

MINISTERUL EDUCAȚIEI NAȚIONALE
UNIVERSITATEA "POLITEHNICA" DIN TIMIȘOARA
FACULTATEA DE ELECTRONICĂ ȘI TELECOMUNICAȚII

**CONTRIBUȚII LA PRELUCRAREA NUMERICĂ A
ELECTROCARDIOGRAMEI ÎN PROBA DE EFORT**

Ioan P. MIHU

Teza pentru titlul de
Doctor inginer

619.478
366 C

BIBLIOTECA CENTRALĂ
UNIVERSITATEA "POLITEHNICA"
TIMIȘOARA

Conducător Științific

Prof. Dr. Ing. Anton POLICEC

1997

PREFAȚĂ

Obiectivul major al acestei teze de doctorat este cercetarea și realizarea unui sistem complet și performant de monitorizare automată a probei de efort, destinat "Laboratorului de Investigații Clinice prin Proba de Efort" din cadrul Secției Cardiologie a Spitalului Județean Sibiu.

Trebuie menționată buna structurare a pregătirii prin doctorat, asigurată de conducătorul științific, domnul prof. dr. ing. Anton Policec. În acest sens cele două referate susținute "Procesarea numerică a semnalului ECG" și "Monitorizarea în Unități de Terapie Intensivă", au constituit demersuri extrem de utile pentru elaborarea ulterioară a tezei. Un suport important al obiectivului tezei l-a constituit și experiența dobândită de autor cu ocazia stagiului de șase luni (ianuarie-iulie 1995), efectuat la IUT Rennes și la Laboratoire de Traitement du Signal et de l'Image, Université de Rennes I Franța. Acest stagiul a fost rezultatul colaborării dintre Universitatea "Lucian Blaga" din Sibiu și Université de Rennes I.

Tematica acestei teze acoperă un domeniu vast, interdisciplinar, cuprinzând între granițele sale electronica, tehnica numerică de calcul și nu în ultimul rând medicina. Tema s-a născut dintr-o necesitate reală, iar colaborarea cu beneficiarul și mai ales rezolvarea efectivă a solicitărilor, dau o valoare în plus cercetării științifice efectuate, încununează efortul depus pentru finalizarea tezei. Sarcinile de bază solicitate de medici sistemului de monitorizare automată au fost:

- 1. Compatibilitatea a ceea ce se va realiza cu baza materială existentă și cu programele pentru evidența pacienților și pentru studii statistice, în scopul valorificării experienței existente.*
- 2. Controlul automat al regimului de efort, prin parametrii mecanici ai covorului rulant: viteză și înclinare.*
- 3. Urmărirea permanentă a traseului ECG în timp real, începând din perioada anterioară startului și inclusiv pe durata revenirii, cu următoarele obiective: urmărirea ritmului cardiac, evoluția segmentul ST, cu informații despre subdenivelare / supradenivelare, cuantificarea tipului și a amplitudinii denivelării, corelarea evoluției segmentului cu timpul de derulare a probei; urmărirea aritmiilor și clasificarea celor detectate.*
- 4. Măsurarea la intervalele de timp prevăzute de protocol a tensiunii arteriale. (În prima etapă măsurarea se va face manual și se va introduce la tastatură).*
- 5. Posibilitatea introducerii în program de la tastatură a simptomelor și semnelor pe care medicul le observă, sau pe care pacientul le reclamă și corelarea cu momentul apariției lor.*
- 6. Alarmarea sonoră și optică privind oprirea probei în situațiile: atingerea frecvenței cardiace maxime; atingerea unei denivelări maxim admisibile a segmentului ST; apariția unor aritmii severe.*



7. *Reținerea automată a segmentelor ECG de interes la fiecare interval prestabilit al protocolului (3 min), precum și posibilitatea ca medicul să aleagă orice alte segmente considerate de interes.*
8. *Posibilitatea de analiză "off line" a rezultatelor testului, precum și posibilitatea de transferare a datelor către programele existente.*
9. *Includerea în raportul probei a porțiunilor de traseu ECG de interes.*
10. *Includerea la cererea medicului a înregistrărilor și rezultatelor testului de efort în baza de date creată în scopul urmăririi evoluției pe termen lung a bolnavilor și pentru cercetarea ulterioară a rezultatelor testului.*

Deși pe plan mondial există realizări deosebite în analiza automată a ECG, acest domeniu rămâne în continuare un subiect tentant de cercetare, mai ales în cazul abordării unor aspecte cum ar fi:

- obținerea unor performanțe notabile în condițiile unui preț de cost scăzut;
- rezolvarea problemelor în contextul valorificării unor situații locale, specifice, cum ar fi dotarea deja existentă (covorul rulant realizat într-o etapă anterioară) și cercetările de informatică medicală existente, legate de proba de efort.
- formarea unui centru de preocupări în domeniul procesării semnalelor biomedicale, în beneficiul Facultății de Medicină și al Facultății de Inginerie ale Universității "Lucian Blaga" din Sibiu, dar mai ales în beneficiul serviciilor medicale ale Spitalului Județean, în special al secției de Cardiologie.

Structurarea materialului s-a făcut pornind de la cerințele impuse sistemului de monitorizare, marcând în mare și principalele etape ale cercetării.

Capitolul 1 constituie cadrul general de prezentare a domeniului. Aici sunt stabilite principalele sarcini ale cercetărilor, în spiritul unor principii și al unor tendințe generale actuale bine definite în domeniul monitorizării și procesării semnalelor biomedicale, argumentate cu trimiteri bibliografice.

Capitolul 2 este consacrat stadiului actual al cercetărilor în aria preocupărilor tezei. După prezentarea din literatura de specialitate a aspectelor de fond ale achiziției de date având la bază un calculator IBM-PC, urmează un amplu studiu bibliografic al cărui scop este clasificarea principiilor consacrate ale detectoarelor QRS și evidențierea performanțelor notabile ale acestora. Tot în acest capitol sunt ordonate și trecute în revistă principalele metode folosite la compresia datelor ECG.

Capitolul 3, cel mai vast, cuprinde în prima sa parte, prezentarea sistemului de achiziție de date folosit de autor. Este dedicat apoi un spațiu amplu cercetărilor privind procesarea numerică a semnalului ECG, în primul rând pentru implementarea detectoarelor QRS în timp real în monitorizarea probei de efort și apoi pentru optimizarea unor algoritmi de compresie de date. Ultimul paragraf al capitolului, este cel în care cercetările efectuate pe parcursul tezei sunt puse în slujba aplicației clinice de monitorizare a probei de efort.

Capitolul 4 prezintă în sinteză rezultatele tuturor cercetărilor prezente în teză, accentul punându-se pe identificarea soluțiilor originale și a elementele de progres care stau la baza atingerii obiectivului tezei.

MULȚUMIRI

În primul rând mulțumesc domnului prof. dr. ing. Anton Policec, conducătorul științific, pentru sprijinul acordat atât la rezolvarea unor probleme, cât și la descoperirea altora noi, a căror soluție nu o întrezăream în primul moment. Când marea majoritate a problemelor fuseseră abordate, însă trebuiau integrate, armonizate cu interesul Laboratorului de Investigații Clinice prin Proba de Efort, vizita efectuată de domnul prof. dr. ing. Anton Policec la Sibiu, la Facultatea de Inginerie a Universității "Lucian Blaga" și la Secția de Cardiologie a Spitalului Județean Sibiu, a constituit unul dintre cele mai importante momente ale elaborării acestei teze.

Țin să mulțumesc în mod deosebit domnului prof. dr. ing. Anton Policec pentru demersurile privind organizarea judicioasă a activității mele de pregătire și cercetare științifică prin doctorat. În acest sens, amintesc cele două referate ("Procesarea numerică a semnalului ECG" și "Monitorizarea în Unități de Terapie Intensivă") și nu în ultimul rând, ajutorul concret pentru structurarea și elaborarea tezei.

În afara sprijinului științific nu pot să nu apreciez în mod egal sprijinul moral, lipsa de formalism, tactul și înțelegerea deosebită cu care domnia sa m-a condus la acest final.

Mulțumesc de asemenea domnului dr. Ioan Manițiu, medic specialist cardiolog, șeful Secției Cardiologie a Spitalului Județean Sibiu pentru deschiderea deosebită față de actul de cercetare realizat prin mijloace cât mai moderne posibil, pentru eforturile deosebite făcute pentru echiparea Laboratorului de Investigații prin Proba de Efort, aceste mulțumiri adăugându-se celor ale pacienților tratați aici. Preocupările domniei sale în studiul probei de efort datează de mai multă vreme, la începerea colaborării noastre existând deja rezultate în analiza statistică a parametrilor probei, rod al colaborării cu domnișoara asistent Liana Bera, informatician, la Facultatea de Medicină a Universității "Lucian Blaga".

Jamais je n'oublierais de remercier a la ...Providence, qui m'a offert la chance unique d'effectuer le stage a l'IUT Rennes, France mais la Providence a été aidée par les démarches de M. Olivier Bonnaud, professeur des Universités, Directeur de Recherches aux CNRS.

J'exprime mes remerciements aux M. professeur Jean Pierre Camps, Directeur de l'IUT Rennes, et M. Jean Pellen, chef du département Électronique de l'IUT Rennes, pour les conditions d'études et de travail et aussi pour l'aide dans la programmation de l'ordinateur.



Je remercie tout particulièrement a M. professeur Rene Ollivier. spécialiste en filtrage numérique, pour son amitié et pour le fait qu'il m'a relevé toujours le fil rouge dans le traitement du signal.

Je remercie aussi aux M. Guy Carrault et M. Louis Thoraval chercheurs au Laboratoire de Traitement du Signal et de l'Image, Université de Rennes 1, pour toutes leurs conseils dans le domaine du traitement de ECG et pour la base de données ECG, qui m'a permis de vérifier les résultats.

Mulțumesc tuturor colegilor și prietenilor care m-au ajutat cu sfaturi, sugestii, întrebări, material bibliografic sau încurajări, în plan științific sau uman, în mod deosebit domnului asistent ing. Macarie Breazu, pentru chestiuni legate de programare și de compresia de date, iar doamnei Rodica Lungu și domnului Tom Oprean, pentru sprijinul acordat la redactarea lucrării.

Pentru Felicia, Ioana și Cantemir !

Sibiu, 10 aprilie 1997

CUPRINS

Capitolul 1. PREZENTAREA DOMENIULUI

1.1	Introducere	1
1.2	Semnalul ECG și starea de efort	3
1.2.1	Fiziologia și fiziopatologia stării de efort	4
1.2.2	Electrocardiografia cordului bolnav în timpul efortului	5
1.3	Proba de efort	8
1.4	Privire generală a sarcinilor pentru cercetare	10
1.4.1	Conversia Analog Numerică	11
1.4.2	Preprocesarea	12
1.4.3	Detecția undelor	13
1.4.4	Compresia datelor	14
1.4.5	Extragerea parametrilor	14
1.4.6	Clasificarea	15

Capitolul 2. STADIUL ACTUAL AL CERCETĂRILOR

2.1	Achiziția de date ECG	17
2.1.1	Semnalul ECG	18
2.1.2	Amplicarea semnalului	20
2.1.3	Condiționări ale semnalului	20
2.1.4	Interfața	21
2.1.5	Cuantificarea semnalului	21
2.1.6	Frecvența de eșantionare	22
2.3	Detecția QRS	23
2.2.1	Clasificări	23
2.2.2	Metode folosite în detecția QRS	24
2.2.3	Preprocesări	25
2.2.3.1	Transformări liniare	25
2.2.3.2	Identificarea punctului fiducial	26
2.2.3.3	Transformări neliniare	27
2.2.3.4	Corelația cu o formă "șablon"	28
2.2.3.5	Modelare de complex QRS	29
2.2.4	Prezentarea aparatului matematic (FTS, FTJ, FTB, Transf.Lungime, Transf.Energie, Corelație)	30
2.2.5	Evaluarea performanțelor algoritmilor de detecție QRS	34

2.3 Compresia datelor ECG	35
2.3.1 Tehnici de codare ireversibilă	36
2.3.1.1. Tehnici de codare ireversibilă în domeniul timp	36
<i>a. Extragerea parametrilor</i>	37
<i>b. Aproximări liniare și poligonale</i>	37
<i>c. Compresia "ciclu cu ciclu"</i>	39
<i>d. Codarea diferențială cu prag</i>	39
2.3.1.2 Transformări ortogonale	40
2.3.2 Tehnici de codare reversibilă	40
2.3.2.1 Metode predictive	41
<i>a. Predicția liniară</i>	41
<i>b. Codarea diferențială</i>	42
2.3.2.2 Codarea entropică	42
<i>a. Codarea HUFFMAN</i>	42
<i>b. Codarea aritmetică</i>	43
2.3.4 Evaluarea algoritmilor de compresie	43
2.4 Criterii pentru diagnosticarea automată a ischemiei	44

Capitolul 3. REALIZĂRI ȘI CONTRIBUȚII PERSONALE

3.1 Arhitectura propusă pentru cercetare	49
3.2 Sistemul de achiziție	51
3.2.1 Prezentarea părții hard a sistemului de achiziție utilizat	51
3.2.2 Achiziția de date în contextul programului de monitorizare	52
3.2.3 Afișarea	55
3.2.3.1 <i>Compresia de date pentru afișaj</i>	56
3.2.3.2 <i>Reprezentări grafice ale traseului ECG</i>	58
3.3 Detectoare QRS	60
3.3.1 Caracteristicile de frecvență ale algoritmilor de procesare ECG	60
3.3.2 Detector QRS bazat pe derivata de ordinul 2, "patru puncte"	61
3.3.3 Detector QRS bazat pe derivata de ordinul 3, "patru puncte"	64
3.3.4 Detector QRS utilizând corelația cu funcții "șablon"	67
3.3.5 Detector QRS utilizând corelația cu funcții "șablon" + FTS	69
3.3.6 Detector QRS cu "șablon" creat din semnal	77
3.3.7 Detector QRS utilizând transformarea "lungime"	81
3.3.8 Identificarea complexului QRS	83
3.3.9 Detector neliniar QRS, "amplitudine - durată"	84
3.3.10 Detector neliniar QRS, cu reaxare după înfășurătoarea inferioară	89
3.3.11 Evaluarea performanțelor algoritmilor de procesare, la detecția QRS	92
3.3.12 Concluzii cu privire la utilizarea algoritmilor pentru detecția QRS	96
3.4 Algoritmi de compresie	97
3.4.1 Optimizarea predicției liniare pentru semnale ECG	97
3.4.1.1 Calculul coeficienților predictorului prin Metoda Entropiei Maxime	100
3.4.1.2 Implementarea metodei. Rezultate	103
3.4.1.3 Concluzii	109
3.4.2 Optimizarea algoritmului de compresie "evantai"	111
3.4.2.1 Prezentarea principiului	111

3.4.2.2 Criteriu de optimizare.....	113
3.4.2.3 Concluzii.....	114
3.4.3 Concluzii la studiul de compresie a datelor ECG.....	115
3.5 Filtrarea semnalelor codate	116
3.5.1 Filtrarea nerecursivă a semnalelor codate prin predicție.....	116
3.5.2 Codarea DELTA cu pantă constantă utilizată la semnale ECG.....	118
3.5.3 Implementarea algoritmului de filtrare a semnalelor codate DELTA.....	119
3.6 Aplicații clinice	122
3.6.1 Automatizarea derulării protocolului probei de efort.....	122
3.6.2 Creare prototip ST prin mediere cu supraeșantionare.....	124
3.6.3 Analiza aritmiilor.....	127
3.6.4 Aprecierea rezultatelor testului.....	129

Capitolul 4. CONCLUZII

4.1 Sinteză.....	131
4.2 Contribuții.....	132
4.3 Direcții de cercetare.....	137

BIBLIOGRAFIE	139
---------------------------	-----

ANEXE	151
--------------------	-----

Anexa A1. Schema electrică a sistemului de achiziție.....	151
Anexa A2. Program pentru trasarea caracteristicilor de frecvență; CAR_FRECV.PAS.....	155
Anexa A3. Program de analiză și procesare of line al semnalelor ECG. QRS.PAS.....	167
Anexa A4. Program de monitorizare a probei de efort. Procedurade mediere SABLON.PAS.....	179

Capitolul 1

PREZENTAREA DOMENIULUI

1.1 INTRODUCERE

Prin tematica abordată lucrarea de față se încadrează în categoria căutărilor de rezolvare cu ajutorul cercetării științifice a unei necesități sociale. Monitorizarea investigațiilor medicale cu ajutorul calculatorului este astăzi nu doar o chestiune de modă, de curent de opinie, ci reprezintă o cerință obligatorie în abordarea și rezolvarea cu maximum de eficiență, atât pentru medic cât mai ales pentru pacient, a actului medical.

Preocupările de avangardă în domeniul monitorizării și procesării semnalelor biomedicale au ca scop asigurarea unui standard cât mai ridicat pentru unitățile medicale, potrivit exploziei tehnologice și informaționale care are loc în zilele noastre. În acest scop Comunitatea Europeană a lansat un program amplu numit "Advanced Informatics in Medicine" (AIM). Obiectivele de bază ale acestui program au fost:

- structurarea și sinteza cunoștințelor generale privind procesarea semnalelor biomedicale și experiența în acest domeniu, extrem de mult cercetat ;
- structurarea și sintetiza cunoștințelor generale privind procesarea semnalelor ECG și a celor referitoare la presiunea sanguină;
- informații privitoare la echipamentul indicat a constitui suportul implementării soluțiilor ;

Concluziile programului reprezintă niște puncte "ținte" ce trebuie vizate de orice cercetare în domeniu.

În cadrul programului amintit, s-a lansat un proiect cu participarea mai multor centre universitare din Europa, (Balochi ș.a, 1989, Carault ș.a, 1990, Siregar ș.a, 1990) cunoscut sub numele KISS (Knowledge-based Interactive Signal monitoring System). Obiectul proiectului KISS era de a pune bazele dezvoltării unui sistem evoluat de monitorizare a activității cardiace, pentru prima etapă fiind vizată realizarea prototipului unei unități de terapie intensivă pentru coronarieni. În acest scop proiectul amintit a avut în vedere următoarele sarcini:

- cunoașterea și sintetizarea cunoștințelor clasice despre procesarea semnalului ECG ;
- evidențierea problemelor nerezolvate sau a celor a căror soluție prezintă anumite limite;

- stabilirea caracteristicilor unui posibil prototip al unui sistem de monitorizare, precum și evaluarea soluțiilor posibile care să faciliteze alegerea căilor de urmat;
- definirea pragurilor dintre metodele numerice clasice și metodele simbolice în vederea definirii, pe cât posibil, a rolului fiecăreia în vederea integrării și exploatării lor.

Sistemul nu trebuie să se mulțumească cu cunoașterea "de suprafață" care leagă direct observațiile asupra pacientului de concluzii, ci să integreze în acest lanț cunoștințe profunde de anatomie fiziologie și patologie (Thull ș.a. 1992;).

După definirea obiectivelor precise ale unui sistem de monitorizare inteligent (Lage ș.a. 1988; Siregar ș.a. 1993), s-au degajat patru mari funcții care au dat naștere unui sistem multi-expert compus din :

- procesarea de nivel scăzut a semnalului;
- interpretor om-mașină evoluat;
- expert, bazat pe cunoștințe "de suprafață" ;
- model profund.

Sistemul este organizat după o arhitectură hibridă ce permite comunicații rapide între elementele sale. De menționat că o astfel de structură presupune și un suport tehnic adecvat. Raportul KISS (Carrault 1990) precizează clar necesitatea implementării pe sisteme de calcul puternice, sub sistemul de operare UNIX.

Este cred momentul de a aminti că țările dezvoltate alocă între 5 și 14 % din Produsul Național Brut cheltuielilor pentru sănătate, pe când în alte țări el se situează sub 4% (Schmitt, Al-Fadel, 1989). Aceste cifre sunt suficiente pentru a înțelege diferențele majore care fac atât de diferită utilizarea unei instrumentații medicale de înaltă tehnicitate în țările dezvoltate față de celelalte țări.

De aceea, fără a minimaliza concluzia amintită în raportul KISS, trebuie luat în considerare un alt punct de vedere și anume, că este posibil să se aducă o calitate nouă îngrijirii cardiacilor, cu sisteme de calcul accesibile și deci la prețuri de cost acceptabile. Această optică trebuie susținută preponderent prin cercetare și eforturi de dezvoltare locale. Se pot face astfel pași premergători în direcția utilizării echipamentelor de foarte mare performanță, atunci când resursele financiare o vor permite. Alegerea între cercetarea fundamentală, de avangardă dar fără șansa implementării sau cercetarea orientată spre implementare, în condițiile specifice țărilor mai puțin dezvoltate, trebuie făcută potrivit principiului că nu există viitor fără prezent. Consider că, în ceea ce ne privește, în perspectiva adevăratei economii de piață, alegerea nu poate fi decât una singură și anume varinta a doua, "cercetare țintită pe implementare, în condițiile noastre specifice".

Această alegere devine acceptabilă și deloc dezonorantă în contextul cercetărilor pe plan mondial, prin aplicarea în paralel a unei politici manageriale adecvate în domeniul cercetărilor medicale interdisciplinare, care să vizeze:

- dezvoltarea unor programe de cercetare cu cele mai mari șanse de aplicare practică;
- exploatarea tuturor resurselor de finanțare;
- integrarea în această importantă activitate socială, alături de actul medical în sine, a tuturor resurselor umane din domeniul cercetării interdisciplinare precum și a unui suport material de cât mai înaltă tehnicitate și nouitate.

Se poate afirma în concluzie că, în condițiile actuale, acest gen de compromis între cercetarea fundamentală și cercetarea aplicativă ținută, între instrumentație clasică și tehnologie de ultimă oră, va aduce cel mai bun răspuns atât necesităților imediate cât și celor de perspectivă, în domeniul monitorizării evaluate a investigațiilor medicale (Sukuvaara ș.a. 1993).

1.2 SEMNALUL ECG ȘI STAREA DE EFORT

Electrocardiograma normală (ECG) reprezintă înregistrarea potențialelor electrice generate de activitatea electrică a inimii, transmise la suprafața corpului, pentru un subiect sănătos.

Pe parcursul lucrării prin notația "ECG" se va înțelege semnalul electric preluat de la suprafața corpului în scopul prelucrării.

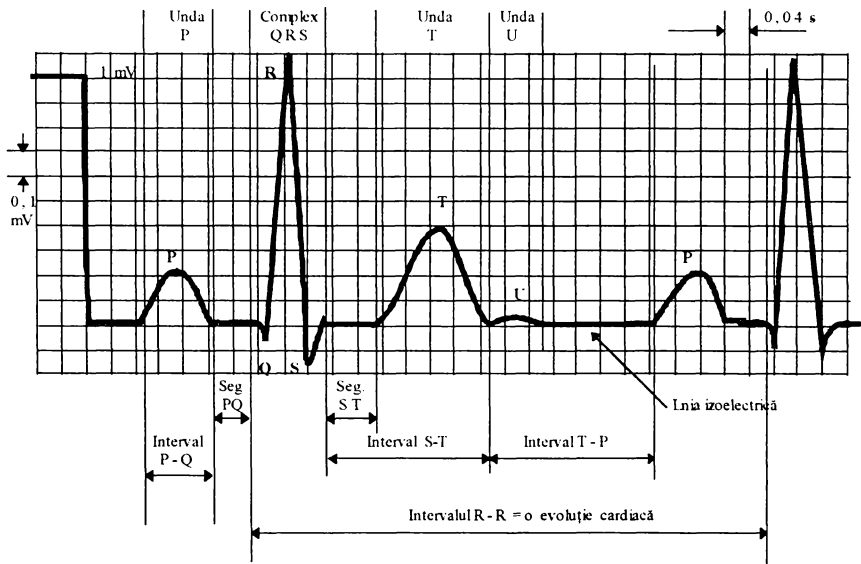


fig 1.1 ECG normală

Dintre elementele unei înregistrări ECG normale, voi evidenția doar elementele de interes pentru demersurile viitoare. Evenimentul cel mai semnificativ al activității cardiace îl constituie complexul de unde QRS, corespunzător depolarizării ventriculare. Având o amplitudine de ordinul 1 mV, el este utilizat fără excepție pentru identificarea ritmului cardiac. Morfologiile posibile ale complexului QRS, normale sau patologice sunt prezentate în fig. 1.2. (Rappaport ș.a. 1982).

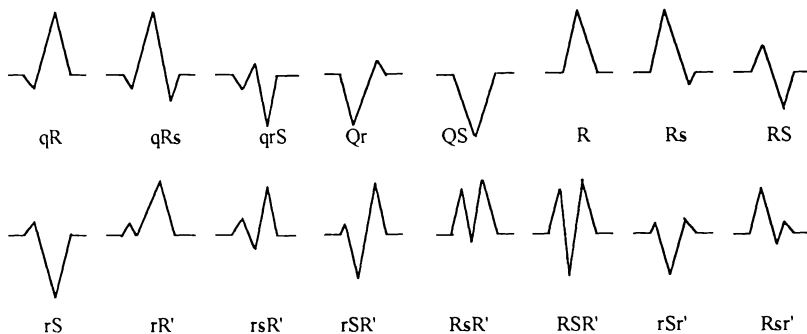


fig. 1.2. Morfologia complexului QRS

Al doilea element de interes major îl constituie segmentul ST. El are începutul notat cu J care reprezintă punctul de joncțiune cu complexul QRS, sfârșitul său fiind greu de precizat. Segmentul ST, corespunde repolarizării lente ventriculare. Diferențele de potențial care se produc au amplitudine mică și se anulează reciproc ceea ce face ca în mod normal segmentul să se afle pe linia izoelectrică. În situații patologice însă, el poate avea diverse forme, ca în fig.1.3.

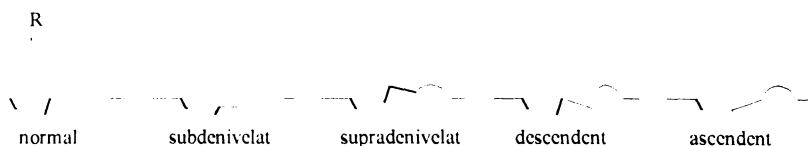


fig. 1.3. Morfologii posibile ale segmentului ST

Forma segmentului ST reprezintă unul din semnele de bază în identificarea infarctului acut (supradenivelarea ST) și de asemenea în ischemie.

1.2.1 Fiziologia și fiziopatologia stării de efort

Exercițiul fizic dinamic provoacă modificări importante ale funcției cardiace și ale repartiției debitului sanguin între diferite teritorii. Adaptarea circulatorie la efort are trei aspecte (Cotoi S., 1988):

- creșterea cantității de oxigen la nivelul zonei musculare în activitate;
- eliminarea excesului de căldură;

- păstrarea unei circulații adecvate la nivelul organelor viscerale.

În timpul efortului se disting patru faze:

- o fază de anticipare având originea în sistemul nervos central;
- debutul efortului, cu creșterea rapidă a frecvenței cardiace, a întoarcerii venoase, a debitului cardiac și cu scăderea rezistențelor periferice;
- faza de echilibru, când se realizează un consum maxim de oxigen constant, cu TA și ritm cardiac stabile;
- faza de epuizare, manifestată prin creșterea frecvenței cardiace, scăderea volumului ventricular ejectat și a vasodilatației cutanate.

Ca rezultat al creșterii frecvenței cardiace și a volumului sistolic, debitul cardiac crește în timpul efortului. Inițial crește întoarcerea venoasă sub influența hiperventilației și a stimulării simpaticе. Crește vasodilatația coronariană direct proporțional cu nivelul efortului. Creșterea debitului în zona miocardului duce automat la creșterea consumului propriu de oxigen, care depinde deci de frecvența cardiacă, contractilitatea și presiunea sistolică ventriculară.

Tensiunea arterială sistolică crește progresiv la efort iar cea diastolică rămâne nemodificată sau poate chiar să scadă ușor la efort ca urmare a scăderii rezistenței periferice totale. La hipertensivi crește TA diastolică prin creșterea rezistenței periferice, atât în timpul efortului, cât și după efort. De aici importanța clinică pentru diferențierea subiectului normal de cel hipertensiv, testul de efort permițând, în cazul hipertensiunii de graniță, să indice pe adevărații hipertensivi, la care crește TA diastolică.

Consumul miocardic de oxigen depine direct proporțional de debitul sanguin ce străbate coronarele. Deși este greu de determinat cu exactitate, s-a constatat că produsul frecvență cardiacă-presiune sanguină este o măsură a acestui debit și în același timp un indicator fidel al efortului fizic al miocardului. În acest context, produsul frecvență cardiacă-presiune se corelează bine cu consumul de oxigen miocardic și cu fluxul sanguin coronar. La același pacient acest produs și debitul anginei pectorale sunt în strânsă legătură. Raportul dintre produsul amintit și consumul de oxigen se schimbă când apar factori care modifică volumul ventricular și contractilitatea.

Consumul maxim de oxigen, pe care îl realizează pacientul fără să aibă simptome sau semne care să limiteze efortul, cuantifică cel mai bine starea de sănătate a aparatului cardiovascular, fiind și o măsură reală a capacității fizice de performanță a unei persoane.

Electrocardiografia reprezintă sursa majoră de date obiective în cursul testului de efort. În timpul efortului se produc o serie de modificări ale ECG ce se pot încadra în limitele normalului. Dintre acestea amintesc (Cotoi S., 1988):

- intervalul PR scade, fiind la frecvența de 150 /min sub 0,15sec;
- intervalul QT evoluează invers proporțional cu pulsul;
- crește amplitudinea undei P;
- axul electric QRS deviază progresiv spre dreapta;
- scade amplitudinea undei T;
- punctul "J" (care marchează contactul dintre QRS și segmentul ST), coboară progresiv, odată cu creșterea pulsului. Partea terminală a segmentului ST este mai puțin afectată.

1.2.3 Electrocardiografia cordului bolnav în timpul efortului

Dacă există o afectare coronară, în timpul unui efort fizic deosebit, circulația la nivelul miocardului scade, zona subendocardică fiind prima care devine ischemică. Ca urmare a acestui fenomen se produce un așa numit "potențial de leziune diastolică", manifestat vectorial ca un fenomen ce se opune vectorului electric ce generează complexul QRS. Acest vector este practic rezultatul lipsei de contribuție la vectorul QRS în momentul depolarizării, a zonei afectate din miocard. Din punctul de vedere al ECG, fenomenul se manifestă ca o subdenivelare a segmentului ST. Subdenivelarea segmentului ST poate fi de mai multe tipuri, ca în fig. 1.3.

Convențional, se utilizează drept criteriu de ischemie o deplasare inferioară a segmentului ST de 0,1 mV (1 mm) sau mai mare. Amplitudinea deplasării segmentului ST, luată drept referință pentru a decela cazurile normale de cele patologice, trebuie să fie o balanță între sensibilitate și specificitate. Astfel o subdenivelare a segmentului ST de numai 0,05 mV, luată drept criteriu de ischemie, deși cuprinde majoritatea bolnavilor coronarieni, include și o mare parte de persoane sănătoase, așa-ziși fals pozitivi. O subdenivelare a segmentului ST la 2 mm reduce mult numărul sănătoșilor suspecți a avea modificări ischemice, având astfel o mare specificitate dar o mai mică sensibilitate. În aprecierea subdenivelării ST orice criteriu este arbitrar, urmărindu-se un compromis optim între posibilitatea de a detecta adevărații bolnavi și de a exclude pe cei sănătoși.

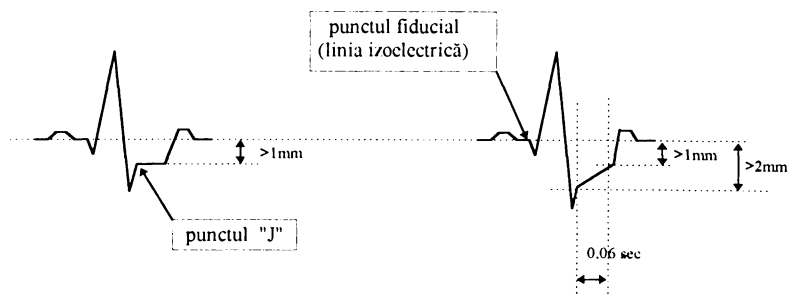


fig. 1.4 . Măsurarea denivelării segmentului ST

Din aceste motive se consideră că o subdenivelare orizontală a segmentului ST, în platou, de peste 1 mm sub linia de bază ce trece prin începutul undei Q la două-trei complexe QRS succesive, sau o subdenivelare oblică descendentă a segmentului de peste 1 mm, măsurată la 0,08 sec după punctul J, semnifică probă pozitivă. Modul principal de măsurare este înfățișat în fig. 1.4.

Supradenivelarea segmentului ST apare mai frecvent după infarctul miocardic anterior sau la bolnavii cu stenoza unui singur vas coronarian major, indicând ischemia transmurală.

Subdenivelarea sau supradenivelarea ST poate să apară în lipsa unei coronaropatii, dacă există situații care măresc dezechilibrul dintre oferta și necesarul de oxigen, cum ar fi cazul stenozei aortice. Modificări ST pot să apară și în sindroamele neurocirculatorii prin simulare simpatică excesivă, care sunt intense la începutul efortului, ca apoi să se diminueze pe măsura creșterii lui. Modificările pot să dispară după medicație cu betablocante. Normalizarea paradoxală a segmentului ST în timpul efortului a fost raportată foarte rar și la coronarieni.

O serie de medicamente ca digitala, procainamida, chinidina și fenotiazinele modifică segmentul ST, putând genera teste fals pozitive care ridică probleme de diagnostic și de aceea, se impune suprimarea lor înaintea efectuării probei.

Este foarte importantă luarea în considerare, alături de aspectul ECG și a simptomatologiei. Astfel, probabilitatea găsirii unei leziuni este foarte mare (95%) dacă se asociază o angină pectorală cu semne ECG certe și mai redusă (numai 65%), dacă apare angina fără semne ECG, sau numai semne ECG fără angină pectorală. La pacienții cu angină tipică cu sau fără semne ECG de ischemie, se practică cu prudență testul de efort pentru determinarea capacității de efort fizic, cu respectarea condițiilor pentru deplina securitate a bolnavului. (Manișiu și Mișu, 1993).

Utilizarea criteriilor modificărilor ST, prin analiza statistică computerizată a complexelor QRS pentru a elimina artefactele, a adus o mai mare precizie în studiul ECG de efort față de măsurătorile electrocardiogramelor convenționale.

Marea răspândire a probelor de efort, dezvoltarea interpretării automate, paralel cu extinderea coronarografiei și corelarea lor sistematică, au permis elaborarea unor *criterii suplimentare*. Acestea sunt:

Aspectul zis "aproape ischemic", care apare în următoarele situații: subdenivelarea segmentului ST de tip lent ascendent spre linia de bază, când atinge 2 mm la 8/100 secunde de punctul J, anunță evenimentele coronariene, la fel ca subdecalajul în platou orizontal. Când segmentul ST are aspect de seceră sau cu "racord spart" la unda T, se estimează un prognostic de peste 10% evenimente coronariene. Valoarea predictivă a subdenivelării ST este mult ridicată în favoarea existenței unei coronaropatii la pacienții simptomatici (Cotoi S. 1988);

Modificarea amplitudinii undei R în cursul efortului. Creșterea amplitudinii undei R în derivația V5 imediat după efort indică o boală coronariană severă, cu disfuncție ventriculară stângă, în timp ce scăderea amplitudinii undei R arată absența bolii coronariene. Unii autori au atribuit o mare valoare predictivă acestui criteriu pentru boala coronariană, el putând fi de folos la recunoașterea testelor fals pozitive și fals negative. Astfel, subdenivelarea ST cu scăderea amplitudinii undei R ar putea reprezenta un test fals pozitiv.

Modificarea undei Q. Scăderea amplitudinii sau lipsa modificării undei Q indică o ischemie septală, deoarece în condiții normale, amplitudinea undei Q crește după efort.

Inversarea undei U ar indica cu mare specificitate (97%) atingerea arterei coronariene interventriculare anterioare cu stenoză mai mare de 75%.

Undele T negative și simetrice în mai multe derivații ECG, fără modificări ale complexului QRS sau ale segmentului ST, apărute spontan, dar mai ales în cursul probei de efort, îndeosebi dacă se asociază cu fenomene anginoase, indică în 25% din cazuri o suferință coronariană serioasă.

În concluzie, modificările anormale ale traseului ECG în timpul efortului reprezintă o sursă valoroasă pentru diagnostic. Atunci când efortul se suprapune unei boli coronariene, cele mai specifice modificări sunt cele ale segmentului ST.

1.3 PROBA DE EFORT

Probele de efort utilizate în bolile cardiovasculare au în principal două obiective:

- să ofere o bază pentru evaluarea obiectivă a capacității de efort prin măsurarea consumului de oxigen;
- să producă simptome și semne care pot explica diferitele mecanisme ce limitează capacitatea de efort.

Pentru aceasta se utilizează tot mai frecvent efortul maximal, atât pentru precizarea diagnosticului, cât și pentru obținerea de informații funcționale cantitative, măsurând consumul de oxigen. Cu toate că efortul depus se poate cuantifica în unități fizice din Sistemul Internațional pentru măsurarea puterii consumate (watt), totuși în practica medicală s-a încetățenit măsurarea efortului fizic prin consumul de oxigen. Compararea consumului de oxigen cu valorile normale permite estimarea obiectivă a gradului de scădere a capacității funcționale a inimii. Consumul de oxigen este folosit și ca element de comparație între rezultatele obținute prin diferite metode.

Suportul tehnic cu ajutorul căruia se realizează proba de efort poate fi bicicleta ergometrică sau covorul rulant. Covorul rulant (fig. 1.5) permite realizarea celui mai mare consum de oxigen, implicând în efort membrele inferioare, toracele și membrele superioare.

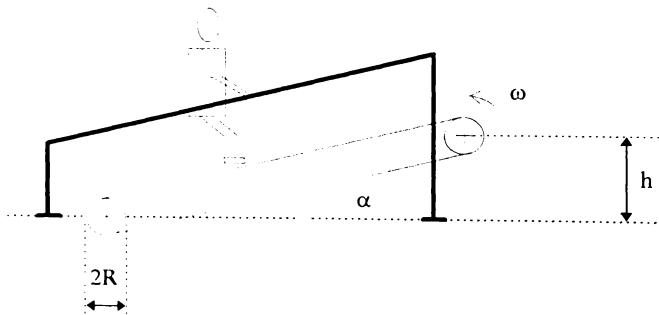


fig 1.5. Covor rulant cu parametrii reglabili.

De asemenea, un alt avantaj al covorului rulant față de bicicletă este faptul că pe covorul rulant se realizează un tip de efort firesc, obișnuit, chiar și pentru persoane neantrenate, acest efort fiind cel mai natural. Dezavantajul principal este

legat de prețul de cost, spațiul mai mare pe care îl ocupă și de faptul că înregistrările ECG efectuate în timpul probei sunt însoțite de artefacte de mișcare.

Din punctul de vedere al derulării în timp a efortului, cercetările medicale evidențiază diferite tipuri de solicitări, ca în fig 1.6, unde sunt prezentate următoarele variante :

- a. rectangulare;
- b. crescător în trepte, cu repaus;
- c. crescător în trepte de scurtă sau lungă durată.

Modul de variație în timp a efortului reprezintă unul din elementele de bază ale așa numitului "protocol de desfășurare a probei".

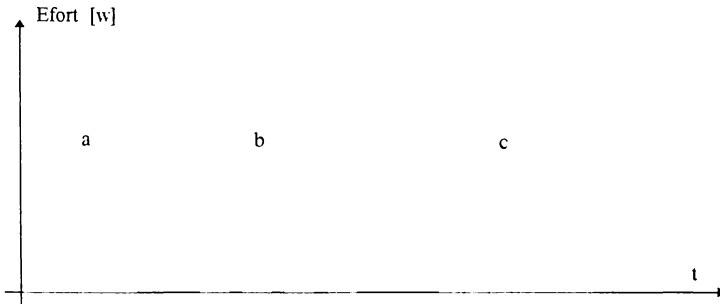


fig. 1.6. Tipuri de solicitare la efort.

Dintre metodele prezentate, metoda intensificării în trepte adecvate a efortului (fig 1.6 - c) este considerată a fi cea mai potrivită scopului urmărit. Cu un astfel de test, efectuat progresiv de la nivele inferioare spre nivele din ce în ce mai ridicate de efort, se pot obține indicații asupra capacității funcționale maxime a pacientului. Durata fiecărui interval depinde de timpul în care pacientul realizează adaptarea circulatorie, după care se poate trece la etapa următoare.

Efectuarea probei de efort în trepte, crescând în intensitate dar cu pauze de "odihnă" între ele, este impusă de măsurarea tensunii arteriale atunci când ea se efectuează manual.

De menționat că efortul pacientului se programează simplu la bicicleta ergometrică, pe când la covorul rulant programarea acestuia presupune modificarea înclinării covorului și a vitezei sale de deplasare în corelație cu greutatea corporală a pacientului și conform protocolului de desfășurare a probei. Acesta este motivul pentru care parametrii "ω" și "α" ai covorului rulant sunt reglabili și pot fi controlați, în scopul controlului puterii de efort P_E dezvoltate de pacient:

$$P_E = P_0 + m g \omega R \sin(\alpha) \tag{1.1}$$

unde:

- P_0 puterea consumată de organism în regim de efort minim ($\alpha = 0$)
- m masa pacientului..... [Kg]

- g accelerația gravitațională.....[m / s²]
- ω viteza unghiulară a cilindrilor de antrenare ai covorului [rad / sec]
- R raza cilindrilor de antrenare..... [m]
- α unghiul făcut de covor cu orizontala..... [rad]

În protocoalele de efectuare a probei de efort este utilizat ca și parametru viteza de deplasare a covorului rulant "v". Exprimarea ei în [Km / h] sau [mile / h], face mai ușoară intuirea de către medic a gradului de efort al pacientului.

$$v = \omega R \tag{1.2}$$

Se constată din relația 1.1 că P_E poate fi modificată prin variația unui singur parametru, α sau ω . Cu toate acestea, practica a arătat că tipurile de efort au un caracter diferit din punct de vedere al organismului, în situațiile în care se fac cu înclinație mare la viteză mică sau cu înclinație mică la viteză mare. De aceea, protocoalele utilizează modificarea în trepte a ambilor parametrii.

Referitor la unitățile de măsură a gradului de efort solicitat pacientului, trebuie menționată utilizarea unei unități de măsură specifice practicii medicale, unitate numită **MET** (prescurtare de la "metabolic equivalent"). 1 MET reprezintă prin definiție, consumul de oxigen al metabolismului bazal. Exprimată în Sistemul Internațional, o treaptă pe scara MET este echivalentă cu aproximativ 30W (Chung, 1984):

Cantitatea de oxigen consumat de organism, în funcție de efortul depus, poate fi plasată pe o scară conform tabelului 1.1, preluat din protocolul "Naughton" (Chung, 1984):

Viteză				Unități MET
	2 mile/h (3,21 km/h)	3 mile /h (4,82 km/h)	3,4 mile/h (5,47 km/h)	
Înclinație (grade)	-	-	-	1
	0	-	-	2
	3,5	0	-	3
	7,0	2,5	2,0	4
	10,5	5,0	4,0	5
	14,0	7,5	6,0	6
	17,5	10,0	8,0	7

Tabelul 1.1

Pentru compararea facilă a rezultatelor testelor cu cele comunicate în literatură, este recomandabilă exprimarea vitezei în mile/h.

1.4 PRIVIRE GENERALĂ A SARCINILOR PENTRU CERCETARE

În schema bloc din fig. 1.7. sunt prezentate sintetic principale sarcini ce revin unei unități de monitorizare ECG pentru proba de efort, sarcini rezultate în urma cooperării cu medicul cardiolog. Ele sunt în acord cu monitorizări propuse în

literatura de specialitate, pentru unități de terapie intensivă pentru coronarieni și clasificate global ca "procesări primare", fiindcă sunt cele care se efectuează aproape de pacient, "la căpătâiul bolnavului".

