0 2
#vcsload 3 4 200 0 switch
vcomload 200 0 pulse -2 10 width 0.5 1
R2 4 0 2

*models
.model dioda #d
.model switch #vcs

.tran 0.1u 20u uic
.probe i(L) v(3)

* relax tolerances
.options reitot=0.01 vntol=1e-4 abstol=1e-9
.options dispmode=ascii

.end

```

## A 29 Simularea unui convertor BUCK-BOOST cu controller fuzzy prin legarea programelor MATLAB și PETS

Fișierul *fuzycr1.m*

```

% Simularea unui controller FUZZY de tip I legand programele MATLAB si PETS.
% Comportarea unui convertor BUCK-BOOST cu controler FUZZY la un impuls in alimentare
% in conditiile in care tensiunea de referinta este constanta si egala cu 5V.
% Se pleaca direct din starea stationara aferenta starii cu Vg=15 (vezi vectorul xisim)
% In PETS se simuleaza o perioada, iar in MATLAB se calculeaza noul factor de umplere pt.
% perioada urmatoare, d(k+1), si furnizeaza fisierului 'cir' durata impulsului de
% comanda, egala evident cu d(k+1)*Ts, precum si valorile variabilelor de stare cu care
% sa inceapa simularea in perioada urmatoare.

% Cea care se modifica este durata impulsului din tensiunea de comanda.
% Se detecteaza pozitia in care apare acest parametru in comanda 'pulse' si apoi permanent
% el se actualizeaza.
% Pt. o detectie usoara in fisierul initial se va scrie la durata impulsului stringul 'width'

% Matricea F1 contine datele in binar ale fisierului analizat (de tip 'cir').
% Matricea 'newline' contine indicii din F1 unde incep liniile
% Matricea 'new' contine indicii din F1 unde incep liniile cu caractere diferite de blank sau TAB
% Matricea 'state' contine numarul liniilor si indicii din F1 unde sunt definite variabilele de stare (L,I, C, sau c)
% Matricea 'location' contine indicii din F1 unde incep definirile conditiilor initiale (ic=anynumber)
% Aceasta matrice ('location')se actualizeaza in urma fiecarei simulari deoarece noile valori
% pt variabilele de stare nu sunt scrise cu acelasi numar de cifre zecimale.
% Matricea 'locwidth' contine indicii din F1 unde incep definirile duratei impulsurilor (width)
% Si aceasta matrice se actualizeaza dupa fiecare simulare a unei perioade.
% Declaratiile 'anynumber' si 'width' trebuie scrise cu litere mici, fara spatiu intre
% litere si fara comentariu anterior (*) deoarece programul nu face aceste verificari.
% In linia 'probe' se vor scrie variabilele de stare IN ORDINEA IN CARE AU FOST ENUNTATE
% ELEMENTELE REACTIVE in fisier.

clear all;
fclose('all');
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

tic

% Deschide fisierul specificat, citeste datele binare in matricea F1, apoi inchide fisierul si transforma
% in string datele binare citite
file=input('Introduceti numele fisierului PETS cu cale si nume (pana la 8 caractere),fara extensie: ','s');
A1=['FID1=fopen("'); A2=['.cir","r");']; B=[A1,file,A2]; eval(B); %FID1=fopen('file.cir','w');
F1=fread(FID1); fclose(FID1); S1=setstr(F1);

```



```

% Gaseste indicii tuturor inceputurilor de linii (vectorul in care se scriu este 'newline')
j=2; newline(1)=1;
for i=1:length(F1)
    if F1(i)==10; % 10 este codul binar pt carriage return
        newline(j)=i+1; j=j+1;
    else
        end
    end
end
% Variabila NL contine numarul de linii din fisier
NL=length(newline);

% Gaseste indicii din F1 ai primelor elemente diferite de blank sau TAB
% din toate liniile, cu exceptia ultimei (vectorul in care se scriu este 'new')
for j=1:length(newline)-1
    k=0;
    while k<=(newline(j+1)-newline(j)-1)
        if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9) % 32 este codul binar pt. blank iar 9 pt TAB
            k=k+1;
        else
            break
        end
    end
    new(j)=newline(j)+k;
end

% Gaseste indexul din F1 al primului element diferit de blank sau TAB din ultima linie
j=length(newline);
k=0;
while k<=(length(F1)-newline(j)-1)
    if (F1(newline(j)+k)==32)|(F1(newline(j)+k)==9)
        k=k+1; % gaseste primele elemente diferite de blank
    else % sau TAB din ultima linie
        break
    end
end
new(j)=newline(j)+k;

% Daca ultimul caracter din fisier este carriage return atunci ultima linie nu are sens a
% se analiza pt. a gasi primul caracter diferit de blank sau TAB
if F1(length(F1))==10 l=length(new)-1;
else l=length(new);
end

% Scrie in vectorul 'state' indicii din F1 unde apar: L.I.C.c
% si care apar ca prime elemente pe linii
k=1;
for j=1:l
    if (F1(new(j))==76)|(F1(new(j))==108)|(F1(new(j))==67)|(F1(new(j))==99) % 76 este codul pt L,
        state(k,1)=j; state(k,2)=new(j); k=k+1; % 108 pt l,
    else % 67 pt C
        end % 99 pt c
end
Nstates=length(state);

% Matricea de verificare V1
for k=1:Nstates
    V1(k)=setstr(F1(state(k,2)));
end
V1';

% Gaseste indexul din F1 de unde incepe valoarea conditiei initiale
target1='ic=anynumber'; % vectorul de comparare
for k=1:length(state(:,1))
    flag=0;
    for j=0:newline(state(k,1)+1)-state(k,2)-length(target1)

        for t=1:length(target1)
            block1(1,t)=S1(state(k,2)+j+(t-1)); % defineste stringul de comparare
        end

        comp1=(block1==target1);
        if sum(comp1)==12 location(k)=state(k,2)+j; flag=1;
        else
            j=j+1;
        end
    end
end

```

```

end
if flag==0
fprintf('Eroare: Conditia initiala in forma "ic=anynumber" nu este definita in linia %i\n',state(k,1));
else
end
end

if flag==0 break
else
end

% Caseste indexul din F1 de unde incepe valoarea conditiei initiale
target2='width'; flag=0; % vectorul de comparare si semnalizarea gasirii stringului
j=1;
for k=1:length(S1)-length(target2)
    for t=1:length(target2)
        block2(1,t)=S1(k+t-1); % defineste stringul curent de comparare
    end

    comp2=(block2==target2);
    if sum(comp2)==5 locwidth(j)=k; j=j+1; flag=1;
    else
        k=k+1;
    end
end
if flag==0
    fprintf('Eroare: Conditia initiala in forma "width" nu a fost gasita'); break;
else
end

% Defineste:
% - starea initiala din care pleaca sistemul
% - acei parametri ai intrarilor care nu pot fi definiti in fisierul *.cir (aici dk_1.)
% - parametrii similarii
% - initializari (timpul curent de simulare, marimi dorite a se afisa, etc)
xisim=[0.4810; 4.9990]; % starea initiala
% Intrari (Ts, factor de umplere anterior si actual, regulator)
Ts=10e-6; dk_1=0.2549; width=[10e-6;dk_1*Ts]; tsalt1=2e-3; tsalt2=8e-3;
% parametrii regulatorului
Vref=5;
be=5; % factorul de normalizare al erorii (in volti). Domeniul controlat este: Vref-be<Vo<Vref+be
bde=0.015; % factorul de normalizare al derivatei erorii (in volti)
eta=0.01; % castigul controlerului
d_max=0.99; d_min=0.01; % limitarile in factor de umplere
Vok_1=5.0014; Vok=5.0014;
% functiile de apartenenta
% intrarile e si de (eroara si derivata erorii)
NB = mf_tri_p([-1 -1 -0.5]);
NS = mf_tri_p([-1 -0.5 0]);
ZE = mf_tri_p([-0.5 0 0.5]);
PS = mf_tri_p([0 0.5 1]);
PB = mf_tri_p([0.5 1 1]);

% variabilele lingvistice
X1 = [PB; PS; ZE; NS; NB]; % variabilele lingvistice pt e
X2 = [NB; NS; ZE; PS; PB]; % variabilele lingvistice pt de

% Tabelul de reguli. X1 se va referi
% la de, iar X2 la e, .
% Deoarece functia fc_mam_1 cere ca prim argument variabila de pe abscisa si
% ca al doilea cea de pe ordonata apelarea se va face cu e si de permutate,
% adica fc_mam_1(de,e,X2,X1,RT)
RT = [-0.30 -0.35 -0.45 -0.65 -1.00
0.00 -0.10 -0.20 -0.35 -0.50
0.20 0.10 0.00 -0.10 -0.20
0.50 0.35 0.20 0.10 0.00
1.00 0.65 0.45 0.35 0.30];

% Parametrii similarii
tmax=16e-3; %eps=[1e-5; 1e-5]; eps=1e-3*ones(Nstates,1);
% Initializari
tcurrent=0; % initializeaza vectorul de timp curent de simulare
N=100*(tmax/Ts);
Vg=zeros(1,N); iL=zeros(1,N); vc=zeros(1,N); vo=zeros(1,N); time=zeros(1,N);

```

```

p=1; % contorul pt. variabile de interes

%Vg=15; iL=xisim(1,1); vc=xisim(2,1); vo=Vok; % initializeaza vectorii curenti ai variabilelor de stare
Xisim=xisim; Xfsim=xisim+eps+ones(Nstates,1); % pentru intrare sigura in bucla while de simulare
countper=0; % contorul de perioade

% Lanseaza o simulare numai pentru a obtine matricea 'index',
% in care sunt corelate ordinea enuntarii variabilelor de stare din linia probe cu cea
% facuta de PETS in fisierul *.nam.

% Extrage din fisierul F1 indicii corespunzatori conditiilor initiale vechi,
% insereaza pe cei corespunzatori conditiilor initiale noi (deoarece valorile de actualizare
% ale conditiilor initiale nu au acelasi numar de cifre si translateaza astfel locatiile)
% Condițiile initiale au 9 cifre semnificative dupa virgula (vezi linia comentata cu ***)

shiftax=9*ones(1,Nstates); %matricea de shiftare anterioara care aici are dimensiunea 9
% pt fiecare element reactiv (stringul 'anynumber' are 9 caractere)
%xisim=zeros(1,Nstates); % valoarea initiala a vectorului de stare de unde incepe simularea
for k=1:length(xisim)
    b=abs(sprintf('% 9f,xisim(k))); % ***
    shiftx(k)=length(b);
    F1prim=F1(1:location(k)+2,1); % scrie valorile lui xisim in fisier la conditiile initiale
    F1secund=F1(location(k)+3+shiftax(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(location)
        location(i)=location(i)+shiftx(k)-shiftax(k);
    end

    for j=1:length(locwidth)
        if location(k)<=locwidth(j)
            locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
        else
            end
        end
    end
    shiftax(k)=shiftx(k);
end

% Efectueaza acelasi lucru pt valorile de actualizare ale latimii impulsurilor
shiftaw=5*ones(1,Nstates); % pt. ca stringul 'width' are 5 caractere
for k=1:length(width)
    b=abs(sprintf('% 9f,width(k))); % *** valoarea pulsului se scrie cu 9 cifre dupa virgula
    shiftw(k)=length(b);
    F1prim=F1(1:locwidth(k)-1,1); % scrie valorile lui xisim in fisier la conditiile initiale
    F1secund=F1(locwidth(k)+shiftaw(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(width)
        locwidth(i)=locwidth(i)+shiftw(k)-shiftaw(k);
    end

    for j=1:length(location)
        if locwidth(k)<=location(j)
            location(j)=location(j)+shiftw(k)-shiftaw(k);
        else
            end
        end
    end
    shiftaw(k)=shiftw(k);
end

A1=[FID1=fopen(''); A2=['.cir','w']; B=[A1,file,A2]; eval(B); %FID1=fopen('file.cir','w');
fwrite(FID1,F1); fclose(FID1);

% Lanseaza simularea pentru a obtine vectorul 'index'
A1=['dos("pets "); A2=[""]; B=[A1,file,A2]; eval(B); %dos('pets file');
statdet;
pack;

% Starteaza simularea propriu-zisa
while tcurrent<tmxax
%while (countper<Nper)&any(abs(Xfsim-Xisim)>eps)
    if tcurrent<tsalt1
        width(1,1)=2e-3; % mai mare decat timpul de simulare scris la .tran (o perioada)
    elseif (tcurrent>=tsalt1)&(tcurrent<=tsalt2)
        width(1,1)=0;
    else
        width(1,1)=2e-3;
    end
end

```

```

% scrie ca valori initiale ale variabilelor de stare pe cele rezultate din simularea precedenta
for k=1:length(xisim)
    b=abs(sprintf('% 9f, xisim(k))); % ***
    shiftx(k)=length(b);
    F1prim=F1(1:location(k)+2,1); % scrie valorile lui xisim in fisier la conditiile initiale
    F1secund=F1(location(k)+3+shiftax(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(location)
        location(i)=location(i)+shiftx(k)-shiftax(k);
    end

    for j=1:length(locwidth)
        if location(k)<=locwidth(j)
            locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
        else
            end
        end
    shiftax(k)=shiftx(k);
end

for k=1:length(width)
    b=abs(sprintf('% 9f, width(k))); % *** valoarea pulsului se scrie cu 9 cifre dupa virgula
    shiftw(k)=length(b);
    F1prim=F1(1:locwidth(k)-1,1); % scrie valorile lui ton in fisier la definirea tensiunii(lor) de comanda
    F1secund=F1(locwidth(k)+shiftaw(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(width)
        locwidth(i)=locwidth(i)+shiftw(k)-shiftaw(k);
    end

    for j=1:length(location)
        if locwidth(k)<=location(j)
            location(j)=location(j)+shiftw(k)-shiftaw(k);
        else
            end
        end
    shiftaw(k)=shiftw(k);
end

A1=[FID1=fopen(""); A2=['.cir', 'w'];]; B=[A1,file,A2]; eval(B); %FID1=fopen('file.cir','w');
fwrite(FID1,F1); fclose(FID1);

% Lanseaza simularea unei perioade din bucla while
A1=['dos("pets "); A2=[""];]; B=[A1,file,A2]; eval(B); %dos('pets file');
A1=['load '); A2=['.txt;']; B=[A1,file,A2]; eval(B); %load file.txt;
%Se defineste timpul curent pt o perioada de simulare
A1=['t=']; A2=['(',1,');]; B=[A1,file,A2]; eval(B); %t=file(:,1)
A1=['varpets=']; A2=['(',2,length(''); A3=['(1,:););]; B=[A1,file,A2,file,A3]; eval(B); %varpets=file(:,2:length(file(1,:)));

    for k=1:length(varpets(1,:))
        var(:,k)=varpets(:,index(k)); % obtine matricea de variabile conforma cu linia 'probe'
    end
for i=1:length(t);
    time(p)=countper*Ts+t(i); % se actualizeaza vectorul de timp curent de simulare
    Vg(p)=var(i,1); iL(p)=var(i,2); vc(p)=var(i,4)-var(i,3); vo(p)=var(i,3); % variabilele curente
    % se esantioneaza pe intervalul de blocare, cu 5x0.1us (0.1us fiind
    % pasul de simulare) inainte de terminarea perioadei, pentru a nu
    % esantiona zgomotele de comutatie (de ex varfurile de tensiune pe
    % sarcina datorate lui rC)

p=p+1;
end
Vok=vo(p-6);
xfsim=[iL(p-1); vc(p-1)]; tcurrent=time(p-1);
% linia de mai sus se va inlocui cu linia '* tran' din *.cir pt alte expresii
% ale variabilelor de stare, linie prin care se calculeaza
% variabilelor de stare, importata din fisierul *.cir
% cu *.transform. Sau pur si simplu se stie expresia variabilei
% de stare respective si se scrie direct in fisierul MATLAB
% (ca aici vc - vezi linia de dupa end-ul de mai sus) fara a mai folosi linia '*tran'

%if any(abs(xfsim-xisim)>eps)
% Se implementeaza algoritmul de interpolare pt calculul starii stationare
% xiinterp=xfsim;
% Interpolarea. Se obtine xfinterp
% xfinterp=xfsim; % numai in cazul in care se doreste simulare perioada cu perioada, fara interpolare

```

```

% Se implementeaza controlerul fuzzy
ek = max(min((Vok-Vref)/be,1),-1); % eroarea curenta normalata si limitata intre -1 si 1
dek = max(min((Vok-Vok_1)/bde,1),-1); % derivata erorii curente, normalata si limitata intre -1 si 1
ddk= eta*fc_mam_1(dek, ek, X2, X1, RT); % incrementul factorului de umplere (iesirea regulatorului)
dk=max(min(dk_1+ddk, d_max), d_min); % formula cu limitare superioara si inferioara a factorului de umplere
%dk=dk_1+ddk; % formula fara limitare superioara si inferioara a factorului de umplere
width(2,1)=dk*Ts;
e=[e;ek]; de=[de;dek]; dd=[dd;ddk]; d=[d;dk];
dk_1=dk; Vok_1 = Vok;
xisim=xfsim;
%else
%Xisim=Xfsim;
%end

countper=countper+1;
clear var; % pt ca de la o simulare a alta nr. de elemente pe coloana difera, PETS-ul variind nr de puncte in jurul comutarii

end

% Scrie din nou 'ic=anynumber' in conditiile initiale si 'width' in sursele de comanda
% pentru ca fisierul sa fie pregatit pentru o noua rulare
b=['anynumber'];
for k=1:length(xfsim)
    shiftx(k)=length(b);
    F1prim=F1(1:location(k)+2,1);
    F1secund=F1(location(k)+3+shiftax(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(location)
        location(i)=location(i)+shiftx(k)-shiftax(k);
    end

    for j=1:length(locwidth)
        if location(k)<=locwidth(j)
            locwidth(j)=locwidth(j)+shiftx(k)-shiftax(k);
        else
            end
        end
    end
    shiftax(k)=shiftx(k);
end

b=['width'];
for k=1:length(width)
    shiftw(k)=length(b);
    F1prim=F1(1:locwidth(k)-1,1); % scrie valorile lui xisim in fisier la conditiile initiale
    F1secund=F1(locwidth(k)+shiftaw(k):length(F1),1);
    F1=[F1prim;b';F1secund];
    for i=k+1:length(width)
        locwidth(i)=locwidth(i)+shiftw(k)-shiftaw(k);
    end

    for j=1:length(location)
        if locwidth(k)<=location(j)
            location(j)=location(j)+shiftw(k)-shiftaw(k);
        else
            end
        end
    end
    shiftaw(k)=shiftw(k);
end

A1=['FID1=fopen(""); A2=['.cir','w']; B=[A1,file,A2]; eval(B); %FID1=fopen('file.cir','w');
fwrite(FID1,F1); fclose(FID1);