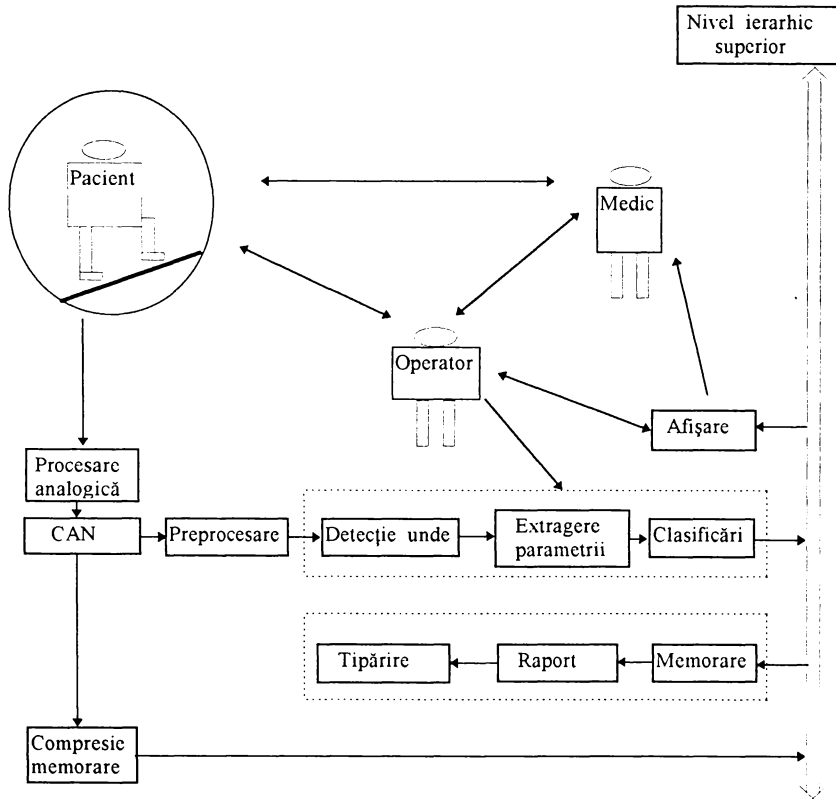


fig. 1. 7. Sarcini la nivelurile de procesare primară.

Prezentarea acestor sarcini, însoțită de scurte comentarii și precizări, are rolul ca de la început de drum să canalizeze, atât documentarea bibliografică, cât și cercetările ulterioare pe un făgaș bun.

1.4.1 Conversia Analog Numerică

Alegerea parametrilor de bază ai CAN, frecvența de eșantionare și numărul de biți nu este guvernată de principii imuabile, ci este cel mai adesea rezultatul unui compromis. Se întâlnesc astfel frecvențe de eșantionare constante, de la 60 Hz la 800 Hz și precizii de la 7 la 12 biți (Merri ș.a. 1988). Deși în literatură sunt

prezentate și soluții de eșantionare neuniformă, sau eșantionare adaptivă (Blanchard și Barr, 1985), am optat pentru o eșantionare cu pas constant.

1.4.2 Preprocesarea

Preprocesarea are ca obiectiv major înlăturarea a tot ceea ce este privit ca perturbație, deci nu provine de la activitatea electrică proprie a cordului, într-un cuvânt "curățarea" semnalului, fără a afecta pe cât posibil informația utilă din acesta. Semnalele biologice sunt de regulă afectate de perturbații (zgomote sau artefacte) de la diferite surse, parte din ele biologice, parte din ele datorate echipamentului de măsură.

- Sursele biologice sunt reprezentate de activitatea musculară (mișcări ale pacientului și respirația).
- Sursele ce nu au origine biologică sunt: variația contactului electrod-pacient, interferența cu rețeaua de alimentare (50 Hz), zgomot al componentelor electronice, zgomot de cuantizare.

Multe din acestea pot fi limitate printr-o pregătire atentă a interfeței cu pacientul. În monitorizarea de lungă durată, este dificil de a menține integritatea semnalului, deoarece pacientul trebuie să se miște iar modificările contactelor electrozi-pacient sunt de neevitat. Situația este prezentă cu atât mai mult în cazul probei de efort, când pacientul este obligat să se miște. Următoarea clasificare a perturbațiilor ține cont de compoziția spectrală a acestora:

- **Perturbații cu o distribuție uniformă a frecvenței**

În cadrul acestora, zgomotul muscular este un exemplu tipic. Tot aici pot fi încadrate forme atipice (artefacte de mișcare). Acestea nu pot fi anulate cu metode generale iar în unele cazuri soluțiile practice de reducere a zgomotului aleatoriu rămân legate de tehnici de mediere (Rompelman și Ross, 1986; Jane și Rix, 1991).

- **Perturbații de înaltă frecvență**

Cea mai comună tehnică folosită pentru reducerea acestora este filtrarea nerecursivă de tip Fourier. Aceste filtre oferă un câștig stabil în interiorul benzii de trecere și caracteristică de fază liniară, ceea ce este cea mai importantă proprietate care permite evitarea distorsiunilor traseului. Aici, există o permanentă dispută între utilizarea unui număr mare de coeficienți reali care asigură însă caracteristici abrupte în zona de tăiere și timpul de calcul. Aceste filtre nu sunt utilizate de regulă pentru procesările în timp real. Această limitare poate fi redusă prin utilizarea unei clase de filtre cu coeficienți întregi (Lynn, 1971; Principe ș.a. 1986)

- **Deriva liniei de bază**

Se poate simula prin însumarea la semnalul ECG a unui semnal de frecvență scăzută iar pentru corecția derivei liniei izoelectrice există două mari categorii de soluții (Grandwohl ș.a. 1989):

În prima categorie sunt incluse interpolări (liniare, neliniare) între câteva puncte de referință, ca de exemplu segmentul PQ, din cicluri consecutive ale ECG. Linia rezultată aproximează deriva liniei de bază și apoi ea este scăzută din semnalul original. Această metodă nu introduce distorsiuni ale formei semnalului, dacă punctele de referință sunt corect alese.

A doua categorie este cea a Filtrelor Trece Sus. Ideea utilizării FTS este considerată convenabilă, chiar dacă se impune o structură complexă de filtru pentru obținerea caracteristicii dorite (McManus ș.a., 1985; van Alste ș.a., 1986). Unii autori consideră că limita inferioară a spectrului util al semnalului ECG și anume 0,5 Hz, recomandată de AHA Committee, nu este cea reală, fiindcă ea a fost stabilită din considerente de liniaritate a filtrelor analogice. Cu FIR se pot obține răspunsuri de fază liniare și la frecvențe mult mai scăzute, dar limita de 0,5 Hz rămâne în continuare reprezentativă și suficientă.

- **Interferențe cu rețeaua de alimentare**

Aceste interferențe cauzează apariția în spectrul semnalului a armonicilor frecvenței rețelei de alimentare, dintre care doar fundamentală (50 Hz) "cade" în spectrul semnalului ECG. Pentru rejecția ei, o primă soluție o constituie utilizarea filtrelor "notch" (Huhta și Webster, 1973; Levcov ș.a. 1984). Acestea sunt filtre numerice, având amplificarea nulă pentru frecvența de 50 Hz. Ca tehnici de proiectare sunt cunoscute transformata Fourier Discretă sau chiar medieri adecvate atunci când frecvența de eșantionare este multiplu întreg al frecvenței rețelei. Metoda permite rejecția totală a zgomotului dar este dificilă realizarea în același timp a unei faze liniare și a unei procesări rapide. O altă soluție este cea numită filtrare optimă sau filtrare adaptivă (Stearns, 1988; Furno ș.a. 1983; Labonski ș.a. 1987; Thakor ș.a. 1991; Laguna ș.a. 1992; Patrick ș.a. 1996).

1.4.3 Detecția undelor

Detecția QRS este punctul central în proiectarea și realizarea oricărui sistem de monitorizare. Literatura în acest domeniu acoperă întreaga perioadă de aplicare a calculatorului în cardiologie (Hamilton și Tompkins, 1986; Thakor ș.a., 1984; Pahlm și Sornmo, 1987). Creșterea accesibilității la tehnica de calcul cu preț tot mai redus a dezvoltat analiza de semnal bazată în totalitate pe tehnici de software. Astfel, dacă în trecut o serie de algoritmi au fost implementați în circuite dedicate datorită necesității de resurse de calcul, azi sunt simplu de implementat în calculatoare aflate la îndemână din punct de vedere al prețului.

Sarcinile unui detector nu sunt simple datorită mării diversități a semnalelor între diferiți pacienți sau chiar la același pacient. Pentru evaluarea performanțelor soluția o reprezintă compararea rezultatelor detectorului cu evenimentele adnotate de medicul cardiolog pe înregistrarea ECG testată. Orice detector include trei secțiuni principale:

1. O transformare liniară având menirea de a accentua proprietățile complexului QRS. O putem denumi și funcție de "evidențiere";
2. O transformare neliniară care să genereze flaguri la coincidența cu evenimentul așteptat, bazat pe o serie de praguri;
3. Criterii de decizie capabile să detecteze dacă sunt într-adevăr "evenimente" QRS, criterii bazate pe recunoașterea trăsăturilor.

Prima parte a detectorului este realizată de obicei printr-un filtru "trece bandă" a căruia ieșire va evidenția acea formă care aduce cele mai multe informații specifice complexului QRS (Ligtenberg și Kunt, 1982; Pahlm și Sornmo, 1984).

Determinarea funcției de "evidențiere" este cea mai importantă operație, de "reușitele" ei depinzând în mare parte calitățile detectorului QRS.

Mulți algoritmi se bazează pe utilizarea de praguri adaptive pentru a putea face față mai bine schimbării caracteristicilor semnalului (Sornmo și Pahlm, 1985). Adaptarea este realizată printr-o "învățare inițială" determinată de specificitatea semnalului pacientului și printr-o "învățare dinamică" pe durata procesului, pentru adaptarea la schimbările semnalului aceluiași pacient. În final, cade în sarcina secțiunii de decizie de a-și adapta regulile în vederea recunoașterii tuturor complexelor, inclusiv în situațiile anormale ca cele legate de aritmii sau cele datorate ischemiilor.

1.4.4 Compresia datelor

Uzual, compresia datelor se referă la tehnici menite să reducă volumul de informație memorată sau transmisă, prin eliminarea redundanței din semnalul ECG, fără a afecta informația utilă.

1.4.5 Extragerea parametrilor

Parametrii cei mai utilizați în analiza ECG, pot fi grupați în două clase:

- parametrii în domeniu timp, potriviți să descrie morfologia semnalului;
- parametrii rezultați în urma unor transformări integrale ale semnalului (cum ar fi transformata Fourier).

a) Parametrii din primă clasă îi vom denumi, "**parametrii descriptivi**", ei constând dintr-un set de valori cu semnificații clinice pentru medic: intervale de timp, amplitudini, suprafețe referitoare la unde, intervale, segmente ale traseului ECG sau pot fi chiar seturi de parametri necesari procesării, ca de exemplu valorile eșantioanelor din ferestre atașate unor evenimente (unde) detectate.

Un exemplu tipic îl constituie definirea limitelor unui complex QRS sau determinarea valorii unui nivel (ST, presiunea sistolică). Aspectul valoros, care compensează dificultățile legate de definirea lor, este faptul că cel mai adesea acești parametrii descriptivi au o semnificație fiziologică precisă și de aici mai este un singur pas pentru diagnosticarea ECG, făcută chiar pe baza acestor parametri descriptivi. Pentru determinarea lor, analiza pretinde un semnal ECG pe termen scurt (câteva zeci de secunde). În acest caz putem controla calitatea semnalului, deoarece și pacientul poate fi menținut într-o poziție stabilă.

Estimarea automată a parametrilor descriptivi este deci de mai mare încredere și astfel, consider că ea reprezintă un pas obligatoriu care poate include o mare parte din experiența valoroasă acumulată în clinică, în interpretarea parametrilor legați de proba de efort.

b) În a doua clasă sunt incluși parametrii rezultați de la transformările Fourier. Walsh, Karhunen-Loeve. (Ahmed și Rao, 1975).

Parametrii din această categorie fiind rezultatul unor transformări integrale, nu depind esențial de precizia procedurii de determinare a lor și de aceea se pretează mai bine metodelor de recunoaștere automată a formelor. Deși nu au o semnificație clinică, ei reprezintă un puternic instrument de investigare a

variabilității diferiților parametri biologici, ca de exemplu prin analiza evoluției spectrelor de putere în timp (Ping și Lin, 1992).

Pe baza acestor considerații problema selecției setului de parametri s-a soluționat preferând pe cei descriptivi, în aplicațiile unde ei au definită o semnificație fiziopatologică și pot fi mășurați cu o precizie suficientă, în vederea diagnosticării. Un exemplu important în acest sens este dat de analiza segmentului ST care are modificări de amplitudine pe durata atacurilor ischemice. Cu unele excepții (Horne, 1984), marea majoritate a studiilor referitoare la analiza segmentului ST sunt realizate (Weisner ș.a., 1982; Akselrod ș.a., 1987) pe baza analizei cu ajutorul parametrilor descriptivi.

Există o serie de alte abordări în scopul de a depăși limitările celor două clase de parametri, printre care amintesc:

- utilizarea unor descriptori geometrici (puncte, curburi) pentru semnalul ECG(Cheng, 1987)
- metode sintactice (Papakonstantinou ș.a., 1981, 1986);

1.4.6 Clasificarea

Clasificarea poate fi definită ca o procedură menită să identifice un obiect ce este cunoscut apriori printr-un set de parametri și despre care la un moment dat putem afla:

- informații statistice referitoare la unii parametri ai săi;
- structura (relații între unii parametri ai săi);
- relații cu alte obiecte.

Obiectivul major al acestei categorii de procesare îl constituie diagnosticarea în sens clinic, pe baza cercetării traseului ECG. În principal există trei categorii de metode în clasificare (Besette, 1989): statistice, sintactice și bazate pe tehnica asocierii.

Din prima categorie, a metodelor statistice, pot fi implementate mai multe variante conform fig.1.8.

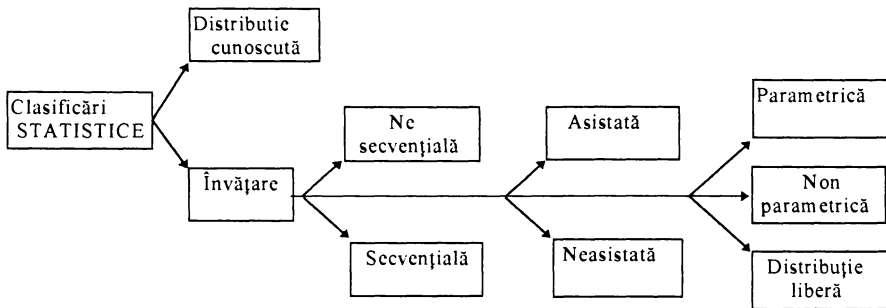


fig. 1.8 Metode de clasificare statistică

Dintre ele, metodele bazate pe o cunoașterea perfectă a distribuției au doar o valoare teoretică, deoarece este practic imposibil de cunoscut parametrii unei distribuții ce caracterizează un "pattern" într-un semnal real, așa cum este cazul ECG. Deci în realitate, putem presupune numai un anumit grad de cunoaștere și de aceea trebuie folosite tehnici de învățare pentru a realiza clasificarea.

Varianta de învățare secvențială este caracterizată de faptul că obiectele intrate în atenție sunt tratate unul câte unul, aceasta permițând clasificarea în timp real, pe când varianta nesecvențială este realizată prin analiza tuturor obiectelor analizate deodată, la sfârșitul etapei de identificare a lor. Învățarea asistată se realizează atunci când este disponibil un set de date de antrenament, considerate de referință, sau când este pusă în practică o procedură interactivă cu un operator uman. O astfel de soluție este indicată în cazul monitorizărilor ECG ambulatorii.

Toate aceste metode de învățare pot fi folosite cu trei tipuri de reguli de decizie: parametrice, non parametrice și cu distribuție liberă. Utilizarea metodelor sintactice presupune reprezentarea "pattern"-ului ca un set de elemente structurale elementare și regula lor de asociere. Procesul de clasificare consistă în acest caz în analiza sintactică a evenimentelor identificate. Rezultatele comunicate sunt încurajatoare (Udupa și Murty, 1980, Skordalakis și Trahanias, 1986) dar metodele sunt pretențioase.

Capitolul 2

STADIUL ACTUAL AL CERCETĂRILOR

Obiectivul acestui capitol este trecerea în revistă, cu referințe bibliografice, a bazelor teoretice și a cercetărilor publicate în literatura de specialitate și care vizează principalele sarcini impuse de realizarea monitorizării probei de efort:

- achiziția de date;
- preprocesare ECG și detecția QRS;
- compresia datelor ECG;

2.1 ACHIZIȚIA DE DATE ECG

În instrumentația medicală, o atenție particulară trebuie acordată achiziției de semnal în scopul evitării zgomotului, interferențelor, derivei liniei de bază și mai ales izolării electrice față de pacient.

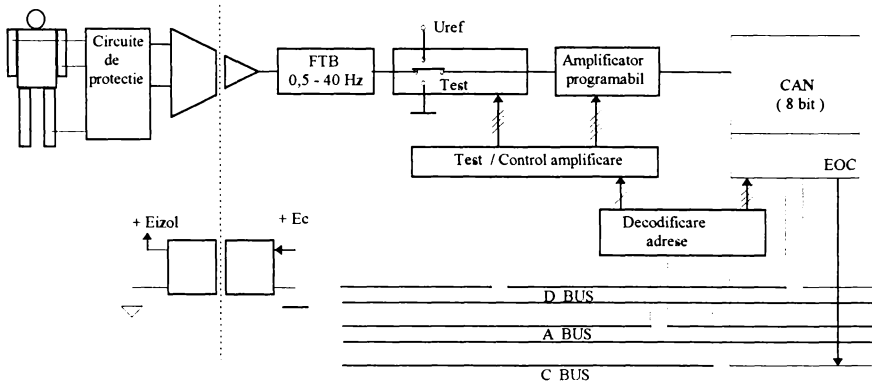


fig. 2.1. Schema bloc a achiziției de date

619.478
366 c

Instrumentația realizată trebuie să răspundă la trei criterii: simplitate, fiabilitate și fidelitate în a înregistra evoluția semnalului. O schemă bloc consacrată pentru achiziția datelor ECG, gestionată de un sistem de calcul, este cea din fig.2.1. Marea majoritate a sistemelor de achiziție publicate sau comercializate se regăsesc în această schemă bloc (Ritter ș.a. 1981; Cupal și Reese 1988; Mișu și Breazu 1992; Mișu 1995, Voiniciuc și Mișu 1996).

Proiectarea tuturor sistemelor de achiziție pornește de la evidențierea caracteristicilor semnalului ECG.

2.1.1 Semnalul ECG

Thakor, Webster și Tompkins (1984) au prezentat studii de referință asupra compoziției spectrale a semnalului ECG normal, provenit de la un cord sănătos precum și spectrele principalelor zgomote: artefacte de mișcare și zgomot muscular (fig. 2.2).

O primă constatare este aceea că spectrul de frecvență al unui cord sănătos se întinde de la frecvențe foarte joase (0.5 Hz) până la aproximativ 40 Hz. La aceasta complexul QRS își aduce cel mai mare aport în intervalul de la 8 la 20 Hz. Undele P și T au o contribuție foarte mică în raport cu cea a complexului QRS, de unde și dificultatea mare în a le identifica direct. O primă concluzie este că, pentru identificarea complexelor QRS, este suficientă o frecvență de eșantionare de 200 Hz.

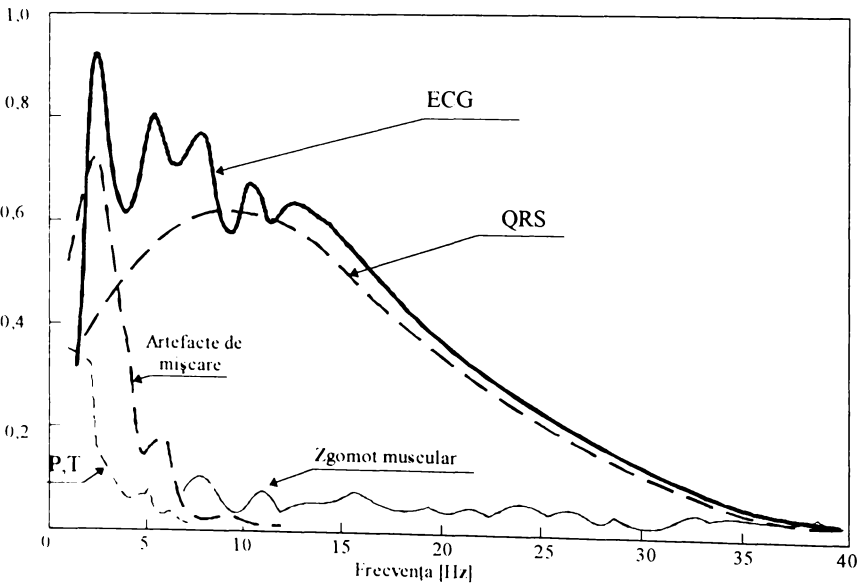


fig. 2.2. Spectrul de frecvență al semnalului ECG, al componentelor sale QRS, P,T, al zgomotului muscular și al artefactelor de mișcare.

Pentru a avea o imagine exactă asupra semnalului real ce ajunge să fie achiziționat, Friesen ș.a. (1990) face o prezentare detaliată a tuturor surselor nedorite care se suprapun peste semnalul provenit din activitatea electrică a inimii.

Interferența rețelei se compune dintr-o componentă de 50 Hz și armonici ale acesteia ce pot fi reprezentate în domeniul timp ca sinusoidă și combinații de sinusoidă cu diferite defazaje și de frecvențe, multipli ai frecvenței de 50 Hz. În spectrul semnalului de interferență cu rețeaua, doar armonica fundamentală "cade" în spectrul semnalului ECG care se ia în analiza preprocesărilor, de aceea zgomotul datorat interferenței cu rețeaua se reprezintă în modele ca o sinusoidă cu frecvența de 50 Hz și cu amplitudine variabilă care poate atinge 50% din amplitudinea semnalului.

Zgomotul dat de contactul electrozilor reprezintă o interferență tranzitorie, cauzată de pierderea temporară a contactului dintre electrod și piele și care are ca efect deconectarea sistemului de măsurare de la subiect. Pierderea contactului cu pielea poate fi intermitentă când se datorează vibrațiilor mecanice. În aceste situații semnalul ECG de măsurat ajunge legat capacitiv la intrarea amplificatorului de măsură și interferența cu rețeaua devine semnificativă. Zgomotul datorat contactului slab al electrozilor poate fi modelat ca o deplasare a liniei izoelectrice, rapidă și aleatoare, care apoi descrește exponențial spre poziția liniei izoelectrice peste care se suprapune și o componentă de 50 Hz. Pentru simularea acestor situații se folosește un impuls de durată 1s, având amplitudinea egală cu amplitudinea înregistrării ECG, și o cădere a palierului cu o constantă de timp de 1s, peste care se suprapune o armonică de 50 Hz. de amplitudine variabilă.

Artefactele de mișcare sunt modificări tranzitorii (dar nu în treaptă) ale impedanței electrod - piele, date de mișcarea electrodului relativ la suprafața pielii, însă cu menținerea contactului. În acest caz "sursa" de semnal ECG va fi divizată prin noua impedanță, astfel că la intrarea amplificatorului va ajunge un semnal util redus. Cauza obișnuită a acestor artefacte este mișcarea pacientului. Pot fi simulate spre exemplu prin modificarea liniei izoelectrice cu o formă de undă provenită prin redresarea bialternantă a unei sinusoidă, care trebuie să fie însoțită de o atenuare corespunzătoare a semnalului util. Amplitudinea și durata artefactului sunt variabile dar tipic, pentru simulare, se folosesc durate de 100 - 500 ms iar amplitudinea de până la 200% din amplitudinea QRS mediu.

Activitatea musculară produce semnale cu un spectru larg de frecvență și amplitudine redusă. Aceste semnale, datorate atât contracțiilor musculilor netezi cât și celor striati, pot fi approximate ca trenuri de impulsuri de zgomot Gaussian de bandă limitată și de valoare medie nulă. De menționat că acest gen de zgomot nu afectează niciodată linia izoelectrică. Parametrii tipici ai semnalului sunt: durata 50 ms; conținutul de frecvențe până la 10 kHz; amplitudinea până la 50% din amplitudinea ECG vârf la vârf.

Deplasarea liniei izoelectrice poate fi reprezentată de o componentă sinusoidală cu frecvența scăzută adăugată semnalului ECG. Cauza acestui gen de deplasare a liniei izoelectrice o constituie respirația, efectul fiind o "modulație" a amplitudinii ECG cu frecvența actului respirator ($0,15 \text{ Hz} \div 0,5 \text{ Hz}$). Datorită modificărilor de volum ale toracelui amplitudinea semnalului ECG variază cu aproximativ 15% pe durata unui ciclu respirator.

Alte zgomote care apar ocazional sunt: zgomotul electrochirurgical (100 kHz \div 1 Mhz) și zgomotul produs de defibrilatoare în acțiune sau de "pace maker".

2.1.2 Amplificarea semnalului ECG

Semnalul ECG reprezentând o diferență de potențial între puncte predefinite, implică utilizarea unui amplificator de tip instrumental (Jurca, 1996), prezăcut cu reacție selectivă, astfel încât caracteristica sa globală de frecvență să faciliteze eliminarea zgomotelor și interferențelor menționate (Ciocloada, 1981). O dată amplificat, semnalul este transferat printr-o cale ce realizează izolația galvanică față de pacient spre blocul de Conversie Analog Numerică (CAN). Caracteristicile amplificatorului, precizate de AHA (American Heart Association) și IEC (International Electrotechnical Commission) sunt următoarele:

- intrare diferențială;
- ieșire cu referință la masă;
- câștig în tensiune: 60 dB;
- bandă de trecere: 0,1-40 Hz;
- CMMR (factor de rejecție a modului comun) >80 dB;
- imunitate la radiofrecvență ;
- impedanță de intrare diferențială $Z_i = 6 \div 10 \text{ M}\Omega$. Deși există dispozitive semiconductoare și amplificatoare de măsură capabile să asigure impedanțe de ordinul 10 G Ω , impedanța optimă de intrare Z_i trebuie să fie

$$Z_e < Z_i < Z_p \quad (2.1)$$

unde Z_e este impedanța tipică a unui electrod iar Z_p este impedanța de cuplaj între pacient și sursa de interferențe (Winter și Webster 1983);

- condiții severe în ceea ce privește izolarea galvanică între pacient și partea de monitorizare (>2kV), corespunzătoare clasei I, conform încadrării date de IEC în Publication No. 601-1 "Safety of Medical Electrical Equipment". Se ține cont de faptul că mai multe aparate alimentate de la rețea pot fi conectate simultan pe pacient (Cook și Webster 1980). În literatura de specialitate și în implementările comerciale se întâlnesc soluții de izolare prin cuplaj capacitiv, optic sau electromagnetic, acestea din urmă rezolvând simultan și problema alimentării cu energie electrică a părții electronice izolate;
- protejarea intrărilor la vârfuri de tensiune produse de defibrilatoare sau încărcare electrostatică. Se poate realiza practic cu ajutorul unor tuburi eclatoare cu gaz (neon), sau cu circuite de limitare cu diode. Pe lângă protecția pacientului la eventuale accidente prin electrocutare, izolarea galvanică asigură eficient și protecția părții de achiziție împotriva impulsurilor de tensiune provenite de la un defibrilator.

2.1.3 Condiționări ale semnalului

Conversia analog-numerică necesită aducerea nivelului semnalului analogic la un nivel adecvat. Pentru aceasta trebuie asigurată o filtrare analogică și o ampli-

ficare adaptativă. Filtrul este de tip FTB (cel mai indicat Butterworth de ordinul 2) și prezintă o bandă de trecere $0,5 \div 40$ Hz. Rolul lui este de a adapta caracteristica globală de frecvență cu spectrul semnalului ECG.

Rolul amplificatorului programabil este de a asigura semnalului analogic o dinamică între limite impuse de convertorul analog numeric. Un amplificator programabil asigură reglarea amplificării, fiind adresat direct de unitatea de centrală. Fără excepție, amplificatoarele programabile sunt realizate pe principiul modificării elementelor rețelei de reacție negativă a unui amplificator operațional, cu ajutorul unor comutatoare electronice comandate de un registru latch de control, a cărui setare se face soft prin intermediul magistralei de date (fig. 2.1). Tensiunea de ieșire depinde de tensiunea de intrare după relația:

$$V_{eș} = A_{(N)} \cdot V_{in} \quad (2.2)$$

Unde $A_{(N)}$ poate lua 2^N valori discrete. S-a constatat că este suficient un număr de $4 \div 8$ trepte de amplificare, ceea ce poate fi realizat cu $N=3$ biți transmiși amplificatorului programabil.

2.1.4 Interfața

Un bus intern de 8 biți este direct accesibil memoriei sau porturilor I/O. Două componente sunt adresabile direct: convertorul analog-numeric și registrul de control. Rezultatul conversiei (eșantionul de intrare) se citește direct pe bus-ul de date intern. Registrul de control mai poate asigura comenzi necesare unui lanț de achiziție performant: calibrarea cu o tensiune de referință (1 mV), aducerea la zero a amplificatorului, pentru o ulterioară corecție soft a tensiunii de offset.

Alte arhitecturi permit: compensarea rapidă a derivei liniei de bază, programarea frecvențelor de tăiere ale filtrului analogic, schimbarea manuală sau automată a derivațiilor de înregistrare.

Semnalul EOC (End Of Conversion) furnizat de convertor se poate utiliza fie pentru modul de lucru prin întreruperi, fie pentru citirea de către calculator a stării CAN.

2.1.5 Cuantificarea semnalului

Mai mulți factori determină nivelul de cuantificare (numărul de biți al CAN), între care: conservarea tuturor informațiilor din semnalul ECG, o dinamică suficientă a semnalului analogic, nivelul de zgomot propriu al amplificatorului. În literatură se apreciază că numărul de biți necesari CAN poate fi între 6 și 10 în funcție de cerințele aplicației, pentru monitorizarea aritmiilor fiind des întâlnită cuantificarea pe 8 biți, considerată suficientă. Pentru alte aplicații cum ar fi studiul post-potențialelor, necesarul de biți crește. Jalaliddine (1990) propune o formulă pentru determinarea numărului minim de biți necesari procesării.

$$N_{\min} \geq \frac{20 \log \frac{V_{ivv}}{V_{ecg}} + 20 \log \frac{V_{ecg}}{q}}{6} \quad (2.3)$$

unde

- V_{viv} - tens. vârf-vârf permisă de intrarea CAN
- V_{ecg} - amplitudinea QRS (în general 1 mV vârf-vârf)
- q - rezoluția dorită (variația semnalului corespunzător LSB).

Nu are sens creșterea numărului de biți peste valoarea la care zgomotul de cuantificare ar scădea sub valoarea zgomotului echivalent total din semnalul ce trebuie cuantificat. De aceea, valoarea maximă a numărului de biți se determină punând condiția ca valoarea efectivă a zgomotului de cuantificare să fie egală cu zgomotul semnalului. Valoarea efectivă totală a zgomotului la intrarea CAN.

$$V_{ZG_TOT}^2 = V_{ZG}^2 + V_{ZG_EL}^2 \quad (2.4)$$

unde:

- V_{ZG} - tensiunea efectivă de zgomot a amplificatorului ECG, datorată în principal componentelor electronice ale amplificatorului.
- V_{ZG_EL} - tensiunea de zgomot datorată electrozilor.

$$N_{max} < \frac{20 \log \frac{V_{viv}}{V_{ZG_TOT}}}{6} - 1 \quad (2.5)$$

În concluzie, numărul de biți pe care se face conversia analog-numerică trebuie ales între limitele impuse de relațiile (2.3) și (2.5). În alegere, un factor de care trebuie ținut cont îl constituie utilizarea eficientă a resurselor hard și soft. De aceea, așa cum rezultă din literatura de specialitate, sunt întâlnite rezoluții de la 7 la 12 biți. În cazul monitorizării însă, nu este necesar un nivel de cuantificare superior lui n=8. Aceasta face posibilă achiziția și prelucrarea datelor rapid și eficace, vehiculând soft informația ca variabile BYTE (un octet) în partea de achiziție.

2.1.6. Frecvența de eșantionare

Alegerea frecvenței de eșantionare se face în funcție de nivelul de cuantificare, de frecvența superioară F_s din spectrul semnalului ECG și de condițiile impuse de filtrul trece bandă din lanțul de achiziție. Din punct de vedere al ponderii în densitatea spectrală de putere se arată că frecvența maximă reprezentativă este F_s = 40 Hz. (Thakor ș.a. 1984), pentru un cord sănătos. În cazuri patologice însă (postpotențialele ventriculare, fibrilație) pot apare frecvențe mult mai mari (sute de Hz).

Jalaleddin ș.a.(1990) determină pentru frecvența minimă de eșantionare o condiție mai restrictivă decât cea impusă de teorema Shanon:

$$F_e > 2 \cdot F_s \cdot \frac{1}{\log \frac{6 \cdot N - K_{ecg} \cdot \log \frac{F_s}{F_{ecg}}}{K_{ecg} + K_f}} \quad (2.6)$$

unde: F_e - frecvența de eșantionare;
 F_s - frecvența superioară de tăiere (-3dB) a filtrului;
 F_{ecg} - frecvență maximă superioară din spectrul ECG;
 K_{ecg} - panta caracteristicilor spectrului din zona de tăiere exprimată în dB /dec. (aproximativ 60 dB/dec);
 K_f - panta caracteristicii filtrului, în zona de tăiere exprimată în dB/dec;
 N - reprezintă numărul de biți pentru cuantificare.

Se poate concluziona că, pe lângă nivelul de cuantificare, alegerea frecvenței de eșantionare este un factor esențial în atingerea scopului final, de ea depinzând atât acuratețea reprezentării numerice a semnalului, cât și performanțele de procesare ulterioară a acestuia.

2.2 DETECȚIA QRS

În orice analiză a înregistrării ECG, detecția complexului QRS reprezintă primul pas, obligatoriu și necesar. Complexul QRS este cea mai importantă caracteristică a ECG, prezentând o mare amplitudine și forme demult studiate în medicina clasică. Cu toate acestea, este poate cea mai provocatoare problemă cu care orice cercetător al ECG se confruntă.

2.2.1. Clasificări

În ultimii ani clasificarea detectoarelor QRS a fost subiectul multor lucrări. Clasificările propuse nu sunt satisfăcătoare în totalitate, căci încercările de generalizare în problematica detecției QRS nu pot cuprinde diversitatea principiilor și tehnicilor folosite.

Pahlm și Sornmo (1984), propun o soluție unanim acceptată de împărțire a detectorului QRS în două părți: preprocesarea și regula de decizie.

Gritzali (1988) folosește cea mai generală clasificare a detectoarelor: non-sintactice, sintactice și hibride și face o generalizare matematică a transformărilor neliniare propuse anterior de Pahlm și Sornmo.

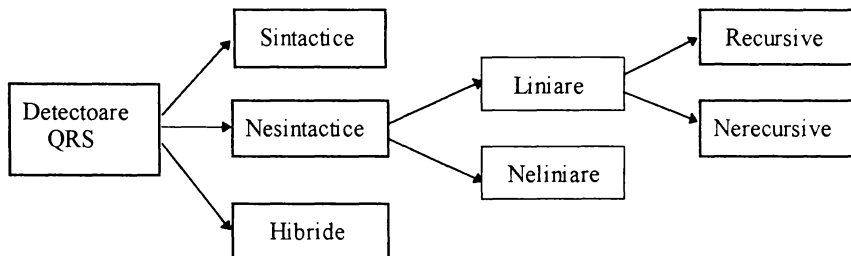


fig. 2.3 Clasificarea detectoarelor QRS (Gritzali 1988)

Jalaledine și Hutchens (1987) au propus gruparea principiilor de detecție QRS în două categorii: în prima parte sunt cuprinse detectoarele adaptive și neadaptive împreună cu criteriile de decizie pentru recunoașterea QRS iar în a doua parte ar fi cuprinse toate detectoarele care folosesc domeniul de frecvență pentru filtrare precum și tehnicile de corelare.

Deși există și alte încercări de a încadra diferitele tipuri de detectoare în anume clasificări (de exemplu Friesen ș.a., în 1990, identifică și compară nouă tipuri de detectoare semnificative), considerăm ca suficient de acoperitoare clasificarea făcută de Gritzali (1988) și o prezentăm în sinteză în fig. 2.3.

De remarcat că, în nici o referire la clasificări ale detectoarelor QRS, nu se insistă asupra existenței unui criteriu de clasificare "utilizarea în timp real", și aceasta deoarece acest criteriu nu este astăzi unul de principiu de funcționare ci unul de comparare a detectoarelor între ele, legat de timpul de calcul și resursele necesare implementării lor.

2.2.2 Metode folosite în detecția QRS

Ne propunem în continuare o trecere în revistă a principalelor realizări publicate, fără a avea pretenția de a epuiza nici pe departe domeniul. Principala sursă de inspirație în a realiza algoritmi numerici de detecție a complexului QRS este prelucrarea analogică a semnalului constând de obicei dintr-un filtru trece bandă urmat de un detector liniar sau neliniar cu prag (Ciocloada, 1981).

Detecția simplă a QRS. A fost implementată într-un spectru larg de tehnici mergând de la simple praguri neadaptive (Mora 1984) până la tehnici probabilistice (Carrault 1992).

În cele mai multe cazuri căutarile au mers în următoarele direcții:

- asigurarea operării în timp real
- stabilitate în localizarea punctului fiducial
- modularitate și siguranță pentru procesarea ulterioară

Astăzi detecția QRS este considerată bună dacă asigură un procent de siguranță mai bun de 99 % în situația de raport semnal/zgomot ridicat (Mark și Moddy, 1988).

Detecții QRS multiple. Este folosită în analiza ortogonală dar și în sistemele de diagnostic standard în detecția potențialelor tîrzii (Berbari, 1992). În aceste cazuri detecția QRS poate combina informațiile provenite de la mai multe canale simultan pentru a amplifica complexul QRS. Sunt și situații în care există mai multe canale de achiziție dar detecția se face pe unul singur și anume este selectat cel cu raportul semnal/zgomot cel mai redus.

Un aspect interesant ce mai trebuie menționat este acela că în categoria procesărilor cu scop de filtrare pentru ECG, nu sunt utilizate metode matematice consacrate cum ar fi: transformata Fourier, transformata Fourier discretă, transformata Z, transformata Biliniară. Deși acestea pot atinge performanțe deosebite în ceea ce privește caracteristica de frecvență realizată, motivul pentru care nu sunt folosite este acela că utilizând calcule în virgulă flotantă, este necesar un volum mare de calcul spre deosebire de algoritmi de calcul specifici în analiza ECG, care necesită un volum redus de calcul, fiind implementați cu coeficienți întregi

2.2.3 Preprocesare

Din punctul de vedere al schemei propuse de Palm și Sornmo, obiectivul preprocesărilor îl constituie obținerea unui "vârf" pozitiv pentru fiecare QRS, vârf care să permită ulterior implementarea de reguli de decizie simple. Pentru aceasta trebuie ca în preprocesarea semnalului ECG, să fie utilizate metode care să evidențieze puternic caracteristicile complexului QRS în detrimentul celorlalte componente ale semnalului: interferențe, artefacte, deriva liniei de bază, nivelele undelor P și T, etc (Shaw și Savard 1995). O încercare de clasificare a metodelor de preprocesare ar putea fi următoarea:

- transformări liniare;
- identificarea punctului fiducial;
- transformări neliniare;
- corelația cu o formă "șablon";
- modelarea complexului QRS.

2.2.3.1 Transformări liniare

Așa cum s-a precizat mai înainte, spectrul semnalului ECG este compus din contribuția undelor P, QRS, T, la care se adaugă interferențele de joasă frecvență, produse de artefactele de mișcare și respirație, zgomotul muscular și altele. Aceste date cunoscute apriori, referitoare la spectrele semnalelor și zgomotelor, sunt exploatare în identificarea evenimentului QRS și constituie baza de rezolvare a problemei detecției în multe articole.

Lucrări anterioare au concluzionat că cea mai mare parte a energiei conținute în QRS cade în banda 8-20 Hz (Brydon 1976; Thakor ș.a., 1984), în timp ce celelalte interferențe sunt scăzute în acest interval. Conținutul de amplitudini mari și conținutul de frecvențe înalte sunt reprezentate în domeniul timp prin pante cu creșteri rapide și cu deflexii importante. Din aceste rațiuni, au fost utilizate în determinarea QRS-ului metode care folosesc în parte metode de filtrare liniară combinate cu diferențierea semnalului în domeniul timp. Astfel, primele detectoare folosesc simple diferențiatoare "2 puncte" care au într-adevăr o viteză de calcul foarte mare (Holsinger ș.a. 1971). Folosirea acestei metode simple de procesare este evidențiată și de clasificările consacrate (Friesen ș.a., 1990). Aceste diferențiatoare, foarte sensibile la zgomot de înaltă frecvență sunt o aproximare "săracă" a derivatei matematice (Alshtröm și Tompkins, 1985). În ideea de a îmbunătăți sensibilitatea la zgomot a diferențiatorului simplu "2 puncte", el este urmat în multe aplicații de un FTJ.

Tompkins și Webster (1981), propun o metodă "trei puncte de diferență centrată" pentru implementarea diferențiatorului cu bandă limitată pentru a obține un răspuns în frecvență mai bun decât diferențiatorul anterior și reduc numărul de calcule folosind un număr mic de coeficienți. Această configurație duce la folosirea unui FTB cu o lărgime de bandă de 12-21 Hz cu o frecvență centrală de 17 Hz.

Thakor ș.a. (1984), Ruiz ș.a. (1984), Laguna ș.a. (1990), Usui și Amidror (1981) au prezentat metodologii pentru FIR (FTJ) și diferențiatoare pentru aplicații în domeniul biomedical unde, apriori, se cunoaște spectrul semnalului util și cel al zgomotului.

Usui și Amidror (1982), Ligtenberg și Kunt (1983) au propus o clasă de "diferențiatore trece jos" care combină diferențierea (FTS), cu o mediere variabilă pentru netezire (FTJ). Având coeficienți întregi (-1,1) algoritmi propuși sunt mult mai rapizi decât "greoiul" calcul cu virgulă flotantă sau întregi foarte mari (double int.), pretins de soluțiile de filtrare numerică clasice.

O altă sarcină a preprocesării este și medierea mai multor cicluri ale semnalului (Jane și Rix, 1991). Medierea se face atunci când se are în vedere o ulterioară analiză mai profundă a ECG, ca de exemplu în analiza morfologică a undelor P sau T. Scopul medierii este de a reduce cât mai mult zgomotul aleatoriu din traseul ECG și de a releva caracteristicile provenite din activitatea electrică a cordului. Alegerea ciclurilor cardiace pentru mediere se face astfel încât bătăile ectopice să fie excluse. Medierea presupune identificarea în prealabil a unor puncte reprezentative. Sunt folosite în general pentru mediere 10 -20 de bătăi.

2.2.3.2 Identificarea punctului fiducial (Extragerea punctelor esențiale)

Punctul fiducial se definește, în cazul ECG, ca punct de referință al unei evoluții R-R (un ciclu cardiac) astfel ales încât toate celelalte cicluri să poată fi aliniate avându-l ca referință.

Foarte mulți autori au creat metode pentru identificarea și extragerea unor puncte importante. De exemplu, vârfurile ECG au fost studiate prin metoda "maximelor și minimelor locale" (Dumpala ș.a. 1982, Skordalakis și Trahanias 1987, Horowitz 1977). Metoda intitulată "fereastră glisantă cu trei puncte" aplicată secvenței de intrare localizează de asemenea maxime și minime locale (Escalona 1986, 1993).

Odată punctele extrase, poate fi făcută liniarizarea ECG prin seturi de segmente de dreaptă și astfel semnalul va fi complet determinat prin lungimi de segmente și unghiuri (pante) ale acestora. O condiție de bază în realizarea acesteia este orizontalitatea liniei de bază. În continuare, se utilizează detecția standard folosind praguri adaptive sau neadaptive pentru identificarea complexelor QRS (Hsiung ș.a. 1989). O idee interesantă este aceea ca după identificarea punctelor esențiale, să se utilizeze și o tehnică de corelare cu o formă șablon (Cheng ș.a. 1987), soluție prezentată mai pe larg în paragrafele următoare.