% Ploteaza variabilele de stare pt o perioada din starea stationara
A1=['t=']; A2=['(:,1)']; B=[A1,file,A2]; eval(B); % t=file(:,1);

for k=1:length(index)
    A1=['steady(:,k)=']; A2=['(:,index(k)+1)']; B=[A1,file,A2]; eval(B); % steady(:,k)=file(:,index(k)+1);
end
iL_st=steady(:,1); vc_st=steady(:,2); % trebuie stiuta pozitia in linia 'probe' a variabilelor dorite a se afisa

for i=1:countper
    disctime(i)=i*Ts;
end

M=0; j=2;

```

```

while ~(time(j)==0)&(j<=N)
    j=j+1;
end
M=j-1;
if M==0
    fprintf('Error: Valoarea initiala a dimensiunii variabilelor prea scurta'); break;
else
    end
time=time(1:M); Vg=Vg(1:M); iL=iL(1:M); vc=vc(1:M); vo=vo(1:M);
save fuzzycon disctime e ce dd d time Vg iL vc vo;

figure
subplot(411); stairs(disctime, e); title('eroarea (e)'); grid;
subplot(412); stairs(disctime, ce); title('derivata erorii (ce)'); grid;
subplot(413); stairs(disctime, dd); title('iesirea regulatorului (dd)'); grid;
subplot(414); stairs(disctime, d); title('factorul de umplere (d)'); grid;
figure;
subplot(411); plot(time,Vg,'k'); title('tensiunea de intrare (Vg)'); grid;
subplot(412); plot(time,iL,'k'); title('curentul prin bobina (iL)'); grid;
subplot(413); plot(time,vc,'k'); title('tensiunea pe capacitate (vo)'); grid;
subplot(414); plot(time,vo,'k'); title('tensiunea de iesire (vo)'); grid;
save fuzzycon disctime e ce dd d time Vg iL vc vo;
toc

```

*Fișierul fc\_mam1.m* - fișierul are comentariile în limba engleză, pentru ca să a păstrat forma în care a fost scris de mai mulți autori pentru rețeaua de calculatoare a catedrei

```

function y=fc_mam_1(x1, x2, X1, X2, RT)
% FC Mamdani, parametric MFs used by So, Tse & Lee
%
% x1 - first crisp input (scalar) for the first variable
% x2 - second crisp input (scalar) for the second variable
% X1 - antecedent MFs for x1 (matrix; each row represents a MF)
% X2 - antecedent MFs for x2 (matrix; each row represents a MF)
% Q - consequent MFs (matrix; each row represents a MF)
% RT - rule table (matrix)
% - column indices: terms for the first variable
% - row indices: terms for the second variable
% - elements: normalised singleton values of the change of duty cycle
% output: crisp output (scalar)
%
% Note: the following dimensions should agree (no checking is performed):
% - # rows of X1 = # columns of RT
% - # rows of X2 = # rows of RT
% - elements in RT >=0 and <= # rows in Q

[ntx2,ntx1] = size(RT); % ntx1 = # terms of x1
% ntx2 = # terms of x2

% Fuzzification, singleton

% Determine the vector alpha1 containing the membership degree of x1 to all
% the fuzzy sets defined for the first variable
for i = 1:ntx1 % index i selects all the fuzzy sets defined for the first variable
    % extract MF parameters. It goes up to A1(i,5) because the triangular and
    a = X1(i,1); % trapezoidal membership functions are defined by 5 parameters
    b = X1(i,2); % (see the files "mf_tri_p.m" and "mf_trp_p.m")
    c = X1(i,3);
    d = X1(i,4);
    h = X1(i,5);

    % compute the membership degree of x1 to the fuzzy set selected by index i for the first variable
    if x1 < a
        alpha1(i) = 0; % Note: the order of comparisons
    elseif x1 < b % is important, to avoid division
        alpha1(i) = h*(x1-a)/(b-a); % by zero, when computing alpha1(i),
    elseif x1 <= c % if b=a;
        alpha1(i) = h;
    elseif x1 < d
        alpha1(i) = h*(d-x1)/(d-c);
    else
        alpha1(i) = 0;
    end
end
end

```

```

% Determine the vector alpha2 containing the membership degree of x2 to all
% the fuzzy sets defined for the second variable
for j = 1:ntx2 % idem pt a doua intrare
    % extract MF parameters
    a = X2(j,1);
    b = X2(j,2);
    c = X2(j,3);
    d = X2(j,4);
    h = X2(j,5);

    % compute the membership degree of x2 to the fuzzy set selected by index j
    if x2 < a
        alpha2(j) = 0; % Note: see above;
    elseif x2 < b
        alpha2(j) = h*(x2-a)/(b-a);
    elseif x2 <= c
        alpha2(j) = h;
    elseif x2 < d
        alpha2(j) = h*(d-x2)/(d-c);
    else
        alpha2(j) = 0;
    end
end

% Defuzzification
numer = 0, denom = 0; % initialize numerator and denominator in the COG formula
for i = 1:ntx1 % loop over x1 terms
    for j = 1:ntx2 % loop over x2 terms
        Ci = RT(j,i); % fetch consequent degree of change, Ci
        wi = min(alpha1(i),alpha2(j)); % compute the weighting factor wi
        % update denominator and numerator
        numer = numer + wi*Ci;
        denom = denom + wi;
    end
end
y = numer/denom;

```

*Fișierul mf\_tri\_p.m* - fișierul are comentariile in limba engleza, pentru ca s-a pastrat forma in care a fost scris de mai multi autori pentru rețeaua de calculatoare a catedrei

```

function f=mf_tri_p(coord)
% MF generation, triangular, parametric
%
% coord=[a b c] - corner coordinates, as shown below
%
% 1 .
%  / \
% 0 ___/ \___
%   a b c
%
% output: triangular MF, (parametric representation)
%
% EXAMPLE:
% f = mf_tri_p([10 20 30]);
% plotmf_p(f)
%
% fill parameters
f(1) = coord(1);
f(2) = coord(2);
f(3) = coord(2);
f(4) = coord(3);
f(5) = 1; % heigth = 1, by default
end

```

*Fișierul bb\_fuzzy.cir*

- \* Convertor BUCK-BOOST cu controler FUZZY implementat in MATLAB
- \* Comportarea la un puls in tensiunea de alimentare
- \* Controlerul este analog cu un PI cu be=5, bce=0.015, eta=0.01

```

Vg 1 0 pulse 15 20 width 0.5 1
* Vg 1 0 DC 15

```

```

#vcs1 1 2 100 0 switch
#D 3 2 dioda

L 2 7 100u ic=anynumber
rL 7 0 0.1
* switch control
vcom 100 0 pulse -2 10 0 width 10u

C 6 3 100u ic=anynumber
rC 6 0 0.1
Rs 3 0 10
* #vcsload 3 4 200 0 switch
* vcomload 200 0 pulse -2 10 width 0.5 1
* R2 4 0 2

*models
.model dioda #d Ron=0.01
.model switch #vcs Ron=0.1

.tran 0.1u 10u uic
.probe v(1) i(L) v(3) v(6)

* relax tolerances
.options reltol=0.01 vntol=1e-4 abstol=1e-9
.options dispmode=ascii

.end

```

### A 30 Calculul și reprezentarea funcțiilor de transfer control-ieșire exacte pentru principalele moduri de funcționare și tipuri de comanda

Funcția de transfer control-ieșire exactă pentru un convertor SEPIC DCVM - *fișierul dcvm.m*

% Aplicarea metodei Newton-Raphson la calculul exact al stării staționare  
 % la un convertor dc-dc de tip DCVM SEPIC și funcția de transfer control-ieșire exactă.

clear % șterge variabile

Fer=4;

for i=1:Fer

close % șterge ultimele Fer ferestre grafice

end

format long;

D=0.6; Ts=20e-6; Dprime=1-D; fs=1/Ts; %comanda

Vg=12; L1=500e-6; L2=500e-6; C=100e-6; C1=220e-9; R=10; %elementele de circuit

Ron=1e-4; % se considera ca D e ideala iar S are Roff infinita și Ron nenula

% Matricile aferente celor trei stări topologice ale convertorului:

A1=[ -Ron/L1 -Ron/L1 0 0;

-Ron/L2 -Ron/L2 0 1/L2;

0 0 -1/(R\*C) 0;

0 -1/C1 0 0];

B1=[1/L1; 0; 0; 0]; E1=[0 0 1 0]; F1=[0];

A2=[ 0 0 -1/L1 -1/L1;

0 0 -1/L2 0;

1/C 1/C -1/C\*(1/R+1/Ron) -1/(C\*Ron);

1/C1 0 -1/(C1\*Ron) -1/(C1\*Ron)];

B2=[1/L1; 0; 0; 0]; E2=[0 0 1 0]; F2=[0];

A3=[ 0 0 -1/L1 -1/L1;

0 0 -1/L2 0;

1/C 1/C -1/(R\*C) 0;

1/C1 0 0 0];

B3=[1/L1; 0; 0; 0]; E3=[0 0 1 0]; F3=[0];

Ns=3; % numărul stărilor topologice în stare staționară

n=4; % numărul variabilelor de stare

In=eye(n);

l=eye(n\*(Ns+1))+Ns);

e1=l(1:4,:); % extrage pe X(T1)

e2=l(5:8,:); % extrage pe X(T2) % cele Ns+1 matrici extractoare ale

e3=l(9:12,:); % extrage pe X(T3) vectorilor de stare X(Ti), i=1,(Ns+1)

e4=l(13:16,:); % extrage pe X(T4)

e5=l(17,:); % extrage pe T1p

e6=l(18,:); % extrage pe T2p % cele Ns matrici extractoare



```
e7=1(19,:); % extrage pe T3p timpilor Tip=Ti+1-Ti
h=[-Ron -Ron 1 1]; % matricile extractoare ale componentelor din vectori ce satisfac relatia de stare (anulare)
```

```
% Se estimeaza cu modelul de semnal mare starea stationara aproximativa
kv=2*(C*C1/(C+C1))*R*fs;
Vo=Vg/(1-D)*sqrt(kv); VC=Vo; % marimile de stare
VC1=Vg; IL1=Vg/(1-D)^2*2*(C*C1/(C+C1))*fs; IL2=Vg/(1-D)*sqrt(2*(C*C1/(C+C1))*fs/R);
```

```
D1=sqrt(kv) ;
T1p=D1*Ts; T2p=(D-D1)*Ts; T3p=(1-D)*Ts; % timpii
```

```
Xmed=[IL1; IL2; VC; VC1;]; T=[T1p; T2p; T3p]
xstart=[Xmed; Xmed; Xmed; Xmed; T;];
% xstart=[2.5*ones((Ns+1)*n,1); D*Ts*ones(Ns,1)];
xa=xstart; xc=zeros(n*(Ns+1)+Ns,1); flag=ones(1,n*(Ns+1)+Ns);
Xa=xstart; Xc=xc; err=[1e-9*ones((Ns+1)*n,1); 1e-12*ones(Ns,1)];
iter=0;
while ~(Xa==Xc)
    f1=(e2*xa)-(expm(A1*(e5*xa))*(e1*xa)+inv(A1)*(expm(A1*(e5*xa))-ln)*B1*Vg);
    f2=(e3*xa)-(expm(A2*(e6*xa))*(e2*xa)+inv(A2)*(expm(A2*(e6*xa))-ln)*B2*Vg); % cele Ns relatii de recurenta
    f3=(e4*xa)-(expm(A3*(e7*xa))*(e3*xa)+inv(A3)*(expm(A3*(e7*xa))-ln)*B3*Vg);
```

```
f4=(e4-e1)*xa; % relatia care egaleaza vectorul de stare de la inceputul si sfarsitul perioadei
```

```
f5=h*(e2*xa);
f6=(e5+e6+e7)*xa-Ts; % relatiile ce se refera la timpii
f7=e7*xa-(1-D)*Ts;
fxa=[f1;f2;f3;f4;f5;f6;f7];
```

```
f1prim=e2-(A1*expm(A1*(e5*xa))*(e1*xa)*e5+expm(A1*(e5*xa))*e1+expm(A1*(e5*xa))*B1*Vg*e5);
f2prim=e3-(A2*expm(A2*(e6*xa))*(e2*xa)*e6+expm(A2*(e6*xa))*e2+expm(A2*(e6*xa))*B2*Vg*e6);
f3prim=e4-(A3*expm(A3*(e7*xa))*(e3*xa)*e7+expm(A3*(e7*xa))*e3+expm(A3*(e7*xa))*B3*Vg*e7);
```

```
f4prim=e4-e1;
```

```
f5prim=h*e2;
f6prim=e5+e6+e7;
f7prim=e7;
fprimxa=[f1prim;f2prim;f3prim;f4prim;f5prim;f6prim;f7prim];
```

```
xc=xa-inv(fprimxa)*fxa; iter=iter+1; % procesul iterativ Newton
```

```
dif=abs(xc-xa);
for i=1:n*(Ns+1)+Ns;
    if dif(i)>err(i) flag(i)=1;
    else flag(i)=0;
    end
end
m=sum(flag);
if ~(m==0), xa=xc;
    else Xa=Xc;
    end
end
X=xc, iter,
```

```
% Calculul functiei de transfer control-iesire exacte
k=401; % numarul de puncte de pe axa frecventei in care se reprezinta caracteristicile
f=logspace(1,6,k); % axa frecventei de la 10^0 la 10^5, logaritmica
w=2*pi*f; % definire omega
```

```
T1p=X((Ns+1)*n+Ns-(Ns-1),1); T2p=X((Ns+1)*n+Ns-(Ns-2),1); T3p=X(((Ns+1)*n+Ns-(Ns-3),1); % calculii timpilor stationari
[phi1, psi1]=c2d(A1,B1,T1p); % modelul analogic in spatiul starilor se
[phi2, psi2]=c2d(A2,B2,T2p);
[phi3, psi3]=c2d(A3,B3,T3p); % converteste in model discret
```

```
XT1=X(1:n,1); % valoarea din stare stationara la T1
XT2=X(n+1:2*n,1); % valoarea din stare stationara la T2
XT3=X(2*n+1:3*n,1);
XT4=XT1;
```

```
gamma1=A1*XT2+B1*Vg; %matricile necesare pt. perturbatii
gamma2=A2*XT3+B2*Vg;
gamma3=A3*XT4+B3*Vg;
```

```
csi1=(A1-A2)*XT2+(B1-B2)*Vg;
csi2=(A2-A3)*XT3+(B2-B3)*Vg;
```

```

zeta1=(E1-E2)*XT2+(F1-F2)*Vg;
zeta2=(E2-E3)*XT3+(F2-F3)*Vg;

h=[0 0 1 1]; % matricea extractoare
miu1=inv(h*gamma1);
Gamma1=ln-miu1*csi1*h;

% Bucla principala pentru calculul functiei de transfer cu formula exacta si cu modelul mediat
for m=1:k
    beta1=expm(-j*w(m)*T1p)*phi1;
    beta2=expm(-j*w(m)*T2p)*phi2;
    beta3=expm(-j*w(m)*T3p)*phi3; % Aici se definesc toti coeficientii care depind de w
    alpha1=E1*(j*w(m)*ln-A1)/(ln-beta1);
    alpha2=E2*(j*w(m)*ln-A2)/(ln-beta2);
    alpha3=E3*(j*w(m)*ln-A3)/(ln-beta3);
    hex(m)=(alpha3+alpha1*beta3+alpha2*Gamma1*beta1*beta3-zeta1*miu1*h*beta1*beta3)*inv(ln-beta2*Gamma1*beta1*beta3)*csi2+zeta2;

% Functia de transfer folosind modelul mediat propus de Maksimovic
M=(1/(1-D))*sqrt(kv); miu0=M/(M+1); lon=iL1+iL2;
ac=2*miu0*(1-miu0)/(1-D);
ai=-miu0*(1-miu0)/lon;
av=kv/R*(1-miu0)^2/(1-D)^2*1/lon;

Amiu=miu0*A1+(1-miu0)*A3; % Atentie! A1 si A3 sunt matricile aferente starilor ce se regasesc si in CCM.
Bmiu=miu0*B1+(1-miu0)*B3; % In starea aferenta lui A2 atat S cat si D conduc, deci nu e stare pt CCM
Emiu=miu0*E1+(1-miu0)*E3;
Fmiu=miu0*F1+(1-miu0)*F3;
Xmiu=inv(Amiu)*Bmiu*Vg;
Jmiu=(A1-A3)*Xmiu+(B1-B3)*Vg;
Hmiu=(E1-E3)*Xmiu+(F1-F3)*Vg;

E11=[1 1 0 0]; F11=[0]; % matrici necesare pt calculul fdt de la miu la iL=iL1+iL2
E13=[1 1 0 0]; F13=[0];
E1=miu0*E11+(1-miu0)*E13;
F1=miu0*F11+(1-miu0)*F13;
H1=(E11-E13)*Xmiu+(F11-F13)*Vg; % matricile X si J sunt aceleasi cu cele pe calculul lui vo/miu

Eap1=[0 0 1 1]; Fap1=[0]; % matrici necesare pt calculul fdt de la miu la vap=vC+vC1
Eap3=[0 0 1 1]; Fap3=[0];
Eap=miu0*Eap1+(1-miu0)*Eap3;
Fap=miu0*Fap1+(1-miu0)*Fap3;
Hap=(Eap1-Eap3)*Xmiu+(Fap1-Fap3)*Vg; % matricile X si J sunt aceleasi cu cele pe calculul lui vo/miu

hmiu(m)=Emiu*inv(j*w(m)*ln-Amiu)*Jmiu+Hmiu;
hmiu_il(m)=E1*inv(j*w(m)*ln-Amiu)*Jmiu+H1;
hmiu_vap(m)=Eap*inv(j*w(m)*ln-Amiu)*Jmiu+Hap;

hjos(m)=ac*hmiu(m)/(1-ai*hmiu_il(m)-av*hmiu_vap(m));
end
magex=20*log10(abs(hex));
phaseex=180/pi*unwrap(angle(hex));
magjos=20*log10(abs(hjos));
phasejos=180/pi*unwrap(angle(hjos));

semilogx(f,magex,'k'); hold on;
semilogx(f,magjos,'k-'); hold on; %axis([1 10^4 -200 50])
title('Convertor SEPIC PWM DCVM')
xlabel('Frecventa [Hz]'); ylabel('Amplitudine [dB]');

figure;
semilogx(f,phaseex,'k'); hold on;
semilogx(f,phasejos,'k-'); hold on; %axis([1 10^4 -200 50])

title('Convertor SEPIC PWM DCVM')
xlabel('Frecventa [Hz]'); ylabel('Faza [°]');

```

**Funcția de transfer control-ieșire exactă pentru un convertor BUCK DIVM - fișierul divm.m**

```

% Functia de transfer control-iesire exacta
% a unui convertor dc-dc de tip BUCK in DIVM.
% Se aplica metoda Newton-Raphson pt. calculul exact al starii stationare,
% necesara in estimarea functiei de transfer control-iesire exacta.

clear % sterge variabile
Fer=4;

```

```

for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end
format long;