După extragerea punctelor esențiale, forma de undă poate fi reprezentată prin șiruri de simboluri care să descrie pozițiile și caracteristicile acestor puncte. Urmează ca apoi complexul QRS să fie descris în termeni structurali cum sunt: unghiuri, curbe, vârfuri, segmente (Lee ș.a. 1987). Aceste caracteristici pot fi explorate apoi prin metode sintactice. Una dintre acestea, "recunoașterea paternurilor" descrie semnalul în termeni de paternuri primitive, folosind o gramatică în mod similar ca în limbajele de programare (Giakoumakis ș.a. 1987). Ca exemplu alfabetul [PZ; PP; PN; PM; PS] poate reprezenta un șir de linii a căror pantă este nulă (PZ), pozitivă (PP), negativă (PN), mare (PM), scăzută (PS). Astfel în acest limbaj un triunghi se descrie în felul următor: [PZ; PN; PP; PZ] (Birman 1983).

După codare, forma de undă va fi reprezentată deci prin șiruri de caractere ale alfabetului ales. Clasa șirului, corespunzătoare formei de undă codificate, poate fi recunoscută prin căutarea de paternuri particulare în interiorul semnalului codificat. Prima încercare de folosire a analizei sintactice

a ECG a fost în domeniul analizei ritmului cardiac (Belforte 1979, Udupa și Murthy 1980). Alți autori au adus contribuții la aceste propuneri inițiale, în special prin adăugarea de reguli semantice la regulile sintactice (Skordalakis 1984; Papakonstantinou 1981, 1986; Trahanias 1989, 1993). Pasul următor îl constituie aplicarea tehnicilor de inteligență artificială la analiza ECG.

2.2.3.3 Transformări neliniare

S-au dovedit ca eficiente metodele zise neliniare, care operează asupra secvenței ECG cu algoritmi precum: modul, ridicare la pătrat, radical (ordinul 2), diferențiere sau medieri prin transformările "lungime" sau "energie" (Murthy și Rangaraj 1979, Taylor 1979, Okada 1979, Fraden și Neuman 1980, Nygard și Sornmo 1982, Ligtenberg și Kunt 1983, Alhstrom și Tompkins 1985). Alte aplicații care au implementat "filtrări prin mediere" au fost realizate (Yu ș.a. 1985, Chu și Delp 1989) pentru identificarea QRS.

Tehnicile de transformare neliniară, așa cum rezultă din bibliografia citată, sunt frecvent utilizate în detecția QRS din cauza complexității scăzute, a numărului scăzut de calcule, a eficienței mari și a șansei mari de a fi utilizate la procesarea în timp real, fiind indicate pentru aplicațiile pe calculatoare PC. În figura 2.4 sunt prezentate sintetic câteva variante întâlnite în literatură.

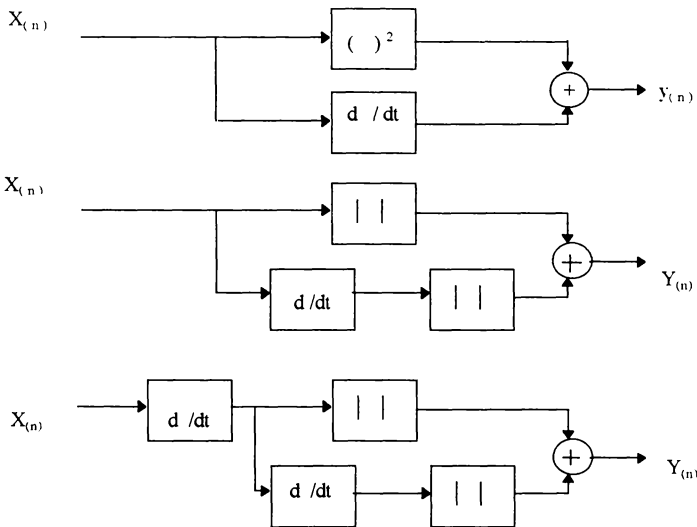


fig. 2.4. Transformări neliniare în detecția QRS:

Prima variantă (Taylor, 1979) însumează pătratul eșantionului curent cu derivata de ordinul 1 a semnalului de intrare. A doua (Nygard și Sornmo, 1983) însumează modulul semnalului de intrare cu modulul derivatei de ordinul 1.

Alhstrom și Tompkins (1983 și 1985), utilizează a treia combinație de procesări neliniare. De menționat de asemenea, că nici o dată tehnicile neliniare nu apar singure, ci în combinații cu cele liniare.

O problemă interesantă o constituie analiza spectrală aferentă unei transformări neliniare. Orice transformare liniară aplicată unui semnal având un spectru de frecvențe dat, are ca efect modificarea spectrului în sensul că pot fi amplificate sau atenuate doar frecvențele existente în spectrul inițial. Spre deosebire de acestea, transformările neliniare duc la apariția de noi componente spectrale.

2.2.3.4 Corelarea cu o formă "șablon" (template matching)

Obiectivul acestei transformări este acela de a compara eșantion cu eșantion o formă de undă necunoscută cu o formă de undă cunoscută ("șablon") și este apropiat modului uman de a identifica cu ochiul liber o formă de undă cunoscută într-o înregistrare unidimensională.

O primă variantă se bazează pe proprietățile funcției de corelație (Collins și Arzbaeher 1981). Fiind dat un semnal zis "șablon" $S(n)$ și un semnal de studiat $X(n)$, funcția de corelație dintre semnal și șablon este dată de relația:

$$C(n) = \sum_{k=1}^N S(k) \cdot X(k-n) \quad (2.7)$$

unde N = numărul de puncte al semnalului "șablon".

Maximul relației (2.7) semnifică cea mai mare asemănare dintre funcție și șablon. Dacă prelucrarea se face în timp real, atunci implementarea acestei funcții se face deplasând fereastra ce conține șablonul în ritmul achiziției, la fiecare nou eșantion calculându-se funcția $C(n)$. Procedura este considerată greoaie din punctul de vedere al numărului de calcule matematice dacă șablonul conține multe puncte (Dobbs și a. 1984). Dacă la aceasta se mai adaugă și memorarea a mai multe forme "șablon", atunci în timp real funcția de corelație presupune înmulțirea foarte rapidă a unor vectori de mari dimensiuni. Jalaledine și Hutchens (1987), folosind mai multe procesoare legate în paralel și în care fiecare procesor este desemnat să detecteze un singur "șablon", au realizat procesări pentru identificare de diverse patologii în înregistrarea ECG.

Limitarea conturului este o altă cale pentru implementarea corelației cu un șablon dat (Goovaerts și a., 1976; Van den Akker și a., 1981). De această dată șablonul nu mai este un contur singular, bine precizat, ci este un domeniu cu limite superioare și inferioare, ca în fig 2.5. În acest caz nu înmulțirile și adunările sunt cele care ocupă rolul central, ci comparațiile care sunt făcute între semnal și limitele conturului șablonului.

Pentru implementarea practică, cele două șabloane sunt construite ca două tablouri de date de lungime N iar regula de decizie a apariției complexului QRS este

$$C_{sup}(N-k) > X(n-k) > C_{inf}(N-k) \quad (2.8)$$

unde: $X(n)$ reprezintă eșantionul curent al semnalului ECG cercetat, $C_{sup}(k)$ este conturul superior impus iar $C_{inf}(k)$ conturul inferior impus. Relația 2.8 trebuie iterată pe întreaga lungime a șablonelor, deci pentru $k = 0 \div N-1$

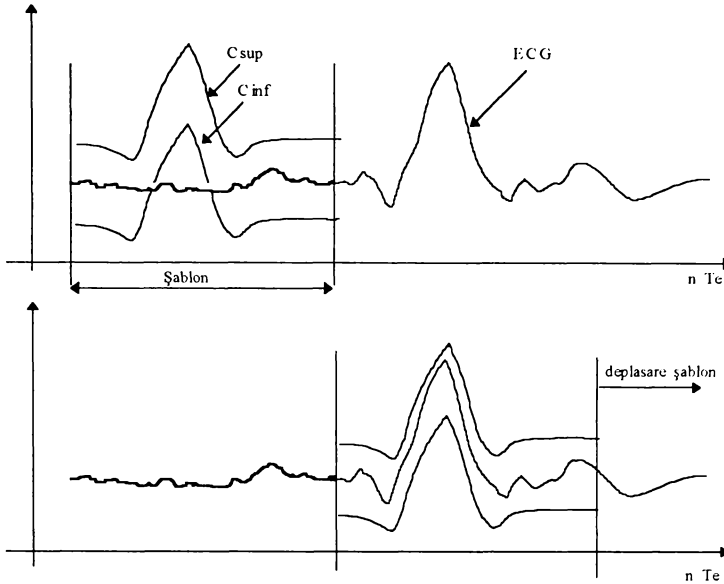


fig 2.5. Algoritm de "limitare a conturului" pentru detecția QRS:

Această diversitate de posibilități de a încadra șabloane QRS normale sau patologice, face ca metoda să fie o bună alternativă la alte soluții mai sofisticate.

2.2.3.5 Modelarea semnalului ECG

Dacă este nevoie de o mare cantitate de informație despre QRS atunci trebuie folosite alte tehnici care să dezvăluie și alte proprietăți ale înregistrării ECG. Astfel modelarea ECG devine necesară când se cere detecția undelor P sau T. Aceasta nu exclude însă utilizarea modelării și la detecția QRS. Scopul modelării este reducerea proceselor complicate, cu mulți parametri, la unul simplu cu un număr mic de parametri (McClellan,1988). Un semnal poate fi modelat ca ieșire a unui sistem liniar, invariant în timp sau adaptiv, sistem condus printr-un semnal fix la intrare. Dacă de exemplu un eșantion unitar este folosit ca excitație, la ieșire va apare răspunsul la impuls al sistemului, care va fi chiar semnalul modelat. Modelarea poate fi făcută direct, așa cum se vede în figura 2.6, în care parametrii funcției $H(z)$ vor fi aleși astfel încât să minimizeze eroarea de modelare

$$A(z) = 1 + \sum_{k=0}^p a_k z^{-k} \tag{2.9}$$

$$B(z) = \sum_{k=0}^q b_k \cdot z^{-k} \quad (2.10)$$

Acest procedeu implică o mare cerere de putere de calcul, de aceea au fost dezvoltate un număr mare de metode indirecte (Stearns, 1988). Mai mult, deoarece semnalele sunt nestaționare și nu pot fi modelate numai cu un singur sistem invariant $H(z)$, este adesea posibil să fie modelate mici porțiuni din semnal, ceea ce implică însă ca parametrii modelului să se schimbe de la o porțiune la alta. Filtrul adaptiv reprezintă un exemplu în care coeficienții modelului evoluează în timp după o metodă dată, pentru a minimiza eroarea.

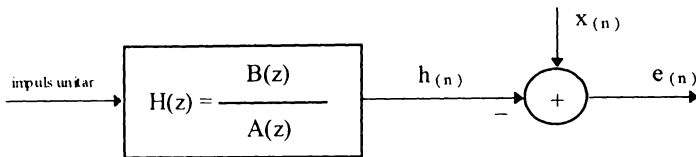


fig. 2.6 Schema bloc a unui modelator direct

Murthy și Rangaraj (1979) au modelat ECG după acest model, vizând în același timp și problema sintezei semnalului ECG. Predicția liniară a fost de asemenea utilizată pentru extragerea proprietăților ECG și identificarea QRS (Lin și Chang 1987). Mai recent Carault și Senhadji (1990), consideră semnalul ECG ca o suprapunere de părți staționare, fiecare având propriul model. Sarcina dificilă este acum detectarea schimbărilor dintr-un anumit model. Rezultatele sunt promițătoare pentru detecția undelor P și T.

2.2.4 Prezentarea aparatului matematic

În acest paragraf vor fi prezentate mai multe filtre întâlnite în studiul bibliografic, însoțite de o sinteză a aparatului matematic utilizat. Se vor prezenta funcția de transfer $H(z)$ și relația de implementare a filtrului, iar acolo unde este cazul, particularități ale folosirii aceluși tip de filtru.

Acești algoritmi elementari pot constitui o bună sursă de inspirație mai ales pentru realizarea de preprocesări rapide. Tot așa de bine se pretează a fi înglobate în structuri de filtrare mai complexe, ca de exemplu în filtre adaptive.

2.2.4.1 Filtre Trece Sus

- Derivata de ordinul 1. Soluția reprezintă implementarea derivatei "clasice" de ordinul 1 definită pe secvența numerică discretă X_n :

$$H(z) = (1-z^{-1})/T \quad (2.11)$$

$$Y_n = X_n - X_{n-1} \quad (2.12)$$

- Derivata de ordinul 1 "trei puncte" (Ahlstrom și Tompkins, 1985)

$$H(z) = (1-z^{-2})/T \quad (2.13)$$

$$Y_n = X_n - X_{n-2} \quad (2.14)$$

- Derivata de ordinul 1 "cinci puncte" (Pan și Tompkins, 1985)

$$H(z) = (-z^{-4} - 2z^{-3} + 2z^{-1} + z) / 8T \quad (2.15)$$

$$Y_n = (-X_{n-3} - 2X_{n-2} + 2X_{n-1} + X_n) / 8 \quad (2.16)$$

- Derivata de ordinul 2 "cinci puncte" (Perriyalwar ș.a., 1990)

$$H(z) = (z^{-4} - 8z^{-3} + 8z^{-1} - z) / 12T \quad (2.17)$$

$$Y_n = (X_{n-3} - 8X_{n-2} + 8X_{n-1} - X_n) / 12 \quad (2.18)$$

- Derivata de ordinul 2 "șapte puncte" (Perriyalwar ș.a., 1990)

$$H(z) = (-3z^{-6} - 2z^{-5} - z^{-4} + z^{-2} + 2z^{-1} + 3z) / 28T \quad (2.19)$$

$$Y_n = (-3X_{n-6} - 2X_{n-5} - X_{n-4} + X_{n-2} + 2X_{n-1} + 3X_n) / 28 \quad (2.20)$$

- Derivata de ordinul 3 "șapte puncte" (Perriyalwar ș.a., 1990)

$$H(z) = (22z^{-6} - 67z^{-5} - 58z^{-4} + 58z^{-2} + 67z^{-1} - 22z) / 252T \quad (2.21)$$

$$Y_n = 22X_{n-6} - 67X_{n-5} - 58X_{n-4} + 58X_{n-2} + 67X_{n-1} - 22X_n / 252 \quad (2.22)$$

2.2.4.2 Filtre Trece Jos

- Filtrul de mediere numit Hanning (Ahlstrom și Tompkins 1985)

$$H(z) = (z^{-2} + 2z^{-1} + z) / 4T \quad (2.23)$$

$$Y_n = (X_{n-2} + 2X_{n-1} + X_n) / 4 \quad (2.24)$$

Acest filtru realizează o mediere a trei eșantioane, centrată pe cel din mijloc.

- FTJ recursiv (Pan și Tompkins, 1985)

$$H(z) = (1 - z^{-6})^2 / (1 - z^{-1})^2 \quad (2.25)$$

$$Y_n = 2Y_{n-1} - Y_{n-2} + X_n - 2X_{n-6} + X_{n-12} \quad (2.26)$$

- FTJ recursiv generalizat (Ahlstrom și Tompkins, 1985)

$$H(z) = (1 - z^{-m})^k / (1 - z^{-n})^p \quad (2.27)$$

Acest tip de filtru prezintă o proprietate deosebit de interesantă: deși este recursiv, el are o caracteristică de fază liniară. Prezența unui "zero" chiar în punctul $z = 1$, face ca frecvența zero (componenta continuă) să treacă. Prezența polilor pe cercul unitar, egal distanțați, face ca respectivele frecvențe să nu treacă. Deci se pot implementa odată cu filtrul trece jos și rejectoare ale unor frecvențe nedorite, prin alegerea adecvată a tipului de funcție de transfer și a frecvenței de eșantionare. Ca dezavantaj, filtrul nu prezintă o bună tăiere în apropierea frecvenței rejectate și nici un palier constant în banda de trecere.

O altă idee de obținere a unui FTS este scăderea din semnalul original a semnalului trecut printr-un FTJ, ca în exemplul următor (Pan și Tompkins, 1985):

$$H(z) = (1 - z^{-256}) / (1 - z^{-1}) \quad (2.28)$$

$$Y_n = 2Y_{n-1} - Y_{n-2} + X_n - 2X_{n-6} + X_{n-12} \quad (\text{FTJ}) \quad (2.29)$$

$$Y'(n) = X_n - Y(n) / 256 \quad (\text{FTS}) \quad (2.30)$$

2.2.4.3 Filtre Trece Bandă

Modul obișnuit de implementare îl constituie legarea în serie a două filtre: unul FTS iar celălalt FTJ.

- FTS (derivata de ord 1 generalizată) plus FTJ tip mediere (Usui și Amidror, 1982)

$$Y_n = \frac{\sum_{n=L}^1 X_{k-n-N} - \sum_{n=L}^1 X_{k+n-N}}{2NT(2L+1)} \quad (2.31)$$

Pentru ca relația 2.31 să devină cauzală se face schimbarea de variabilă:
 $k+N+L \rightarrow k$

$$Y_n = \frac{\sum_{n=L}^1 X_{k+n-L} - \sum_{n=L}^1 X_{k+n-L-2N}}{2NT(2L+1)} \quad (2.32)$$

Modul de operare al relației 2.32 este ilustrat în fig 2.7. Acest tip de combinație, FTS+FTJ, reprezintă un filtru trece bandă extrem de util în preprocesare.

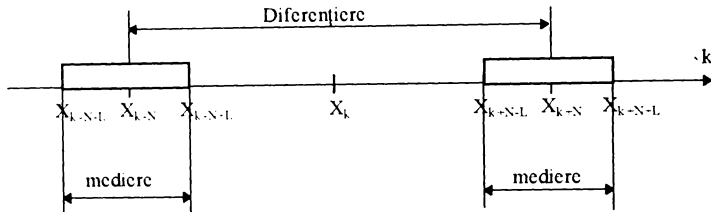


fig 2.7. Algoritm de mediere și diferențiere (Usui și Amidror,1982)

- FTB recursiv (Ahlstrom și Tompkins 1985)

$$H(z) = \frac{(1-z^{-12})^2}{(1-2(\cos q)z^{-1}-z^{-2})^2} \quad (2.33)$$

Pentru $q = \pi/3, \pi/2, 2\pi/3$ se obțin FTB. Pentru $q = 0$ sau π , se obțin FTJ respectiv FTS. Pentru $q = \pi/3$, structura filtrului devine

$$Y_n = 2Y_{n-1} - 3Y_{n-2} + 2Y_{n-3} - Y_{n-4} + X_n - 2X_{n-12} + X_{n-24} \quad (2.34)$$

Trebuie menționat că toate transformările liniare prezentate până acum au coeficienți întregi și se pretează foarte bine la procesarea în timp real. Se pare că soluțiile cu filtre rezultate prin metode ca: Transformata Fourier, Transformata Fourier Discretă, Transformata Z, Transformata Biliniară sau altele nu prezintă interes în domeniul preprocesării ECG în vederea detecției QRS. Aceasta datorită, pe de o parte, valorilor reale ale coeficienților utilizați iar pe de alta, lungimii algoritmilor de calcul comparativ cu filtrele prezentate, pentru obținerea de performanțe identice.

2.2.4.4 Transformarea LUNGIME

Fie o curbă C definită printr-o funcție vectorială X(u), reală, în spațiul n-dimensional, cu derivate continue. Prin definiție lungimea curbei în intervalul [t, t+q] este:

$$L_{(n,q,t)} = \int_t^{t+q} \sqrt{\sum_{i=1}^n (X_i)^2} \quad (2.35)$$

unde: n dimensiunea spațiului vectorial (numărul canalelor de înregistrare)
 q lungimea ferestrei în care se calculează lungimea

În cazul discretizării funcției X , cu o perioadă normalizată de eșantionare ($T = 1$), avem:

$$L_{(n,q,j)} = \sum_{k=j}^{j+q-1} \sqrt{\sum_{i=1}^n (X_{i,k} - X_{i,k-1})^2} \quad (2.36)$$

sau sub formă recursivă:

$$L_{(n,q,j)} = L_{(n,q,j-1)} + \sqrt{\sum_{i=1}^n (X_{i,j+q+1} - X_{i,j+q})^2} - \sqrt{\sum_{i=1}^n (X_{i,j-1} - X_{i,j-2})^2} \quad (2.37)$$

Pentru o fereastră de analiză q fixată la lățimea estimată a complexului QRS (cca 135 ms) și pentru o curbă ale cărei coordonate sunt valorile rezultate în urma înregistrării mai multor derivații standard ale ECG, transformarea 2.37 va fi maximă în momentul în care apare complexul QRS și va fi scăzută peste tot în altă parte. Principalul inconvenient este că transformarea pune în evidență și zgomotul de înaltă frecvență suprapus peste semnal, putând duce la identificări false, așa cum este ilustrat în fig. 2.8.

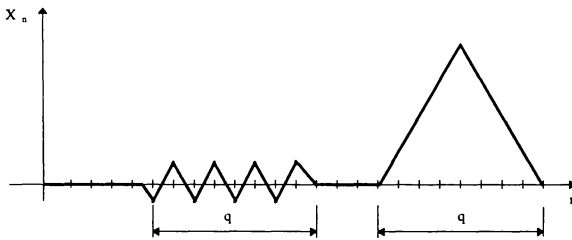


fig 2.8 Identificare falsă cu transformarea lungime

Din acest motiv, este indicată folosirea acestei transformări în cooperare cu altele ca de exemplu o transformare neliniară cu caracter integral care să ofere informații și asupra ariei cuprinse între axa de bază și curbă sau cu o metodă liniară.

2.2.4.5 Transformarea ENERGIE

Prin analogie cu energia cinetică a unui corp de masă m , transformarea "energie" definită pe un interval $[t ; t+q]$ este:

$$E_{(n,q,t)} = \int_t^{t+q} \frac{m}{2} \sum_{i=1}^n (X_i)^2 \quad (2.38)$$

În procesarea discretă, alegând perioada de eșantionare unitară, și o masă $m=2$ vom avea:

$$E_{(n,q,j)} = \sum_{k=j}^{j+q-1} \left(\sum_{i=1}^n (X_{i,k} - X_{i,k-1})^2 \right) \quad (2.39)$$

care conduce la relația de recurență:

$$E_{(n,q,j)} = E_{(n,q,j-1)} + \sum_{i=1}^n (X_{i,j+q+1} - X_{i,j+q})^2 - \sum_{i=1}^n (X_{i,j-1} - X_{i,j-2})^2 \quad (2.40)$$

Cele două transformări, "lungime" și "energie", generalizează mai multe transformări propuse în literatură ca de exemplu:

- pentru $n = q = 1$, transformarea 2.37 devine echivalentă derivatei de ordinul 1 a semnalului redresat (Holsinger ș.a. 1971);
- pentru $n = 1$ și $q = 65$, transformarea 2.40 devine transformarea propusă de Ligtenberg și Kunt, 1983;
- pentru $n = 3$ și $q = 1$ transformarea 2.37 devine transformarea utilizată frecvent pentru analiza vectocardiogramelor.

2.2.4.6 Transformarea MEDIANĂ

Pentru a evita confuzia dintre valoarea medie a unei secvențe și transformarea "mediană", le vom prezenta pe ambele. Valoarea medie este o transformare liniară, valoarea eșantionului la ieșire calculându-se cu relația 2.41 (numită și medie aritmetică):

$$Y(n) = \frac{1}{q} \sum_{i=0}^{q-1} X_{n-i} \quad (2.41)$$

unde q reprezintă lățimea ferestrei în care se face medierea.

Transformarea "mediană" este o transformare neliniară al cărei algoritm este următorul: în fereastra de lungime $q=2k+1$ (impar), se ordonează toate eșantioanele în ordine crescătoare. Ieșirea $Y(n)$ va lua valoarea eșantionului aflat în centrul ferestrei, după ordonare. În cazul valorilor pare ale numărului de eșantioane, mediana se calculează ca medie aritmetică a celor două valori aflate în centrul ferestrei.

2.2.5 Evaluarea performanțelor în detecția QRS

De la debutul procesării analogice a semnalului ECG, problema evaluării performanțelor era rezolvată de cele mai multe ori după criterii specifice fiecărui autor. Aceasta și datorită diversității extrem de mari a înregistrărilor ECG, datorate patologieilor sau diferitelor tipuri de zgomote, interferențe și artefacte. Problemele erau două:

- raportarea diverselor metode de procesare numerică la aceleași înregistrări;
- găsirea unor soluții pentru cuantizarea rezultatelor.

Prima problemă și-a găsit astăzi o rezolvare cvasiunanimită acceptată prin utilizarea bazei de date MIT / BIH (Massachusetts Institut of Technology), deși mulți autori utilizează și prezintă încă rezultate pe înregistrări ECG aflate în posesia instituțiilor pe care le reprezintă. De menționat că baza de date MIT / BIH conține pe lângă înregistrările propriu-zise și alte informații codificate, cum ar fi numărul și poziționarea fiecărui complex al fiecărei înregistrări ECG. Datele sunt

digitizate pe 12 biți, cu semn, ceea ce în limbajul Pascal obligă la utilizarea variabilelor de tip INTEGER pentru introducerea datelor.

Referitor la structura fișierelor de date, există mai multe variante. În fișierele MIT / BIH, datele sunt înregistrate pe doi octeți dintre care primul (L) în ordinea citirii conține pe cei mai puțin semnificativi 8 biți ai eșantionului iar în următorul octet, pe primele 4 poziții se află cei mai semnificativi 4 biți ai eșantionului. Ceilalți 4 biți sunt folosiți pentru codificarea datelor adiacente traseului ECG.

Soluția celei de-a doua probleme o constituie utilizarea cu precădere a două mărimi exprimate în procente, conform relațiilor:

- detectii fals pozitive (FP)

$$FP = \frac{\text{numărul complexelor QRS inexistente în realitate dar "identificate"}}{\text{numarul total de complexe QRS}} \cdot 100 \quad (2.42)$$

- detectii fals negative (FN).

$$FN = \frac{\text{numarul complexelor QRS existente în realitate dar neidentificate}}{\text{numarul total de complexe QRS}} \cdot 100 \quad (2.43)$$

În relațiile 2.42 și 2.43, "numărul total de complexe QRS" este furnizat de către medicul specialist, prin analiza traseului ECG testat și de detectorul QRS.

2.3 COMPRESIA DATELOR ECG

Primul pas pentru minimizarea volumului de date ECG îl constituie minimizarea frecvenței de eșantionare și a nivelului de cuantizare. În aplicații în care este generat un volum mare de date și sunt necesare stocări sau transmisii ale acestora, se impune utilizarea tehnicilor de compresie a datelor. Cea mai generală schemă bloc de compresie urmată de o decompresie este cea din fig.2.9.

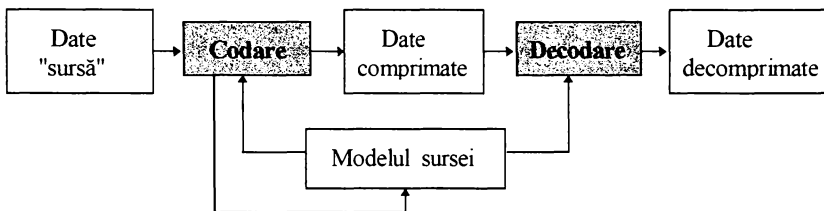


fig.2.9. Schema bloc a compresiei / decompresiei de date

Clasificarea metodelor de compresie. Un prim criteriu îl constituie eroarea de refacere. În raport cu acest criteriu există două categorii de metode:

- **metode ireversibile** ("lossy") unde pierderea de informație este permisă în limitele unui nivel de erori prestabilit;

- **metode reversibile** ("lossless") la care nu există pierdere de informație și semnalul numeric original se va putea reface cu exactitate. Acestea elimină redundanțele statistice pe baza corelării și predictibilității datelor și se adresează în general bazelor de date numerice (fișiere surse de programe, fișiere .EXE, fișiere text). Pot fi aplicate și ambele categorii în cascadă.

Orice proces de compresie pleacă de la o cunoaștere a structurii statistice a setului de date ce vor fi comprimate. Performanțele codării sunt deci determinate și de modelul datelor "sursă". În raport cu evoluția în timp a modelului sursei, metodele de compresie se pot împărți în :

- **metode statice**, în care modelul sursei nu se modifică în timp, fiind stabilit apriori pe baza unui set de date "tipice" din punct de vedere statistic;
- **metode semi-statice**, în care modelul sursei este stabilit la începutul procesului de compresie pe baza unui set de date "tipice" din punct de vedere statistic și apoi nu se mai modifică pe parcursul compresiei;
- **metode adaptive** în care modelul evoluează în timp o dată cu evoluția datelor de codat. După codare și refacere, starea modelului se actualizează în același fel, atât la codare, cât și la decodare.

Un alt criteriu de clasificare îl constituie natura informațiilor pe baza cărora se construiește modelul. După acest criteriu avem:

- **metode lingvistice** în care se construiește un dicționar al limbajului iar ceea ce se transmite este o referință la poziția în dicționar a cuvântului curent codat;
- **metode statistice** în care se alocă cuvinte de cod cu un număr mai mic de biți cuvintelor cu o probabilitate de apariție mai mare.

Obiectivele urmărite în cuprinsul acestui subcapitol sunt:

- prezentarea unei sinteze a studiului bibliografic din domeniu;
 - încercarea de clasificare cuprinzătoare a tehnicilor de compresie a datelor ECG
- Pentru evaluarea performanțelor algoritmilor de compresie se definește rata medie de compresie (average Compression Ratio) definită $CR = A : 1$ unde A reprezintă raportul dintre totalitatea datelor înainte și după compresie.

2.3.1 Tehnici de codare ireversibilă

În procesarea ECG conservarea morfologiei este fundamentală pentru a fi acceptată clinic la reconstrucția semnalului. Conceptul de eliminare a redundanței la această categorie de tehnici se realizează în general prin:

- reținerea doar a acelor eșantioane care vor servi la aproximarea semnalului cu distorsiuni acceptabile;
- cuantizări predictive;
- folosirea transformărilor ortogonale.

Aceste principii pot fi grupate în două mari categorii, după cum ele se dezvoltă în domeniul timp sau în alte domenii obținute prin transformări ortogonale.

2.3.1.1. Tehnici de codare ireversibilă în domeniul timp

Această categorie de tehnici este larg folosită, deoarece algoritmiile pot fi aplicați direct semnalului achiziționat, obținându-se compresii eficiente și timpi de

procesare scăzuți. În acest cadru am identificat și clasificat din literatura de specialitate metodele următoare:

a. Extragerea caracteristicilor

Ideea ce stă la baza acestei metode este aceea de a privi traseul ECG ca o succesiune de linii, curbe, unghiuri care să fie memorate ca atare iar la reconstrucție să fie utilizat un proces de reaşezare a structurilor elementare în ordine. Udupa și Murthy (1980) au propus o procedură bazată pe descrierea traseului ECG prin segmente de dreaptă cărora să le fie precizată lungimea și unghiul pe care-l fac cu direcția liniei de bază. Aceasta a reprezentat primul pas spre analiza ritmului cardiac folosind descrierea sintactică. O altă metodă folosește marcarea traseului ECG cu "puncte semnificative" (Lee și, 1987). Prin studiul curburilor semnalului, pot fi identificate punctele semnificative în scopul reprezentării semnalului sau pentru a fi folosite în procese de pattern recognition. Giakoumakis și Papakonstantinou (1987) definesc pentru descrierea semnalului niște structuri numite "unde". O astfel de undă are vârfuri pozitive și negative conform regulilor sintactice stabilite. Reconstrucția se face aproximând undele cu segmente de dreaptă. Un concept similar folosește Lachiver și. (1987): aici sunt detectate punctele extreme, pantele segmentelor și un set de reguli pentru identificarea eşantioanelor ce trebuie reținute. Aceste reguli sunt stabilite printr-o procedură iterativă al cărei scop este să minimizeze eroarea medie pătratică dintre semnal și versiunea sa reconstruită prin aproximații cu funcții spline. Obține rate de compresie de 5:1 la o frecvență de eşantionare de 400Hz și 12 biți rezoluție.

b. Aproximări liniare sau poligonale

Ideea de bază este eliminarea acelor eşantioane care pot fi refăcute prin aproximații liniare sau poligonale. Cele mai importante tehnici de aproximare liniară sau poligonală folosite pentru compresia ECG pot fi catalogate după cum urmează:

- **SAPA** (Scan Along Polygonal Approximation) sau "Evantai"

Este frecvent utilizată de diverși autori (Ishjima și, 1983; Blanchard și Bar, 1985; Lee și 1987; Giallorenzo și, 1988; Bohs și Barr, 1988; Bohs și Bar, 1989; Jalaledine și Hutchens, 1990). Principiul metodei constă în faptul că traseul ECG este modelat pe porțiuni prin segmente de dreaptă sau segmente poligonale iar dacă de-a lungul unui astfel de segment "distanța" dintre valoarea unui eşantion și valoarea rezultată prin interpolare este mai mică decât un prag prestabilit, atunci acel eşantion se elimină. Lamberti și Coccia(1989) au propus utilizarea interpolării cu funcții spline. Metoda este eficientă din punctul de vedere al compresiei, dar faptul că lungimea segmentelor de aproximare a traseului este diferită face ca în transmisia semnalului ECG astfel compresat, rata de compresie să nu fie constantă pe durata transmisiei. Bohs și Barr (1988) au propus o soluție de segmentare a semnalului de intrare și un algoritm care optimizează eroarea de reconstrucție, menținând în același timp raportul de compresie constant. Tot ei implementează algoritmul pe un procesor DSP (TMS 32010), iar în 1989 prezintă un circuit VLSI pentru dezvoltarea algoritmului. În scopul obținerii unei rate constante s-au propus algoritmi care adaptează eroarea la variațiile statistice ale semnalului, algoritmi utilizat la

procesări cu frecvențe mari de eșantionare. Giallorezzo ș.a., (1988) obține cu un algoritm similar rate de 8:1 la o frecvență de 200 Hz și o rezoluție de 8 biți într-o aplicație în timp real.

- **AZTEC** (Amplitude Zone Time Epoch Coding). Acest algoritm este larg utilizat în realizarea echipamentelor comerciale, fiind primul implementat de mari companii (Hewlett Packard, Electronics for Medicine, etc..) în preprocesarea ECG în analizoarele de aritmie. Acest algoritm utilizează trei variabile pentru codarea semnalului referitoare la timp, pante și zone plate. Deci, traseul ECG este aproximat între eșantioane cu segmente de dreaptă dintre care cele horizontale (zone plate) se definesc ca fiind media valorilor minime și maxime găsite într-un domeniu. Un eșantion adiacent unei zone plate va fi atașat acesteia numai dacă diferența (în modul) dintre valoarea sa și valoarea zonei plate este mai mică decât o valoare prestabilită. Dacă nu se încadrează între pragurile astfel stabilite, acel eșantion va genera un segment înclinat. Zonelor plate li se memorează lungimea, iar celor înclinate li se va memora panta, semnul și lungimea. Algoritmul are avantajul că utilizează numai operații de adunare, scădere și logice ceea ce îl face ușor de implementat în timp real. Cox ș.a., (1968) obțin rezultate remarcabile în domeniul compresiei: $CR = 10:1$ la $F_e = 500$ Hz și 12 biți. Furth și Perez, (1988) propun praguri variabile pentru zonele plate, adaptate la evoluția statistică a semnalului. Jalaliddine și Hutchens, (1990) combină metoda AZTEC pentru complexele QRS cu metoda SAPA pentru regiuni izoelectrice și pentru unde P și T. Această schemă permite îmbunătățirea calității undelor P și T în semnalul reconstruit. Cu această combinație ei au obținut $CR = 5,9:1$, pentru $F_e = 166$ Hz și 10 biți.
- **Puncte de inflexiune. (Turning point -TP)**. Se bazează pe deplasarea unei "ferestre" cu trei puncte, de-a lungul semnalului pentru detectarea maximelor și minimelor locale, în scopul reținerii a două puncte din trei care conservă panta semnalului original, având fixată o rată de compresie dată. Altfel spus metoda constă în reducerea numărului de eșantioane fără a afecta însă pe cele care reprezintă minime sau maxime locale (vârfuri) pe parcursul traseului ECG. Metoda comportă numai operații de scădere și comparație, ceea ce o face ușor de implementat (Tompkins și Webster, 1981; Abenstein și Tompkins, 1982). De Lucia, (1990) propune un algoritm care având la bază acest principiu realizează o reducere a frecvenței de eșantionare de două ori. Algoritmul se poate aplica și în cascadă, rata de compresie dublându-se la fiecare nouă iterație, însă cu deteriorarea corespunzătoare a semnalului. Fiind în fond o eșantionare adaptivă, algoritmul prezintă față de metoda SAPA avantajul că nu trebuie reținute eșantioane și timpi de referință, deoarece frecvența de eșantionare este constantă. Moody ș.a. (1989), propun un algoritm de cercetare a punctelor de extrem urmat de o aproximare liniară și apoi de o codare cu cuvinte de lungime variabilă, obținând $CR = 4,8:1$ la o frecvență de 120 Hz și 8 biți.
- **CORTES** (Coordinate Reduction Time Encoding System). Acest metodă este o combinație a algoritmilor AZTEC (pentru regiuni plate = linia izoelectrică) și TP pentru undele P,Q,R,S,T. Se bazează pe operații matematice foarte simple

dar implementarea sa este mai complexă decât AZTEC. Abstein și Tompkins (1982) obțin astfel $CR = 4:1$ pentru $F_e = 200$ Hz și 12 biți.

- **Codarea DELTA.** Ideea metodei este aceea de a aproxima conturul semnalului cu segmente de dreaptă de pantă pozitivă sau negativă dar constante în modul. Codul semnalului va rezulta prin atribuirea de 1 intervalelor cu pantă pozitivă și de 0 celor cu pantă negativă. În felul acesta în locul valorii curente a unui eșantion se transmite doar un singur bit. Reconstrucția semnalului se face ușor cunoscând valoarea modulului pantei utilizate la codare. Se pot obține deci rate de compresie care se apropie de $N_b : 1$ unde N_b este numărul de biți cu care s-a făcut achiziția semnalului. O variantă mai performantă utilizează pante constante la început iar, dacă aproximarea devine nesatisfăcătoare (segmente prea lungi), se modifică valoarea pantei cu valori constante, după o logică ce are în vedere lungimea ultimului segment. Dacă aceeași logică este utilizată și la recepție atunci nu mai este nevoie de a transmite informații referitoare la schimbarea pantei. Reprezintă una dintre cele mai utilizate și performante tehnici de compresie a semnalului audio, existând circuite integrate specializate în acest sens (Motorola-MC 3517, MC 3518) care codifică / decodifică direct semnalul analogic.
- **Codarea DELTA-SIGMA.** Principial metoda este similară cu cea prezentată anterior doar modalitatea efectivă de obținere a codului diferă: este integrată diferența dintre semnal și semnalul codat, pe când la metoda delta este integrat doar semnalul de ieșire (circuitul integrat AD 1879).

c. Compresia "ciclu cu ciclu"

Principiul metodei este realizarea inițială a unui "șablon" prin analiza unui anumit număr de cicluri cardiace și apoi utilizarea diferențelor ce apar în evoluție față de acest șablon pentru compresie. Acest lucru este posibil prin considerarea traseului ECG ca unul cvasiperiodic în care schimbările de morfologie apar doar o dată cu tulburările funcțiilor cardiace. Smith și Platt (1989) utilizează o clasificare a fiecărui ciclu QRS într-o familie de șabloane anterior construită.

Jalaleddine ș.a. (1990) descriu o metodă aplicată înregistrărilor Holter în următoarele etape: eliminarea ciclurilor anormale; aplicarea unui algoritm de compresie "evantai" sau AZTEC; crearea unui șablon corespunzător unui ciclu cardiac; calcularea diferenței fiecărui ciclu față de șablon; înlocuirea fiecărui ciclu QRS cu aceste diferențe; aplicarea din nou a unui algoritm de compresie evantai. De notat că această metodă implică practic utilizarea unui algoritm de detecție QRS și calculul corelației, ceea ce o face "consumatoare" de timp. Hamilton și Tompkins (1991) realizează șablonul prin medierea ciclurilor și obțin $CR = 4,6:1$, pentru $F_e = 100$ Hz, și 8 biți

d. Codarea diferențială cu prag

Această metodă este cunoscută în literatură sub numele de Differential Pulse Cod Modulation with threshold (DPCM-WT). Algoritmul memorează diferențele de amplitudine între eșantioanele succesive, atunci când aceste diferențe depășesc un prag prestabilit, precum și momentele la care se întâmplă acest lucru. Bertrand ș.a.

(1977) obțin $CR = 4:1$ și $CR = 10:1$ la 300 Hz și 8 biți impunând un prag de 2% respectiv 5% din maximul domeniului dinamic de conversie a CAN. Metoda este principial deosebită față de metoda DELTA prin aceea că acum aproximarea se face pentru fiecare interval de eșantionare, neavând un caracter integral. La reconstrucție, semnalul se aseamănă cu cel rezultat după o compresie AZTEC. Pornind de la ideea de bază a utilizării diferenței dintre eșantioane succesive Kao ș.a. (1989), simplifică codarea astfel: dacă diferența este pozitivă, ieșirea codorului este 1 iar dacă este negativă ieșirea este 0. Metoda este simplă de implementat, permite obținerea unor compresii remarcabile: $CR = 12:1$, la $F_c = 500$ Hz și 12 biți, dar urmărește cu erori mari deviațiile lente ale liniei de bază.

2.3.2. Transformări ortogonale

La acestea (Cetin ș.a.1993), eliminarea redundanței și ajustarea parametrilor sunt obținute în domenii diferite (ex: frecvență), față de cel în care s-a făcut achiziția (timp). Au fost cercetate și utilizate în domeniul semnalelor ECG diferite transformări matematice ca:

- Transformata Karhunen-Loeve, "KLT" (Ahmed ș.a., 1975)
- Transformata Cosin, "CT" (Ahmed ș.a., 1975)
- Transformata Hilbert (Chang ș.a, 1992)
- Transformata Haar "HT" (Ahmed ș.a., 1975)
- Transformata Walsh, "WT" (Adrian ș.a., 1987)
- Transformata Fourier, "FT" (Reddy și Murty,1986; Chang ș.a.1992)

Pentru fiecare din aceste transformări compresia se efectuează în două etape prin aplicarea transformării propriu-zise semnalului original, după care urmează compresia efectivă în sensul reducerii parametrilor necesari descrierii semnalului în noul domeniu. În acest caz redundanța este căutată (și apoi eliminată) în distribuția spectrală sau în distribuția energiei semnalului transformat. Reconstrucția se obține prin aplicarea transformării inverse, rezultatul fiind obținerea semnalului original afectat de erori predefinite.

Ca și criteriu general, este considerată ca optimă transformarea care duce la minimum de erori la reconstrucție, simultan cu o rată maximă de compresie în raport cu parametrii noului domeniu. În acest sens, cea mai performantă pare a fi transformarea Karhunen-Loeve, dar ea implică un volum mare de calcule. Transformările pot fi grupate în două categorii: armonice (FT, CT) și nearmonice (HT, WT, KLT). În implementarea tehnicilor bazate pe transformări ortogonale, printre problemele ce trebuie avute în vedere amintesc:

- Criteriul de a decide care parametri ai semnalului transformat se pot reduce. Acesta trebuie să plece de la o analiză statistică a semnalului. În acest sens Womble ș.a. (1977) au obținut acești parametri prin analiza a 900 de cicluri cardiace. Performanțele algoritmilor sporesc în timp astfel ajungându-se la numai 200 de cicluri pentru a deduce acești parametri, dar în cazul unor cicluri foarte diferite de cele care au stat la baza regulilor de reducere, compresia nu este eficientă și de calitate.
- Existența unui mare volum de calcule matematice pentru fiecare transformare. Acest lucru face dificilă implementarea lor în timp real, singura soluție în acest sens constituind-o folosirea procesoarelor de semnal specializate (DSP).

- Aplicarea unei transformări de lungime fixă (asupra unui număr constant de eşantioane) înseamnă practic segmentarea traseului ECG în "porțiuni" egale. Cum însă complexul QRS, unda P, unda T nu au durate egale, apar două mari probleme: detecția QRS și obținerea unui timp de referință normalizat pentru fiecare ciclu cardiac.

Compresia propriu-zisă în algoritmi ce utilizează transformări ortogonale aplicate semnalelor ECG, are loc în două etape diferite:

- Prima o constituie reducerea numărului de date în domeniul transformat. Un segment de semnal ECG cu N eşantioane este transformat într-un domeniu unde sunt reținute numai M componente ale funcției transformate (cu $M < N$). Acest pas este însoțit deci de încă un aspect al compresiei : fixarea numărului de biți pentru cuantificarea celor M coeficienți.
- A doua se referă la codarea efectivă a celor M coeficienți. O soluție o constituie lungimea de codare variabilă pentru coeficienții obținuți prin transformare, folosind o lege de compresie exponențială. În majoritatea transformărilor ECG, 90% din energie este concentrată în componentele joase ale spectrului (Adrian, 1987; Ahmed ș.a, 1975).