D=0.345; Ts=30e-6; Dprime=1-D; fs=1/Ts; %comanda
Vg=15; L=212e-6; L1=970e-6; C=2659e-6; C1=104e-9; R=5.2; %elementele de circuit
Ron=1e-4; % se considera ca S e ideal iar D are Roff infinita si Ron nenula

% Matricile aferente celor trei stari topologice ale convertorului:
A1=[ 0 0 -1/L 1/L;
    0 0 0 -1/L1;
    1/C 0 -1/(R*C) 0;
    -1/C1 1/C1 0 0 ];
B1=[0; 1/L1; 0; 0]; E1=[0 0 1 0]; F1=[0];

A2=[ 0 0 -1/L 1/L;
    0 0 0 -1/L1;
    1/C 0 -1/(R*C) 0;
    -1/C1 1/C1 0 -1/(Ron*C1)];
B2=[0; 1/L1; 0; 0]; E2=[0 0 1 0]; F2=[0];

A3=[-Ron/L 0 -1/L 0;
    0 0 0 -1/L1;
    1/C 0 -1/(R*C) 0;
    0 1/C1 0 0 ];
B3=[0; 1/L1; 0; 0]; E3=[0 0 1 0]; F3=[0];

Ns=3; % numarul starilor topologice in stare stationara
n=4; % numarul variabilelor de stare
In=eye(n);
I=eye(n*(Ns+1))+Ns);

e1=l(1:4,:); % extrage pe X(T1)
e2=l(5:8,:); % extrage pe X(T2) % cele Ns+1 matrici extractoare ale
e3=l(9:12,:); % extrage pe X(T3) vectorilor de stare X(Ti), i=1,(Ns+1)
e4=l(13:16,:);% extrage pe X(T4)

e5=l(17,:); % extrage pe T1p
e6=l(18,:); % extrage pe T2p % cele Ns matrici extractoare
e7=l(19,:); % extrage pe T3p timpilor Tip=Ti+1-Ti
h=[0 0 0 1]; % matricile extractoare ale componentelor din vectori ce satisfac relatia de stare (anulare)

% Se estimeaza cu modelul de semnal mare starea stationara aproximativa
b=Ts/(2*C1*R)*(1-D)^2;
Vo=Vg*2/(1+sqrt(1+4*b)); VP=2*Vg/(1-D)*(1-2/(1+sqrt(1+4*b)));
VC=Vo; VC1=VP; IL=VC/R; IL1=Vo^2/(R*Vg);
D1=(1-D)*(1+sqrt(1+4*b))/(2*b);
T1p=D1*Ts; T2p=(D-D1)*Ts; T3p=(1-D)*Ts;

Xmed=[IL; IL1; VC; VC1]; T=[T1p; T2p; T3p]
xstart={Xmed; Xmed; Xmed; Xmed; T};
xa=xstart, xc=zeros(n*(Ns+1)+Ns,1); flag=ones(1,n*(Ns+1)+Ns);
Xa=xstart; Xc=xc; err=[1e-9*ones((Ns+1)*n,1); 1e-12*ones(Ns,1)];
iter=0;
while ~(Xa==Xc)
    f1=(e2*xa)-(expm(A1*(e5*xa))*(e1*xa)+inv(A1)*(expm(A1*(e5*xa))-In)*B1*Vg);
    f2=(e3*xa)-(expm(A2*(e6*xa))*(e2*xa)+inv(A2)*(expm(A2*(e6*xa))-In)*B2*Vg); % cele Ns relatii de recurenta
    f3=(e4*xa)-(expm(A3*(e7*xa))*(e3*xa)+inv(A3)*(expm(A3*(e7*xa))-In)*B3*Vg);

    f4=(e4-e1)*xa; % relatia care egaleaza vectorul de stare de la inceputul si sfarsitul perioadei

    f5=h*(e2*xa);
    f6=(e5+e6+e7)*xa-Ts; % relatiile ce se refera la timp
    f7=e7*xa-(1-D)*Ts;
    fxa=[f1;f2;f3;f4;f5;f6;f7];

    f1prim=e2-(A1*expm(A1*(e5*xa))*(e1*xa)+e5+expm(A1*(e5*xa))*e1+expm(A1*(e5*xa))*B1*Vg*e5);
    f2prim=e3-(A2*expm(A2*(e6*xa))*(e2*xa)+e6+expm(A2*(e6*xa))*e2+expm(A2*(e6*xa))*B2*Vg*e6);
    f3prim=e4-(A3*expm(A3*(e7*xa))*(e3*xa)+e7+expm(A3*(e7*xa))*e3+expm(A3*(e7*xa))*B3*Vg*e7);

    f4prim=e4-e1;

    f5prim=h*e2;
    f6prim=e5+e6+e7;
    f7prim=e7;

```

```

fprimxa=[f1 prim;f2prim;f3prim;f4prim;f5prim;f6prim;f7prim];

xc=xa-inv(fprimxa)*fxa; iter=iter+1;    % procesul iterativ Newton

dif=abs(xc-xa);
for i=1:n*(Ns+1)+Ns,
    if dif(i)>err(i) flag(i)=1;
    else flag(i)=0;
    end
end
m=sum(flag);
if ~(m==0), xa=xc;
else Xa=Xc;
end
end
X=xc, iter,

% Calculul functiei de transfer control-iesire exacte
k=401; % numarul de puncte de pe axa frecventei in care se reprezinta caracteristicile
f=logspace(0,5,k); % axa frecventei de la 10^0 la 10^5, logaritmica
w=2*pi*f; % definire omega

T1p=X((Ns+1)*n+Ns-(Ns-1),1); T2p=X((Ns+1)*n+Ns-(Ns-2),1);; T3p=X((Ns+1)*n+Ns-(Ns-3),1);;    % calculi timpilor stationari
[phi1, psi1]=c2d(A1,B1,T1p); % modelul analogic in spatiul starilor se
[phi2, psi2]=c2d(A2,B2,T2p);
[phi3, psi3]=c2d(A3,B3,T3p); % converteste in model discret

XT1=X(1:n,1); % valoarea din stare stationara la T1
XT2=X(n+1:2*n,1); % valoarea din stare stationara la T2
XT3=X(2*n+1:3*n,1);
XT4=XT1;

gamma1=A1*XT2+B1*Vg;    %matricile necesare pt. perturbatii
gamma2=A2*XT3+B2*Vg;
gamma3=A3*XT4+B3*Vg;

csi1=(A1-A2)*XT2+(B1-B2)*Vg;
csi2=(A2-A3)*XT3+(B2-B3)*Vg;
zeta1=(E1-E2)*XT2+(F1-F2)*Vg;
zeta2=(E2-E3)*XT3+(F2-F3)*Vg;

h=[0 0 0 1];    % matricea extractoare
miu1=inv(h*gamma1);
Gamma1=ln-miu1*csi1*h;

% Bucla principala pentru calculul functiei de transfer cu formula exacta
for m=1:k
    beta1=expm(-j*w(m)*T1p)*phi1;
    beta2=expm(-j*w(m)*T2p)*phi2;
    beta3=expm(-j*w(m)*T3p)*phi3; % Aici se definesc toti coeficientii care depind de w
    alpha1=E1*(j*w(m)*In-A1)/(ln-beta1);
    alpha2=E2*(j*w(m)*In-A2)/(ln-beta2);
    alpha3=E3*(j*w(m)*In-A3)/(ln-beta3);
    hex(m)=(alpha3+alpha1*beta3+alpha2*Gamma1*beta1*beta3-zeta1*miu1*h*beta1*beta3)*inv(ln-beta2*Gamma1*beta1*beta3)*csi2+zeta2;

    P1=-(2*D1*L1*Vg)/((1-D)*(1-D+D1)); P0=Ts*D1*Vg/C1;
    K3=L*L1*C; K2=L*L1/R+Ts*D1^2*L1*C/(2*C1)+Ts*L*C/(2*C1)*(1-D+D1)^2;
    K1=L+Ts/(2*R*C1)*(D1^2*L1+(1-D+D1)^2*L)+(Ts/(2*C1))^2*D1^2*(1-D)^2*C;
    K0=Ts/(2*C1)*(1-D+D1)^2+(Ts/(2*C1))^2*D1^2*(1-D)^2/R;
    hjos(m)=(P1*(j*w(m))+P0)/(K3*(j*w(m))^3+K2*(j*w(m))^2+K1*(j*w(m))+K0);
end
magex=20*log10(abs(hex));
phaseex=180/pi*unwrap(angle(hex));
magjos=20*log10(abs(hjos));
phasejos=180/pi*unwrap(angle(hjos));

semilogx(f,magex,'k-'); hold on;
semilogx(f,magjos,'r'); hold on; %axis([1 10^4 -200 50])
title('Converter BUCK PWM DIVM')
xlabel('Frecventa [Hz]'); ylabel('Amplitudine [dB]');

figure;
semilogx(f,phaseex,'k-'); hold on;
semilogx(f,phasejos,'r');hold on; %axis([1 10^4 -200 50])

title('Converter BUCK PWM DIVM')

```

```
xlabel('Frecventa [Hz]'); ylabel('Faza [°]');
```

### Funcția de transfer control-ieșire exactă pentru un convertor BOOST PWM CCM cu feedforward - *fișierul ff\_ccm.m*

```
% Functia de transfer control-iesire exacta la un convertor BOOST PWM CCM feedforward.
% Se considera feedforward clasic si feedforward integrativ (Arbeter)
% Comparatie cu rezultatele din modelul mediat in spatiul starilor propus Arbeter

format long
clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

k=401; % numarul de puncte de pe axa frecventei in care se reprezinta caracteristicile
f=logspace(1,6,k); % axa frecventei de la 10^-1 la 10^5, logaritmica
w=2*pi*f; %definire omega
j=sqrt(-1);
% Marimile de circuit:
fs=50e3; Ts=1/fs; Vm=4; A=12.5; %comanda
Vg=35; Vo=50; R=250; C=10e-6; L=718e-6; rL=0.4; Ron=0.7; % (pt. tranz.) elementele de circuit
D=1-Vg/Vo; Dprime=1-D;
% Matricile aferente celor doua stari topologice ale convertorului:
A1=[ -(rL+Ron)/L    0;
      0    -1/(R*C)];
A2=[ -rL/L    -1/L;
      1/C    -1/(R*C)];
B1=[1/L; 0];
B2=[1/L; 0];
E1=[0 1];
E2=[0 1];
F1=[0];
F2=[0];
I=eye(2);
p1=0; p0=1; q1=-1; q0=1; % coeficientii polinoamelor P si Q dupa care se face feedforward

% Matricile si toate marimile ce nu depind de w, necesare in calculul functiei exacte
T1=0; T2=D*Ts; T3=Ts;
[phi1, psi1]=c2d(A1,B1,(T2-T1)); %modelul analogic in spatiul starilor se
    % converteste in model discret
[phi2, psi2]=c2d(A2,B2,(T3-T2));

XT1=inv(I-phi2*phi1)*(phi2*psi1+psi2)*Vg; % valoarea din starea stationara la kTs
XT2=inv(I-phi1*phi2)*(phi1*psi2+psi1)*Vg; % valoarea din starea stationara la kTs+DTs

Phi=phi2*phi1; % matricea de la recurenta, a nu se confunda cu phi de la comanda.
csi1=(A1-A2)*XT2+(B1-B2)*Vg;
zeta1=(E1-E2)*XT2+(F1-F2)*Vg;

% Bucla principala pentru calculul functiei de transfer
for n=1:k
    phi=w(n)*D*Ts;
    eps1=1;
    eps2=(sin(phi/2))/(phi/2)*exp(-j*phi/2);
    const1=1/(p1*Vg/A-q1*Vm)*exp(j*phi)*(q0+q1*D*eps1);
    const2=1/(p1*Vg/A-q1*Vm)*exp(j*phi)*(q0+q1*D*eps2);
    beta1=exp(-j*w(n)*D*Ts)*phi1; % bucla pt. calculul functiei de transfer.
    beta2=exp(-j*w(n)*Dprime*Ts)*phi2; % Aici se definesc toticoeficientii care depind de w
    a1=E1*inv(j*w(n)*I-A1)*(I-beta1);
    a2=E2*inv(j*w(n)*I-A2)*(I-beta2);
    hexmul(n)=const1*((a1+a2*beta1)*exp(-j*w(n)*Ts)*inv(I-beta2*beta1)*phi2*csi1+(zeta1+a2*csi1)*exp(-j*phi));
    hexint(n)=const2*((a1+a2*beta1)*exp(-j*w(n)*Ts)*inv(I-beta2*beta1)*phi2*csi1+(zeta1+a2*csi1)*exp(-j*phi));
end
magexmul=20*log10(abs(hexmul));
phaseexmul=180/pi*unwrap(angle(hexmul));
magexint=20*log10(abs(hexint));
phaseexint=180/pi*unwrap(angle(hexint));

% Functia de transfer utilizand modelul mediat pt convertor (Kassakian)
AD=D*A1+Dprime*A2;
BD=D*B1+Dprime*B2;
ED=D*E1+Dprime*E2;
FD=D*F1+Dprime*F2;
X=-inv(AD)*BD*Vg;
```

```

J=(A1-A2)*X+(B1-B2)*Vg;
H=(E1-E2)*X+(F1-F2)*Vg;
for i=1:k
    have(i)=(q1*D+q0)/(p1*Vg/A-q1*Vm)*(ED*inv(j*w(i)*I-AD)*J+H); % bucla pt. calculul functiei.
end
magave=20*log10(abs(have));
phaseave=180/pi*unwrap(angle(have));

difmag1=magexmul-magexint; % diferenta dintre spectrele de amplitudine
difphase1=phaseexmul-phaseexint; % si de faza intre modelul exact si cel integrativ, respectiv mediat
difmag2=magexmul-magave;
difphase2=phaseexmul-phaseave;

% plot the magnitude (amplitude)and phase
semilogx(f,magexmul,'k');hold on;
semilogx(f,magexint,'r');hold on;
semilogx(f,magave,'g');
title('Convertor BOOST PWM CCM cu feedforward')
xlabel('Frecventa [Hz]'); ylabel('Amplitudine [dB]');

figure;
semilogx(f,phaseexmul,'k'); hold on
semilogx(f,phaseexint,'r');
semilogx(f,phaseave,'g');
title('Convertor BOOST PWM CCM cu feedforward')
xlabel('Frecventa [Hz]'); ylabel('Faza [°]');

```

### Funcția de transfer control-ieșire exactă pentru un convertor BUCK PWM CCM cu comanda prin sarcina - *fișierul charge.m*

```

% Functia de transfer control iesire exacta la un convertor BUCK cu comanda prin sarcina

clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

k=401; % numarul de puncte de pe axa frecventei in care se reprezinta caracteristicile
f=logspace(2,6,k); % axa frecventei de la 10^2 la 10^5, logaritmica
w=2*pi*f; % definire omega

% Marimile de circuit:
Vg=12; Vo=5; D=Vo/Vg; Dprime=1-D; fs=90e3; Ts=1/fs; %comanda
L=37.5e-6; R=3*L*fs; C=380e-6; Rc=20e-3; CT=733e-9; Ron=0.1; %elementele de circuit

flag=0;
t1=ceil(2*f(length(f))/fs)+1;
for i=1:length(f)
    for l=1:t1
        if f(i)==l*fs/2
            flag=1;
        else
            end
        end
    end
end

if flag==1
    error('Vectorul frecventa impropriu ales - exista valori egale cu p*fs/2, cu p numar natural ')
else
    end
end

% Matricile aferente celor doua stari topologice ale convertorului:
A1=[ -1/L*(R*Rc/(R+Rc)) -R/(L*(R+Rc)) 0;
     1/C*(R/(R+Rc)) -1/(C*(R+Rc)) 0;
     1/CT 0 0];
A2=[ -1/L*(R*Rc/(R+Rc)) -R/(L*(R+Rc)) 0;
     1/C*(R/(R+Rc)) -1/(C*(R+Rc)) 0;
     0 0 -1/(Ron*CT)];

B1=[1/L; 0; 0];
B2=[0; 0; 0];
E1=[R*Rc/(R+Rc) R/(R+Rc) 0];
E2=[R*Rc/(R+Rc) R/(R+Rc) 0];
F1=[0];
F2=[0];
I=eye(3);

```

```

% Matricile si toate marimile ce nu depind de w, necesare in calcul (dupa prof.Tymerski)
T1=0; T2=D*Ts; T3=Ts; % calculul timpilor stationari
[phi1, psi1]=c2d(A1,B1,(T2-T1)); % modelul analogic in spatiul starilor se
[phi2, psi2]=c2d(A2,B2,(T3-T2)); % converteste in model discret

XT1=(1-phi2*phi1)*(phi2*psi1+psi2)*Vg; % valoarea din stare stationara la T1
XT2=(1-phi1*phi2)*(phi1*psi2+psi1)*Vg; % valoarea din stare stationara la T2
XT3=XT1; % valoarea din stare stationara la T3
gamma1=A1*XT2+B1*Vg; %matricile necesare pt. perturbatii
gamma2=A2*XT3+B2*Vg;
csi1=(A1-A2)*XT2+(B1-B2)*Vg;
zeta1=(E1-E2)*XT2+(F1-F2)*Vg;
h1=[0 0 1]; % matricea extractoare
miu1=inv(h1*gamma1);
Gamma1=I-miu1*csi1*h1;

AD=D*A1+Dprime*A2; % matricile necesare pt. functia de transfer
ED=D*E1+Dprime*E2; % la joasa frecventa obtinuta din cea exacta

% Bucla principala pentru calculul functiei de transfer cu formula exacta
for n=1:k
    beta1=expm(-j*w(n)*(T2-T1))*phi1;
    beta2=expm(-j*w(n)*(T3-T2))*phi2; % Aici se definesc toti coeficientii care depind de w
    alpha1=E1*(j*w(n)*I-A1)\(I-beta1);
    alpha2=E2*(j*w(n)*I-A2)\(I-beta2);
    hex(n)=(miu1/Ts)*(alpha1*beta2+alpha2-zeta1*miu1*h1*beta1*beta2)/(I-Gamma1*beta1*beta2)*csi1+zeta1);
end
magex=20*log10(abs(hex));
phaseex=180/pi*unwrap(angle(hex));