Tehnicile bazate pe transformate ortogonale asigură rapoarte de compresie mari. În literatură de specialitate pot fi găsite $CR=12:1$, ($F_e=250$ Hz și 12 biți) folosind KLT; $CR = 5,33:1$, ($F_e=250$ Hz și 8 biți) folosind WT; $CR = 3:1$, ($F_e=400$ Hz și 8 biți) folosind FT.

2.3.2. Tehnici de compresie reversibilă

Tehnicile de compresie reversibilă, numite și "fără pierderi", permit o reconstrucție exactă a datelor comprimate. Obiectivul tehnicilor este de a elimina redundanța statistică și sunt bazate pe corelația și predictibilitatea datelor. Aceste tehnici furnizează un raport scăzut de compresie și din acest motiv, ele sunt găsite în aplicații ECG împreună cu alte metode ireversibile deja amintite.

2.3.2.1. Metode predictive

Aceste metode se bazează pe dependența semnalelor adiacente. Acest fapt permite estimarea valorii unui eşantion la un moment dat pe baza eşantioanelor sale din trecutul apropiat. Prin codare se transmite sau se memorează doar diferența dintre eşantionul curent și valoarea sa estimată. În literatură aceste metode sunt grupate sub numele "Differential Pulse Coding Modulation" (DPCM).

a. Predicția liniară și interpolarea

Predicția reprezintă estimarea valorii unui eşantion la un moment dat pe baza ultimelor N eşantioane, pe când interpolarea estimează valoarea eşantionului pe baza a N eşantioane anterioare și a M eşantioane ulterioare lui, prin decalarea corespunzătoare în timp. Ruttiman și Pipberger (1979) au obținut rezultate bune folosind predicția liniară: $CR = 3,8:1$, pentru $F_e=500$ Hz și o rezoluție de 8 biți. Metoda poate servi în același timp și la extragerea unor parametri ai undelor importante (Lin și Chang, 1989).

b. Codarea diferențială

Semnalul codat reprezintă diferența dintre eșantionul prezent și cel anterior, diferență care este în medie mai mică decât amplitudinea semnalului original și poate fi deci cuantificată cu un număr mai mic de biți. În termeni de predicție lineară acest algoritm poate fi privit ca cel mai simplu predictor, la care eșantionul prezent estimat este chiar eșantionul real anterior. Bertrand ș.a.(1977) au obținut folosind această metodă simplă $CR = 1,4:1$ la $F_e = 500$ Hz și 10 biți rezoluție. Dacă se ține seama că rezultatul codării va fi în mare parte pe 2 sau 3 biți, pornind de la o reprezentare inițială pe 10 biți, atunci se poate obține o rată maximă de compresie $CR = 3,33:1$.

2.3.2.2 Codarea entropică

Codarea entropică se bazează pe comportarea statistică a datelor de intrare. Datele cu o probabilitate mai mare de apariție se codează cu un număr mai mare de biți decât cele ce au o probabilitate mai mică. Procedeele se mai numește codare cu cuvânt de lungime variabilă.

Această metodă generează un set de cuvinte-cod care vor da o versiune comprimată a setului de date de intrare și vor fi un model al comportării lor statistice. Dificultatea majoră în acest caz o constituie lungimea fluctuantă a cuvântului de cod ceea ce creează probleme la emisie. Această fluctuație necesită un buffer între sursă și canalul de transmisie. În plus o eroare chiar de numai un bit poate provoca imposibilitatea de a decoda mesajul dacă nu se folosește un protocol și un cod detector de erori adecvat.

a. Codarea Huffman

Această metodă de codare necesită cunoașterea apriori a distribuției probabilităților datelor de intrare. Pentru aceasta sunt necesare două treceri prin datele de intrare: o trecere pentru calcul probabilităților și o a doua pentru obținerea alfabetului de codare. Există însă și posibilitatea de obținere a probabilităților în timpul apariției datelor din proces. Această a doua variantă se numește "Huffman adaptivă" (nu am întâlnit referiri la folosirea ei practică în aplicații ECG).

Folosirea codului Huffman-static în compresia de date ECG a fost utilizată în general în combinație cu alte metode de compresie, de obicei ireversibile. Ruttiman ș.a. (1979) utilizează codarea de acest fel în combinație cu un predictor, iar Jalaladdine ș.a.(1990) comunică $CR = 5,75 : 1$, pentru $F_e = 500$ Hz și 12 biți combinând codarea cu o dublă diferențiere a ECG original. O altă categorie de căutări o constituie cele referitoare la determinarea probabilităților bazate în general pe descompunerea setului de date de intrare în subseturi. De exemplu Jalaladdine ș.a., 1990, au divizat datele în subseturi "frecvente" și "rare". Aplicând codarea Huffman la datele "frecvente" și codarea cu cuvinte-cod de lungime fixă pentru cele "rare" s-a obținut un $CR = 2,8 : 1$, $F_e = 250$ pe 10 biți.

b. Codarea aritmetică

Este considerată o tehnică de codare eficientă care se poate apropia mai mult de entropia sursei decât codarea Huffman. Se fac dese comparații între cele două metode în aplicații ECG.

Witten șa., (1987) arată că principalul atu al acestei metode este reducerea timpului de codare / decodare. În plus, codarea este realizată printr-o singură trecere, ceea ce-i sporește atractivitatea. Utilizând codarea aritmetică în combinație cu un algoritm SAPA, rezultatele datorate numai codării aritmetice sunt CR = 1,52:1, pentru Fe = 360 Hz și 8 biți rezoluție.

2.3.3 Evaluarea algoritmilor de compresie a datelor ECG

Nu există criterii absolute pentru evaluarea eficienței unui algoritm de compresie de date. Fiecare autor folosește procedura care se potrivește mai bine aplicației sale: stocare, transmisie sau procesare. Fiindcă performanțele algoritmilor depind în primul rând de specificul semnalului, sunt frecvent folosite pentru evaluarea performanțelor algoritmilor de compresie semnale din bazele de date AHA (American Heart Association Arrhythmia Database) și MIT-BIH.

Primul criteriu de performanță este rata de compresie (CR). Aceasta se obține comparând numărul datelor din semnalul original cu parametrii memorajii sau transmiși ai codării. Se întâlnesc diferite modalități de exprimare a lui:

- CR = A : 1; A reprezintă raportul dintre totalitatea datelor înainte și după compresie;
- CR = B % ; unde B este raportul A anterior, prezentat procentual;
- C [biți/canal/sec] . C reprezintă viteza de transmisie ce poate fi atinsă după compresie. Cunoscând frecvența de eșantionare (Fe) și numărul de biți ai achiziției (N) , raportul de compresie este CR = (Fe N) / C : 1 ;

Doar simpla exprimare a raportului de compresie nu oferă o imagine completă asupra algoritmului de compresie dacă acesta nu se completează cu informații referitoare la: banda de trecere, frecvența de eșantionare, lungimea cuvântului după compresie, nivelul erorilor posibile la reconstrucție, mărimea bazei de date sau nivelul de zgomot.

Calitatea semnalului refăcut după compresie reprezintă în fond criteriul esențial în a aprecia algoritmul ca "acceptabil clinic". În cazul metodelor ireversibile este standardizată exprimarea acestei calități prin folosirea distanței medii pătratice dintre semnalul original și cel refăcut, raportată procentual la semnalul original (Percent Mean Square Difference -PRD).

$$PRD = \sqrt{\frac{\sum_{i=1}^N (ECG_{orig(i)} - ECG_{ref(i)})^2}{\sum_{i=1}^N (ECG_{orig(i)})^2}} \times 100 \quad (2.44)$$

Există și alte criterii pentru aprecierea cantitativă a asemănării dintre semnalul original și cel refăcut, ca de exemplu: calculul coeficienților de corelație, raportul dintre puterea semnalului de eroare și puterea semnalului original, estimarea distorsiunilor prin analiza spectrală (Hamilton și Tompkins,1991). Aceste criterii vizează mai mult aspecte legate de procesarea semnalelor și în general nu este o bună corelare între acestea și acceptabilitatea clinică.

O altă abordare a estimării modificărilor aduse traseului ECG prin compresie este una globală care încearcă să stabilească măsura în care compresia influențează

diagnosticul. Moody ș.a (1989) propun ca și criteriu de acceptabilitate clinică compararea diagnosticelor efectuate pe ECG originale și apoi pe cele compresate.

În evaluarea performanțelor algoritmilor de compresie un loc aparte îl au criteriile referitoare la complexitatea implementării și a eficienței în timp. Deși utilizarea procesoarelor de semnal fac posibile prelucrări aritmetice complexe și rapide, în general autorii fac referiri și la aceste criterii.

În concluzie, pentru evaluarea și compararea algoritmilor de compresie trebuie cunoscute întotdeauna elementele esențiale ale bazei de date originale: frecvență de eșantionare, nivel de cuantizare, caracteristicile zgomotului și eventual patologia. Performanțele algoritmilor de compresie ECG pot fi apreciate prin CR, PRD, eficiență în timp, complexitate și acceptabilitate clinică.

2.4 Criterii pentru diagnosticarea automată a ischemiei.

Toți autorii recomandă utilizarea derivației V5 pentru analiza segmentului ST. Prima etapă o constituie identificarea punctului fiducial, aflat la debutul undei Q (înainte de începerea depolarizării ventriculare). Acest punct este considerat ca aflându-se pe linia izoelectrică a traseului ECG normal, fără artefacte sau interferențe. Determinarea exactă a poziției sale se face după ce a fost identificată poziția undei R, față de care punctul fiducial se află înainte cu 50 ± 60 ms. Acest punct se determină pentru fiecare complex în parte (fig.2.10).

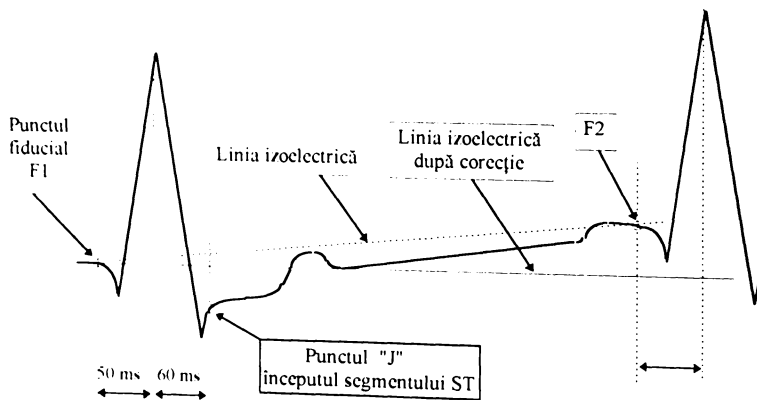


fig. 2.10. Axare după linia izoelectrică.

Linia izoelectrică poate fi construită din segmentele de dreaptă ce unesc câte două puncte consecutive, sau cu mai multă exactitate poate fi construită cu funcții spline sau polinomiale. Acestea "obligă" curba să treacă prin punctele determinate și să se racordeze în dreptul punctelor cu aceleași valori ale pantei. Am constatat că practic prima soluție, mult mai simplă, oferă siguranță și este satisfăcătoare permițând lucrul în timp real, după fiecare QRS având deja la dispoziție linia izoelectrică

Urmează apoi corectarea traseul ECG, axându-l față de linia izoelectrică determinată anterior. Din valoarea fiecărui eșantion se va scădea valoarea corepunzătoare a liniei izoelectrice având coordonata acelui eșantion.

Referitor la definirea segmentului ST și la modalitatea de cuantificare a morfologiei sale am întâlnit în literatură mai multe variante. Toate au un element comun și anume definirea punctului J, punct care marchează sfârșitul complexului QRS, respectiv începutul segmentului ST. Stabilirea poziției sale se face tot față de poziția undei R și anume la $55 \div 60$ ms.

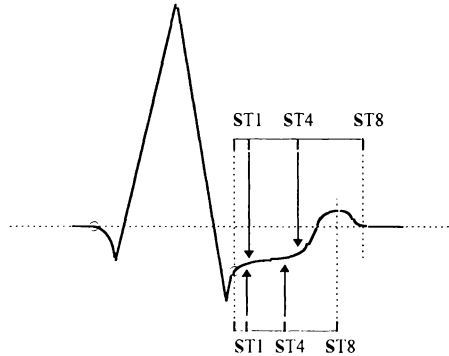


fig. 2.11. Măsurarea segmentului ST (Blomqvist, Simoons)

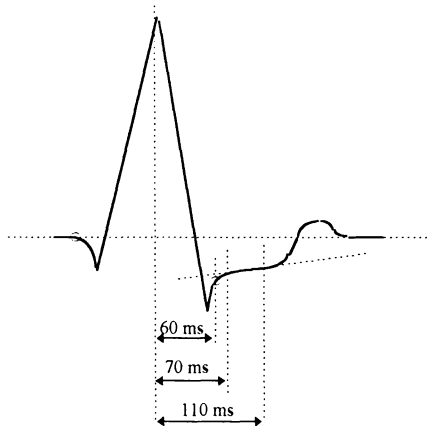


fig. 2.12. Măsurarea segmentului ST (McHenry)

O primă modalitate de a cuantiza starea segmentului ST este sugerată de Blomqvist, 1965. Se împarte intervalul dintre punctul J și sfârșitul undei T în 8

părți (fig.2.11). Valorile eșantioanelor având coordonatele ST1 și ST4 (deci la jumătatea intervalului definit anterior), definesc valorile de început și de sfârșit ale segmentului ST. Simoons (1975) utilizează același algoritm dar definește sfârșitul intervalului inițial ca fiind maximum undei T. Ambele soluții sunt dificil de generalizat datorită variațiilor formei undei T.

McHenry,1968 introduce o mărime numită "index ST" pentru cuantificarea globală a subdenivelării segmentului ST și a caracterului ascendent-descendent al acestuia. Astfel, acest indice K_{ST} se calculează empiric ca sumă a denivelării ST exprimată în mm și măsurată la 10 ms după punctul J și panta ST exprimată în mV/sec. Panta se calculează între eșantioanele de coordonate 110 ms și 70 ms.

Dacă acest indice este negativ, este semnalul unei ischemii a cărei gravitate este de asemenea exprimată prin valoarea indicelui K_{ST} .

De remarcat că rezultatele depind esențial de gradul de filtrare a semnalului, mai ales de zgomotele de înaltă frecvență. Ca soluție este recomandată medierea mai multor porțiuni de traseu (25) cuprinzând segmentul ST. Acest lucru este însă dificil în cazul monitorizării probei de efort unde trebuie urmărită evoluția ciclului cu ciclul a segmentului ST iar o mediere ar reduce șansa surprinderii unor fenomene ce pot deveni dramatice pentru pacient.

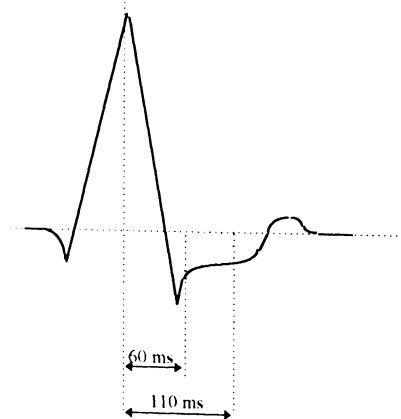


fig. 2.13. Măsurarea segmentului ST (Sheffield)

O altă variantă, propusă de Sheffield,(1969), ilustrată în fig.2.13, utilizează tot o măsură globală a morfologiei segmentului ST și anume, suprafața închisă între linia izoelectrică și traseul ECG, între punctele de coordonate 60 ms și 110 ms, măsurate față de unda R. Suprafața se exprimă în " $\mu V \cdot Sec$ ". Se admit ca normale valori cuprinse între 0 și $-7 \mu V \cdot Sec$, iar în diverse patologii, au fost găsite valori între -7 și $-20 \mu V \cdot Sec$.

O variantă a soluției anterioare o constituie cea din fig. 2.14 , inclusă deja în unele sisteme comerciale de monitorizare. Acum integrarea se efectuează până la momentul în care traseul intersectează linia izoelectrică. Ultimele două variante au

avantajul relativei imunități la zgomot, datorat principului de măsură integral presupus de calculul ariei. Este recomandată o frecvență de eșantionare de 500 Hz pentru a spori acuratețea calculului.

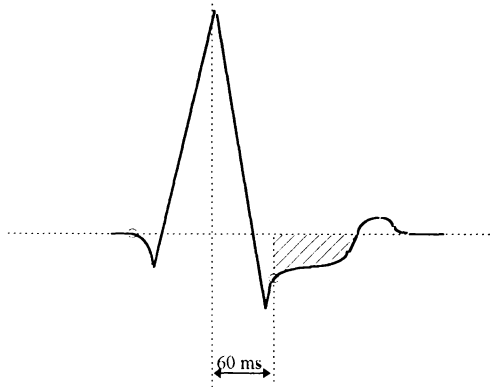


fig. 2.14. Măsurarea segmentului ST (var. Sheffield)

În varianta din fig.2.15 (Forlini,1975), este măsurată aria triunghiului format între axa izoelectrică și dreapta ce unește punctele extreme ale segmentului ST. Pentru a exista acest triunghi trebuie ca segmentul ST să aibă un aspect ascendent.

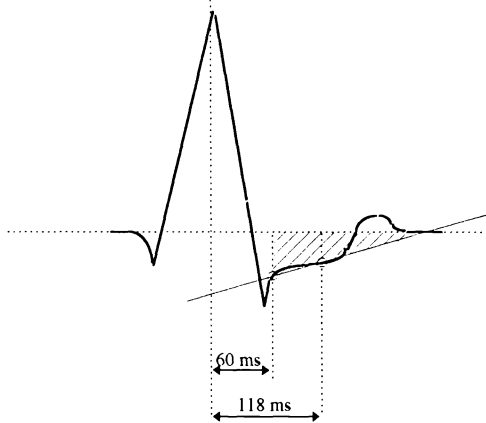


fig. 2.15. Măsurarea segmentului ST (Forlini)

De aceea este "forțată" puțin lungimea segmentului ST, al doilea punct fiind ales la 118 ms față de unda R. Această soluție înglobează și informațiile despre panta segmentului ST, informații pe care McHenry le introducea empiric.

Toții autorii menționați fac referire la numărul de cazuri studiate și prezintă date referitoare la eficiența variantei propuse. În general se caută corelații ale datelor despre segmentul ST cu numărul de vase afectate și cu amploarea zonelor ischemice din cord, rezultate obținute prin angiografii coronariene făcute aceluiași subiecți. Rezultatele nu sunt niciodată de încredere absolută (100%), mai ales în cazul afectării unor porțiuni mici din cord, la aceasta concurând de obicei și medicația administrată bolnavilor coronariani.

Capitolul 3

REALIZĂRI ȘI CONTRIBUȚII PERSONALE

3.1 ARHITECTURA PROPUȘĂ PENTRU CERCETARE

În ideea dezvoltării cercetărilor, am gândit ca activitatea din laboratorul pentru proba de efort, să fie integrată unei structuri complexe de monitorizare pentru secția de cardiologie, conform schemei bloc prezentată în fig 3.1.

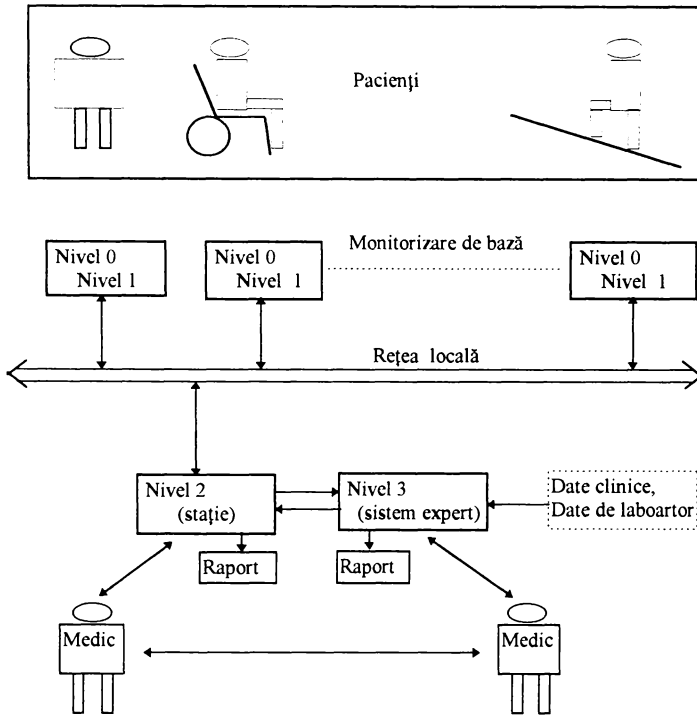


fig. 3.1 Arhitectura de monitorizare cardiacă complexă

Soluția propusă are în vedere în primul rând monitorizarea permanentă a saloanelor de terapie intensivă pentru coronarieni urmând ca prin stația centrală de lucru să fie asigurată supravegherea medicală de specialitate în permanență (serviciul de gardă).

Structura de monitorizare și procesare a semnalelor ECG este o structură suplă ce permite o dezvoltare progresivă, pe etape. Un astfel de sistem cu sarcini și subsarcini modulare permite facilități de cercetare în fiecare segment al său și se pretează la rețehnologizare pe etape, de aceea un lucru esențial trebuie avut în atenție și anume, aspectul compatibilităților permanente ale modulelor acestei arhitecturi structurate, în toate etapele de dezvoltare.

În schema bloc a unei astfel de arhitecturi, prezentată în fig 3.1, semnificația părților componente este următoarea:

Nivelul "0"

Denumirea de nivel "0" de procesare este încetățenită în literatura de specialitate și se referă la procesarea efectuată în imediata apropiere a pacientului, "la căpătâi". Nivelul "0" răspunde necesităților clasice, la un nivel minim pentru monitorizarea cardiacă, chiar și într-o unitate de terapie intensivă și presupune:

- achiziție de date;
- afișaj în timp real;
- măsurarea ritmului cardiac.

Nivelul "1"

Reprezintă o extindere a nivelului precedent prin mijloace suplimentare de procesare și de gestiune de date. Include de asemenea interfețe de comunicație cu stația centrală din nodul rețelei locale. Acest nivel presupune existența unui sistem de calcul pe bază de microprocesor ale cărui funcții sunt:

- pupitru de comandă multifuncțional;
- afișaj de mesaje și grafice pe ecran;
- execuția de rutine de procesare a semnalului în funcție de necesitățile monitorizării;
- memorarea de date, caracteristici ale complexului QRS, P, T pentru estimarea tendințelor de evoluție;
- alarmare locală sau la distanță;
- transmiterea prin rețea a semnalelor brute sau a parametrilor extrași.

Nivelul "2"

La acest nivel se realizează gestiunea simultană a semnalelor și a datelor de la mai mulți pacienți. Principalele caracteristici sunt următoarele:

- preluarea segmentelor de observație și a informațiilor de la nivelurile inferioare;
- aplicarea de algoritmi sofisticati pe aceste segmente;
- generarea și editarea de grafice și de texte mijlocite de o interfață prietenoasă;
- gestiunea alarmelor;
- interfața comunicației cu rețeaua locală;
- transferul informației către sistemul expert.

Nivelul " 3 "

El reunește toate informațiile dobândite la nivelurile inferioare și le integrează cu date provenind de la alte unități ale clinicii. Sarcinile ce sunt prevăzute pentru acest nivel sunt:

- interpretarea și înregistrarea în timp real a observațiilor, începând cu acelea care pot determina starea pacientului;
- selecția modalităților de monitorizare;
- indicații terapeutice;
- facilități de diagnosticare, rapoarte de stare;
- comunicare cu nivelul inferior.

O astfel de realizare nu poate fi avută în vedere decât pe termen lung. Global ea reprezintă un obiectiv ambițios de cercetare care trebuie să reunească obligatoriu specialiști în inginerie biomedicală, specialiști în inteligență artificială și experți clinici în domeniu.

Un pas important într-o etapă ulterioară va fi și acela al monitorizării în paralel a bolnavilor cu ajutorul camerelor TV plasate în saloane și în camera destinată probei de efort.

3.2. PREZENTAREA SISTEMULUI DE ACHIZIȚIE

Fiind cronologic prima problemă rezolvată din ansamblul demersurilor făcute pe parcursul tezei, sistemul de achiziție a reprezentat în ansamblul tezei un mijloc de atingere a obiectivelor tezei, nu un scop în sine.

3.2.1. Prezentarea părții hard a sistemului de achiziție utilizat.

Pentru a asigura pacienților securitatea maximă și pentru depășirea problemelor legate de obținerea autorizațiilor referitoare la respectarea normelor de protecție pentru aparatura electronică utilizată ce vine în contact cu bolnavul, am recurs la folosirea unui electrocardiograf clasic din dotarea secției de cardiologie pentru soluționarea părții de izolare galvanică și de procesare analogică, a semnalului ECG. Astfel am utilizat ieșirea analogică a electrocardiografului înregistrator tip MONIPAT-102B, la care am conectat sistemul de achiziție de date conectat la magistrala unui microcalculator IBM-PC.

Pentru punerea la punct a monitorizării, am utilizat un sistem de achiziție propriu (Mihu P. I., Breazu M., 1992), bazat pe un CAN cu aproximații succesive, montat pe bus-ul intern (magistrala ISA). Menționez că același sistem este în exploatare și actualmente pentru monitorizarea probei de efort. Elementele lui de bază, ilustrate în detaliu în partea de anexe a tezei sunt:

- două multiplexoare 4051 care permit achiziția a 16 tensiuni monopolare sau 8 tensiuni diferențiale;
- circuit de eșantionare și memorare AD583;
- convertor numeric analogic AD7541, cu rețea R-2R, pe 12 biți;
- comparator LM 311;
- decodificatorul de adrese care este realizat cu ajutorul circuitelor 74LS138 și 74LS30. Există posibilitatea de a selecta cu ajutorul unui comutator, două zone

de adrese din spațiul de adrese I/O, rezervate prin standardul IBM-PC pentru astfel de aplicații. Acestea sunt 278÷27F sau 2F8÷2FF;

- memorie PROM (82S147, 82S129). La aceasta, rezultatul furnizat de compararea tensiunii de convertit cu ieșirea CNA, formează cel mai semnificativ bit al cuvântului de adresă. Conținutul memoriei, de la adresa astfel formată constituie pe de o parte noua aproximare a valorii tensiunii analogice iar pe de altă parte viitoarea adresă ce se va forma, fără cel mai semnificativ bit al său.
- registru latch 74LS174 pentru conectarea ieșirilor de date ale memoriei la intrarea de adrese a memoriei și la intrarea CNA;
- amplificator de magistrală bidirecțional 74LS244 pentru conectarea ieșirii memoriei la magistrala de date a calculatorului;

Pentru controlul plăcii de achiziție, calculatorului îi revine sarcina de a realiza într-o procedură specială o secvență de scriere / citire de la adresele rezervate 278÷27F care prin decodificare hard pe placa de achiziție vor forma secvența de impulsuri electrice necesare funcționării ei. Achiziția propriu-zisă din acest program validează citirea memoriei cu viteza permisă de lanțul: PROM, CNA, comparator iar după al 12-lea impuls rezultatul este disponibil ca și conținut al ultimei adrese generate în urma celei de-a 12-a comparații. După al 4-lea impuls se citesc cei mai semnificativi 4 biți, iar după următoarele 8 se citesc și următorii 8 biți mai puțin semnificativi.

Timpul total de achiziție pentru un canal, obținut pentru 12 biți este de 80μs, utilizând un calculator IBM-PC funcționând la 100 MHz.

3.2.2. Stabilirea frecvenței de eșantionare și a nivelului de cuantificare.

Pentru alegerea frecvenței de eșantionare am utilizat relația 2.6. A rezultat în final $F_e=200$ Hz, rezultat folosit de mai mulți autori pentru monitorizare.

Pentru determinarea numărului minim de biți necesari, o analiză mai atentă a relației 2.3 (Jalaleddine, 1990), de referință în studiile bibliografice, duce însă la un rezultat diferit. Cu notațiile din paragraful 2.1.5, valoarea efectivă a zgomotului de cuantizare este (Toma, 1996):

$$e_{er} = \sqrt{\frac{1}{q} \int_{-q/2}^{q/2} e^2 de} = \frac{q}{\sqrt{12}} \quad (3.1)$$

Raportul semnal zgomot al cuantizorului pe N biți, se determină raportând mărimea domeniului acoperit de cuantizor (S), la tensiunea de zgomot:

$$SNR = \frac{S}{e_{er}} = \frac{q \cdot 2^N}{\frac{q}{\sqrt{12}}} = 2^N \sqrt{12} \quad (3.2)$$

$$[SNR]_{dB} = 20 \log(2^N \sqrt{12}) \cong 6 \cdot N + 10,8 \text{ dB} \quad (3.3)$$

Relația 3.3 arată că orice nou bit în plus introdus pentru cuantizarea unui semnal îmbunătățește raportul semnal zgomot cu 6 dB.

Probabilitatea de apariție a unor valori ale tensiunii ECG care să atingă marginile domeniului este scăzută. De aceea, se poate lăsa o plajă de siguranță

de $\pm 5\%$ pentru posibile valori accidentale peste cele scontate, la marginile inferioară și superioară. Rezultă o scădere a raportului semnal zgomot cu

$$20 \log \left(\frac{2^N}{2^N - \frac{10}{100} \cdot 2^N} \right) = 0,93 \text{ dB.} \quad (3.4)$$

deci raportul semnal zgomot va fi

$$[\text{SNR}]_{\text{dB}} = 6 \cdot N + 9,8 \text{ dB.} \quad (3.5)$$

Pe de altă parte, acest raport se exprimă ca fiind

$$[\text{SNR}]_{\text{dB}} = 20 \log \frac{V_{\text{ivv}}}{q} \text{ dB.} \quad (3.6)$$

Din relațiile 3.5 și 3.6 rezultă:

$$N_{\text{min}} \geq \frac{20 \log \frac{V_{\text{ivv}}}{q} - 9,8}{6} \quad (3.7)$$

Valorile date de relația 2.3 (Jaleledin ș.a.,1990), sunt acoperitoare însă calculul mai exact cu rel 3.7 oferă rezultate absolut satisfăcătoare. Aceste rezultate diferă în medie cu un bit, calculul exact ducând la un bit mai puțin. Consecința a celor prezentate și în acord cu literatura în domeniu vom utiliza 8 biți pentru achiziție.

3.2.3. Achiziția de date în contextul programului de monitorizare.

Prima problemă a constituit-o stabilirea sarcinilor principale și a ordinii priorităților, în distribuirea bugetului de timp al sistemului de achiziție (fig 3.5):

1. achiziția ($T_a=80 \mu\text{s}$ /canal, pentru placa utilizată);
2. procesarea (T_p);
3. alarmarea (inclusă în T_p);
4. selecție meniuri (T_m);
5. grafica, afișare date, (T_g).

Pentru lansarea achiziției la intervale de timp T_e foarte exacte, am utilizat modul de lucru prin întreruperi de tipul INT8, transmise prin IRQ0 circuitului de control al întreruperilor 8259. Deoarece frecvența de eșantionare trebuie să fie de 200 Hz ($T_e = 5 \text{ ms}$), se impune reprogramarea circuitului temporizator 8253 al calculatorului.

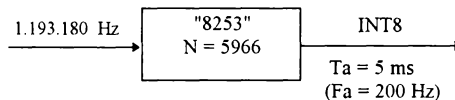


fig. 3.2. Reprogramarea INT8

În mod normal numărătorul "0" al circuitului temporizator este programat să furnizeze impulsuri pentru ceasul intern al calculatorului, astfel: numărătorul este încărcat inițial cu valoarea $N = 65.536$ și decrementat cu o frecvență $f_{ck} = 1.193.180$ Hz. La fiecare trecere prin zero, circuitul 8253 va furniza un impuls de întrerupere IRQ0 iar apoi se va reîncărca. Frecvența acestor impulsuri de întrerupere este în acest caz de 18,2 Hz ($T_C = 54,92$ ms).

Pentru obținerea frecvenței de 200 Hz va trebui reprogramată soft încărcarea numărătorului cu o nouă valoare N, (fig. 5.2)

$$N = \frac{f_{ck}}{F_e} = \frac{1.193.180}{200} = 5.966 = 174E \text{ [hexa]} \quad (3.8)$$

În același timp va trebui redirectat vectorul de întreruperi INT8, astfel ca aceste întreruperi să apeleze procedura de achiziție, în locul ceasului intern. Pentru aceasta se înlocuiește adresa procedurii de ceas din vectorul de întreruperi cu adresa procedurii de achiziție.

O problemă ce trebuie soluționată o dată cu modificarea făcută este aceea a menținerii funcționării ceasului intern al sistemului pe durata achiziției (Toma L., 1996) Acest lucru este necesar atât pentru funcționarea corectă a ceasului sistem cât mai ales pentru derularea corectă a activităților sistemului, care folosește IRQ0 și pentru alte scopuri (testare unități de disc). Pentru aceasta am utilizat schema bloc din fig. 3.3. în care întreruperile INT8 decrementează două contoare realizate soft. Primul contor va trebui să lanseze procedura de achiziție la fiecare 5 ms iar al doilea procedura de ceas la fiecare 54,92 ms.

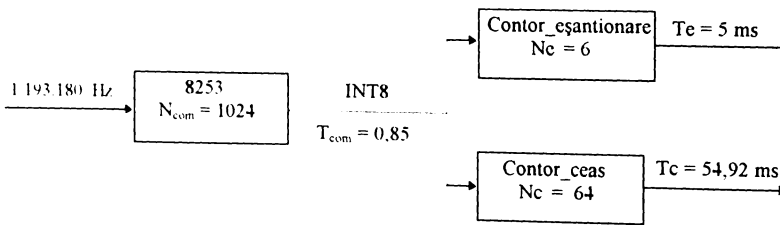


fig 3.3. Programare întreruperi INT8 cu menținerea ceasului intern.

Am determinat constantele de contorizare pentru obținerea celor două intervale de timp T_e și T_c , cât mai apropiate de valorile cerute. Rezultă sistemul nedeterminat de ecuații 3.9 ce trebuie rezolvat în numere întregi, unde prin $[[x]]$, am notat funcția "cel mai apropiat întreg de x":

$$\begin{cases} N_{com} \cdot N_c = 65.536 \\ N_e = \left[\left[\frac{5966}{N_{com}} \right] \right] \end{cases} \quad (3.9)$$

În alegerea soluției optime am avut în vedere și respectarea condiției ca pe durata unei achiziții, să nu mai sosească alte întreruperi INT8, deoarece rutinele lansate de ceasul intern fiind prioritare, ar întrerupe procesul de achiziție atunci când o întreruperea de ceas este lansată de primul impuls INT8 după lansarea achiziției.

$$T_{com} > k \cdot T_a \tag{3.10}$$

În relația 3.10, k reprezintă numărul de canale achiziționate. Câteva dintre seturile de valori care satisfac cerințele impuse sunt prezentate în tabelul 3.1.

N com	T com [μ s]	Nc	Ne	Fe [Hz]
2048	1716	32	3	194,20
1024	858	64	6	194,20
512	429	128	12	194,20
256	214	256	23	202,64
128	107	512	47	198,33

Tabelul 3.1. Selecția constantelor de programare contoare.

Prin alegerea făcută (în ipoteza a 5 canale de achiziție), frecvența de eșantionare nu va fi 200 Hz ci 194,20 Hz, ceea ce implică o eroare de sub 3% față de valoarea propusă inițial, însă aceasta nu influențează negativ calitatea rezultatelor ulterioare.

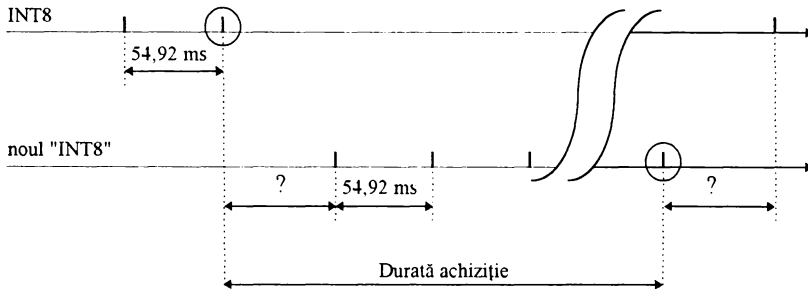


fig. 3.4. Sincronizarea redirectării vectorului de întrerupere.

Am studiat în continuare problema sincronismului momentului redirectării vectorului de întrerupere, pentru a nu exista discontinuități ale intervalului la care este lansată rutina de ceas.

Așa cum se observă din fig. 3.4, evitarea unui prim interval prea mare pentru ceasul sistemului, se poate face sincronizând momentul setării noii valori a constantei pentru contoare și redirectării vectorului de întrerupere cu apariția ultimului impuls INT8. În acest fel primul interval va fi mai lung decât cei 54,92 ms

doar cu durata necesară redirectării vectorilor, care este neglijabilă. Același lucru se întâmplă și la revenire, când sincronizarea trebuie să se facă cu ultimul "INT8". După terminarea părții de achiziție a programului, se revine la funcționare obișnuită prin reprogramarea temporizatorului 8253 cu $N=65.536$ și prin readucerea adresei procedurii de ceas în locația INT8 a vectorului de întreruperi.

În continuare (fig.3.5.) este prezentat modul în care este distribuit fondul de timp existent pe durata unei perioade de eșantionare, între sarcinile stabilite la început achiziție (T_a), procesare (T_p), selecție meniuri (T_m) și grafică (T_g).

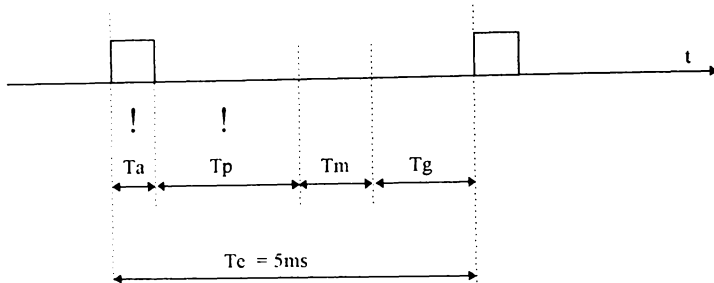


fig. 3.5. Structura fondului de timp.

De remarcat că din timpul total al unei perioade de eșantionare (T_e), trebuie alocat obligatoriu timp acoperitor pentru achiziție (T_a) și pentru procesare (T_p)

T_a reprezintă timpul total de derulare a procedurii achiziție pentru toate canalele. Timpul total pentru achiziția a trei canale (două derivații ECG și un canal pentru tensiune arterială), este de aprox. $240 \mu s$. Determinarea acestuia s-a făcut într-un program de estimare a duratelor unor secvențe de program, prin rulare acelei secvențe de un număr foarte mare de ori iar apoi determinarea duratei s-a făcut prin medierea intervalului de timp dat de ceasul intern al calculatorului. De este aproximativă, soluția folosită este acoperitoare, și asigură formarea unor imagini de ansamblu referitoare la împărțirea timpului unei perioade de eșantionare între "utilizatori". Pentru protecția timpului de achiziție, am invalidat întreruperile de la tastatură pe întreaga durată de execuție a procedurii de achiziție.

Timpul pentru procesare depinde de obiectivul și complexitatea obiectivelor procesării, el depinzând concret de meniul selectat.

Timpul necesar pentru selecție meniuri apare doar în momentul când de tastatură se intervine pentru a schimba o opțiune din meniul.

3.2.4 Afișarea

Inițial, am pornit de la ipoteza că afișajul reprezintă o sarcină secundară orientativă în raport cu celelalte procesări avute în vedere, așa cum a rezultat din analiza anterioară a fondului de timp. Cu toate acestea, din colaborarea cu medic a rezultat necesitatea acordării unei atenții deosebite și acestui aspect. Pentru a fi bine receptat de către personalul medical, afișajul ECG în timp real trebuie să :

facă cu vitezele standard de 25 mm/s sau de 50 mm/s, aceleași cu cele de la înregistratoarele ECG clasice.

3.2.4.1 Compresia de date pentru afișaj

Afișarea unui ECG la viteza $v = 25$ mm/s pe întreaga lățime a unui monitor cu diagonală $d = 13$ ", permite vizualizarea a aproximativ 9,3 sec din semnal, conform relației 3.11:

$$t = \frac{2,54 \cdot d}{\sqrt{2}} \cdot \frac{1}{v} \quad (3.11)$$

Aceasta înseamnă că la o frecvență de eșantionare de $F_e = 200$ Hz (avută în vedere din alte considerente, prezentate anterior), vor trebui afișate

$$N = t \cdot F_e = 1860 \text{ eșantioane.} \quad (3.12)$$

Fiindcă utilizăm o placă VGA (640 x 350), ce oferă

$$c = \frac{640}{\frac{2,54 \cdot d}{\sqrt{2}}} = 2,74 \text{ pix / mm} \quad (3.13)$$

nu este posibilă o astfel de rezoluție pe orizontală. Este evident că se impune un proces de compresie a datelor cu o rată de compresie:

$$CR = \frac{N}{640} : 1 = 2,9:1 \text{ la } 25 \text{ mm/s} \quad (3.14)$$

Din motive de eficiență și timp de calcul este mai interesantă obținerea unei rate de compresie de 4:1 sau 2:1, mai simplu de implementat. Deoarece placa VGA oferă un număr constant de pixeli/cm, soluția de obținere a unei rate de compresie simple de implementat și a unei viteze standard de afișare o reprezintă adaptarea ratei de eșantionare conform relației 3.15:

$$F_e = c \cdot v \cdot CR \quad (3.15)$$

Rezultă din 3.15 următoarele seturi de valori pentru F_e , v și CR , care rămân în discuție spre a fi alese

Fe	205,5 Hz		274 Hz	
v	25 mm/s	50 mm/s	25 mm/s	50 mm/s
CR	3:1	3:2	4:1	2:1

Tabelul 3.2

În varianta $F_e = 274$ Hz pentru realizarea unei rate de compresie de 2:1 o primă soluție ar fi subeșantionarea corespunzătoare a datelor de afișat, în cazul nostru cu o rată de 2:1 adică, pentru afișare ar trebui luat tot al doilea eșantion. Aceasta metodă are un dezavantaj major și anume faptul că pot fi "scăpate" puncte esențiale și în special virfurile care marchează amplitudinea diferitelor unde, deci această soluție ar afecta calitatea semnalului afișat.

Ținând cont de acestea, am ales o soluție care este derivată din metoda de compresie prezentată la 2.3.1.1 și anume "extragerea punctelor esențiale", datorită simplității sale și deoarece nu este nevoie de un raport de compresie mare.

Regula după care un eșantion este considerat "esențial", se stabilește cu ajutorul situațiilor posibile ale unei secvențe de trei eșantioane, descrise în fig. 3.6.

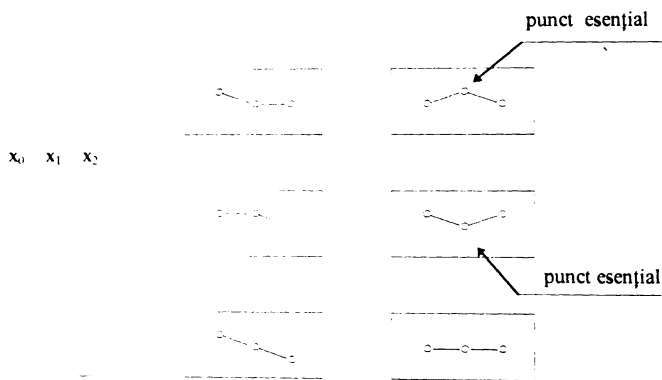


fig. 3.6. Definirea eșantioanelor esențiale

Ca principiu, algoritmul analizează tendința de evoluție a valorii eșantioanelor și memorează unul din fiecare pereche de două consecutive. Primul eșantion X_0 este preluat ca punct de referință. Acesta, împreună cu următoarele două consecutive X_1 și X_2 , se constituie într-o fereastră de analiză. Din fig. 3.6 se observă că în două cazuri am definit în cadrul ferestrei de analiză puncte esențiale, atunci când:

$$x_0 > x_1 < x_2 \quad \text{sau} \quad x_0 < x_1 > x_2 \tag{3.16}$$

În toate cazurile se va reține eșantionul X_2 excepție făcând cazul în care X_1 este punct de extrem conform definiției 3.9, situație în care se va reține X_1 .