% Functia de transfer control iesire folosind modelul mediat Ridley si modelul generalizat pt convertorul PWM de putere
Fm=CT/(Vg*(D/R+D*Dprime*(2*L*fs))*Ts);
Ri=D*Ts/CT; wn=pi/Ts; Qz=-2/pi; Kr=(D*Dprime*Ts^2)/(2*L*CT); %parametrii functiei
Ad1=[-1/L*(R*Rc/(R+Rc)) -R/(L*(R+Rc)); 1/C*(R/(R+Rc)) -1/(C*(R+Rc))]; % matricile numai
Ad2=[-1/L*(R*Rc/(R+Rc)) -R/(L*(R+Rc)); 1/C*(R/(R+Rc)) -1/(C*(R+Rc))]; % pt. convertor (2X2)
Bd1=[1/L; 0; ];
Bd2=[0; 0; ];
Ed1=[R*Rc/(R+Rc) R/(R+Rc)];
Ed2=[R*Rc/(R+Rc) R/(R+Rc)];
Fd1=[0];
Fd2=[0];
I2=eye(2);
AD2=D*Ad1+Dprime*Ad2;
BD2=D*Bd1+Dprime*Bd2;
ED2=D*Ed1+Dprime*Ed2;
X=-inv(AD2)*BD2*Vg;
csid=(Ad1-Ad2)*X+(Bd1-Bd2)*Vg;
zetad=(Ed1-Ed2)*X+(Fd1-Fd2)*Vg;
h2=[1 0]; % matricea extractoare a lui iL din x pt convertor

for m=1:k
    He(m)=1+j*w(m)/(wn*Qz)+(j*w(m))^2/(wn^2);
    Hd(m)=ED2*inv(j*w(m)*I2-AD2)*csid+zeta;
    havegen(m)=(Fm*Hd(m))/(1+Fm*Ri*He(m)*h2/(j*w(m)*I2-AD2)*csid-Fm*Kr*Hd(m)); % calculul la modul general, care nu tine cont de tipul de
    convertor
end

magavegen=20*log10(abs(havegen));
phaseavegen=180/pi*unwrap(angle(havegen));
% Ploteaza caracterele
semilogx(f,magex,'k-'); hold on;
semilogx(f,magavegen,'r'); hold on;
title('Convertor BUCK PWM CCM cu comanda prin sarcina')
xlabel('Frecventa [Hz]'); ylabel('Amplitudine [dB]');

figure;
semilogx(f,phaseex,'k-'); hold on;
semilogx(f,phaseavegen,'r'); hold on;
title('Convertor BUCK PWM CCM cu comanda prin sarcina')
xlabel('Frecventa [Hz]'); ylabel('Faza [°]');

```

**Funcția de transfer control-ieșire exactă pentru un convertor BUCK PWM CCM cu comanda mixta - *fișierul mixt.m***

% Funcția de transfer control iesire exacta la un convertor BUCK PWM CCM cu comandă mixtă.

% Comparatie cu rezultatele din modelul mediat propus la PCIM'95.

```
clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end
format short
```

```
k=401; % numarul de puncte de pe axa frecventei in care se reprezinta caracteristicile
f=logspace(1,6,k); % axa frecventei de la 10^1 la 10^6, logaritmica
w=2*pi*f; % definire omega
```

```
% Marimile de circuit:
Vg=15; D=0.4; Dprime=1-D; fs=30.2e3; Ts=1/fs; % comanda
L=0.48e-3; Co=30e-6; R=25;
Rint=25e3; Cint=1e-9; Rdesc=0.1; % elementele de circuit
tau=Rint*Cint; Rtrad=50; % traductor de curent de 50 ohm transrezistentă.
p=[Rtrad 0 0]; q=1; % pt BUCK cu comandă mixtă;
n=2; % numarul variabilelor de stare
```

% Matricile aferente celor doua stari topologice ale convertorului:

```
Aq1=[0 -1/L;
     1/Co -1/(R*Co)];
Bq1=[1/L; 0]; Eq1=[0 1]; Fq1=[0];
Aq2=[0 -1/L;
     1/Co -1/(R*Co)];
Bq2=[0; 0]; Eq2=[0 1]; Fq2=[0];
```

```
A1=[[Aq1 zeros(n,1);
     1/tau.*p ];
     B1=[Bq1; q/tau]; E1=[Eq1 0]; F1=Fq1;
     A2=[[Aq2 zeros(n,1);
     zeros(1,n) -1/(Rdesc*Cint)];
     B2=[Bq2; 0]; E2=[Eq2 0]; F2=Fq2;
     I=eye(3);
```

% Matricile si toate marimile ce nu depind de w, necesare in calcul (dupa prof.Tymerski)

```
T1=0; T2=D*Ts; T3=Ts; % calculul timpilor stationari
[phi1, psi1]=c2d(A1,B1,(T2-T1)); % modelul analogic in spatiul starilor se
[phi2, psi2]=c2d(A2,B2,(T3-T2)); % converteste in model discret
```

```
XT1=(1-phi2*psi1)/(phi2*psi1+psi2)*Vg; % valoarea din stare stationara la T1
XT2=(1-phi1*psi2)/(phi1*psi2+psi1)*Vg; % valoarea din stare stationara la T2
XT3=XT1; % valoarea din stare stationara la T3
gamma1=A1*XT2+B1*Vg; % matricile necesare pt. perturbatii
gamma2=A2*XT3+B2*Vg;
csi1=(A1-A2)*XT2+(B1-B2)*Vg;
zeta1=(E1-E2)*XT2+(F1-F2)*Vg;
h1=[0 0 1]; % matricea extractoare pt tensiunea pe Cint
miu1=inv(h1*gamma1);
Gamma1=I-miu1*csi1*h1;
```

% Bucla principala pentru calculul functiei de transfer cu formula exacta

```
for n=1:k
    beta1=expm(-j*w(n)*(T2-T1))*phi1;
    beta2=expm(-j*w(n)*(T3-T2))*phi2; % Aici se definesc toti coeficientii care depind de w
    alpha1=E1*(j*w(n)*I-A1)*(I-beta1);
    alpha2=E2*(j*w(n)*I-A2)*(I-beta2);
    hex(n)=(miu1/Ts)*((alpha1*beta2+alpha2-zeta1*miu1*h1*beta1*beta2)/(1-Gamma1*beta1*beta2)*csi1+zeta1);
end
magex=20*log10(abs(hex));
phasecx=180/pi*unwrap(angle(hex));
```

% Exprimarea cu modelul mediat pt. intreg sistemul (inclusiv He(s))- model de ordinul 4, publicatla PCIM

```
AD=D*Aq1+Dprime*Aq2;
BD=D*Bq1+Dprime*Bq2;
ED=D*Eq1+Dprime*Eq2;
FD=D*Fq1+Dprime*Fq2;
X=inv(AD)*BD*Vg;
csiD=(Aq1-Aq2)*X+(Bq1-Bq2)*Vg;
zetaD=(Eq1-Eq2)*X+(Fq1-Fq2)*Vg;
```

```
I2=eye(2); hi=[1 0]; hc=[0 1]; wn=pi/Ts; Qz=-2/pi;
ILT2=X(1)+1/2*Vg*D*(1-D)*Ts/L;
Fm=tau*Ts*1/(Rtrad*ILT2+q*Vg); Ri=Rtrad*(D*Ts/tau); kr=D*(1-D)*Ts^2/(2*L*(tau/Rtrad));
```



```

for l=1:k
HD(l)=ED/(j*w(l)*I2-AD)*csiD+zetaD;
He(l)=1+j*w(l)/(wn*Qz)+(j*w(l))^2/wn^2;
Have(l)=HD(l)*inv(1-(Fm*kr*hc-Fm*ri*He(l)*hi)*inv(j*w(l)*I2-AD)*csiD)*Fm;
end

magave=20*log10(abs(Have));
phaseave=180/pi*unwrap(angle(Have));

% plot the magnitude (amplitude)and phase
semilogx(f,magex,'k-'); hold on;
semilogx(f,magave,'r-');hold on;
title('Convertor BUCK PWM CCM cu comanda mixta')
xlabel('Frecventa [Hz]'); ylabel('Amplitudine [dB]');

figure;
semilogx(f,phaseex,'k-'); hold on;
semilogx(f,phaseave,'r-'); hold on;
title('Convertor BUCK PWM CCM cu comanda mixta')
xlabel('Frecventa [Hz]'); ylabel('Faza [°]');

```

### A 31 Aplicații ale convertoarelor QRC în circuite de corecție a factorului de putere

*Fișierul fn\_wt\_m.m*

```

% Plotarea caracteristicilor fN=f(wt) cu Mmin parametru pt lambda=1, sarcina constanta
% si structuri fara transformator.
clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

disp('Alegeti tipul de convertor dorit a se analiza:');
disp(' 1 - FORWARD (BUCK)');
disp(' 2 - BOOST');
disp(' 3 - FLYBACK, CUK sau SEPIC');
disp(' 0 - Abandonati analiza. ');
tip=input('Selectati un numar si apoi apasati tasta "Enter": ');
fprintf('\n');
while (tip~=0)&(tip~=1)&(tip~=2)&(tip~=3);
    disp('Numar incorect (numarul trebuie sa fie 1,2,3 sau 0)');
    type=input('Selectati un numar si apoi apasati tasta "Enter": ');
fprintf('\n');
end;

if(tip==0)

Mmin=input('Introduceti valorile pentru parameterul Mmin intre "[ ]" si separate de blankuri sau virgule: ');

n=[1,2]; % se face plotare atat pt HW cat si pt FW
N=40; % numarul de puncte in care se face reprezentarea
w=2*pi*50;
t=pi/(2*w)/N:pi/(2*w)/N:pi/(2*w);
nT=1;

for i=1:length(n)
    for j=1:length(Mmin)
        for k=1:length(t)
            fN(k)=2*pi*(miu(Mmin(j)/sin(w*t(k)),nT,tip))/(quasi(abs(sin(w*t(k))),n(i)));
        end
        if n(i)==1
            plot(w*t, fN,'k-');
        else
            plot(w*t, fN,'k--');
        end
        axis([0 pi/2 0 1]); xlabel('wt [rad]'); ylabel('fN');
        if tip==1
            title('Convertor ZCS QRC de tip FORWARD (BUCK)')
        elseif tip==2
            title('Convertor ZCS QRC de tip BOOST')
        else
            title('Convertor ZCS QRC de tip FLYBACK, CUK sau SEPIC')
        end
    end
end

```

```

end
hold on
end
end
grid

else
disp('Parasire program')
end

```

#### Fișierul fn\_wt\_lm

```

% Plotarea caracteristicilor fN=f(wt) cu lambda parametru (sarcina variabila) pt Mmin=ct
% si structuri fara transformator.
clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

disp('Alegeti tipul de convertor dorit a se analiza :');
disp(' 1 - FORWARD (BUCK)');
disp(' 2 - BOOST');
disp(' 3 - FLYBACK, CUK sau SEPIC');
disp(' 0 - Abandonati analiza. ');
tip=input('Selectati un numar si apoi apasati tasta "Enter": ');
fprintf('\n');
while (tip~=0)&(tip~=1)&(tip~=2)&(tip~=3);
    disp('Numar incorect (numarul trebuie sa fie 1,2,3 sau 0)');
    type=input(' Selectati un numar si apoi apasati tasta "Enter": ');
fprintf('\n');
end;

if(tip~=0)
Mmin=input('Introduceti valoarea pentru Mmin: ');
lambda=input('Introduceti valorile pentru parametrul lambda intre "[" separate de blankuri sau virgule: ');

n=[1,2]; % se face plotare atat pt HW cat si pt FW
N=40; % numarul de puncte in care se face reprezentarea
w=2*pi*50;
t=pi/(2*w)/N:pi/(2*w)/N:pi/(2*w);
nT=1;

for i=1:length(n)
    for j=1:length(lambda)
        for k=1:length(t)
            fN(k)=2*pi*(miu(Mmin/sin(w*t(k)),nT,tip))/(quasi(abs(sin(w*t(k)))/lambda(j),n(i)));
        end
        if n(i)==1
            plot(w*t, fN,'k-');
        else
            plot(w*t, fN,'k--');
        end
        axis([0 pi/2 0 1]); xlabel('wt [rad]'); ylabel('fN');
        if tip==1
            title('Convertor ZCS QRC de tip FORWARD (BUCK)')
        elseif tip==2
            title('Convertor ZCS QRC de tip BOOST')
        else
            title('Convertor ZCS QRC de tip FLYBACK, CUK sau SEPIC')
        end
        hold on
    end
end
end
grid

else
disp('Parasire program')
end

```

#### Fișierul fn\_wt\_fp.m

% Plotarea caracteristicilor fN=f(wt) cu fNP parametru, pt lambda=1 (sarcina constanta)

```

% si structuri cu transformator.
clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

disp('Alegeti tipul de convertor dorit a se analiza :');
disp(' 1 - FORWARD (BUCK)');
disp(' 2 - BOOST');
disp(' 3 - FLYBACK, CUK sau SEPIC');
disp(' 0 - Abandonati analiza. ');
tip=input('Selectati un numar si apoi apasati tasta "Enter": ');
fprintf('\n');
while (tip~=0)&(tip~=1)&(tip~=2)&(tip~=3);
    disp('Numar incorect (numarul trebuie sa fie 1,2,3 sau 0)');
    type=input('Selectati un numar si apoi apasati tasta "Enter": ');
    fprintf('\n');
end;

if(tip~=0)

fNP=input('Introduceti valorile pentru parametrul fNP intre "[ ]" si separate de blankuri sau virgule: ');
lambda=1;

n=[1,2]; % se face plotare atat pt HW cat si pt FW
N=40; % numarul de puncte in care se face reprezentarea
w=2*pi*50;
t=pi/(2*w)/N:pi/(2*w)/N:pi/(2*w);

for i=1:length(n)

    for j=1:length(fNP)
        for k=1:length(t)
            fN(k)=2*pi*fNP(j)/(fNP(j)+(1-fNP(j))*abs(sin(w*t(k))))*1/(quasi(abs(sin(w*t(k))),n(i)));
            end
            if n(i)==1
                plot(w*t, fN,'k-');
            else
                plot(w*t, fN,'k--');
            end
            axis([0 pi/2 0 1]); xlabel('wt [rad]'); ylabel('fN');
            if tip==1
                title('Convertor ZCS QRC de tip FORWARD (BUCK)')
            elseif tip==2
                title('Convertor ZCS QRC de tip BOOST')
            else
                title('Convertor ZCS QRC de tip FLYBACK, CUK sau SEPIC')
            end
            end
            hold on
        end
    end
    grid

else
    disp('Parasire program')
end

```

#### *Fișierul fn\_wt\_1.m*

```

% Plotarea caracteristicilor fN=f(wt) cu lambda parametru (sarcina variabila) si fNP=ct
% si structuri cu transformator.
clear % sterge variabile
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

disp('Alegeti tipul de convertor dorit a se analiza :');
disp(' 1 - FORWARD (BUCK)');
disp(' 2 - BOOST');
disp(' 3 - FLYBACK, CUK sau SEPIC');
disp(' 0 - Abandonati analiza. ');
tip=input('Selectati un numar si apoi apasati tasta "Enter": ');

```

```

fprintf('\n');
while (tip~=0)&(tip~=1)&(tip~=2)&(tip~=3);
    disp('Numar incorect (numarul trebuie sa fie 1,2,3 sau 0)');
    type=input('Selectati un numar si apoi apasati tasta "Enter": ');
fprintf('\n');
end;

if(tip~=0)

fNP=0.4;
lambda=input('Introduceti valorile pentru parametrul lambda intre "[ ]" si separate de blankuri sau virgule: ');

n=[1,2]; % se face plotare atat pt HW cat si pt FW
N=40; % numarul de puncte in care se face reprezentarea
w=2*pi*50;
t=pi/(2*w)/N:pi/(2*w)/N:pi/(2*w);

for i=1:length(n)

    for j=1:length(lambda)
        for k=1:length(t)
            fN(k)=2*pi*fNP/(fNP+(1-fNP)*abs(sin(w*t(k))))/(quasi(abs(sin(w*t(k)))/lambda(j),n(i)));
        end
        if n(i)==1
            plot(w*t, fN,'k-');
        else
            plot(w*t, fN,'k--');
        end
        axis([0 pi/2 0 1]); xlabel('wt [rad]'); ylabel('fN');
        if tip==1
            title('Convertor ZCS QRC de tip FORWARD (BUCK)')
        elseif tip==2
            title('Convertor ZCS QRC de tip BOOST')
        else
            title('Convertor ZCS QRC de tip FLYBACK, CUK sau SEPIC')
        end
        hold on
    end
end
grid

else
disp('Parasire program')
end

```

#### Fișierul miu.m

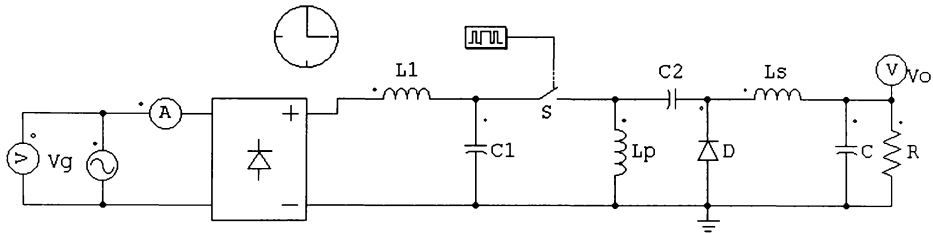
```

% Functia miu definita functie de m, raport de transformare si tip de convertor
function y=miu(m,nT,tip)
if tip==1 % BUCK sau FORWARD
    y=m/nT;
elseif tip==2 % BOOST
    y=(m-1)/m;
elseif tip==3
    y=m/(m+nT); %FLYBACK, CUK si SEPIC
else
    disp('Valoare incorecta pentru variabila "tip", care trebuie sa fie 1,2, sau 3') ;
end

```

## A 32 Aplicații ale convertoarelor DIVM în circuite de corecție a factorului de putere

Fișierul *divm\_pfc.sch* - fișier PSIM (Simcad)



Formele de unda au fost scrise într-un fișier ASCII, care a fost apoi importat în MATLAB pentru afișare iar analiza spectrală se obține importând și afișând și analizând datele din fișierul ASCII creat de opțiune FFT din programul PSIM.

Fișierul *pfc\_divm.m*

```
% Ploteaza formele de unda ale tensiunii de alimentare si ale curentului de alimentare dintr-un fisier
% (more than one period can be contained in that data file)
% si afisaza spectrul curentului folosind datele din fisierul *.fft creat de programul PSIM.
% (exactly pentru o perioada)
% Datele pentru formele de unda sunt in fisierul "file1" (cu extensia .dat)
% iar datele pentru spectru in fisierul "file2" (extensia .fft).
function pfc_spec(file1,file2,Nharm)
format long
Fer=4;
for i=1:Fer
    close % sterge ultimele Fer ferestre grafice
end

eval(['load ' file1 '.dat']);
wav=eval(file1); M=length(wav(:,1)); tmin=wav(1,1); tmax=wav(M,1);
subplot(211), plot(wav(:,1),wav(:,5),'k'); xlabel('time [s]'); ylabel('Line voltage [V]'); axis([tmin tmax -320 320]);
subplot(212), plot(wav(:,1),wav(:,3),'k'); xlabel('time [s]'); ylabel('Line current [A]'); axis([tmin tmax -1.3 1.3]);

eval(['load ' file2 '.fft']);
a=eval(file2);

X=a(:,2);
Y=a(:,3);
N=length(X);
figure; stem1(X,Y,'k'); title('Line current spectra'); xlabel('Frequency [Hz]'); ylabel('Amplitude [V]'); axis([X(1) X(Nharm) min(Y)-0.1*abs(min(Y))
max(Y)+0.1*abs(max(Y))]);
```