Eșantionul reținut va constitui eșantion de referință pentru următoarea testare. Astfel, din fiecare două eșantioane consecutive, se reține unul. Excepție făcând primul eșantion luat de referință, rezultă o rată de compresie de 2:1. Pentru implementare se aplică algoritmul:

```
IF (( $x_2 - x_1$ ) · ( $x_1 - x_0$ ) < 0) THEN  $x_0 = x_1$ 
ELSE  $x_0 = x_2$ 
```

În cazul în care se dorește afișarea cu viteza de 25 mm/s și trebuie realizată o compresie de 4:1 algoritmul se aplică de două ori. Deși nu este o transformare liniară, aplicarea consecutivă a algoritmului conservă proprietatea sa de bază, aceea de a reține eșantioanele esențiale și atunci când din patru eșantioane se reține unul.

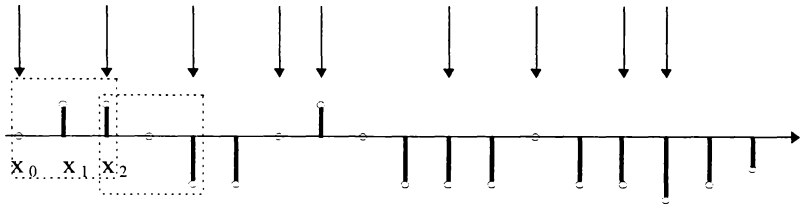


fig. 3.10. Compresie 2:1 cu reținerea punctelor esențiale.

Pentru ilustrare, în fig. 3.10 sunt marcate eșantioanele reținute dintr-o secvență oarecare. De remarcat faptul că algoritmul este extrem de rapid și în plus păstrează nealterate amplitudinile undelor importante ale traseului ECG afișat.

3.2.4.2 Reprezentări grafice ale traseului ECG

Vizualizarea unui traseu ECG pe monitorul PC se poate face în două variante, ambele prezentând asemănări cu înregistrarea tradițională făcută de electrocardiografe pe benzi de hârtie.

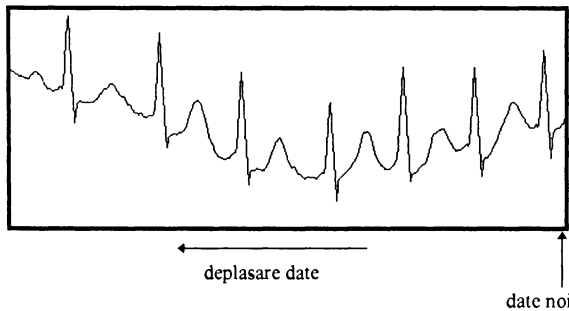


fig. 3.11 Afișarea ECG cu fereastră glisantă

Prima soluție, "fereastră glisantă", este agreată de personalul medical care o asimilează cu modul tradițional de înregistrare pe hârtie și constă în a afișa o fereastră cuprinzând un număr convenabil de cicluri cardiace (aproximativ 10 la un puls de 70 cicluri / minut), în ritmul sosirii eșantioanelor. Eșantionul nou sosit se

înregistrează mereu în dreapta ferestrei de afișare, toate celelalte mutându-se cu un pixel spre stânga (fig. 3.11). În acest fel se dă impresia că înregistrarea "curge" înspre stânga într-un cadru static, ultimele date ECG venind în partea dreaptă a cadrului. Dezavantajul major constă în aceea că pentru fiecare nou eșantion reprezentat trebuie reîmprospătat tot ecranul, ceea ce presupune un timp mare solicitat bugetului de timp al sistemului. Acest timp neputând fi luat din timpul necesar achiziției sau procesării de semnal, soluția nu este adecvată în cazul monitorizării ECG în timp real în proba de efort.

A doua metodă (fig. 3.12), prezintă în cadrul ferestrei de afișaj aleasă, o bară glisantă de la stânga la dreapta. Bara șterge graficul vechi iar valorile noi se vor afișa imediat în stânga barei, odată cu deplasarea ei. "Ștergerea" se face prin trasarea unei linii de aceeași culoare cu a fondului, la coordonata la care dorim să afișăm noul eșantion, după care se plasează efectiv eșantionul, cu o culoare diferită pentru grafic. Pentru a fi mai sugestiv, este util ca simultan cu desenarea pixelului corespunzător eșantionului, să se traseze o bară, de o altă culoare, cu un pixel înaintea coordonatei eșantionului curent, ceea ce va da mai bine senzația de derulare a înregistrării.

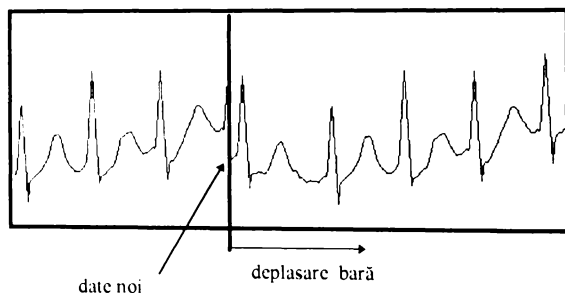


fig. 3.12 Afișarea ECG cu bară glisantă

Față de prima soluție, acum se reîmprospătează doar o linie având lungime egală cu înălțimea ferestrei de afișaj ceea ce reduce mult timpul necesar afișării.

O variantă a soluției cu bară glisantă o constituie ștergerea doar a pixelului care fusese desenat anterior la coordonata eșantionului curent. Aceasta presupune să se memoreze într-un tablou toate valorile existente la un moment dat în fereastra de afișaj. Necesarul de memorie utilizat în plus este justificat de reducerea spectaculoasă a timpului de afișaj în raportul k dat de relația 3.17:

$$k = \frac{1}{H} \quad (3.17)$$

unde H reprezintă numărul de pixeli al "înălțimii" ferestrei de afișaj.

Afișarea grafică realizată pentru monitorizarea probei de efort utilizează metoda cu bară glisantă, deoarece necesită un timp mai redus decât altele. Chiar și așa, afișarea rămâne "marea consumatoare" de timp. Timpul necesar afișării unui

pixel este în medie de 36 μ s, al procedurii de setare și trasare a unei linii necesare ștergerii graficului pe toată lățimea sa este de 297 μ s, iar al procedurii de afișare a unui mesaj alfanumeric poate ajunge chiar la 1 ms! Cu toate acestea nu este o sarcină de importanță majoră și de aceea prin structura programului, afișarea este "sacrificată" în momentul în care există un deficit de timp în cadrul unei perioade de eșantionare. Acest lucru se manifestă prin lipsa câte unui pixel, sau chiar prezența unei linii neșterse, dar pe parcursul tuturor testelor afișarea a fost absolut satisfăcătoare din punct de vedere clinic.

Sistemul de achiziție prezentat a permis realizarea unei baze de date proprii care a servit dezvoltării ulterioare a cercetărilor, precum și implementarea ulterioară a tuturor programelor de monitorizare prezentate în lucrare.

3.3 IMPLEMENTAREA DETECTOARELOR QRS

În paragraful de față sunt implementate detectoare QRS incluzând partea de preprocesare precum și detectorul propriu-zis. Programele realizate se constituie ca o etapă de cercetare în obiectivul major al demersurilor pe care le vizează tema tezei: concretizarea unui modul pentru investigarea complexă în timp real a probei de efort.

Ca primă etapă, este necesară abordarea și cunoașterea de fond a aspectelor teoretice și de implementare ale tuturor modulelor care vor concura în final la crearea unui sistem performant și mai ales eficient. Modulul de realizat îl constituie "procesarea de nivel primar", adică acea parte situată în apropierea pacientului. În cadrul acesteia un rol esențial îl are realizarea detecției QRS. Deși performanțele și eficiența detecției nu vor putea fi judecate decât după integrarea în ansamblul sistemului de monitorizare totuși, în interiorul acestui modul pot fi realizați algoritmi care vor fi o garanție pentru eficiența globală a soluțiilor. Tot în această fază, am avut în vedere analiza aritmiilor și realizarea unor alarmări.

Toate programele sunt scrise în limbaj PASCAL. Am avut permanent în vedere ca aceste programe să poată constitui un instrument eficient de cercetare, fără a neglija însă prezentarea facilă a modului de utilizare prin meniuri, care să le facă ușor accesibile personalului medical sau utile și în scop didactic.

În continuare vor fi prezentate soluțiile practice de implementare a algoritmilor mai importanți, clasificați în cap.2. Ei sunt grupați în același program spre a putea fi ușor testați și apoi comparați din punct de vedere al eficacității în evidențierea complexului QRS.

3.3.1 Caracteristica de frecvență a procesărilor ECG.

Ideea realizării unui program care să permită trasarea facilă a caracteristicilor de frecvență ale funcțiilor de preprocesare propuse pentru ECG, s-a impus datorită necesității de a corela analiza temporală cu cea frecvențială și pentru a stabili eventuale concluzii mai puternice în favoarea utilizării uneia sau alteia din soluțiile propuse pentru procesarea ECG, oricare ar fi scopul ei.

Din punct de vedere spectral toate funcțiile de procesare liniară sunt în fond filtre numerice iar caracteristica de frecvență se poate trasa ușor prin aplicarea funcției de transfer semnalului complex de intrare:

$$Xdat(i) = e^{j(2\pi i f \cdot T e - \varphi)} \quad (3.18)$$

Cu toate acestea am găsit utilă și am concretizat soluția de față din următoarele considerente:

- necesitatea de trasare facilă a caracteristicilor de frecvență pentru transformări liniare (nerecursive sau recursive);
- posibilitatea de a evalua simplu comportarea în domeniul frecvență a transformărilor neliniare;
- posibilitatea de a trata cu aceleași instrumente din punct de vedere al analizei spectrale procesări care au fost create pornind de la analize morfologice de semnal în domeniul timp și de a vedea rapid legătura dintre ele;
- posibilitatea ca funcția de procesare să se introducă simplu în program prin chiar expresia ei analitică.

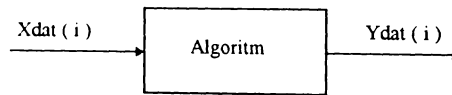


fig. 3.11. Algoritmul de procesare

Considerăm procesarea ca fiind cea descrisă în fig. 3.11. $Xdat(i)$ reprezintă secvența de date ECG ce urmează a fi procesate iar $Ydat(i)$ reprezintă secvența de date rezultată în urma procesării furnizate de algoritmul propus spre cercetare. Algoritmul poate fi unul liniar recursiv sau nerecursiv.

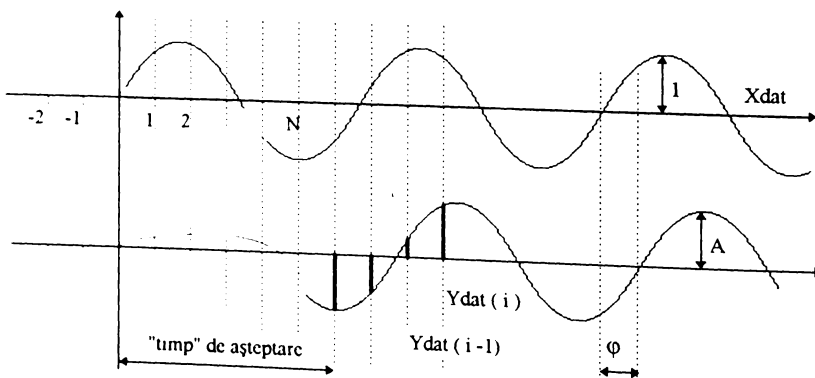


fig 3 12 Alegerea eșantioanelor din secvența de ieșire

Considerând secvența de intrare o sinusoidă reală de amplitudine unitară și fază nulă, evoluția în timp a intrării și a ieșirii filtrului este cea prezentată în fig.3.12. Pentru orice transformare liniară, secvența de ieșire este tot armonică, deci două eșantioane consecutive din secvența de ieșire Y_{dat} se exprimă astfel:

$$Y_{dat}(i) = A \cdot \sin\left[2\pi i \frac{f}{F_e} + \varphi\right] \quad (3.19)$$

$$Y_{dat}(i-1) = A \cdot \sin\left[2\pi(i-1) \frac{f}{F_e} + \varphi\right] \quad (3.20)$$

Relațiile 3.19 și 3.20 formează un sistem de două ecuații în care:

- A și φ necunoscute
- $Y_{dat}(i), Y_{dat}(i-1)$ două eșantioane consecutive din secvența de ieșire
- F_e frecvența de eșantionare a semnalului
- f frecvența semnalului sinusoidal de intrare.

Rezolvarea sistemului duce la:

$$\varphi = \arctg \frac{\sin \alpha - \frac{Y_{dat}(i)}{Y_{dat}(i-1)} \cdot \sin \beta}{\frac{Y_{dat}(i)}{Y_{dat}(i-1)} \cdot \cos \beta - \cos \alpha} \quad (3.21)$$

$$A = \frac{Y_{dat}(i)}{\sin[\alpha + \varphi]} \quad (3.22)$$

unde am folosit notațiile:

$$\alpha = 2\pi i \frac{f}{F_e} \quad ; \quad \beta = 2\pi(i-1) \frac{f}{F_e} \quad (3.23)$$

Deci pentru o frecvență dată a semnalului de intrare, se pot determina amplitudinea și faza semnalului de ieșire cunoscând valoarea a două eșantioane consecutive. Nu rămâne decât să "trecem" pe rând prin algoritmul căruia dorim să-i stabilim caracteristica de frecvență, semnale de intrare având frecvențe dintr-un domeniu dorit. Se obțin astfel dependențe

$$A = A(f) \quad ; \quad \varphi = \varphi(f) \quad (3.24)$$

care reprezintă chiar caracteristicile de frecvență referitoare la domeniul ales pentru algoritmul (filtrul) propus. Trebuie făcute câteva mențiuni referitoare la aplicarea soluției propuse:

- Cele două eșantioane de ieșire luate în calcule trebuie prelevate din secvență abia după trecerea prin filtru a unui număr suficient de date de intrare. În cazul algoritmilor nerecursivi, numărul de date trebuie să depășească "lungimea" N a filtrului. Prin "lungime" (sau ordin al filtrului) se înțelege numărul ultimelor N

eșantioane din secvența de intrare asupra cărora se va opera în timpul filtrării. Abia din acel moment secvența de ieșire devine armonică (filtru cu răspuns finit la impuls - FIR) și rezultatele vor fi corecte. În cazul algoritmilor recursivi (filtre cu răspuns infinit la impuls - IIR), abia după un număr foarte mare secvența de ieșire devine armonică însă practic se obțin rezultate bune după un număr acceptabil de date trecute prin filtru. Legat de cele amintite trebuie avut în vedere ca la implementare, variabila tablou ce conține secvența de intrare Xdat, să fie completată în partea stângă cu cel puțin N eșantioane Xdat (-N), Xdat (-N + 1), ..., Xdat (-1), pentru ca procesarea să poată începe cu Xdat (0). Valoarea acestora nu afectează rezultatul, dacă prelevarea celor două eșantioane de ieșire se face corect.

- Evitarea situațiilor în care cele două eșantioane sunt de valori egale (când se ajunge în situații de împărțire la zero) se poate face în program, testând la început diferența dintre ele și în cazul în care ea ajunge sub o limită impusă, se va alege o altă pereche de eșantioane. Se vor corecta corespunzător și valorile α și β din sistem, în funcție de "i" și de distanța dintre eșantioane.

3.3.2 Detector QRS bazat pe derivata de ordinul 2, "patru puncte"

Pentru realizarea acestui algoritm am pornit de la analiza morfologică a înregistrării ECG. Din studiul asupra bazei proprii de date realizată cu o frecvență de eșantionare de 200 Hz, se constată că cea mai evidentă proprietate a complexului QRS este panta mare pozitivă pe durata medie a patru eșantioane, corespunzătoare intervalului ascendent al undei R și panta negativă mare pe durata a patru sau cinci eșantioane corespunzător fazei descendentă a undei R.

Această constatare reprezintă o condiție asiguratorie fiindcă în situații atipice (fibrilație ventriculară) se întâlnesc pentru faza ascendentă mai rar trei și mai des cinci eșantioane iar pentru faza descendentă se întâlnesc mai frecvent cinci și mai rar chiar șase eșantioane. Această ascendență de patru eșantioane asociată cu o descendență pe durata următoarelor patru eșantioane constituie elementul morfologic ce va fi exploatat. Matematic, se exprimă panta dintre cele trei puncte considerând ca referință eșantionul X_n , vârful undei R:

$$\text{tg}^{\text{"4+"}} = (X_n - X_{n-4}) / 4Te \quad (\text{ascendentă}) \quad (3.25)$$

$$\text{tg}^{\text{"4-"}} = (X_{n-4} - X_n) / 4Te \quad (\text{descendentă}) \quad (3.26)$$

Se constată astfel că vârful undei R se încadrează cu patru perioade de eșantionare în stânga și patru în dreapta, deci pe un interval de:

$$8 Te = 8 \cdot 5 \text{ ms} = 40 \text{ ms} \quad (3.27)$$

Durata tipică a complexului QRS este de 100 ms din care fazelor ascendente și descendente le corespund împreună aproximativ 60 -70 ms. Dacă la aceasta se mai adaugă faptul că eșantionarea nu "cade" exact în vârful undei R și că apare astfel o imprecizie de plasare a vârfului undei R față de eșantionul X_n de

+/- 5ms (10 ms), rezultă că alegerea celor 8 intervale de eșantionare prin analiza morfologică se încadrează acoperitor în intervalele cunoscute ale undei R.

Pentru evidențierea alăturării celei mai mari pante pozitive și a celei mai mari pante negative, trebuie studiată derivata a doua "patru puncte".

$$Y'' = -(X_n - X_{n-4}) + (X_{n-4} - X_{n-8}) = -X_n + 2X_{n-4} - X_{n-8} \quad (3.28)$$

Se constată că relația 3.28 provine din cele două precedente printr-o translație în timp de patru intervale pentru ca expresia Y'' să devină cauzală, permițând în acest fel aplicarea algoritmului și în timp real. De asemenea în transformarea 3.28, din motiv de viteză de calcul nu mai include divizarea cu 4Te.

Implementarea relației 3.28 s-a făcut în cadrul meniului "Transformări specifice" al programului dedicat studiului detectoarelor QRS, prezentat în anexă. Modalitatea efectivă de realizare a transformării o constituie procedura "Filtru_QRS" din UNIT-ul "Id_QRS". Din programul principal se transmite prin variabila de legătură "tip_filtru" valori corespunzătoare fiecărui tip de filtru implementat. Pentru a implementa un nou tip de transformare trebuie introdusă în meniul inițial numele noului tip de transformare și o constantă nouă de apel pentru meniul Turbo Vision. În continuare se atribuie variabilei de transfer "tip_filtru" o nouă valoare în Event.Command iar apoi se introduce relația dorită cu instrucțiunea IF în cadrul procedurii "Filtru_QRS".

Rezultatul acțiunii preprocesării este accesibil imediat, prin afișarea pe ecran a fișierului de date ECG original și a celui procesat, pe același sistem de axe de coordonate.

O soluție simplă de apreciere a caracteristicii de frecvență a acestui tip de filtru este "trecerea" prin această transformare a unui semnal de frecvență uniform crescătoare între 1 Hz și 100 Hz. Semnalul este generat matematic și este depus în fișierul FVAR. Se constată că transformarea propusă reprezintă în termeni de analiză spectrală un FTB.

3.3.3 Detector QRS bazat pe derivata de ordinul 3, "patru puncte".

Acest tip de procesare este un model dezvoltat pe scheletul celui precedent. Astfel, din analiza proprietăților temporale ale semnalului Y'' , rezultă că panta sa are o variație maximă în mijlocul intervalului de 8 eșantioane. Derivata acestui semnal va fi maximă în momentul apariției undei R. Aceasta înseamnă matematic:

$$Y''' = -Y''_n + 2Y''_{n-4} - Y''_{n-8} \quad (3.29)$$

Relația 3.22 devine după dezvoltare :

$$Y''' = X_n - 4X_{n-4} + 6X_{n-8} - 4X_{n-12} + X_{n-16} \quad (3.30)$$

Concluzii:

- Rezultatele procesării unui fișier cu date ECG cu ajutorul celor două tipuri de derivate sunt prezentate în fig. 3.13. Se constată eficiența mai crescută a celui de-al doilea filtru deoarece amplitudinea semnalului rezultat corespunzător

complexului QRS este de aproximativ 1,3 ori mai mare și în același timp celelalte evenimente produc la ieșire valori mult mai mici în comparație cu valoarea corespunzătoare complexului QRS.

- De menționat că întârzierea semnalului astfel procesat este de $8T_e$ față de $4T_e$ în cazul celui precedent.

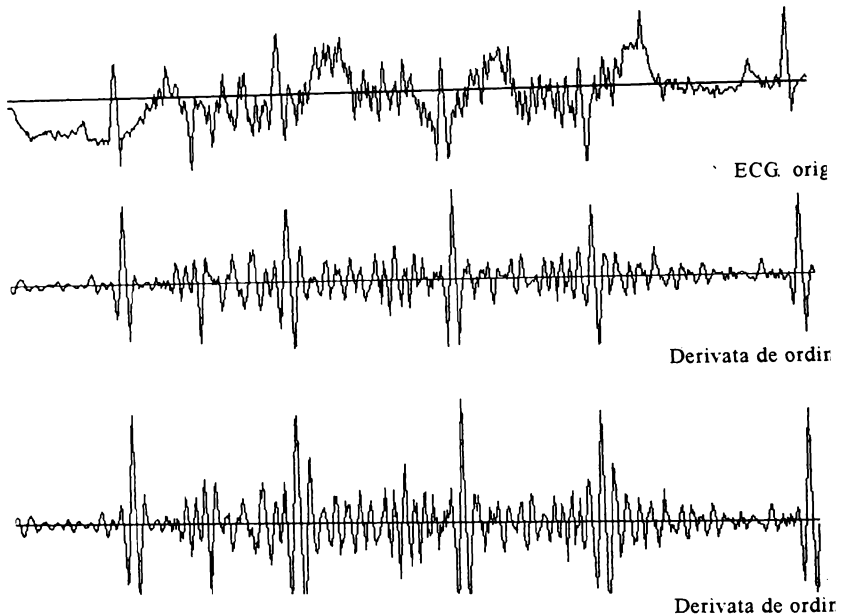


fig 3.13 Rezultatele procesării cu derivate "4 puncte"

- Pentru analiza în domeniul frecvență, am utilizat programul special prezentat în paragraful 3.3.1, cu ajutorul căruia am obținut caracteristicile de frecvență (amplitudine și fază) pentru cei doi algoritmi (3.28 și 3.30) analizați, caracteristici prezentate în fig 3.14 și 3.15.
- La o primă analiză a acestora, se constată că, atât derivata de ordinul 2, cât și cea de ordinul 3 au caracteristici de filtre trece bandă și au aproximativ aceeași bandă de trecere. Primul are amplificarea la mijlocul benzii 12 iar cel de-al doilea 24. Acestui fapt i se datorează și evidențierea mai bună a complexului QRS de către cel de-al doilea și de o manieră care face această transformare indicată pentru o ulterioară operație de identificare printr-un algoritm cu prag a QRS.

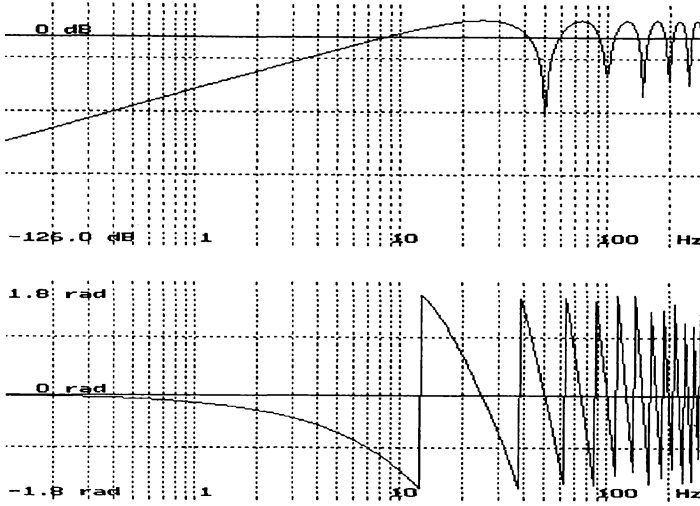


fig. 3.14. Caracteristica de frecvență a filtrului "derivata de ordinul 2"

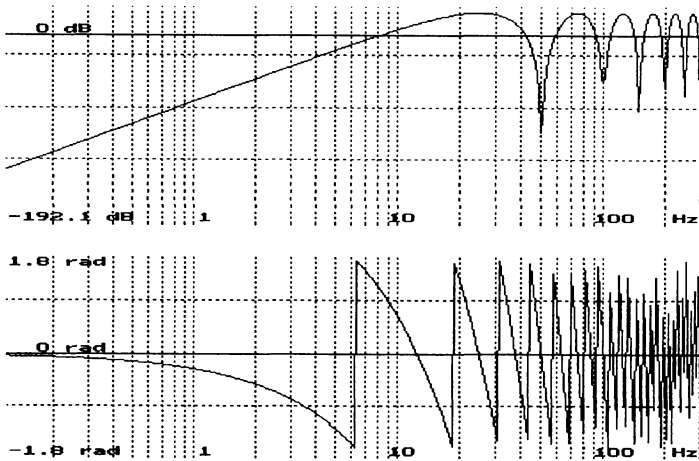


fig. 3.15 Caracteristica de frecvență a filtrului "derivata de ordinul 3".

- Frecvența centrală a ambelor filtre este 12,5 Hz, ceea ce este în acord cu datele referitoare la conținutul spectral pe care complexul QRS îl aduce semnalului ECG.

- Se constată de asemenea că atunci când $F_e = 200$ Hz, ambele filtre au calitatea de a rejecta frecvența de 50 Hz.
- Referitor la caracteristica de fază, se observă că pe intervalul frecvențelor utile au loc variații foarte importante ale fazei, însă în cazul procesării făcute doar cu scopul accentuării complexului QRS, aceasta nu deranjează.
- Se constată că ambele filtre amplifică și semnale având frecvența mai mare decât 100 Hz, ceea ce obligă la limitarea spectrului de frecvență al semnalului la 100 Hz, ținând cont că frecvența de eșantionare este de 200Hz.
- Timpul de calcul necesar transformării 3.28 este redus: 3 înmulțiri și 2 adunări. În acest fel se întrevide utilizarea algoritmului la procesarea în timp real a ECG
- Efectul celei de-a doua transformări s-a dovedit mai bun decât al primeia și pentru testele realizate pe semnale conținând: deriva liniei de bază, zgomot 50 Hz, zgomot muscular, semnal afectat de respirație (0,3 Hz).
- Din analiza rezultatelor reies concluzii utile pentru următoarea etapă care o constituie identificarea efectivă a "evenimentului" QRS prin metoda comparației cu praguri. Astfel, pentru această transformare, analizând maximele locale ce apar în afara complexului QRS și care ajung la 0,6 din valoarea maximum util datorat QRS, rezultă ca suficient un prag de valoare 0,75 din valoarea maximum precedent.

3.3.4 Detector QRS utilizând corelația cu funcții "șablon"

Ideea realizării acestui gen de filtre o constituie stabilirea gradului de asemănare a unei ferestre din înregistrarea ECG cu o formă "șablon" prestabilită. Pornind de la prezentarea făcută în paragraful 2.2.3.4, este utilă sinteza unor șabloane cu o formă simplă, descrisă pe cât posibil cu coeficienți numere întregi sau puteri ale lui 2

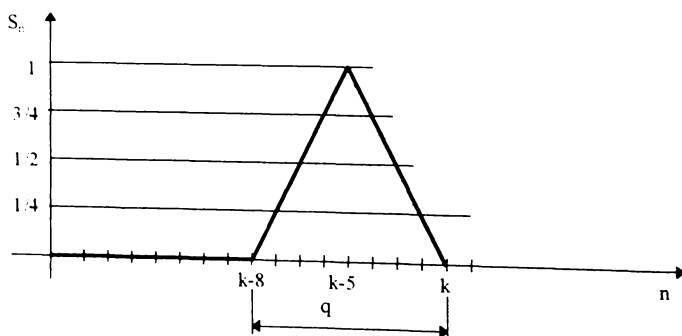


fig 3.16. Șablonul simplu SS1.

Pentru început voi prezenta rezultatul corelării semnalului ECG cu o formă care încearcă să copieze forma undei R în punctul său de vârf, o formă de triunghi

isoscel de amplitudine unitară. Baza triunghiului se întinde pe durata a 8 intervale T_e , consecință a celor prezentate în paragraful anterior. Șablonul este prezentat în fig 3.16 și este botezat "șablon simplu SS1" pentru a-l putea identifica în meniul programului pentru testarea comportării algoritmilor.

Expresia funcției de corelație este:

$$Y(n) = \sum_{i=1}^8 (X_{n-i} S_i) \tag{3.31}$$

Valorile funcției șablon cu care se va face corelarea sunt cele ce rezultă din fig. 3.16, și creează forma explicită 3.32:

$$Y_n = X_{n-1}/4 + X_{n-2}/2 + 3 X_{n-3}/4 + X_{n-4} + 3 X_{n-5}/4 + X_{n-6}/2 + X_{n-7}/4 \tag{3.32}$$

Pentru a studia proprietățile procesării cu astfel de funcții, al doilea șablon sintetizat este cel din fig. 3.17, șablon pentru care expresia funcției de corelare este cea din relația 3.33.

$$Y_n = X_n/2 + X_{n-2}/4 + X_{n-3}/2 + 3X_{n-4}/4 + X_{n-5}/2 + X_{n-6}/4 - X_{n-8}/4 \tag{3.33}$$

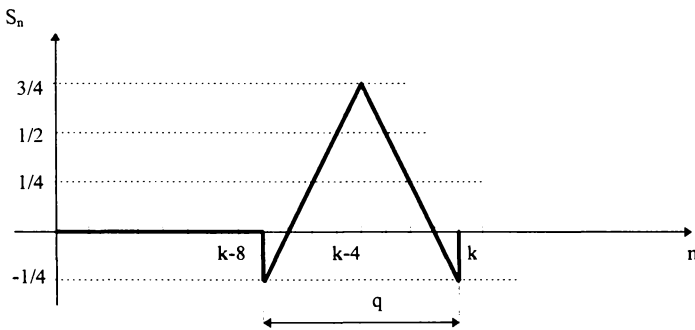


fig 3.17 Șablonul simplu SS2.

Relațiile 3.32 și 3.33 sunt implementate în submeniul "procesări speciale " iar rezultatele efective sunt prezentate în fig 3.18.

Se constată că, într-adevăr, sunt evidențiate complexe QRS dar în același timp sunt evidențiate în mod nedorit și zonele cu variație lentă respectiv porțiuni ce conțin o componentă continuă de mare valoare. De asemenea algoritmul 3.33 duce la un rezultat de ansamblu mai bun față de algoritmul 3.32.

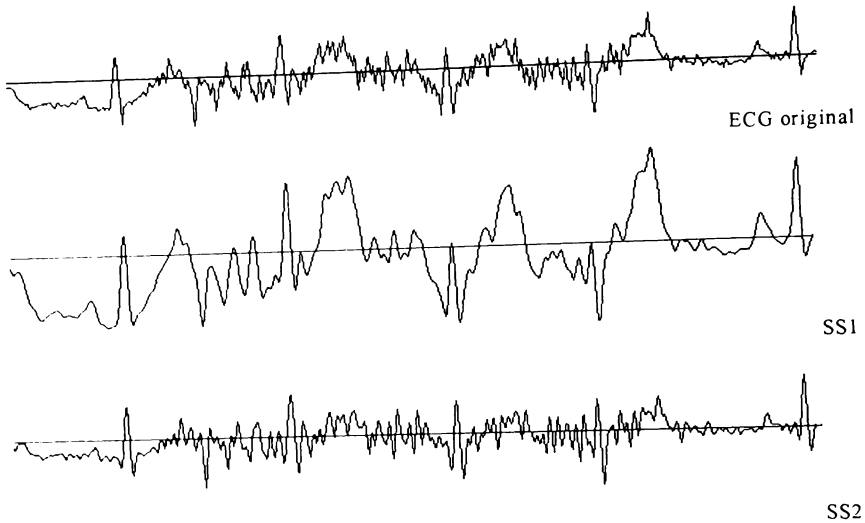


fig 3.18 Rezultatul corelării cu sabloanele SS1 și SS2

Această situație face improprie utilizarea acestui șablon la identificări ulterioare cu praguri, datorită prezenței componente continue de valoare imprevizibilă.

3.3.5 Detector QRS utilizând corelația cu funcții "șablon" + FTS

Ideea realizării acestui tip de transformare a plecat de la cea precedentă, în încercarea de a realiza simultan, atât eliminarea componente continue, cât și corelarea cu un șablon dat. Pentru aceasta este limpede că valorile funcției șablon trebuie să satisfacă condiții în plus față de cazul șabloanelor din fig. 3.12 și fig. 3.13. O analiză extrem de simplă va rezolva problema astfel: Presupunem la intrarea filtrului o secvență de valori X_i toate egale, de valoare constantă K . Dacă dorim ca rezultatul procesării acestei componente continue să producă un rezultat nul la ieșire va trebui ca în relația 3.31 să fie pusă condiția:

$$Y(n) = \sum_{i=1}^n (K S_i) = K \sum_{i=1}^n S_i = 0 \quad (3.34)$$

Rezultă deci că suma coeficienților funcției șablon trebuie să fie nulă dacă dorim să realizăm simultan o corelare în sensul dat de relația 3.24, și în același timp o filtrare pentru a elimina componenta continuă. Considerând o perioadă de eșantionare unitară, relația 3.34 mai poate fi interpretată astfel: *șablonul a căru arie cuprinsă între el și axa timp este nulă, va produce automat prin corelare și o filtrare eficientă pentru rejectia componente continue.*

Conform condiției 3.34 am sintetizat mai multe funcții șablon, ale căror grafice sunt prezentate în fig. 3.19 -3.20. Scopul creării lor este acela de a le testa comportarea în condițiile a diverse înregistrări ECG și de a decide în alegerea celu

mai potrivit pentru preprocesarea ECG în timp real, în vederea identificării complexului QRS.

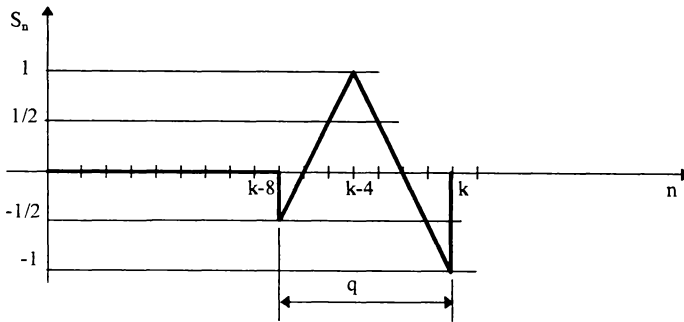


fig 3.19 Șablonul S1

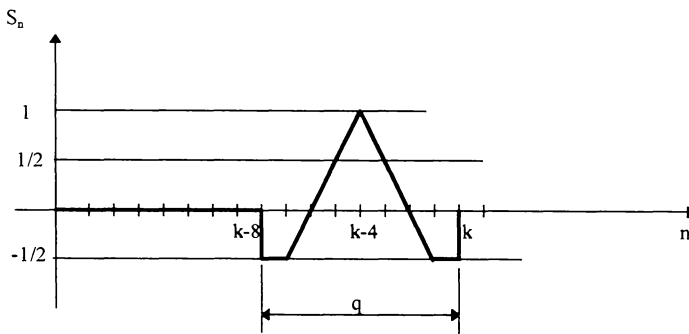


fig 3.20 Șablonul S2

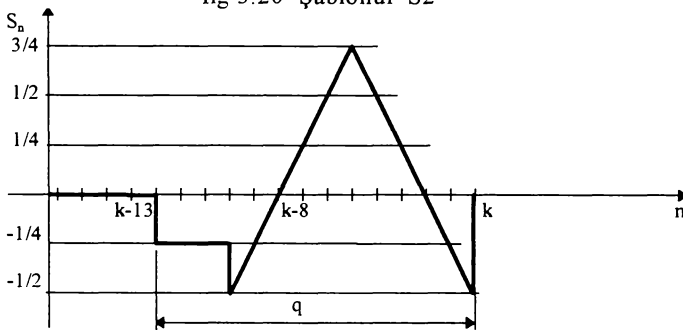


fig 3.21. Șablonul S3

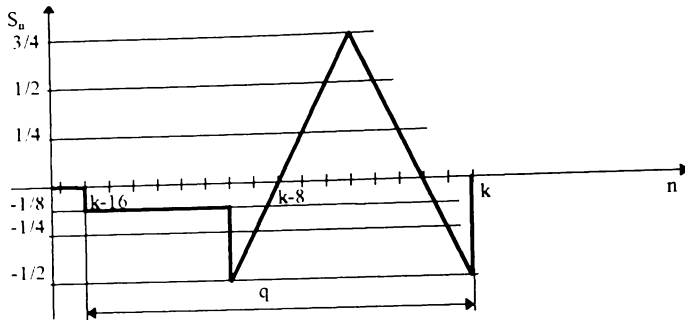


fig 3.22. Șablonul S4

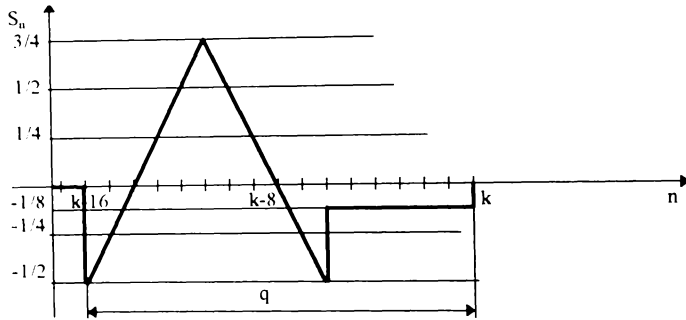


fig 3.23. Șablonul S5

De menționat că în definiția acestor funcții șablon am utilizat doar puteri întregi ale lui 2 în scopul unei ulterioare implementări a algoritmului în limbaj de asamblare, știut fiind că o împărțire cu 2 se realizează printr-o simplă deplasare de un bit la dreapta.

O a doua idee de realizare a unor funcții șablon care să efectueze simultan și rejecția componentei continue, deci o filtrare trece sus, este următoarea: se realizează funcția șablon având forma cea mai apropiată de cea a complexului QRS iar apoi aceasta este supusă unei transformări ce are ca rezultat comportarea spectrală dorită. Astfel funcția șablon din fig. 3.12 care așa cum s-a văzut nu are o comportare satisfăcătoare, supusă unei filtrări de tip FTS realizată prin diferențierea simplă de ordinul 1 devine:

$$Y(n) - Y(n-1) = \sum_{i=1}^q (X_{n-i} S_i - X_{n-1-i} S_i) = \sum_{i=1}^q (X_{n-i} - X_{n-1-i}) S_i \quad (3.35)$$

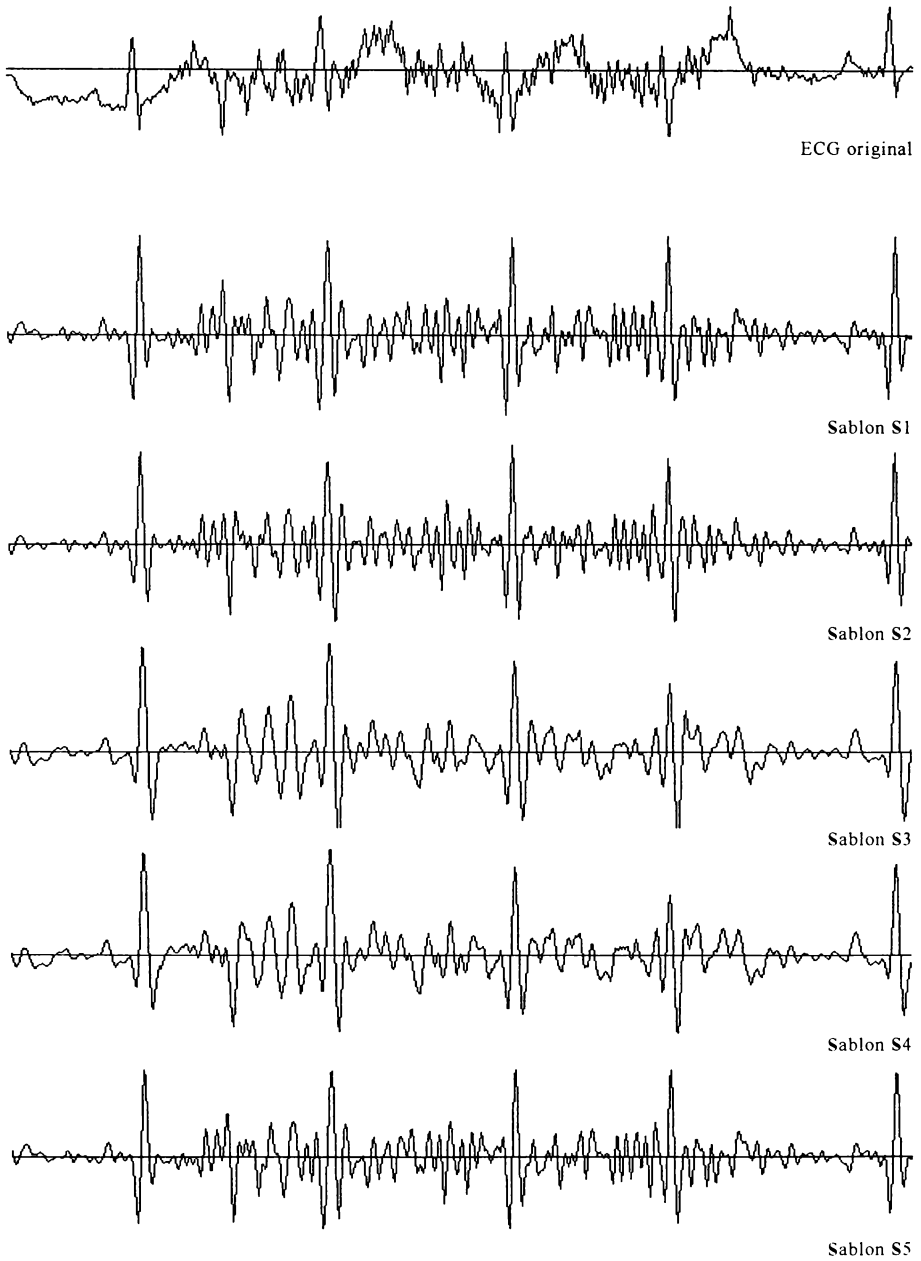


fig.3. 24 Efectul procesării prin corelare cu sabloane + FTS

Concluzii:

- Rezultatele aplicării transformării prezentate în acest capitol sunt deosebit de încurajatoare în a le aplica la o ulterioară identificare QRS, datorită eficacității în a evidenția aceste complexe.
- Pe înregistrarea prezentată în fig. 3.24 (S4.dat din baza proprie de date) înregistrare cu importante artefacte de mișcare, se remarcă o comportare remarcabilă pentru toate aceste șabloane, cu o notă în plus pentru înregistrările procesate cu S2 și S5.
- Față de rezultatele prezentate pentru o înregistrare "dificilă" cum este cea din fig. 3.24, testele făcute pe înregistrări cu derivate lente ale liniei izoelectrice, sau cu zgomot de 50 Hz sunt net superioare.
- Înregistrările din fig. 3.24 sunt făcute la aceeași scară, atât pentru ECG original cât și pentru procesări. Sunt evidențiate evenimentele QRS de o astfel de manieră, încât aceste trasee să poată fi utilizate ulterior la identificarea complexelor QRS, cu ajutorul unui detector cu prag.
- Este bine înlăturată deriva liniei izoelectrice chiar și la variații de scurtă durată, comparativă cu a complexelor QRS.
- Pentru analiza proprietăților în domeniul frecvență ale filtrărilor cu funcții șablon, am ales pentru analiză funcțiile S1 și S5, ale căror caracteristici de frecvență sunt reprezentate în fig 3.25 și 3.26

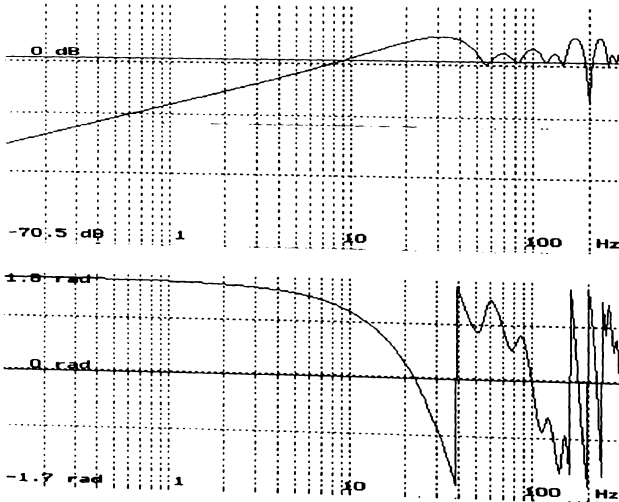


fig. 3.25 Caracteristica de frecvență a filtrului S1.

- Din punct de vedere spectral ambele procesari reprezintă filtre trece bandă, având frecvența centrală de 30 Hz pentru șablonul S1 și de aprox 10 Hz pentru șablonul S5

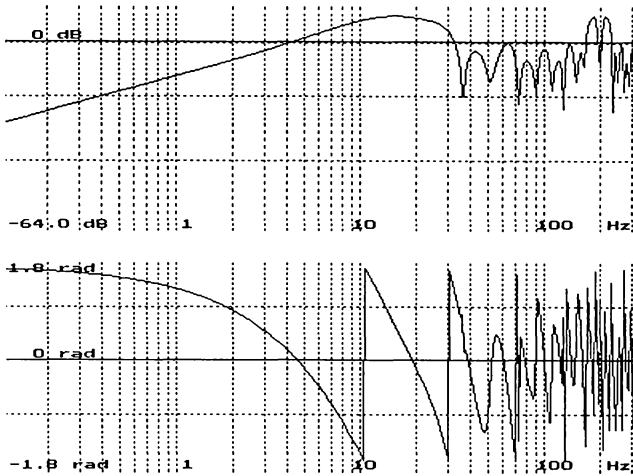


fig. 3.26. Caracteristica de frecvență a filtrului S5.

- Amplificarea la mijlocul benzii este de 11,2 pentru S1 și de 10,4 pentru S5.
- S1 amplifică și frecvențe mai mari decât 40 Hz, pe când S5 nu le amplifică.
- Referitor la caracteristica de fază, se observă că pe intervalul frecvențelor utile au loc variații foarte importante ale fazei, însă în cazul procesării făcute doar cu scopul accentuării complexului QRS, aceasta nu deranjează.
- Se constată că ambele filtre amplifică și semnale având frecvența mai mare decât 100 Hz, ceea ce obligă la limitarea spectrului de frecvență al semnalului la 100 Hz, ținând cont că frecvența de eșantionare este de 200Hz.



fig. 3.27. Șablonul S5 aplicat unui semnal cu derivă lentă a liniei de bază.

- Așa cum am arătat, S5 este cel mai performant dintre detectoarele prezentate până acum iar ca și concluzie se poate afirma faptul că o procesare cu scopul evidențierii complexului QRS este cu atât mai performantă cu cât caracteristica sa de frecvență este mai apropiată de spectrul complexului QRS.

Rezultatele aplicării transformărilor prezentate în acest capitol sunt deosebit de încurajatoare în a le aplica pentru identificarea complexului QRS. Pentru a releva acest lucru am ales procesarea cu șablonul S5 pe care am aplicat-o unui set de semnale, reprezentate împreună cu rezultatele procesărilor în fig. 3.27 ÷ 3.31.

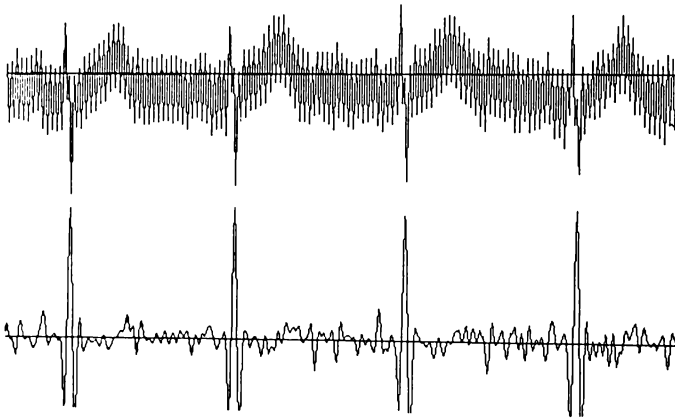


fig. 3.28. Șablonul S5 aplicat unui semnal cu zgomot de 50Hz (10,4 dB)

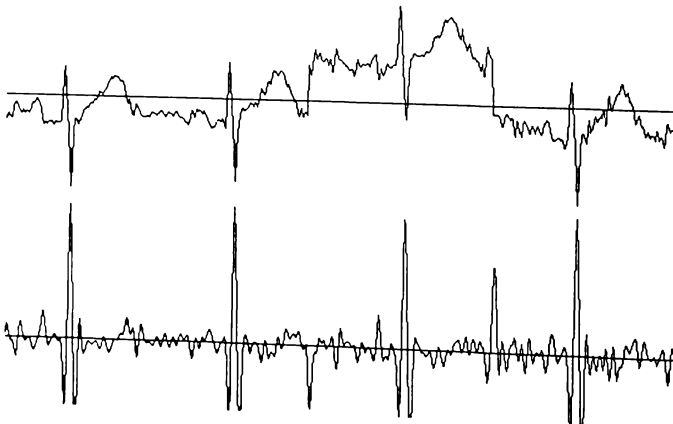


fig. 3.29. Șablonul S5 aplicat unui semnal cu derivă bruscă a liniei de bază.(6,9 dB)

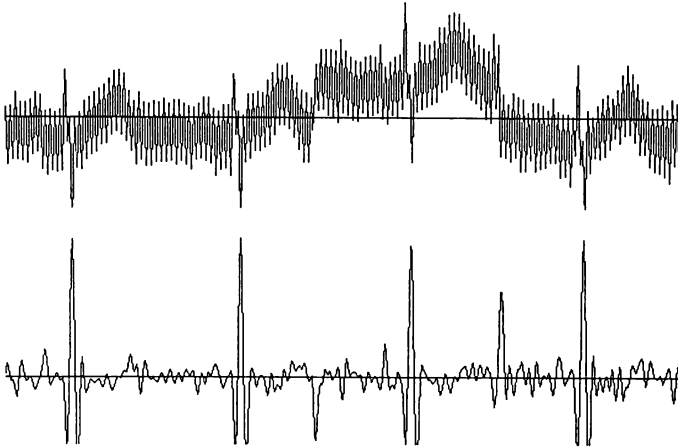


fig. 3.30. Șablonul S5 aplicat unui semnal cu derivă bruscă a liniei de bază (6,9 dB), plus zgomot de 50Hz (10,4 dB).

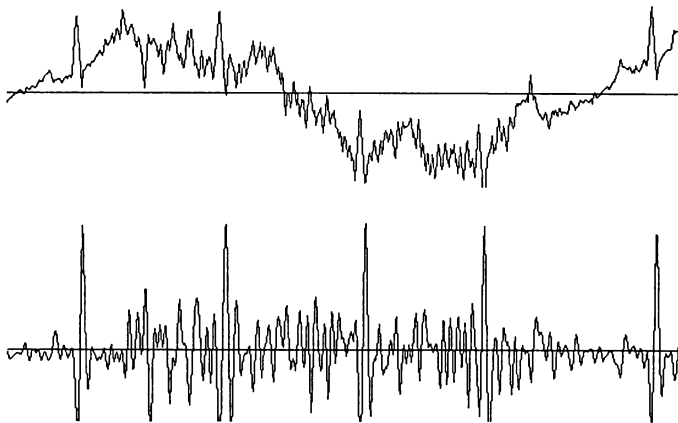


fig 3.31. Șablonul S5 aplicat unui semnal cu fibrilații, afectat de respirație (joasă frecv, 1 Hz; 4,4 dB)

În urma acestor cercetări a apărut necesitatea evaluării eficienței acestor procesări în vederea stabilirii unor "clasamente" care să constituie criteriul de alegere a lor în lanțul de procesare în timp real al ECG în proba de efort. Această problemă, finalizată prin cuantificarea rezultatelor procesărilor este prezentată în finalul paragrafului 3.3 pentru toți algoritmi cercetați.

3.3.6 Detector cu șablon creat din semnalul ECG ("autocorelație")

Rezultatele încurajatoare obținute cu șabloanele prezentate anterior arată că rezultatele sunt cu atât mai bune cu cât asemănarea șablonului cu forma reală a unui complex QRS este mai mare. De aceea pentru proba de efort este posibilă derularea unei secvențe de program care în primele momente ale probei, chiar înainte de începerea înregistrărilor, să aibă drept obiectiv determinarea unui șablon prelevat chiar din semnalul ECG provenit de la pacient.

Acest șablon va fi utilizat apoi pe toată durata probei de efort, după o prealabilă "cosmetizare".

Soluțiile prezentate până în prezent converg cu soluția clasică de determinare a periodicității unui semnal cu ajutorul funcției de autocorelație. Problema de fond este în cazul procesării în timp real aceea că nu dispunem de întreg semnalul niciodată și va trebui făcută corelarea semnalului ce "curge", cu porțiuni anterioare din acesta. Fiindcă ritmul QRS poate fi considerat cvasialeator chiar și în lipsa aritmiilor și în mod evident pe durata probei de efort, este indicată utilizarea unei porțiuni care să cuprindă un singur complex QRS.

Soluția propusă în continuare realizează simultan "autocorelarea" semnalului ECG și în același timp înlătură efectul derivei liniei izoelectrice.

Pentru prelevarea porțiunii de traseu care să reprezinte chiar un complex se pornește cu unul din șabloanele propuse anterior după care în mod automat, după câteva cicluri cardiace se reține o secvență având complexul QRS în interior (nu neaparat în centru). Numărul de eșantioane reținut depinde de frecvența de eșantionare. Pentru 200 Hz am ajuns la concluzia că o fereastră $q = 12 \div 30$ de eșantioane reprezintă soluția optimă ca și compromis între volumul de calcule, întârzierea produsă de lungimea șablonului la identificarea efectivă a complexului QRS și calitatea rezultatului. Secvența prelevată este indicat să fie reținută nu din semnalul ECG original și deci posibil zgomotos ci după o prealabilă filtrare.

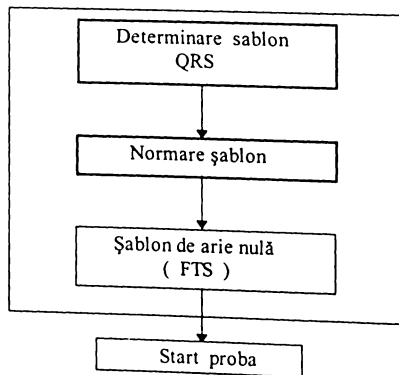


fig. 3.32. Construcția șablonului din semnalul ECG

Normarea are ca scop aducerea valorilor șablonului între niște limite totdeauna aceleași și se face cu scopul de a facilita procesarea prin evitarea depășirilor atunci când se lucrează în virgulă fixă. Astfel vom transla valorile secvenței reținute printr-o funcție Φ conform fig.3.33. Procesul de normare în această etapă prezintă avantajul că la prelevarea secvenței nu mai trebuie ținut cont de deriva liniei izoelectrice.

Funcția Φ poate fi una liniară sau de preferat o funcție neliniară care să distorsioneze complexul QRS în sensul accentuării valorilor mari din secvență, păstrând evident numărul de elemente al domeniului $[0 ; 1]$ același cu al domeniului inițial și anume q elemente.

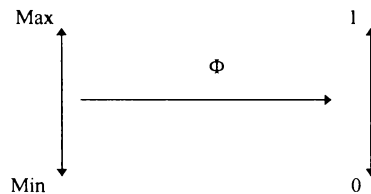


fig. 3.33 Translatarea domeniilor prin funcția de normare

Următorul pas este sugerat de rezultatele favorabile obținute cu șabloane având simultan și funcția de filtru trece sus, de aceea se va transforma șablonul normat într-unul având "aria" nulă. Modul de obținere a acestuia va fi prin adăugarea unei funcții corectoare Ψ , funcție având o asemenea formă, încât rezultatului să-i poată fi impusă ușor condiția de "arie" nulă din rel 3.34

$$S p_i = S n_i + \Psi_i \tag{3.36}$$

Notațiile vor fi următoarele:

S_i secvența prelevată conținând complexul QRS;

$S n_i$ secvența normată;

$S p_i$ secvența procesată;

q numărul de eșantioane al secvențelor.

Impunând condiția 3.34 secvenței procesate vom avea:

$$\sum_{i=1}^q (S n_i + \Psi_i) = 0 \tag{3.37}$$

$$\sum_{i=1}^q \Psi_i = - \sum_{i=1}^q S n_i \tag{3.38}$$

Pentru funcția Ψ_i am luat în considerare mai multe forme posibile, toate având ca și condiție impusă aceea de a nu distorsiona forma șablonului normat în

porțiunea sa centrală. Determinarea exactă a funcției Ψ_i în condiția 3.38 se face ușor dacă ea se definește astfel

$$\Psi_i = k \Omega_i \quad (3.39)$$

unde definim pe Ω_i drept o funcție "unitară" pentru scopul urmărit, iar prin factorul k vom stabili amplitudinea funcției Ψ_i

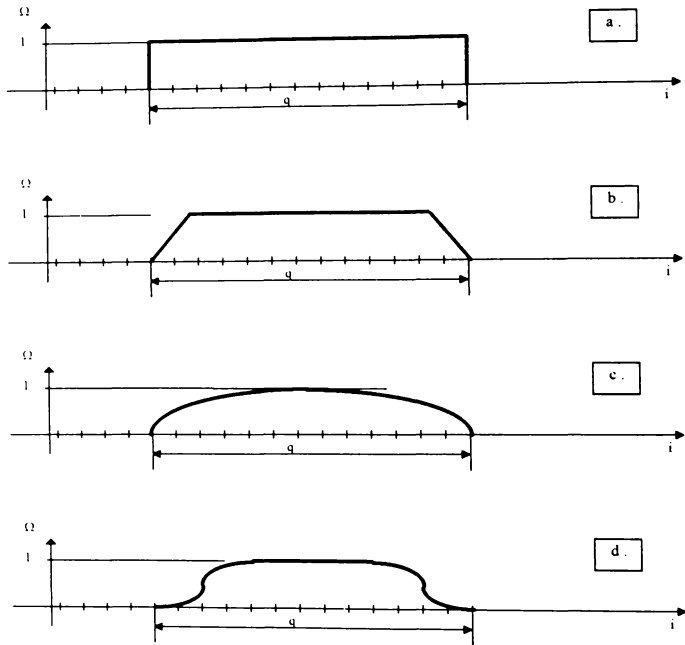


fig. 3.34. Tipuri de bază pentru funcțiile unitare

Funcția Ω_i se construiește analitic sau tabelar după modelele prezentate în fig. 3.34, o sursă de inspirație în acest sens putând fi funcțiile de "windowing" din analiza spectrală: Hamming, Hanning. În aceste condiții, factorul k se determină pentru fiecare secvență reținută ca urmare a relațiilor 3.38 și 3.39:

$$k = \frac{\sum_{i=1}^q S n_i}{\sum_{i=1}^q \Omega_i} \quad (3.40)$$

iar apoi

$$S p_i = S n_i + k \Omega_i \quad (3.41)$$

Deci șablonul procesat pe care-l vom utiliza la identificarea QRS se va construi din șablonul prelevat cu ajutorul relațiilor 3.33 și 3.34.

Pentru varianta "c" din fig. 3.34 a funcțiilor unitare propuse se poate construi o întreagă clasă de funcții având forma:

$$\Omega_i = \alpha_i \sin \frac{\pi i}{q} \quad (3.42)$$

în care α_i este o funcție pentru obținerea unei aplatizări mai mari a funcției sinus la mijlocul intervalului q . O altă formă utilizată a fost:

$$\Omega_i = (1 - e^{-\lambda i}) (1 - e^{-\lambda (q-i)}) \quad (3.43)$$

unde λ este o constantă ce accentuează aplatizarea curbei în zona centrală.

Șablonul "d" din fig. 3.34 are derivate nule la capetele domeniului de definiție și a fost gândit pentru a reduce mai mult efectele eșantioanelor depărtate de complexul QRS. Pentru o formă analitică aceasta se poate exprima ușor pe porțiuni. Oricum diferențe semnificative apar doar la folosirea unui număr mare de eșantioane.

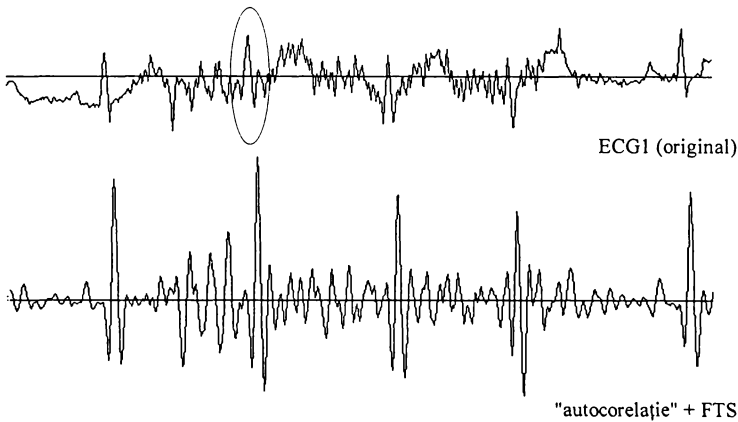


fig.3.35. Rezultatul corelării cu șablon din semnal, pentru ECG1

Rezultatele soluției prezentate sunt ilustrate cu două exemple în fig. 3. 35 și fig. 3.36. Se constată astfel că pentru semnale afectate de zgomot de frecvență ridicată, rezultatele nu sunt spectaculoase și mai mult, depind de gradul de alterare cu zgomot a porțiunii selectate, iar lungimea selectată este hotărâtoare pentru calitatea procesării. În exemplul dat este încercuit complexul QRS selectat și porțiunea aferentă lui. O soluție de evitare a acestei situații este ca în cazul probei de efort, prelevarea șablonului să se facă imediat înainte de start.

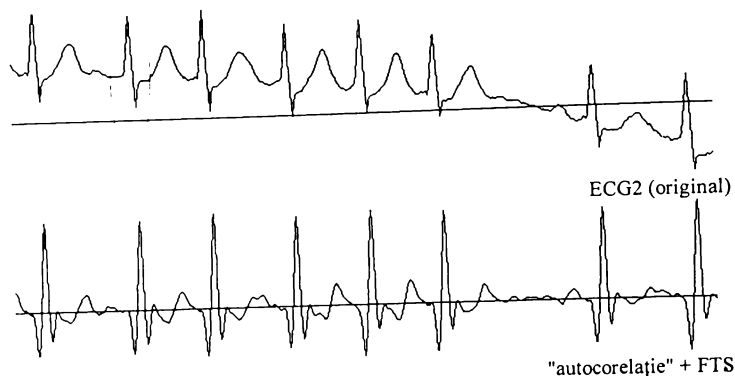


fig.3.36. Rezultatul corelării cu șablon din semnal pentru ECG2.

Semnalele fără astfel de zgomote, chiar cu derivate însemnate ale izoelectrice, se pretează la o procesare de această factură, dovada fiind semnalul ECG2 din fig.3.36

3.3.7 Detector QRS utilizând transformarea "lungime"

Acest detector reprezintă o transformare neliniară și rezultă implementarea relației recursive 3.44 pentru cazul particular $n = 1$ (nu canalelor de înregistrare) iar $q = 8$ (lungimea ferestrei pe care se calculează lungimea.). Pentru implementarea relației recursive este cea indicată datorită reducerii volumului de calcule.

$$L(8,j) = L(8,j-1) + (X_j - X_{j-1})^2 - (X_{j-9} - X_{j-8})^2 \quad (3.44)$$

Concluziile folosirii acestui tip de procesare sunt susținute de rezultate prezentate în fig. 3.37 și 3.38.

Se observă din fig. 3.37 că acest gen de procesare este indicat atunci când semnalul este filtrat de zgomotele de frecvență ridicată, fiindcă se obțin foarte marcante corespunzător complexelor QRS, chiar în prezența unor zgomote importante ale liniei de bază.

Dacă însă semnalul este afectat de un zgomot de înaltă frecvență datorat artefactelor de mișcare sau de contact al electrozilor sau chiar fibrilație, rezultatele nu recomandă algoritmul pentru procesare în vederea evidențierii QRS.

În concluzie, utilizarea lui pentru monitorizarea probei de efort nu este indicată. Aceleași concluzii sunt valabile și pentru transformarea "energie".

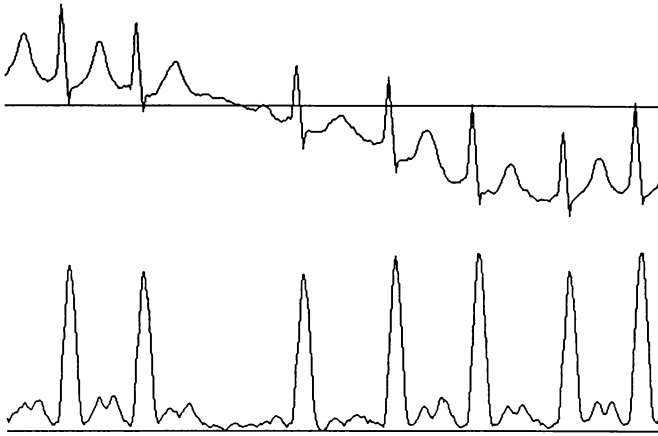


fig. 3.37. Semnal cu derivă lentă a liniei de bază procesat cu transformarea "lungime"

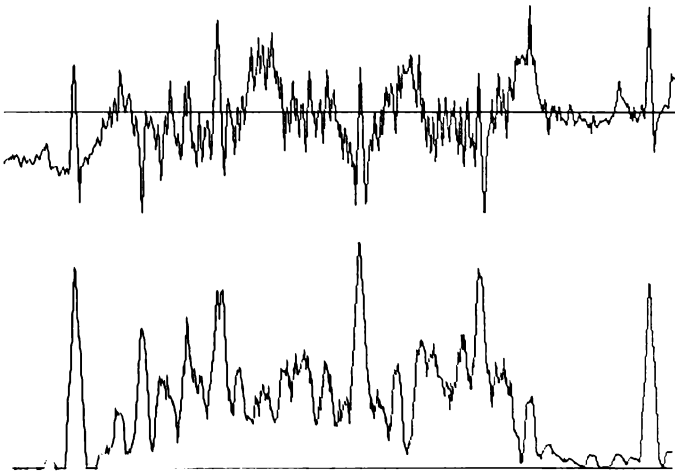


fig. 3.38. Semnal cu zgomot de înaltă frecvență, procesat cu transformarea "lungime"

3.3.8 Identificarea complexului QRS

După partea de preprocesare prezentată anterior, având scopul de a accentua complexul QRS, urmează partea de identificare a acestuia folosind metoda de comparare cu prag adaptativ. Funcția de adaptare are menirea de a face detectorul imun la eventualele fluctuații de amplitudine ale semnalului și acționează astfel:

- se pornește în analiză cu o valoare inițială pentru prag;
- se reține o fracțiune ($0,65 \div 0,75$) din valoarea ultimului maxim obținut;
- valoarea pragului scade cu o pantă constantă în timp, începând din momentul ultimei detecții valide. Această pantă poate fi dependentă chiar de valoarea maximumului detectat;
- validarea propriu-zisă a complexului QRS poate fi condiționată și de alte criterii care să funcționeze în paralel cu această detecție de bază.

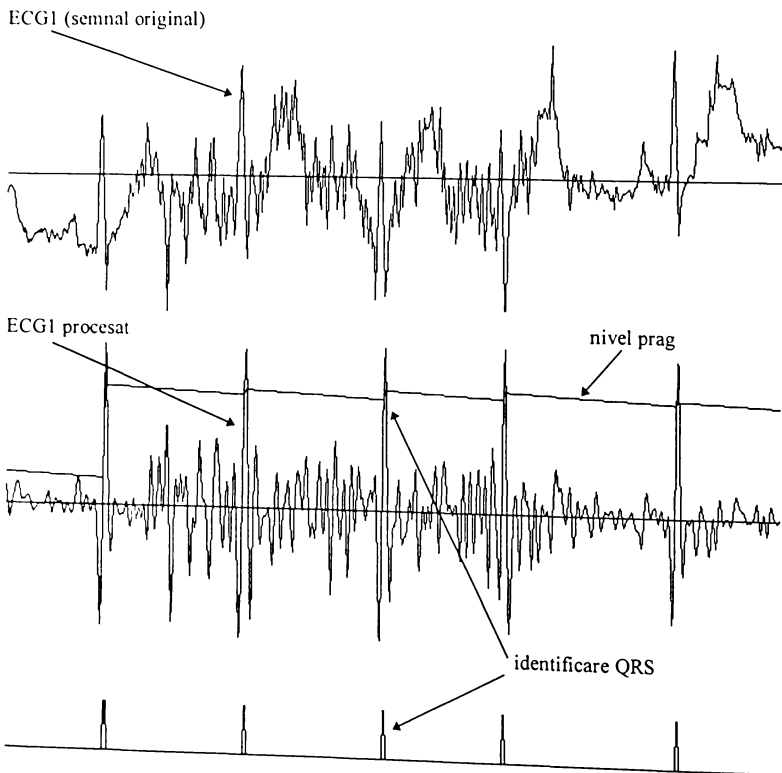


fig3.39. Identificarea complexului QRS cu prag adaptativ

Față de un prag fix, această lege are avantajul că nu produce falsuri negative în ipoteza atenuării semnalului ECG. Riscul de a produce falsuri pozitive este scăzut, prin alegerea unei pante de descreștere mici pentru prag și chiar în ipoteza unei astfel de eventualități, calitatea de adaptare va readuce pragul la valori normale.

Implementarea algoritmului de identificare s-a făcut în meniul numit "Ident_QRS". Realizarea grafică a acestui meniu permite studiul aplicării pragului ales asupra semnalului ECG preprocesat și mai ales permite alegerea optimă a nivelului pragului pentru a avea o cât mai bună identificare QRS.

În fig. 3.39 este prezentat ecranul având reprezentate în prima fereastră semnalul ECG original (ECG1 din baza noastră de date) iar în a doua fereastră semnalul ECG preprocesat și valoarea adaptivă a pragului. În urma validării detecțiilor, este prezentat și semnalul de identificare QRS.

3.3.9. Detector QRS neliniar "AMPLITUDINE - DURATĂ"

Fie $X_{(n)}$ o secvență de eșantionare din înregistrarea ECG. Evenimentul apariției unui extrem local este detectat aplicând criteriile următoare (Policec A., Mișu P.I., 1996): se definește ca maxim local eșantionul care satisface relația (3.45) iar ca minim local eșantionul care satisface relația (3.46)

$$X_{(n-1)} \leq X_{(n)} > X_{(n+1)} \quad (3.45)$$

$$X_{(n-1)} > X_{(n)} \leq X_{(n+1)} \quad (3.46)$$

Plasarea semnului de inegalitate doar într-un sens în cele două relații are ca scop stabilirea de reguli simple de decizie în cazul în care în secvența de date ECG apar paliere orizontale fără ca între ele să se afle și minime sau maxime "evidente". Astfel vor putea apare succesiuni de mai multe minime (sau maxime) unul după altul caz în care de fiecare dată se va considera ca minim (sau maxim) primul din acea succesiune.

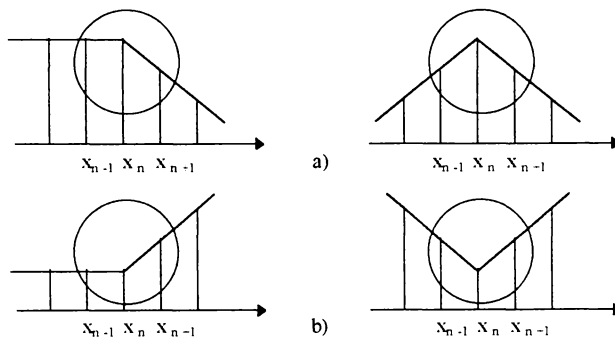


Fig. 3.40 . Definierea unui extrem local cu o fereastră cu trei puncte.

Acest mod de definire a extremelor locale, ilustrat în fig. 3.40 s-a dovedit eficient în cazul înregistrărilor ECG deoarece nu există șansa scăpării de extreme în cazul în care acestea (undele Q, R, S de exemplu) ar fi concretizate în punctele lor extreme prin două eșantioane de aceeași valoare.

Se consideră complexul QRS ca fiind compus din 3 extreme fundamentale (min, max, min) iar intervalele QR sau RS, reprezintă părți "abrupte" aflate între aceste extreme. Vom introduce în continuare două criterii fundamentale de decizie pentru a determina prezența QRS:

a. **Amplitudine** : diferența de amplitudine între un maxim local și primul minim local consecutiv al unei subsecvențe descrescătoare trebuie să fie mai mare decât un prag specificat. Criteriul se formulează similar pentru situația în care semnalul este crescător

b. **Durată** : intervalul de timp dintre punctele de maxim local și cel de minim local consecutiv, care satisface criteriul de amplitudine și este situat într-un interval prescris. Raportul dintre amplitudinea și durata care satisfac criteriile menționate, dă o estimare a pantei înregistrării ECG care poate fi folosită ca și criteriu de recunoaștere QRS. Algoritmul pentru determinarea prezenței "vârfurilor și a văilor" în semnal poate fi imaginat ca o fereastră glisantă cu trei puncte. Aceasta are la bază analiza diferențelor de amplitudine și a intervalului de timp între maxime și minime consecutive ceea ce permite detecția complexului QRS după care intervalele R-R sunt ușor de determinat. Criteriile amintite sunt ușor de aplicat când avem de a face cu semnal liber de zgomot. Totuși nu acesta este cazul tipic pentru o înregistrare ECG provenită de la o probă de efort. În funcție de zgomotul cu care este contaminat semnalul, putem avea două situații distincte.

În primul caz dacă tendința principală a semnalului (scăderea sau creșterea) este datorată artefactelor de joasă frecvență, aplicarea criteriilor de amplitudine și durată, care sunt măsurate local și nu față de linia de bază vor fi satisfăcătoare pentru maximele și minimele locale care aparțin QRS, așa cum este prezentat în fig. 3.41.

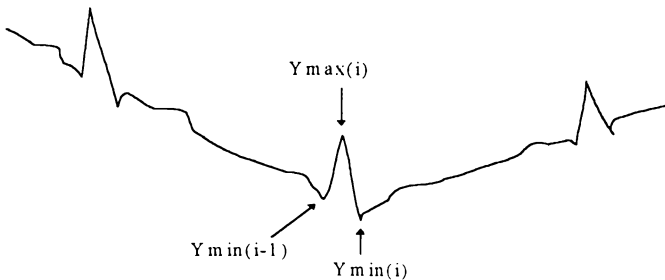


fig. 3.41. Semnal ECG afectat de zgomot de joasă frecvență

În al doilea caz, dacă înregistrarea ECG este afectată de zgomot ce produce extreme locale pe durata pantelor complexului QRS, aplicarea criteriilor va fi insuficientă și se impune completarea algoritmului cu un criteriu în plus pe care l-am denumit "amplitudine cumulativă". O amplitudine cumulativă poate fi definită după câteva extreme locale găsite, care nu satisfac criteriul de amplificarea și durată.

Se observă că dacă semnalul are o tendință generală de descreștere se va ajunge la un minimum extrem (minimorum) $Y_{\min(K)}$, de unde cea mai importantă tendință este inversată. Secvența de amplitudine cumulativă este definită astfel:

$$A_{(K)} = Y_{\max(K)} - Y_{\min(K)} \quad (\text{pt } K=0) \quad (3.47)$$

$$A_{(K)} = A_{(K-1)} + Y_{\min(K-1)} - Y_{\min(K)} \quad (\text{ptr. } K \neq 0) \quad (3.48)$$

Valoarea lui $A_{(K)}$ este cumulată până este satisfăcut criteriul "amplitudine-durată", făcând posibilă detecția QRS înecat în zgomot de înaltă frecvență.

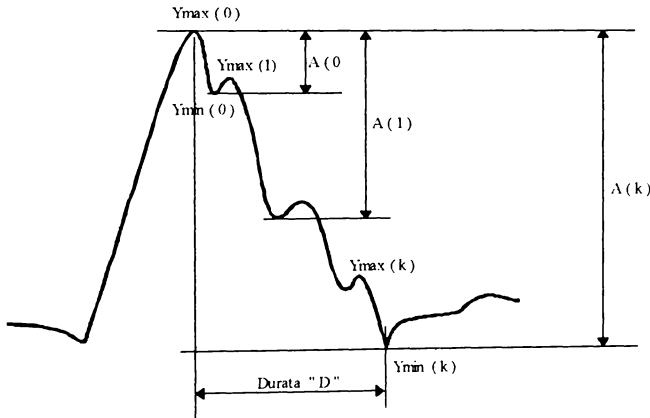


fig. 3.42. Semnal ECG afectat de zgomot de înaltă frecvență

Prezumția că aici $Y_{\max(0)}$ și $Y_{\min(K)}$ sunt extreme, ca în exemplul din fig 3.42 și pot reprezenta undele R și Q, impune ca aceste porțiuni să fie analizate de-a lungul întregului complex QRS. În scopul determinării schimbării de tendință a semnalului, următoarea testare trebuie făcută de fiecare dată:

$$A_{(K)} < A_{(K-1)} \quad (3.49)$$

De aceea fiecare maxim și minim local trebuie să aibă o valoare mai mică ca unul anterior. Criteriul "amplitudine cumulativă", constă numai în adunări, scăderi și comparări simple, fapt ce sporește mult viteza algoritmului. În analiza menționată am presupus că ne aflăm pe o pantă cu tendința generală de descreștere.

Toate criteriile menționate rămân valabile dacă secvența X_n este complementară (crește).

În actuala implementare este recomandabil a avea diferite seturi de praguri de amplitudine și durată pentru pantele pozitive și negative. Aceasta deoarece în QRS patologice, duratele a două pante pot diferi. În algoritmul implementat detecția QRS este confirmată numai după identificarea celor două pante consecutive (negativă și pozitivă) ale complexului.

În plus, în algoritmul de detecție, pragurile criteriului amplitudine-durăată sunt adaptate continuu la schimbările de amplitudine și durată ale QRS anterioare. Am efectuat mai multe seturi de verificări utilizând ca principiu de adaptare a pragurilor medierea ultimelor valori detectate. Astfel, pentru durată compromisul între timp de calcul și eficiență arată că este necesară medierea ultimelor 12 valori pe când la amplitudine sunt suficiente 8.

Am constatat că numărul falsurilor negative scade dacă este introdusă în plus următoarea regulă: valorile medii se vor modifica numai dacă noua valoare diferă față de medie cu o valoare prestabilită un număr mai mare de perioade (trei). După fiecare detecție urmează o perioadă de 100 ms în care nu poate apare un nou complex QRS și de aceea algoritmul de detecție este "blocat".

Semnalul de intrare pentru aplicația avută în vedere este un semnal monocanal ECG obținut de la un amplificator de biopotențial cu o bandă de trecere de $0,1 \div 40$ Hz. Semnalul ECG este eșantionat la fiecare 5 ms (200 Hz) și digitizat prin intermediul unui convertor analog numeric pe 8 biți. Aceste elemente ale procesului de achiziție a semnalului ECG sunt apreciate în literatura de specialitate ca fiind suficiente pentru analiza aritmiilor.

Algoritmul prelucrează fiecare complex prin rularea detectorului de pantă ascendent sau descendent. Orice detecție, găsind prima pantă QRS validă, în termeni de amplitudine și durată, setează un flag și oprește altă detecție de pantă. În detecția de pante negative, fiecare extrem local este testat pentru a verifica dacă el satisface criteriul "amplitudine-durată". Dacă acesta nu este satisfăcut, atunci este calculată amplitudinea și durata cumulată, pentru criteriul "amplitudine-cumulativă".



fig. 3.43. Detectorul în prezența semnalului afectat de zgomot de joasă frecvență

În eventualitatea că testul de monotonie este urmat de amplitudinea cumulată, starea flagurilor va fi resetată și este începută o nouă căutare. Oricare detecție QRS validă, conform criteriului amplitudine-durăată sau amplitudine cumulativă, setează un flag pentru un interval de timp de 100 ms (durata maximă a

unui complex QRS). Intervalul dintre două validări constituie măsura intervalului RR.

Evaluarea performanțelor algoritmului din punct de vedere al calității identificărilor QRS s-a făcut "off line" prin studiul comportării detectorului proiectat asupra unui set de semnale ECG aflate în baza noastră de date. Această bază conține semnale provenind de la cord sănătos sau bolnav, semnale afectate în diferite grade de artefacte sau zgomote.

Detectorul se comportă foarte bine în prezența zgomotelor de joasă frecvență, neavând nici o detecție fals pozitivă (FP) sau fals negativă (FN), ca în fig.3.43, unde a fost testat semnalul ECG1, iar în figură este reprezentată o secvență reprezentativă.

În plus, s-a realizat un modul de program care permite crearea de noi "semnale ECG" realizate prin însumarea la înregistrarea ECG originală a altor semnale. Astfel se poate însuma un semnal sinusoidal de foarte joasă frecvență (1Hz) pentru a simula deriva liniei de bază sau un semnal aleator în banda de frecvență 0,1÷40 Hz ("zgomot alb") pentru a studia comportarea detectorului la diverse situații probabile. Aceste semnale au fost generate matematic și ele pot fi adăugate celui original cu diverse ponderi, realizând în acest fel un raport semnal / zgomot dorit, ca în fig 3.44. Acest lucru se face printr-un program special scris care este ușor accesibil din cadrul meniului general.

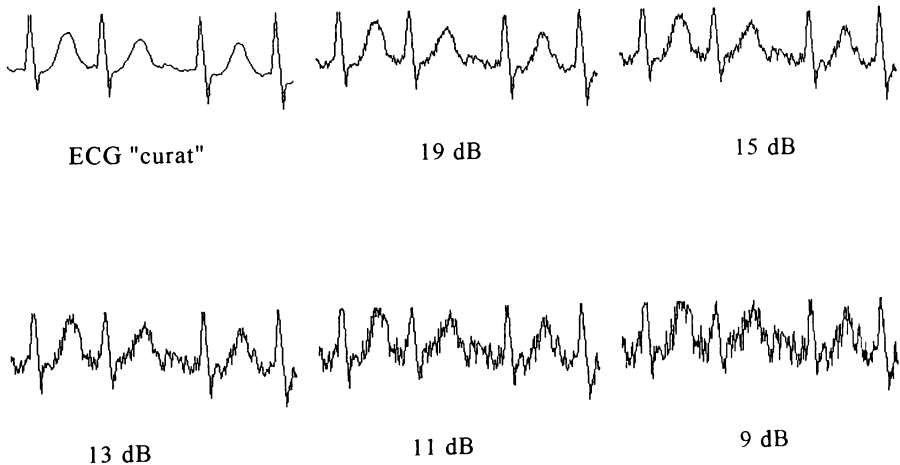


fig. 3.44 Exemple de ECG acoperite de zgomot de înaltă frecvență

Deși este dificil de apreciat cantitativ eficacitatea unui astfel de algoritm, datorită în primul rând diversității extrem de mari a surselor de semnal și a celor de zgomot, iată totuși câteva concluzii deduse în urma încercărilor și sintetizate în parte în tabelul 3.3, unde FP reprezintă detecțiile fals pozitive, FN detecțiile fals negative, iar TP detecțiile totale pozitive.

În domeniul adăunerii zgomotelor conținând frecvențe ridicate, primele detecții false au apărut la un raport semnal / zgomot (SNR) de aproximativ 9 dB (amplitudinea semnalului de zgomot adăugat reprezentând 0,354 din cea a semnalului ECG considerat "curat").

În condițiile în care banda de frecvență a semnalului este cea standard (0,1Hz ÷ 40 Hz) se admite ipoteza că pe durata unui complex QRS este extrem de puțin probabil să existe variații abrupte având același ordin de mărime ca și unda R și deci orice tendință locală de schimbare a pantei este datorată contaminării cu zgomot de înaltă frecvență.

SNR	SNR[dB]	FP%	FN%	TP%
0,112	19	0	0,3	99,7
0,177	15	0	0,4	99,6
0,223	13	0	0,7	99,3
0,281	11	0	3,4	96,6
0,354	9	0,3	7,5	92,5
0,446	7	3,7	13,5	86,5

Tabelul 3.3.

La simulări ale unui salt bruscat și mare al liniei izoelectrice (de 100% din nivelul unde R), adăugat semnalului ECG, algoritmul nu răspunde eficient. Nu vor putea fi detectate complexele QRS ce urmează unui astfel de salt, principiul de mediere a ultimilor 8 complexe nepermițând adaptarea la astfel de salturi. Fiindcă în realitate nu apar astfel de situații și ținând cont de cele prezentate în tabelul 3.3, algoritmul se recomandă totuși ca suficient de performant și în primul rând ca utilizabil în procesările în timp real.

3.3.10. Detector neliniar QRS, cu reaxare după înfășurătoarea inferioară

Ideea realizării acestui tip de detector a rezultat din analiza semnalelor procesate liniar anterior, din care se vede că evenimentul căutat (complexul QRS) generează valori mari în modul și în domeniul numerelor negative, fapt care nu este exploatat de principiul utilizării unui prag pozitiv. Ideea utilizării imediate a unor procesări neliniare ca: ridicări la pătrat sau funcția modul, nu sporesc valoarea globală a procesării, deoarece semnalele negative vor produce acum maxime la alte momente de timp decât cele datorate complexelor QRS.

Principiul metodei propuse, ilustrat în fig.3.45, se bazează pe reaxarea întregului traseu al semnalului procesat față de "înfășurătoarea" inferioară a semnalului. Cel mai simplu mod de a defini înfășurătoarea este din segmente de dreaptă între punctele considerate ca "minime locale semnificative".

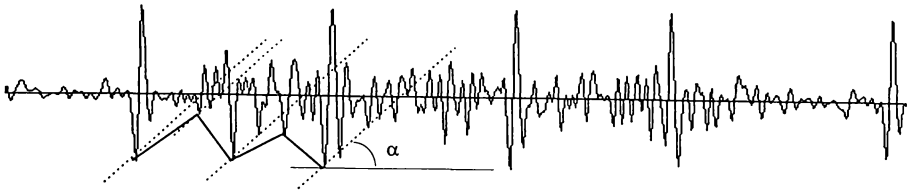


fig. 3.45. Construcția înfășurătoarei inferioare.

Definirea punctelor de "minim local semnificativ" se face impunând în plus acestora, condiția ca valoarea tangentei față de ultimul minim reținut să fie inferioară unei valori minime impuse. Urmează apoi reaxarea traseului între ultimele două minime față de segmentul de dreaptă ce le unește.

$$\text{tg } \alpha < \text{tg_max} \quad (3.50)$$

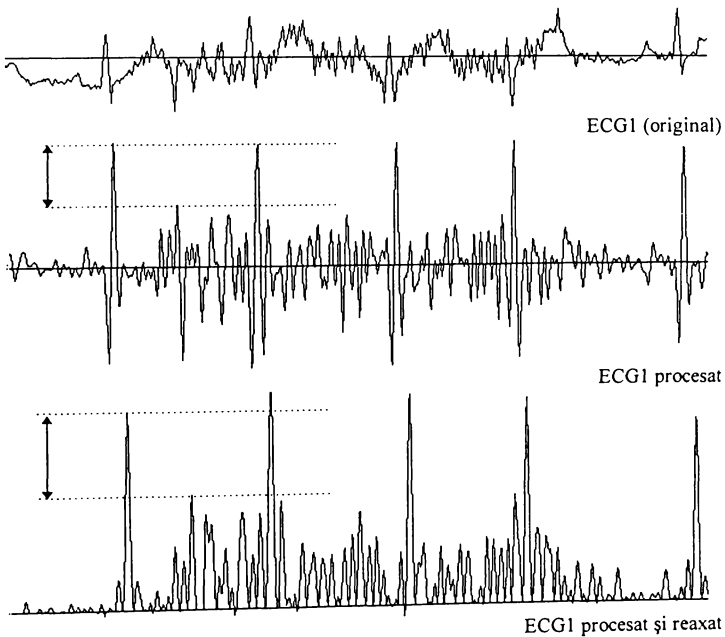


fig. 3.46 Rezultatul reaxării pentru ECG1 procesat.

Metoda are avantajul unui algoritm extrem de rapid, care rezolvă procesarea într-o singură trecere iar ca timp maxim de întârziere nu depășește $10 T_e$, în toate testele făcute pentru un t_{g_max} judicios ales.

În funcție de valoarea maximă a pantei impuse, rezultatul procesării poate fi diferit. Cu cât panta minimă impusă este mai mare, variațiile traseului dintre evenimentele marcante (QRS), sunt mai mici, însă există riscul de a pierde și unele complexe mai mici ca amplitudine.

Se constată la o analiză simplă a fig. 3.46, accentuarea decalajului dintre cel mai puțin semnificativ eveniment căutat, și cel mai semnificativ eveniment necăutat, situație favorabilă ulterioarei detecții cu prag. În acest caz, în care semnalul este afectat atât de derivă linie de bază, cât mai ales de zgomot de frecvență ridicată, raportul celor două evenimente nu crește semnificativ (de la 1,80 pentru semnalul procesat, la 1,82 pentru cel reaxat).

Pentru semnale fără componente de frecvență ridicată, rezultatul procesării este spectaculos, ilustrat de fig. 3.47 unde raportul ajunge de la 4,62 pentru semnalul ECG2 procesat la 5,60 pentru semnalul reaxat.

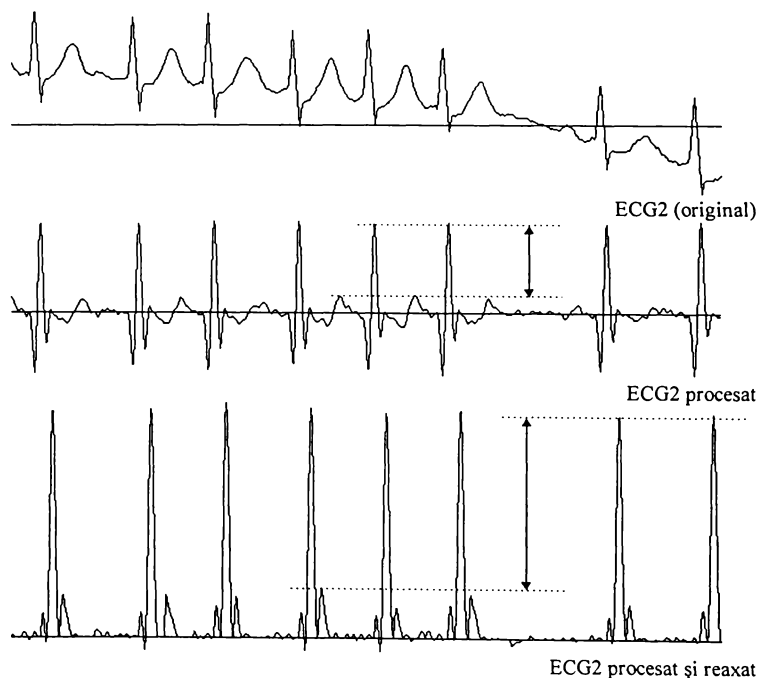


fig. 3.47. Rezultatul reaxării pentru ECG2 procesat.

3.3.11. Evaluarea performanțelor algoritmilor de preprocesare pentru detecția QRS

Așa cum am arătat la 2.3.3, evaluarea performanțelor detectoarelor QRS, în sensul cuantizării lor se face utilizând mărimile FP și FN, respectiv numărul de detecții fals pozitive și detecții fals negative. Acestea reprezintă însă un rezultat global, atât al preprocesării semnalului original, cât și al detectorului propriu-zis.

În general, detecția propriu-zisă presupune compararea semnalului procesat cu un prag fix sau adaptiv, de rezultatul comparării depinzând decizia de existență sau inexistență la un moment dat a complexului căutat. Decizia poate avea un grad sporit de siguranță, dacă în paralel cu procesarea despre care discutăm se fac și altele care să aducă noi criterii pentru evaluarea gradului de certitudine al existenței complexului căutat. Orice noi procesări am aduce în discuție alături de cea despre care discutăm acum, problema căutării unui eveniment marcant într-un semnal se face la un moment dat tot prin compararea cu un prag pentru luarea deciziei.

Acest fapt a dus la ideea încercării de cuantizare a rezultatelor algoritmului de preprocesare, pentru o corectă evaluare și clasificare a performanțelor acestora, în vederea atingerii obiectivului major, acela de identificare cât mai sigură a evenimentului căutat. În realitate, de rezultatul preprocesării depinde esențial calitatea în ansamblu a detectorului, mai exact de gradul în care algoritmul de procesare reușește să transforme semnalul inițial, accentuând evenimentul care ne interesează, respectiv complexul QRS.

Se impune deci găsirea unei modalități de a cuantifica gradul în care un semnal prezintă niște forme dominante, majore și unice în contextul "spectacolului" din jurul lor, iar formularea problemei este: cum se poate cuantifica gradul de unicitate ("singularitate") al unei forme predominante într-un semnal ?

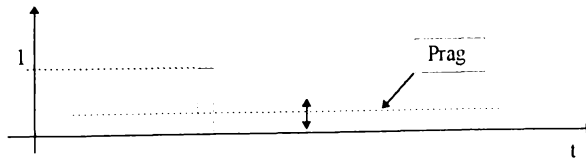


fig. 3.48. Identificarea cu prag.

Spre exemplu, impulsul singular de amplitudine unitară din fig. 3.48, poate fi identificat cu un prag având orice valoare în intervalul $0..1$, și asta cu o certitudine maximă. Are un caracter de unicitate absolut, iar pe o scară de valori ar trebui notat cu un scor maxim posibil.

În schimb în multitudinea de forme rezultate după preprocesare, pragul nu mai poate lua orice valoare, așa cum se vede în exemplul din fig. 3.49

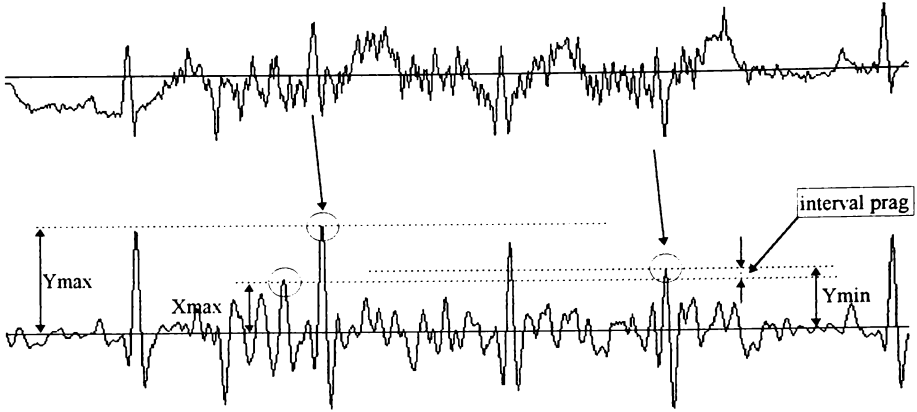


fig3.49. Alegerea pragului.

Pentru alegerea pragului, vom defini următoarele valori determinate pentru întreaga lungime a traseului analizat:

- Ymax valoarea maximă corespunzătoare evenimentului căutat;
- Ymin valoarea minimă corespunzătoare evenimentului căutat;
- Xmax valoarea maximă în afara evenimentului căutat;

Valoarea pragului pentru care nu vom avea nici o detecție fals pozitivă sau fals negativă trebuie să fie cuprinsă între Xmax și Ymin. Cu cât acest interval este mai larg cu atât mai sigură va fi identificarea evenimentului căutat.

Pentru evaluarea algoritmilor de detecție, am definit următorii coeficienți

$$K_{PRAG} = \frac{Ymin}{Xmax} \tag{3.51}$$

$$K_{STAB} = \frac{Ymax}{Ymin} \tag{3.52}$$

Primul coeficient este cel mai important și arată dinamica pe care o poate avea pragul iar al doilea dinamica valorilor maxime, element util pentru proiectarea detectoarelor cu prag adaptiv. Este evident că valoarea acestor coeficienți depinde de semnalul procesat. În cercetările efectuate, pentru evidențierea calităților unui algoritm, l-am aplicat de fiecare dată semnalelor "dificile", (zgomote, fibrilație artefacte), din baza noastră de date. Pentru exemplul ales (Ymax = 202,87; Ymin = 118,62; Xmax = 106,75; valori determinate cu ajutorul programului realizat în acest scop), rezultă: K_{PRAG} = 1,11; K_{STAB} = 1,71. Se poate aprecia că algoritmul prezentat nu este unul performant.

În tabelul 3.4, sunt sintetizate rezultatele cercetărilor efectuate asupra eficacității mai multor algoritmi, în termenii amintiți mai sus. Totodată mai sunt

prezentate și alte elemente care să faciliteze opțiunea pentru unul din ei, în eventualitatea utilizării lor pentru monitorizarea în timp real:

- lungimea algoritmului (ordinul filtrului), reprezintă ordinul cel mai mare al eșantionului luat în calcul de algoritm, față de ordinul eșantionului curent, și arată întârzierea pe care o introduce algoritmul, exprimată în intervale T_e ;
- tipul coeficienților, servește la alegerea reprezentării în program a tipurilor de variabile, cu efect asupra timpului total de calcul cerut de algoritm. Prin 2^K se înțeleg coeficienți "puteri întregi ale lui 2";
- numărul coeficienților nuli, oferă împreună cu lungimea algoritmului, informații referitoare la numărul de înmulțiri și de adunări pe care le implică algoritmul.

	Denumire algoritm (filtru)	Lun- gime	Tip coef.	Coef. nuli	K_{STAB}	K_{PRAG}
1	Derivata ord. 2, patru puncte pe ECG1. (fig. 3.13)	8 T_e	2^K	5	1,20	1,93
2	Derivata ord. 3, patru puncte pe ECG1. (fig. 3.13)	16 T_e	2^K	12	1,51	1,51
3	Șablon S1 pe ECG1. (fig. 3.24)	8 T_e	2^K	2	1,03	1,62
4	Șablon S2 pe ECG1. (fig. 3.24)	8 T_e	2^K	2	1,22	1,84
5	Șablon S4 pe ECG1. (fig. 3.24)	13 T_e	2^K	2	1,71	1,11
6	Șablon S5 pe ECG1. (fig. 3.24)	16 T_e	2^K	2	1,05	1,80
7	Șablon S5 pe ECG1, reaxat. (fig. 3.46)	18 T_e	reali	-	1,13	1,81
8	Șablon S5 pe ECG2.	18 T_e	reali	-	1,08	4,62
9	Șablon S5 pe ECG2, reaxat. (fig. 3.47)	18 T_e	reali	-	1,05	5,60
10	Corelație cu șablon din semnal, pentru ECG1. (fig. 3.35)	16 T_e	reali	-	1,54	1,27
11	Corelație cu șablon din semnal pentru ECG2.	16 T_e	reali	-	1,06	3,84

Tabelul 3.4. Sinteza evaluării algoritmilor.

În urma experimentelor efectuate, putem aprecia ca fiind foarte performanți algoritmi care au $K_{PRAG} > 2$; $K_{STAB} < 1,1$, respectiv cei care asigură un decalaj cel puțin dublu dintre cel mai puțin semnificativ eveniment căutat și cel mai semnificativ eveniment în afara celui căutat și care oferă o stabilitate relativă a evenimentului căutat cu variații de cel mult 10 %.

Pentru implementarea în timp real se remarcă algoritmul 6 (Șablonul S5) din tabelul 3.4 care pe lângă performanțele bune ($K_{PRAG} = 1,80$; $K_{STAB} = 1,05$), se poate implementa performant ca și timp de calcul, având coeficienți puteri întregi ale lui 2.

Pentru a completa spectrul de procesări ECG, se impune comparația rezultatelor oferite de procesările specifice tratate anterior cu procesările liniare clasice cunoscute din tehnica filtrărilor numerice. Plecăm de la observația că cea mai eficientă procesare se obține cu șablonul S5 a cărei caracteristică de frecvență reprezintă un filtru trece bandă având frecvența de trecere inferioară $F_i = 8,7$ Hz și frecvența de trecere superioară $F_s = 26,2$ Hz, determinate prin măsurători în program, pe caracteristica de frecvență din fig.3.26.

Pentru a le putea compara între ele am construit mai întâi filtre liniare trece bandă având același gabarit al caracteristicii de frecvență, prin metoda Fourier. Construcția caracteristicii de frecvență se face în programul QRS:EXE, în acest caz impunând cele două frecvențe F_i și F_s , precum și ordinul filtrului, respectiv numărul de eșantioane asupra cărora se aplică algoritmul. Crescând ordinul filtrului crește panta caracteristicii în zona de tăiere, caracteristica apropiindu-se mai mult de cea ideală a filtrului trece bandă. Rezultatul filtrării este prezentat în fig. 3.50, unde alături de ECG1 (original), este prezentată evoluția filtrării trece bandă de tip Fourier, în funcție de ordinul filtrului.

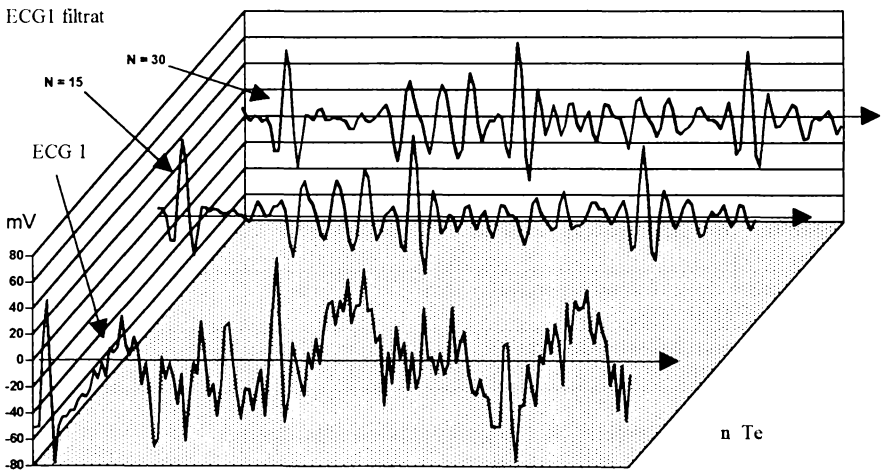


fig 3.50. FTB Fourier de ordin variabil

Rezultate acceptabile în sensul accentuării complexelor QRS într-un context "dificil", reprezentat de semnalul ECG1, se obțin pentru $N > 15$ iar creșterea ordinului până la $N = 30$ nu mai aduce câștiguri majore, așa cum se observă din tabelul 3.5, în care sunt prezentate rezultatele filtrării de bandă impusă, în funcție de ordinul filtrului. Cele mai bune rezultate ($K_{PRAG} = 1,25$; $K_{STAB} = 1,32$) sunt neperformante comparativ cu rezultatele corelării cu S5.

	Denumire algoritm (filtru)	Lun-gime	Tip coef.	Coef. nuli	K_{STAB}	K_{PRAG}
1	FTB - Fourier, (8,7 Hz - 26,2 Hz)	15 Te	reali	-	1,45	1,11
2	FTB - Fourier, (8,7 Hz - 26,2 Hz)	20 Te	reali	-	1,45	1,12
3	FTB - Fourier, (8,7 Hz - 26,2 Hz)	25 Te	reali	-	1,46	1,14
4	FTB - Fourier, (8,7 Hz - 26,2 Hz)	30 Te	reali	-	1,41	1,17
5	FTB - Fourier, (8,7 Hz - 26,2 Hz)	35 Te	reali	-	1,32	1,25
6	Șablon S5 pe ECG1.(fig.3.26)	16 Te	2^K	2	1,05	1,80

Tabelul 3.5. Comparația FTB-Fourier cu Șablon S5

A doua categorie de teste a fost efectuată cu ajutorul filtrelor sintetizate prin metoda Fourier-Discretă (Mihu P.I., Toma E., Breazu M., 1995). Programul permite construcția caracteristicii de frecvență dorite prin puncte, cu ajutorul cărora se încearcă realizarea unui gabarit cât mai apropiat de cel de referință, în cazul nostru de cel al "filtrului S5". Rezultatele obținute nu depășesc în performanță pe cele de la filtrul Fourier. Metoda devine însă utilă dacă se impune realizarea unui filtru de tip "notch", pentru rejectarea unei frecvențe nedorite din semnalul ECG.

3.3.12. Concluzii cu privire la utilizarea algoritmilor pentru detecția QRS

Toți algoritmi prezentați în paragraful 3.3 au fost experimentați "off line", pe semnale din baza de date "Laboratoire de Traitement du Signal et de l'Image" (Rennes, Franța), sau pe baza de date creată în acest scop în cadrul Secției Cardiologie a Spitalului Județean Sibiu și apoi incluși în programul de monitorizare a probei de efort, pentru a fi exploatați în timp real.

O primă concluzie, este faptul că metodele de procesare specifice pentru detecția QRS, bazate în special pe exploatarea proprietăților funcției de corelație cu secvențe "șablon" sintetizate pe baza morfologiei complexului QRS, se dovedesc net mai avantajoase decât metodele de filtrare liniară clasică, atât din punct de vedere al performanțelor cât și din punct de vedere al timpului de calcul necesar.

O altă concluzie importantă este necesitatea selectării algoritmului de procesare (dintr-o "bibliotecă" de algoritmi), potrivit specificității semnalului oferit de pacientul testat în proba de efort. Această concluzie va fi materializată în programul de monitorizare relizat, unde este prezăcută posibilitatea selectării algoritmului optim fiecărui pacient, la începutul testului de efort.

3.4 ALGORITMI DE COMPRESIE PENTRU DATE ECG

Realizarea monitorizării complexe a probei de efort include între obiectivele sale și crearea unei baze de date, fapt ce obligă la utilizarea unor metode de compresie. Din variantele trecute în revistă în subcapitolul 2.3, am optat pentru dezvoltarea unui algoritm reversibil (predicție liniară) și a unuia nereversibil (aproximarea poligonală "evantai"). Argumentele acestor opțiuni precum și soluțiile de optimizare a acestor algoritmi vor fi prezentate în cele ce urmează.

3.4.1 OPTIMIZAREA PREDICȚIEI LINIARE PENTRU SEMNALE ECG

Predicția reprezintă estimarea valorii unui eșantion la un moment dat pe baza eșantioanelor sale din trecut. Deoarece semnalul de intrare variază statistic, semnalul prezis nu este în mod necesar același ca semnalul de intrare, deși va fi foarte apropiat de acesta. Codarea predictivă presupune transmiterea diferenței dintre semnalul de intrare și cel prezis, ceea ce reduce semnificativ volumul de informație transmis.

Obiectivele urmărite în această secțiune sunt :

- prezentarea metodei de calcul al coeficienților predictorului;
- studiul dependenței performanțelor predictorului de parametrii semnalului ECG;
- implementarea unui algoritm pentru eficientizarea predicției în sensul compresiei de date ECG, potrivit scopurilor propuse de contextul general al monitorizării.

Un prim argument în favoarea abordării acestei metode este existența unor referiri bibliografice doar la nivel de principiu în legătură cu utilizarea predicției pentru compresia datelor ECG și cu atât mai puțin la utilizarea metodei Entropiei Maxime. Un alt argument în favoarea utilizării predicției liniare este faptul că în semnale ECG, eșantioanele alăturate sunt puternic corelate, așa cum se vede din fig. 3.51, în care este reprezentată dependența autocorelației semnalului ECG2 din baza proprie de date, de lungimea M a "ferestrei" de corelare, conform relației 2.7.

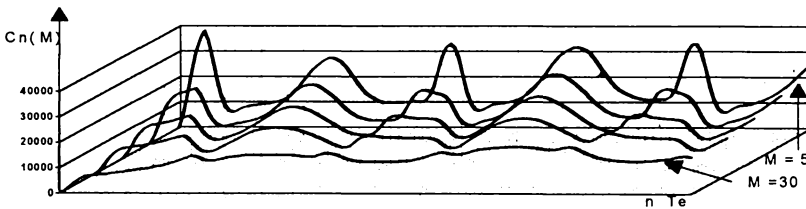


Fig.3.51. Autocorelația semnalului ECG2

În codarea predictivă, rezultatul codării este dat de relația 3.53.

$$\Delta_{(n)} = x_{(n)} - \hat{x}_{(n)} = x_{(n)} - \sum_{i=1}^N a_i x_{(n-i)} \quad (3.53)$$

unde $\hat{x}(n)$ reprezintă rezultatul estimării eşantionului curent utilizând un algoritm liniar pentru ultimele N eşantioane rezultate din procesul de achiziție, iar a_i reprezintă coeficienți de predicție constanți. Schema canonică a unui astfel de predictor este cea din fig. 3.52.a, iar schema bloc de principiu a codării / decodării predictive, conform cu relația 3.53, este cea din fig. 3.52.b.

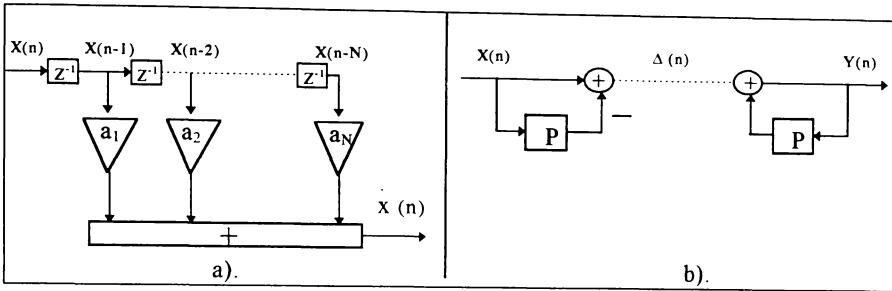


fig. 3.52 Codare / decodare cu predicție liniară.

O a doua variantă a aceluiași principiu de codare o constituie interpolarea. Acum estimarea eşantionului prezent se face cu o combinație liniară de forma (3.54) numită "interpolare în jurul punctului", în care intră atât N eşantioane anterioare cât și M eşantioane posteroare lui.

$$\hat{x}_n = \sum_{i=-M}^N a_i x_{n-i} \tag{3.54}$$

Pentru ca procesarea să fie posibilă (cauzală) trebuie realizată o deplasare în timp a tuturor eşantioanelor, valoarea eşantionului estimat și implicit a rezultatului codării fiind întârziată cu un număr de perioade de eşantionare egal cu numărul eşantioanelor "viitoare" (M) ce intră în formula interpolatorului, conform rel. 3.55.

$$\hat{x}_n = \sum_{i=-M}^N a_i x_{n-M-i} \tag{3.55}$$

Codarea în cele două variante are avantajul că dinamica semnalului diferență transmis este mult mai mică decât dinamica semnalului original ceea ce duce automat la posibilitatea de a coda ieșirea cu un număr mai mic de biți la același raport semnal / zgomot. În alți termeni de procesare numerică, deși nu ne interesează pentru moment proprietățile spectrale ale codorului / decodorului, se observă că la codare este implementat un filtru nerecursiv iar la decodare un filtru recursiv, cu comportare de integrator.

Schema bloc din fig 3.52.b prezintă un dezavantaj major legat de propagarea zgomotului de cuantizare la ieșirea decodorului. Ca alternativă pentru evitarea propagării sale este prezentată schema bloc din fig. 3.53, în care și predictorul de la codare este inclus într-o buclă de reacție.

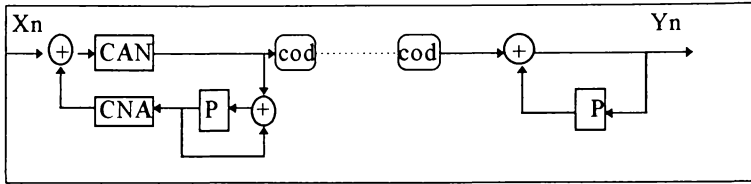


fig. 3.53. Predictor în bucla de reacție

În această schemă de principiu plasarea cuantizatorului Q după comparatorul analogic, obligă la utilizarea unui CNA în bucla de reacție. Soluția este numai aparent mai costisitoare fiindcă altfel, prin plasarea cuantizatorului înainte de comparator, acesta ar fi trebuit să fie realizat pe un număr de biți mai mare. Schema este deosebit de avantajoasă atunci când cuantizarea se face pe un singur bit.

Demonstrația avantajului legat de propagarea zgomotului de cuantizare se poate face simplu prin aplicarea unei transformări integrale sistemelor liniare de la emisie pentru fiecare din cele două variante ale codorului (fig.3.54). Aici Q(s) reprezintă contribuția de zgomot a CAN.

$$Dx(s) = [1-P(s)] \cdot X(s) + Q(s) \tag{3.56}$$

$$Dx(s) = [1-P(s)] \cdot [X(s) + Q(s)] \tag{3.57}$$

Ținând cont de funcția de transfer la recepție, semnalul de ieșire este

$$Y(s) = \frac{Dx(s)}{1 - P(s)} \tag{3.58}$$

Rezultă imediat din 3.56, 3.57 și 3.58 că la prima variantă, zgomotul de cuantizare apare integrat în timp la ieșirea recepției, pe când la cea de a doua (cu predictor în bucla de reacție) acesta apare în aceeași cantitate ca și după emisie.

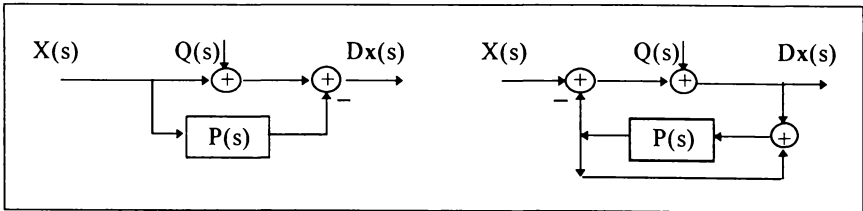


fig. 3.54 Efectul zgomotului de cuantizare

De aceea fără excepție codorul predictiv se implementează într-o configurație cu buclă de reacție ca în fig 3.53, soluție care înlătură inconvenientul legat de propagarea zgomotului de cuantizare.

3.4.1.1 Calculul coeficienților predictorului prin Metoda Entropiei Maxime.

Coeficienții predictorului (a_i) se determină din condiția ca energia semnalului de eroare să fie minimă. Esența metodei derivă din aparatul matematic ce pune în evidență relația existentă între spectrul de putere estimat prin Metoda Entropiei Maxime (MEM) și coeficienții predictorului.

Pentru început trebuie făcute câteva referiri la estimarea spectrului de putere al unui semnal ECG prin Metoda Entropiei Maxime. Transformata Fourier nu este singura cale de a estima spectrul de putere al unui semnal și nici măcar nu este cea mai eficientă, limitele ei fiind legate în ceea ce ne privește acum, și de faptul că estimarea spectrului de putere prin transformarea Fourier se regăsește numai în intervalul de frecvență Nyquist ($-F_e/2 \div +F_e/2$). Considerând un interval de N eșantioane ale unei funcții x_n , eșantionată cu o perioadă T_e , transformata FFT a acestui interval este:

$$C_k = \sum_{j=0}^{N-1} x_j e^{2\pi i j \frac{k}{N}} \quad ; \quad k = 0, 1, \dots, N-1 \quad (3.59)$$

Estimarea spectrului de putere se face pentru $N/2 + 1$ valori, astfel:

$$P(f_k) = \frac{1}{N^2} (|C_k|^2 + |C_{N-k}|^2) \quad ; \quad k = 1, 2, \dots, (N/2 - 1) \quad (3.60)$$

unde f_k este definită numai pentru frecvențe mai mari sau egale cu zero

$$f_k = \frac{k}{N T_e} = F_e \frac{k}{N} \quad ; \quad k = 0, 1, \dots, \frac{N}{2} \quad (3.61)$$

Obținem transformata z prin substituția ce transformă planul complex f în planul z :

$$z = e^{2\pi i f T_e} \quad (3.62)$$

Exceptând factorul de normalizare, relația (3.53) devine acum

$$P(f) = \left| \sum_{k=-N/2}^{N/2-1} x_k z^k \right|^2 \quad (3.63)$$

Analiza relației (3.63) arată că nu avem de a face cu o determinare exactă a puterii spectrale ci cu o estimare din următoarele două motive:

- în planul z relația (3.63) reprezintă o serie Laurent finită, care oferă numai o aproximație a funcției analitice generale (3.64):

$$P(f) = \left| \sum_{k=-\infty}^{\infty} x_k z^k \right|^2 \quad (3.64)$$

- calculul dat relația (3.64) provine de la un set de N eșantioane ale funcției, și este deci cert că acest set de N eșantioane nu conține informații care să caracterizeze spectrul exact al funcției cercetate. Acest aspect este valabil și în cazul estimării prin metoda FFT.

Estimarea spectrului de putere cu ajutorul relației (3.63) care nu conține poli, ci numai zerouri, reprezintă o "metodă directă". Pentru calculul efectiv al relației (3.63) vom utiliza o aproximare ale cărei avantaje vor fi exploatate mai târziu.

$$P(f) \cong \frac{1}{\left| \sum_{k=-M/2}^{M/2} b_k z^k \right|^2} = \frac{a_0}{\left| 1 + \sum_{k=1}^M a_k z^k \right|^2} \quad (3.65)$$

Ideea este de a aproxima seria finită (3.63) cu o alta ai cărei parametri să se afle la numitor. Pentru început trebuie remarcat că diferența dintre (3.63) și (3.65) nu este doar una de exprimare ci ele implică caracteristici foarte diferite și anume: reprezentarea spectrului de putere cu o funcție având numai poli este adecvată pentru situațiile în care în spectru există salturi mari. Aproximarea (3.65) este cunoscută sub diferite denumiri (Childers, 1978; capII):

- modelul "numai poli";
- metoda entropiei maxime (MEM) ;
- modelul autoregresiv.

Teoretic, coeficienții a_k ai aproximării se pot determina din condiția de convergență a seriei (3.63) către (3.65). Evident, gradul de "apropiere" va depinde de lungimea M a aproximării care rămâne la alegere. Practic însă coeficienții a_k se determină folosind teorema Wiener-Khinchin

$$\frac{a_0}{\left| 1 + \sum_{k=1}^M a_k z^k \right|^2} = \sum_{j=-M}^M \Phi_j z^j \quad (3.66)$$

Unde Φ_j reprezintă valori medii ale corelațiilor pentru diferite decalaje conform relației:

$$\Phi_j = \frac{1}{N+1-j} \sum_{i=0}^{N-j} x_i x_{i+j} \quad (3.67)$$

În relația (3.67) funcția de corelație se calculează pe setul de date x_i având lungimea $N+1$ eșantioane. Relațiile (3.66) și (3.67) formează un sistem care permite determinarea efectivă a celor M necunoscute reprezentate chiar de coeficienții a_1, \dots, a_M . Numărul M se numește "ordin" sau "număr de poli" și teoretic el poate fi ales mai mic decât N . De menționat că dacă la transformarea Fourier se poate face o legătură directă între coeficienții rezultați și frecvențele rezultate prin eșantionarea inervalului Nyquist, la această metodă nu există o astfel de legătură. Estimarea spectrului de putere al semnalului analizat se poate face acum prin implementarea relației (3.65). În fig.3.55 este reprezentat spectrul de

putere al semnalului ECG6, estimat prin MEM pe un set de $N_{date}=100$, în funcție de numărul de coeficienți ($N_{poli} = 2, 4, 6, 8, 20, 30$).

Care este însă legătura dintre estimarea spectrului de putere prin metoda entropiei maxime și predicția liniară? Din prezentarea de principiu făcută rezultă că între valoarea prezisă $\hat{x}_{(n)}$ și valoarea exactă $x_{(n)}$, va exista o diferență:

$$\Delta_n = x_{(n)} - \sum_{i=1}^N a_i x_{(n-i)} \quad (3.68)$$

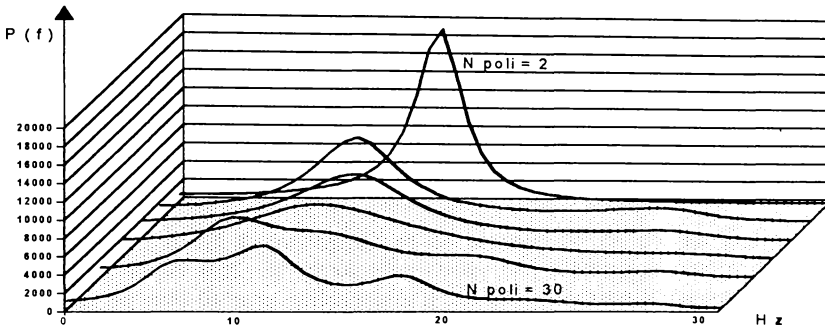


Fig. 3.55. Dependența spectrului de putere de numărul coeficienților (numărul polilor).

Condiția pe care trebuie să o îndeplinească un predictor eficient este ca energia E a semnalului diferență (3.68) să fie minimă.

$$E = \sum_{n=1}^r \left(x_{(n)} - \sum_{i=1}^N a_i x_{(n-i)} \right)^2 \quad (3.69)$$

$$\frac{dE}{da_i} = 0 \quad (3.70)$$

Pentru determinarea coeficienților optimi ai predictorului se rezolvă sistemul de ecuații reprezentat de (3.70). Se arată că se obțin aceiași coeficienți ca și cei determinați prin metoda entropiei maxime. Acest fapt este remarcabil și dovedește că predicția liniară și MEM reprezintă în fond două abordări diferite ale aceleiași probleme:

- MEM caracterizează un semnal dat printr-un număr finit de poli în planul z , cu ajutorul cărora se reprezintă cel mai bine spectrul său de putere
 - Predicția liniară extrapolează semnalul folosindu-se de aceiași poli.
- Fiindcă studiul are ca finalitate implementarea unui lanț de compresie / decompresie a datelor ECG, trebuie făcute referiri și la partea de decompresie. Rezultă din schema 3.53 că

$$y_{(n)} = \Delta_n + \sum_{i=1}^N a_i y_{(n-i)} \tag{3.71}$$

Structura de filtru recursiv (3.71) trebuie analizată din punctul de vedere al stabilității. Pentru aceasta este necesar și suficient ca toate rădăcinile ecuației caracteristice (3.72) să se afle în interiorul cercului unitate

$$z^n - \sum_{i=1}^N a_i z^{n-i} = 0 \tag{3.72}$$

$$|z^k| \leq 1 \quad ; \quad k = 1, 2, \dots, N \tag{3.73}$$

Referitor la implementarea relației (3.71) trebuie făcute precizările:

- Făcând un bilanț global al lanțului compresie / decompresie avem:

$$y_{(n)} = x_{(n)} - \sum_{i=1}^N a_i x_{(n-i)} + \sum_{i=1}^N a_i y_{(n-i)} \tag{3.74}$$

Așa cum rezultă din (3.74), pentru ca reconstituirea să fie fără pierdere de informație trebuie ca primele N-1 valori refăcute să fie copia fidelă a primelor N-1 eşantioane ale semnalului original și anume chiar setul de date pe care s-au determinat coeficienții predictorului. Abia începând de la al N-lea eşantion se va face reconstituirea de semnal conform relației (3.63).

- Respectând condiția amintită mai sus, stabilitatea algoritmului este asigurată.

3.4.1.2 Implementarea metodei. Rezultate

Implementarea are ca obiectiv compresia datelor ECG apărute pe durata monitorizării, pentru a fi stocate ca martore ale evoluției pacientului, în fișiere speciale de date. Problema de fond este stabilirea unui algoritm care să producă cea mai eficientă compresie, pornind de la specificitatea fiecărei înregistrări ECG. Parametrii ce trebuie stabiliți pentru compresie sunt:

- numărul de date ce vor sta la baza calculului coeficienților Ndate;
- poziția setului de Ndate în înregistrarea ECG Ndec;
- numărul coeficienților predictorului Npoli.

De la început trebuie precizat faptul că rezultatul compresiei va fi o înregistrare într-un fișier și care va conține:

- numărul coeficienților: Npoli;
- valorile acestora: $a_1 \div a_{Npoli}$;
- primele Npoli date exacte ale ECG;
- urmează apoi șirul valorilor rezultate în urma procesului de compresie corespunzătoare fiecărui eşantion de date original.

Conform organigramei programului de optimizare a compresiei (fig.3.56), inițial se introduc valorile dorite pentru variabilele Npoli, Ndate, Ndec. Imediat pot fi vizualizate poziția și lungimea setului de date ales în înregistrarea ECG. Alegerea se poate reface și apoi se trece la calculul parametrilor. Criteriul de optim este legat de raportul dintre dinamica globală a semnalului original și a celui transmis.

Atîngerea unei performanțe impuse sau găsirea uneia care nu mai este depășită după un număr de iterații va fi marcată de variabila logică "Optim", care va valida trecerea efectivă la generarea datelor compresate.

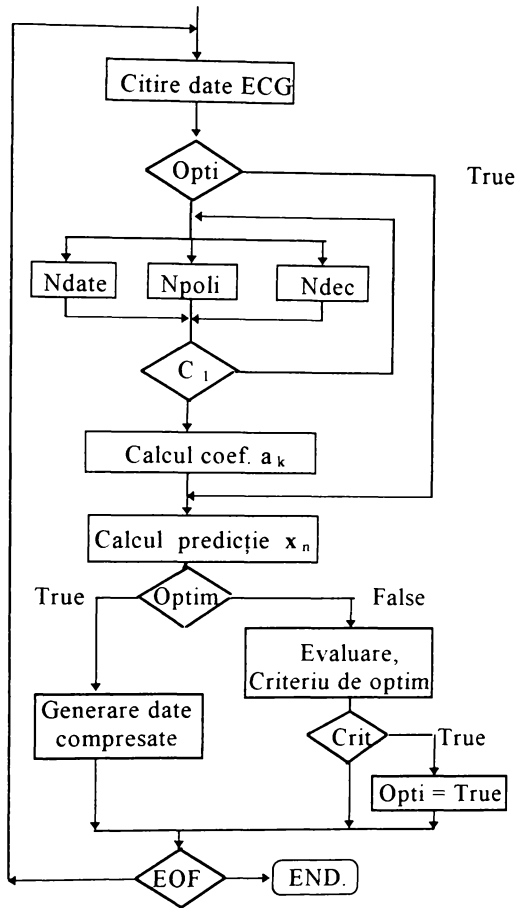


Fig.3.56. Organigrama algoritmului de optimizare a predicției.

Rezultatele analizelor sunt reliefate în diagramele următoare. În fig 3.57 este prezentată evoluția în timp a diferenței dintre semnalul original ECG6 și valoarea semnalului prezis în funcție de numărul coeficienților utilizați pentru predicție și anume, Npoli = 1, 2, 4, 8. Predicția s-a făcut pe un set inițial de lungime Ndate=100 eşantioane. Acest set a fost poziționat prin program la începutul înregistrării ECG prin Ndec = 0. Pe același grafic este reprezentat și semnalul

refăcut după predicție pentru a se putea compara cu amplitudinile semnalelor diferență.



Fig. 3.57 Evoluția semnalului la emisie (ECG6, Ndec = 0)

Pentru a evidenția gradul de dependență de alegerea poziției setului de date pentru calculul coeficienților de predicție, în fig. 3.58 este prezentată evoluția în timp a diferenței dintre semnalul original ECG6 și valoarea semnalului prezis pentru un set de date, decalat față de primul cu un număr de 30 eșantioane (Ndec=30), restul parametrilor rămânând aceiași ca cei din fig. 3.57.

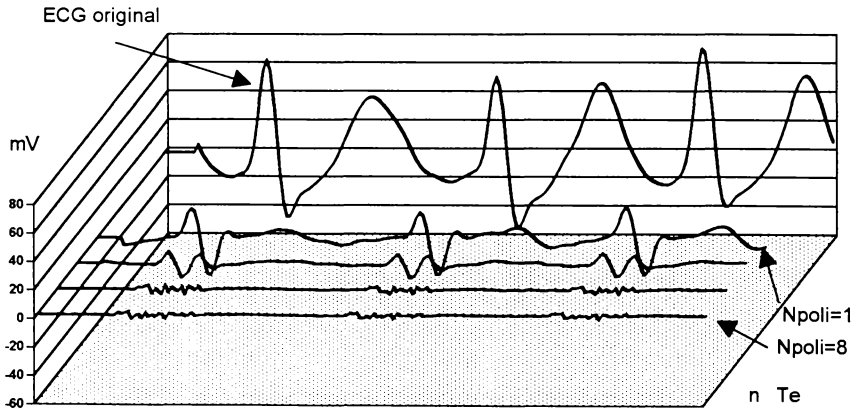


Fig. 3.58. Evoluția semnalului la emisie (ECG6, Ndec = 30)

O sinteză a rezultatelor anterioare, legată în special de contribuția numărului de poli este prezentată în fig. 3.59, de unde se pot trage ușor câteva concluzii. Aici

este prezentată în paralel dinamica unui semnal ECG, alături de a celor rezultate prin compresie, utilizând diferite valori pentru numărul de poli precum și efectul alegerii poziției setului de date în cadrul înregistrării.

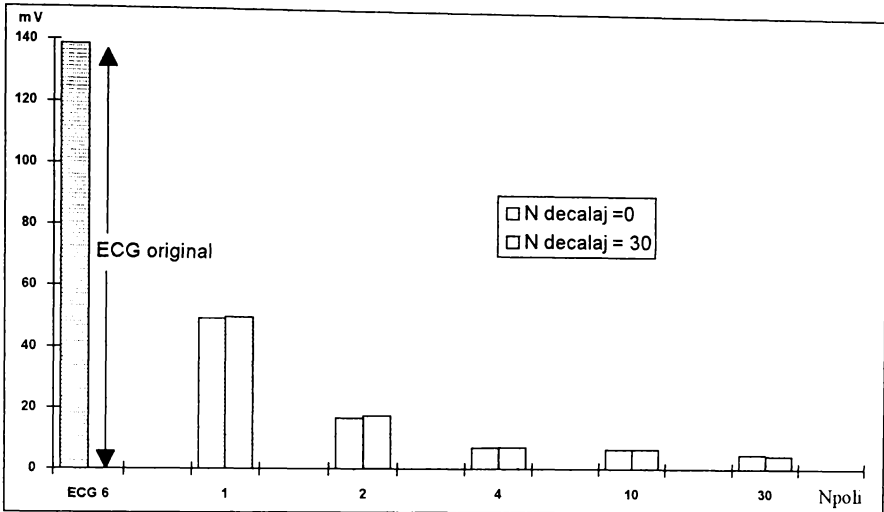


Fig. 3.59. Sinteza studiului de optimizare în funcție de numărul de poli și de poziția setului de date de calcul al coeficienților.

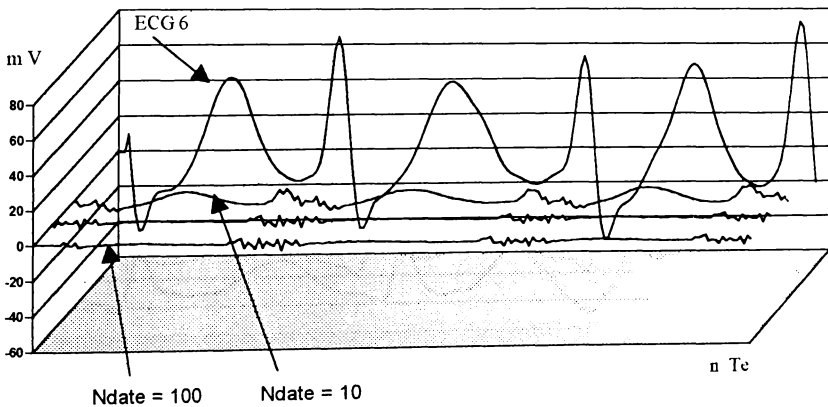


Fig. 3.60. Dependența semnalului emis, de lungimea setului de date folosit pentru calculul coeficienților.

Analiza efectului lungimii setului de date este prezentată sintetic în fig. 3.60. Sunt prezentate comparativ semnalul original (ECG 6) și semnalele compresate pentru valori $N_{date} = 10, 50, 100$, în condițiile $N_{poli} = 4, N_{dec} = 0$. Analiza arată că pentru valori ale lungimii mai mici decât 50 de eșantioane dinamica semnalului diferență crește, dar mai ales devine dependentă de poziția setului de date din înregistrare. Acest lucru nu se mai întâmplă pentru valori care depășesc intervalul unui ciclu cardiac, în cazul nostru începând de la $N_{date}=80$, frecvența de eșantionare fiind 200 Hz. În concluzie, o lungime $N_{date}=100$ asigură o calitate suficientă, creșterea acestui număr nefiind productivă.

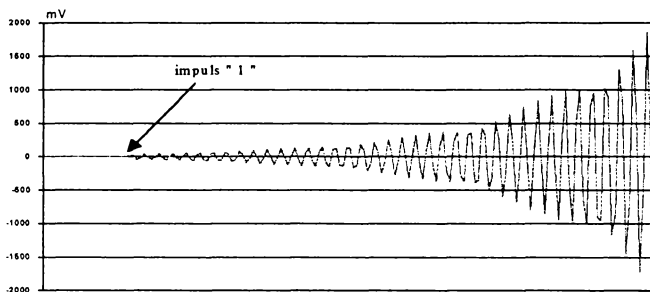


Fig. 3.61. Stabilitatea algoritmului.

O altă problemă pe care am luat-o în considerare a fost aceea a stabilității algoritmului la recepție. Scopul acestei analize este și acela de a stabili influența unor erori de transmisie asupra posibilităților refacerii corecte a semnalului codat.

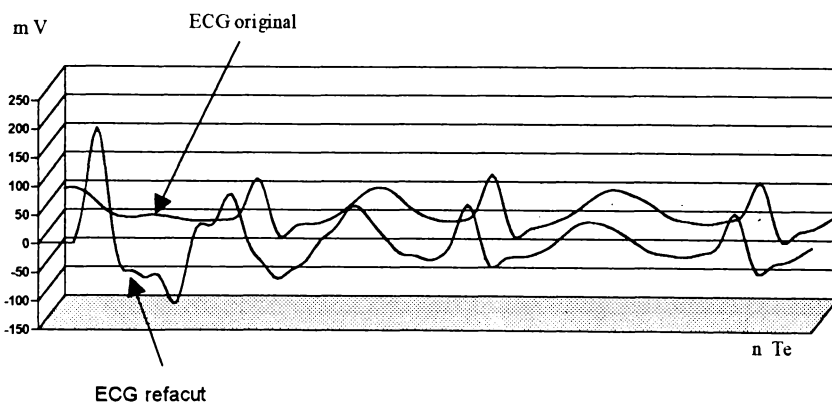


Fig 3.62. Efectul de convergență spre semnalul original

Pentru început s-a verificat stabilitatea algoritmului de refacere a semnalului cu un set de 10 coeficienți determinați pe ECG6 în condițiile $N_{date}=50$, $N_{poli}=10$, $N_{dec}=0$. Verificarea s-a făcut prin trimiterea unui impuls unitar la recepție și "refacerea" sa cu coeficienții rezultați pentru semnalul ECG6. Rezultatul este cel din fig.3.61 și dovedește caracterul de instabilitate al algoritmului pentru acest caz.

Așa cum am arătat anterior însă, sistemul este stabil atunci când se refacă semnalul care a stat la baza calculului coeficienților. Trebuie însă verificat dacă sistemul rămâne stabil în cazul recepționării unor semnale eronate. În exemplul din fig. 3.62 am simulat recepția unui eșantion eronat în setul celor N_{date} trimise inițial la recepție. S-a utilizat același semnal și aceleași condiții de predicție ca pentru exemplul din fig.3.61. Se constată că după câteva cicluri cardiace semnalul refăcut tinde spre cel original, deci astfel de erori nu duc la instabilitate, dimpotrivă, ele sunt repede anulate.

Întrebarea de fond la care trebuie dat un răspuns este cum se realizează efectiv compresia de date? S-a văzut că semnalul ce se transmite, rezultat ca diferență dintre semnalul original și cel prezis, are o dinamică mult mai mică decât cel original, ceea ce duce la ideea reprezentării lui pe un număr mai mic de biți. Dificultatea majoră provine de la faptul că numărul ce se emite, deși are o valoare mică, este un "real" cu un număr nedeterminat de zecimale(!). Aceasta fiindcă algoritmul de calcul al coeficienților prin metoda entropiei maxime duce la obținerea de coeficienți de predicție reali. Mai mult, este indicat calculul acestor coeficienți în dublă precizie atunci când se dorește refacerea fără pierderi a semnalului.

Dacă se codifică o înregistrare ECG disponibilă în format de numere reale, se va transmite tot un număr real, ceea ce presupune același volum de informație ca și pentru transmiterea semnalului original (tot 6 octeți). Dacă înregistrarea ECG este făcută în format de numere întregi (2 octeți) atunci, pentru a refacă semnalul fără pierderi, trebuie transmis un număr real, ceea ce înseamnă o sporire a volumului de informație transmis.

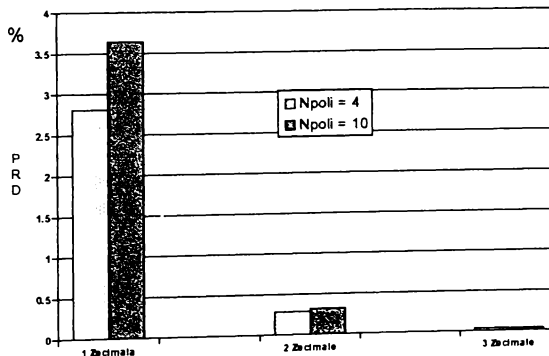


Fig. 3.63. Estimarea erorilor la refacere în cazul trunchierii la emisie

În concluzie, pentru a realiza efectiv o compresie de date trebuie ca numărul ce se va transmite să fie o trunchiere a valorii rezultate prin calcule și apoi reprezentat ca întreg pe un număr corespunzător de biți. Evident, dacă înregistrarea ECG este în format de întregi (2 octeți), trunchierea trebuie făcută până acolo unde cifrele rezultate să poată fi reprezentate pe un număr de biți mai mic decât 16.

Iată deci că, deși predicția liniară se încadrează din punct de vedere al principiului teoretic în categoria metodelor de compresie reversibile, aplicarea ei în practică obligă la trunchierea valorii ce se emite, ceea ce duce automat la apariția de erori la refacerea semnalului, deci metoda devine ireversibilă.

Primul pas al analizei trunchierii trebuie să fie cel referitor la stabilitatea algoritmului, fiindcă trunchierea poate fi privită din acest punct de vedere ca o recepție eronată a semnalului emis. Experimental am constatat pe semnalele ECG1-ECG6 din baza de date că sistemul rămâne stabil chiar dacă trunchierea se face până la nivelul la care mai rămâne doar o singură cifră.

În graficul din fig. 3.63 sunt sintetizate rezultatele analizei efectului trunchierii semnalului la emisie asupra calității semnalului refăcut în condițiile următoare: ECG6, Ndate = 100, Ndec = 0 iar Npoli = 4 și 10. Pentru cuantificarea rezultatelor s-a folosit distanța medie pătratică (PRD) dintre semnalul original și cel refăcut. Rezultatele arată că dacă trunchierea se face la :

- prima zecimală PRD = 2,81 % - 3,61 %
- primele două zecimale PRD = 0,28 %
- primele trei zecimale PRD = 0,03 %

Se constată că în condițiile afișajului pe un monitor PC, cele două semnale sunt observate ca "identice" pentru valori ale trunchierii care conservă minimum 2 zecimale. Ținând cont că rezultatul codării nu depășește unitatea, se vor folosi pentru codare 3 cifre pentru a căror reprezentare sunt suficienți 10 biți.

3.4.1.3. Concluzii

Programul realizat de către autor pentru studiul compresiei semnalului ECG prin predicție liniară este scris în PASCAL și este structurat în două UNIT-uri: **Entrop.pas** și **Predic.pas**. Programul are la bază estimarea spectrului de putere al unui semnal prin Metoda Entropiei Maxime care a servit apoi la obținerea datelor menționate pe parcurs.

În concluzie, se obțin rezultate optime din punctul de vedere al compresiei de date ECG achiziționate pe doi octeți, utilizând Metoda Entropiei Maxime, dacă:

- se utilizează 4 coeficienți;
- setul de date pe care se calculează coeficienții are 100 eșantioane (caz în care poziția acestui set în înregistrarea ECG nu are efect);
- pentru emisie rezultatul codării se trunchiază la 10 biți.

În acest caz se tinde spre o rată medie de compresie având asigurată

$$\begin{aligned} \underline{CR = 1,6 : 1} \\ \underline{PRD < 0,5 \%} \end{aligned}$$

3.4.2. OPTIMIZAREA ALGORITMULUI DE COMPRESIE "EVANTAI"

Metodă de referință a categoriei de metode de compresie realizate prin aproximări poligonale, acest algoritm exploatează bine morfologia specifică traseului ECG. În această secțiune pe lângă implementarea efectivă a algoritmului se rezolvă și problema optimizării lui în raport cu specificitatea semnalului ECG a fiecărui pacient în parte, problemă netratată în literatura aferentă acestei metode de compresie.

3.4.2.1 Prezentarea principiului

Esența acestui algoritm este aproximarea traseului ECG cu segmente de dreaptă construite în felul următor:

- se încadrează traseul inițial cu o "bandă" de toleranță specificată $+E$ și $-E$ [mV];
- pornind de la valoarea tensiunii la un moment dat (eșantionul de pornire) se caută să se construiască un segment de dreaptă cât mai lung, care să se încadreze între limitele benzii de toleranță construite anterior;
- se cercetează pentru fiecare eșantion care urmează dacă dreapta construită între eșantionul de pornire și cel curent se încadrează în limitele benzii de toleranță pe toată lungimea ei;
- în momentul în care dreapta construită nu se mai încadrează, se reține valoarea eșantionului anterior precum și distanța sa față de eșantionul de pornire inițial (în număr de perioade de eșantionare);
- acesta va deveni la rândul său un nou eșantion de pornire pentru construirea unui nou segment de dreaptă care să-l continue pe primul pentru aproximarea traseului ECG.

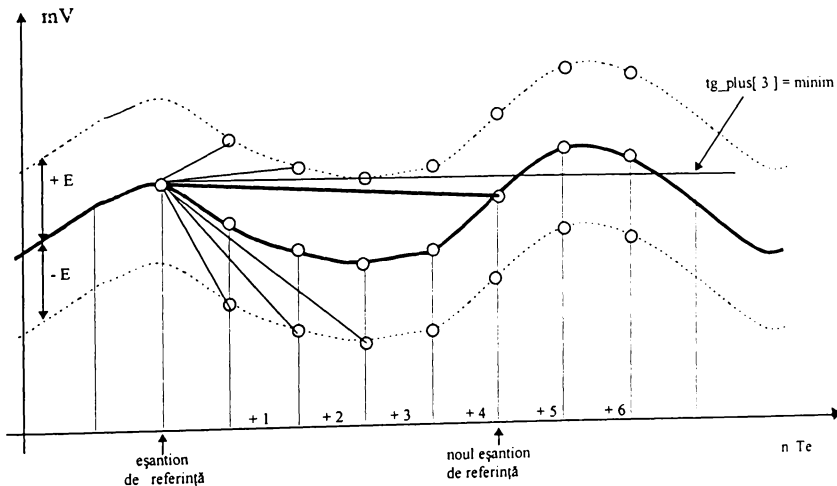


fig. 3.64 Determinarea eșantioanelor ce se rețin

În concluzie utilizând această metodă compresia se realizează prin selecția și transmisia numai a anumitor eșantioane din înregistrare, determinate conform algoritmului "geometric" descris mai sus. La recepție, semnalul original se reface prin aproximarea cu linii drepte trasate între eșantioanele recepționate.

Implementarea acestui algoritm este descrisă în continuare cu ajutorul exemplului din fig 3.64. Pentru a stabili printr-un algoritm matematic care este cel mai lung segment pe care-l putem încadra în banda de toleranță aleasă în limitele +E și -E [mV], pornind de la un eșantion oarecare, se calculează pentru fiecare eșantion ce urmează celui ales ca referință următoarele elemente:

$$tg_plus = \frac{x_{(ref+i)} + E - x_{(ref)}}{i} \quad (3.74)$$

$$tg = \frac{x_{(ref+i)} - x_{(ref)}}{i} \quad (3.75)$$

$$tg_minus = \frac{x_{(ref+i)} - E - x_{(ref)}}{i} \quad (3.76)$$

unde:

- i - reprezintă numărul de ordine al eșantionului testat;
- tg - reprezintă tangenta unghiului făcut de segmentul ce unește eșantionul de referință cu eșantionul curent;
- tg_plus - reprezintă tangenta unghiului făcut de segmentul ce unește eșantionul de referință cu marginea superioară a benzii de toleranță corespunzătoare eșantionului curent;
- tg_minus - tangenta unghiului făcut de segmentul ce unește eșantionul de referință cu marginea inferioară a benzii de toleranță corespunzătoare eșantionului curent.

Odată cu calcularea valorii tangentelor se caută și minimul valorii tangentei cu marginea superioară precum și maximum tangentei cu marginea inferioară a benzii de toleranță. Secvența de căutare se încheie atunci când nu mai este satisfăcută relația :

$$tg_minus \leq tg \leq tg_plus \quad (3.77)$$

În exemplul dat, al 6-lea eșantion este cel care nu mai satisface relația 3.77. În acest moment se ia decizia ca eșantionul 5 să fie reținut ca nou eșantion de referință, dreapta care unește primul eșantion de referință cu al 5-lea fiind cea mai bună aproximare în condițiile alese pentru +E și -E. Se constată că din punct de vedere geometric testarea fiecărui eșantion ulterior celui de referință produce două "evantaie", unul legat de marginea superioară a benzii de toleranță superioare și unul inferior legat de marginea inferioară a benzii de toleranță. Relația 3.77 nu semnifică decât condiția ca segmentul de dreaptă propus pentru aproximare să se încadreze între marginea inferioară a evantaiului superior și marginea superioară a evantaiului inferior.

3.4.2.2. Criteriu de optimizare.

Performanțele compresiei depind doar de marginile +E și -E. Cu cât acestea sunt mai mari, rata de compresie va fi mai ridicată deoarece se pot construi segmente de dreaptă mai lungi pentru aproximarea traseului. Acest lucru duce însă la scăderea performanțelor la refacerea semnalului. Criteriul după care se alege valoarea marginilor este acceptabilitatea clinică sau compromisul făcut între rata de compresie (CR) și eroarea medie pătratică (PRD) la reconstrucție. Pentru a facilita determinarea valorii E a marginilor, în vederea proiectării unui algoritm de compresie eficient, am studiat efectul acestui parametru pe un set de înregistrări din baza noastră de date. Pentru ca rezultatele studiului să aibă un caracter de generalitate am găsit utile următoarele:

- "E" va fi exprimat prin procente din valoarea maximă a valorilor conținute în respectiva înregistrare. Numai aparent această opțiune obligă la trecerea de două ori peste înregistrarea ECG, fiindcă valoarea maximă se poate determina în primele secunde ale monitorizării, înainte de începerea efectivă a compresiei.
- Definirea unui "coeficient de eficiență" (Kef), care să înglobeze atât calitățile de compresie cât și pe cele de refacere a semnalului:

$$Kef = \frac{CR}{PRD} \quad (3.78)$$

Cu cât rata de compresie CR va fi mai mare iar diferența medie pătratică la refacere, va fi mai mică, cu atât algoritmul va fi mai eficient în ansamblu și acest lucru va fi sintetizat de valoarea lui Kef.

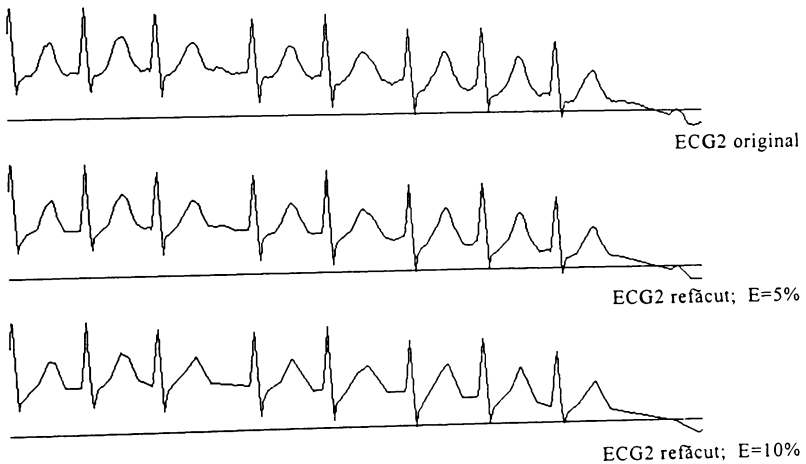


fig. 3.65. Efectul parametrului E asupra refacerii semnalului ECG2.

3.4.2.3 Concluzii

- Algoritmul reține punctele de inflexiune majore, în sensul definit la 2.3.1.1.(b), chiar și pentru $E = 25 \%$. Prin "inflexiuni majore" înțeleg inflexiunile care marchează undele cu amplitudini și pante mari cum ar fi complexul QRS. Acest fapt permite folosirea algoritmului pentru monitorizarea aritmiilor, cu rate de compresie mergând până la $CR = 6:1$, pentru $E = 20 \%$.
- Performanțele algoritmului sunt puternic dependente de conținutul zgomotului de înaltă frecvență. Cu cât este suprapus mai mult astfel de zgomot cu atât posibilitatea de a aproxima porțiuni lungi de traseu cu o linie dreaptă, scade. Exemplul dat în fig. 3.65 și fig. 3.66, ilustrează cele spuse, performanțele în cazul ECG2 fiind de 5 ori mai bune decât cele ale traseului ECG1, zgomotos.

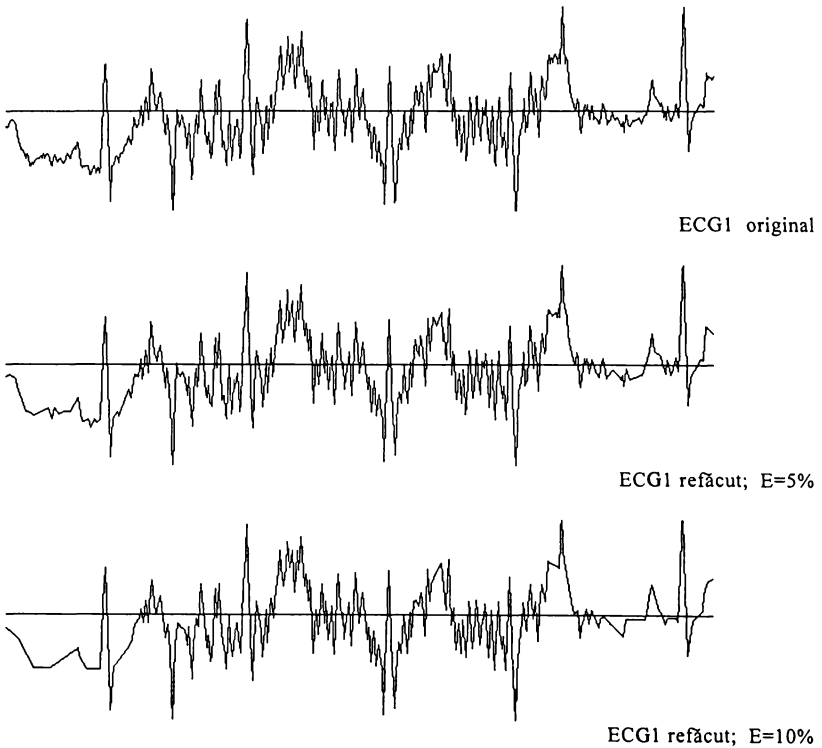


fig. 3.66. Efectul parametrului E asupra refacerii semnalului ECG1.

- Pentru memorarea valorilor indicilor eşantioanelor reținute, este mai avantajos să se genereze un fișier separat și când lungimea segmentului aproximativ nu

depășește 16 Te, (se poate chiar limita lungimea la această valoare), atunci pe un octet se vor putea codifica doi indici vecini.

- Pentru memorarea unui minut de semnal ECG eșantionat la 200 Hz, cu o rezoluție de 8 biți, ar fi necesari 12 Kb de memorie. Cu ajutorul acestei metode de compresie, utilizând o eroare E de 40 μV , se obține o rată medie de compresie de $\text{CR} = 5,5:1$.
- O sinteză a studiilor legate de optimizarea compresiei prin metoda evantai este redată sintetic în figurile 3.67 și 3.68.
- Pentru semnale fără zgomot consider optimă acea valoare a lui E pentru care $\text{Kef} < 1$. Aceasta reprezintă în medie $E = 8\%$, din valoarea E_{max} și înseamnă o "bandă" în jurul traseului ECG de aprox. 40 μV .
- Determinarea lui E pornind de la Kef , are o valoare de generalitate mai mare decât soluțiile propuse de Moody ș.a.(1989) sau De Lucia ș.a.(1990)

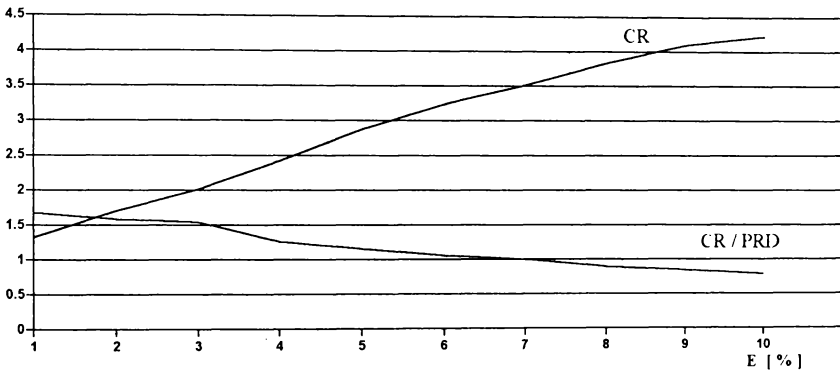


fig. 3.67. Dependenta parametrilor compresiei de limita E[%], pentru ECG2.

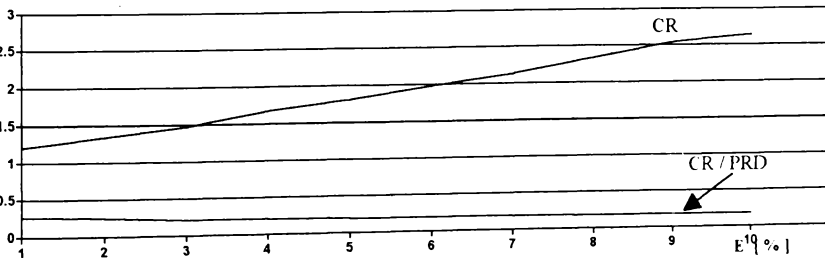


fig. 3.68. Dependenta parametrilor compresiei de limita E[%], pentru ECG1.

3.4.3. Concluzii la studiul de compresie a datelor ECG.

În scopul memorării unei cantități de date ECG cu scopul principal de monitorizare pe termen lung dar și pentru crearea unei baze de date ECG, am utilizat o soluție originală de compresie a datelor. Această soluție este o sinteză a studiilor precedente și combină tehnici ireversibile cu cele reversibile în scopul obținerii unei rate de compresie ridicate și în același timp permite lucrul în timp real, având o complexitate scăzută și dovedească o bună acceptare clinică.

Soluția propusă, folosește trei metode în cascadă: extragerea punctelor esențiale, evantai și codare predictivă

Am utilizat mai multe fișiere cu înregistrări ECG. Datorită utilizării metodei evantai în lanțul de compresie, rata de compresie globală nu este constantă, depinzând în general de caracteristicile semnalului, dar pentru semnale fără zgomot preponderent de înaltă frecvență, am obținut următoarele rezultate, în condiții de acceptabilitate clinică:

- puncte esențiale.....CR = 1,99:1
- evantaiCR = 3,7:1
- codare predictivă.....CR = 1,5:1

Global: rata de compresie obținută a fost de **CR = 10,90 :1**
eroare medie la reconstrucție **PRD = 1,87 %**

3.5 FILTRAREA SEMNALELOR CODATE

În acest subcapitol se urmărește studiul și implementarea unor algoritmi de filtrare numerică ce vor procesa semnale compresate, scopul lor fiind micșorarea volumului de calcule și implicit realizarea unor viteze mari fluxului de date. Totul trebuie subordonat ideii de a realiza aceleași performanțe prin filtrarea semnalului codat ca în cazul filtrării semnalului original.

3.5.1 Filtrarea nerecursivă a semnalelor comprimate prin predicție.

Algoritmul nerecursiv de ordinul N reprezintă algoritmul prin care eșantionul de ieșire se calculează ca și combinație liniară a ultimelor N eșantioane ale secvenței de intrare x_n .

$$y_n = \sum_{k=0}^{N-1} c_k x_{n-k} \quad (3.79)$$

Algoritmul (3.79) poartă numele de filtru numeric, dacă obiectivul lui este acela de a realiza o comportare selectivă față de anumite frecvențe din spectrul semnalului de intrare. Acest lucru este realizabil prin calculul adecvat al coeficienților utilizând una din metodele date de Transformata Fourier sau de Transformata Fourier Discretă.

În sarcina preprocesării apropiate de patul bolnavului cade și aceea de detecție a complexelor QRS, lucru realizabil și prin implementarea de filtre recursive. Fiindcă oricum se dispune de semnalul codat pentru stocare și transmisie consider utilă ideea de a profita de existența acestuia și de a-l procesa în vederea atingerii simultane și a scopurilor de filtrare ce se pun la acest nivel (de exemplu detecția QRS sau anularea derivei liniei izoelectrice). Acest fapt este prezentat sintetic în fig.3.69 unde compresia se realizează prin predicție

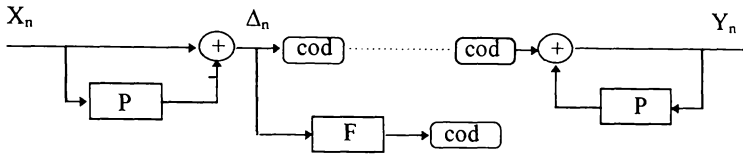


Fig. 3.69. Filtrarea semnalelor codate prin predicție.

Obiectivul cercetării privitoare la filtrarea semnalelor codate este de a stabili eficiența și performanțele combinației algoritmilor de compresie și de filtrare. Considerând un predictor de ordinul P și un filtru de lungime N avem:

$$\Delta_n = x_n - \sum_{i=1}^P a_i x_{n-i} \tag{3.80}$$

$$y_n = \sum_{k=0}^{N-1} c_k \left(x_{n-k} - \sum_{i=1}^P a_i x_{n-k-i} \right) \tag{3.81}$$

Dezvoltând relația 3.81 și schimbând ordinea de însumare se obține:

$$y_n = \sum_{k=0}^{N-1} c_k x_{n-k} - \sum_{i=1}^P a_i \sum_{k=0}^{N-1} c_k x_{n-k-i} \tag{3.82}$$

Notând prima sumă a relației 3.82 cu f_n se observă că ea reprezintă algoritmul de filtrare aplicat semnalului de intrare iar în ansamblul ei se regăsește predicția semnalului filtrat în prealabil.

$$y_n = f_n - \sum_{i=1}^P a_i f_{n-i} \tag{3.83}$$

Comparând relatiile 3.80 și 3.83 rezultă că algoritmul de filtrare a semnalului codat este echivalent unuia de filtrare și apoi de compresie (fig. 3.70).



Fig. 3.70. Echivalența ordinii algoritmilor

Acum apare justificat întrebarea: este de interes utilizarea filtrării semnalelor codate? La prima vedere, relațiile 3.81 și 3.83 conțin același volum de calcule dar plecând de la premisa că semnalul comprimat prin predicție oricum există, se arată în continuare că pot apare situații în care metoda propusă, concretizată în relația 3.81 prezintă avantaj.

3.5.2 Codarea DELTA cu pantă constantă

Cosiderăm semnalul de codat unul analogic. Numit și "modulator delta", schema bloc a codorului DELTA cu pantă constantă, este cea din fig. 3.71 și constă dintr-un comparator în calea directă și un integrator în calea de reacție. Ieșirea comparatorului reflectă semnul diferenței dintre tensiunea de intrare și ieșirea integratorului. Bitul de semn al ieșirii comparatorului este chiar ieșirea digitală a codorului și tot el este cel care controlează semnul pantei integratorului. Ieșirea comparatorului este testată cu o frecvență de eșantionare constantă și identică cu cea de la recepție. În diagrama din fig. 3.72 sunt arătate: semnalul de intrare, semnalul codat și cel refăcut.

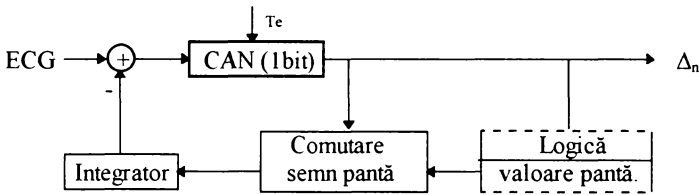


Fig. 3.71. Schema bloc a codorului DELTA

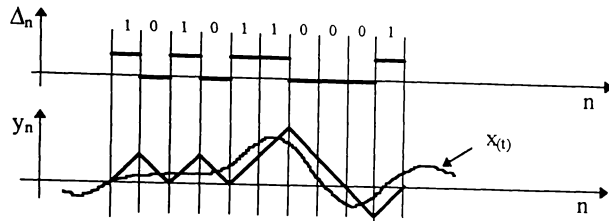
Blocul notat "logică valoare pantă" se folosește în cazul realizării codrului DELTA cu pantă variabilă. În ipoteza realizării codării cu pantă unică, constantă în modul, ieșirea codorului are expresia :

$$\Delta_n = \begin{cases} 1 & \text{daca } x_n > x_{n-k} - \sum_{i=1}^k p(-1)^{\Delta_{n-i}} \\ 0 & \text{daca } x_n < x_{n-k} - \sum_{i=1}^k p(-1)^{\Delta_{n-i}} \end{cases} \tag{3.84}$$

$$y_n = y_{n-1} - p(-1)^{\Delta_n} \tag{3.85}$$

unde:

- k reprezintă lungimea subsecvenței ultimelor valori identice cu Δ_{n-1} , fără întrerupere, anterioare lui Δ_{n-1} în secvența valorilor Δ .
- p reprezintă creșterea / descreșterea semnalului între două eșantioane, respectiv valoarea pozitivă a pantei segmentelor cu care este aproximat semnalul, aceeași cu panta integratorului din schema bloc a codorului DELTA
- 3.85 reprezintă relația de obținere a semnalului reconstituit


 fig. 3.72. Semnal codat (Δ) și cel reconstituit (y_n)

Se constată că la refacere recursivitatea merge până la termenul anterior lui y_n , pe când la codare până la ultimele k valori identice fără întrerupere cu y_{n-1} . Pentru proiectarea codorului DELTA elementele care trebuie determinate sunt T_e și panta p . Pentru a obține o aproximație cât mai bună, trebui ca derivata întâia a semnalului de intrare $x(t)$ să nu depășească valoarea aleasă pentru p . Deci după eșantionare trebuie ca panta să verifice condiția 3.86:

$$p \geq \text{Max} (x_n - x_{n-1}) \quad (3.86)$$

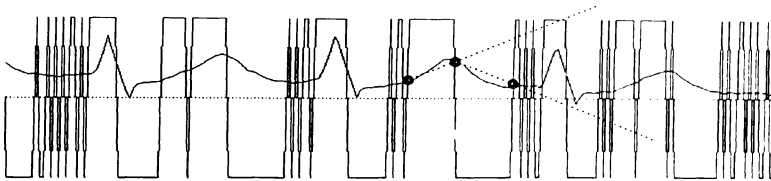


fig. 3.73. Codare cu pantă insuficient de mare.

Acest lucru este confirmat de exemplul de codare din fig.3.73, unde panta mică aleasă duce la pierderea informațiilor de pe porțiuni întinse ale traseului.

Alegerea unei valori mult mai mari pentru p nu este benefică iar practic stabilirea valorii este specifică dinamicii fiecărei înregistrări ECG în parte. Practic am constatat că valori $p = 11 \div 20$, asigură o aproximare satisfăcătoare pentru înregistrările ECG din baza de date. Referitor la frecvența de eșantionare trebuie respectată condiția $F_e \geq 2F_{max}$, practic frecvența de 200 Hz utilizată fiind satisfăcătoare scopurilor propuse.

3.5.3 Implementarea algoritmului de filtrare pentru semnale codate DELTA

Stabilirea algoritmului de filtrare pe semnale codate DELTA pornește de la analiza semnalului codat și a celui reconstituit (fig. 3.72). Din relațiile 3.79 și 3.85 rezultă:

$$Y_n = \sum_{k=0}^{N-1} C_k y_{n-k-1} - p \sum_{k=0}^{N-1} C_k (-1)^{\Delta(n-k)} \quad (3.87)$$

$$Y_n = Y_{n-1} - p \sum_{k=0}^{N-1} C_k (-1)^{\Delta(n-k)} \quad (3.88)$$

Relația (3.88) evidențiază următoarele:

- Reprezintă o structură de filtru recursiv, fiind utilizată o singură celulă de întârziere pentru Y_{n-1}
- Coeficienții C_k ai filtrului obținut sunt aceiași ca ai celui inițial, obținerea lor făcându-se deci cu metodele clasice amintite (Fourier, Fourier Discretă).
- Algoritmul noului filtru conține o singură înmulțire și maximum N adunări/scăderi deoarece coeficienții C_k sunt ponderați de factorii: $+1$ sau -1 după cum semnalul codat este 1 sau 0 .
- Algoritmul filtrului original conține N adunări și N înmulțiri. Deci numărul înmulțirilor flotante se reduce cu $N-1$. Este evidentă eficiența acestui gen de filtrare din punct de vedere al timpului de calcul chiar începând de la filtre de ordinul doi la care numărul înmulțirilor se înjumătățește.
- Pentru implementarea practică a filtrului este utilă observația că dacă semnalul $\Delta_{(k)}$ nu este codat cu "1" și "0" ci cu "+ 1" și "- 1", atunci notând:

$$- (-1)^{\Delta(n-k)} = \Delta^*_k \quad \text{vom avea :} \quad (3.89)$$

$$Y_n = Y_{n-1} + p \cdot \sum_{k=0}^{N-1} C_k \Delta^*_k \quad (3.90)$$

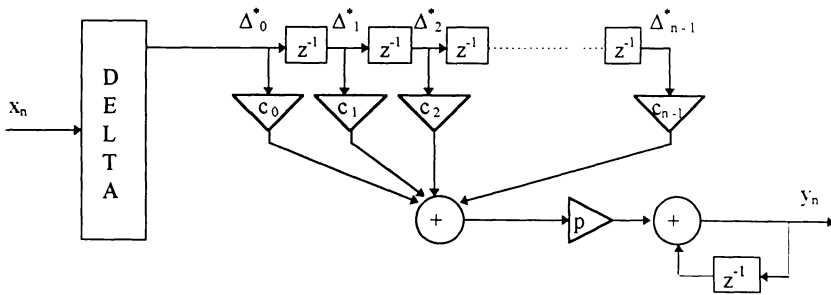


Fig. 3.74. Schema bloc a filtrării semnalelor codate DELTA

Consecință a relației 3.90 implementarea filtrului se face după schema din fig. 3.74. Modulatorul DELTA poate fi realizat hard pentru conversia semnalului analogic $X_{(t)}$ și codarea lui corespunzătoare relației (3.84) sau poate

fi realizat soft, situație justificată cel puțin pentru punerea la punct și testarea performanțelor algoritmului.

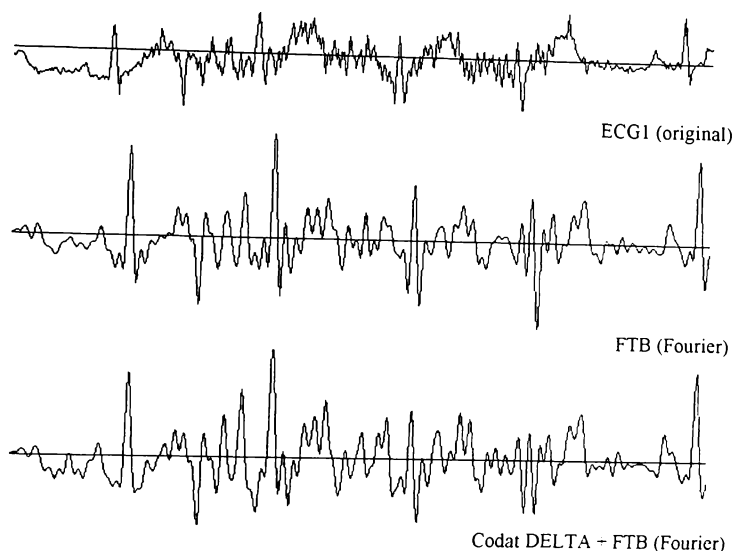


fig. 3.75. Filtrare semnale codate DELTA.

Avantajele filtrării semnalelor codate în acest caz sunt evidente, algoritmul necesitănd:

- o **sigură înmulțire;**
- **N adunări / scăderi ale coeficienților filtrului.**

Cum aceștia au valori cunoscute și constante pe durata filtrării, există varianta mai rapidă de creare a unui "dicționar": cele 2^N combinații posibile ale secvenței Δ_n de lungime N vor reprezenta adresele la care se vor depune valorile sumelor corespunzătoare. Filtrarea devine în acest moment o simplă citire dintr-un tablou.

Verificarea rezultatelor, am realizat-o în unit-ul DELTA.PAS și _DELTA.PAS, (prezentate în Anexa 2) din programul principal TRAIT.PAS. Ele permit codarea Delta cu panta constantă a semnalelor din baza de date, respectiv decodarea lor. Valoarea pantei poate fi prescrisă la valoarea dorită. Un alt modul permite filtrarea acestor semnale prin metode liniare Fourier, Fourier discretă, Z, Biliniară. Programele permit vizualizarea imediată a rezultatelor

În fig. 3.75 este prezentat rezultatul filtrării semnalului ECG1 codat DELTA ($p=20$), comparativ cu rezultatul filtrării semnalului necodat, la aceiași parametrii și anume: filtru trece bandă de ordinul 30, cu $F_j = 3\text{Hz}$, $F_s = 30\text{Hz}$. Se constată similitudinea semnalelor filtrate, cu o eficiență deosebită ca timp de procesare de partea filtrării semnalului codat.

3.6 APLICAȚII CLINICE

Cercetările din cadrul acestei teze s-au bucurat permanent de interesul și susținerea echipei de medici care se ocupă de studiul și aplicarea probei de efort la Spitalul Județean Sibiu, Secția Cardiologie, echipă condusă de dr. Ioan Manițiu.

Rod al acestei colaborări, au rezultat obiectivele și temele la care sistemul de monitorizare trebuie să răspundă.

3.6.1 Automatizarea derulării protocolului probei de efort

Prin protocol se înțelege în contextul temei în discuție, ansamblul tuturor măsurilor și activităților derulate, cu specificarea cronologică a fiecărui pas și a parametrilor ce trebuie fixați sau urmăriți, pentru ca testul să aibă șansa maximă de a furniza rezultate pozitive.

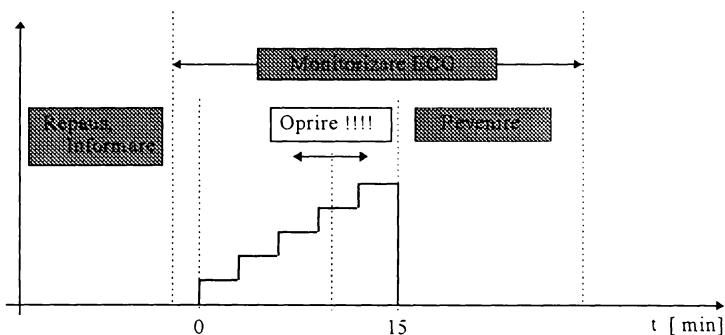


fig. 3.76 Protocolul probei de efort

Protocolul probei este stabilit de medic și el trebuie să fie bine precizat și respectat întocmai, pentru a putea furniza date relevante pe loturi mari de bolnavi. În acest fel, experiența medicală se îmbogățește cu date reale și pertinente, în beneficiul pacienților. Trebuie arătat că nu toți pacienții se supun unui singur protocol, ci medicul va stabili care protocol din setul propriu al laboratorului este cel mai indicat în funcție de vârstă, greutate, antecedente. Cel mai frecvent utilizat protocol este denumit Bruce, ilustrat în fig.3.76 și constă din:

- o etapă de repaus, când pacientul este pregătit și informat asupra desfășurării probei, făcându-i-se și investigațiile obligatorii de debut: electrocardiograma de referință, TA, și frecvența cardiacă în pozițiile culcat și în ortostatism. urmează proba de efort propriu zisă, asistată permanent de medic, în care se asigură la fiecare 3 minute o viteză și o înclinare a covorului bine precizate, conform unui tabel stabilit de protocol. Creșterea de putere la fiecare nou stadiu este specifică fiecărui protocol utilizat (30 W în cazul protocolului Bruce)..

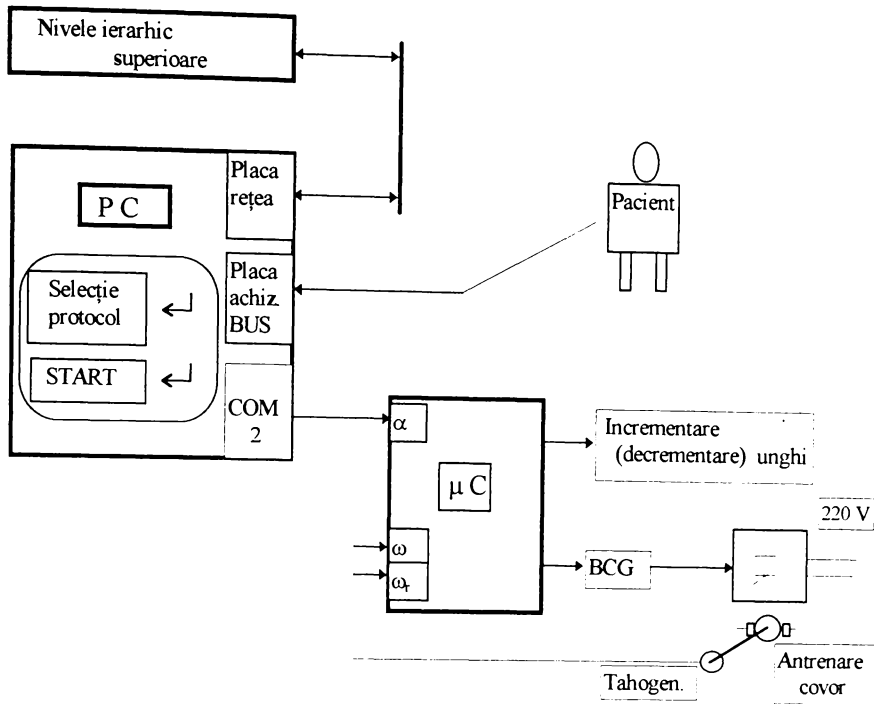


fig. 3.77. Schema bloc a automatizării protocolului probei de efort

- Oprirea se poate face în următoarele cazuri:
 1. s-au epuizat cele 15 minute prestabilite ale testului
 2. s-a atins un procent prestabilit (85..90%) din frecvența cardiacă maxim admisibilă pentru fiecare pacient, calculată pe baza unor tabele (sau formula 3.94) în funcție sex, vîrstă și greutate.
 3. atunci când securitatea bolnavului o reclamă, decizia de oprire este luată de medic:
 - intoleranța la un anumit nivel de efort;
 - dureri anginoase de gravitate medie;
 - nivel de ischemie suficient pentru diagnostic dedus din modificarea segmentului ST.
 - aritmii severe: mai mult de 3 contracții ventriculare premature, tahicardie sau fibrilație ventriculară.
 - semne de ataxie;
 - semne de insuficiență circulatorie periferică (paliditate, piele umedă, cianoză);
 - creșterea TA peste 250 mmHg la valoarea sistolică și peste 130 mmHg la valoarea diastolică;

-scăderea tensiunii în momente de efort cu mai mult de 20 mm Hg pe durata primelor 3 feze de creștere a efortului.

-oboseală extremă ;

-dorința expresă a bolnavului de a se opri testul.

- etapa de revenire în care sunt controlate ECG și TA timp de 10 minute și este indicată și ascultarea cordului. Multe cercetări confirmă importanța deosebită a monitorizării ECG pe durata revenirii, când pot fi observate modificări importante ale morfologiei semnalului (inclusiv a segmentului ST), cu relevanță pentru diagnosticare.

Pentru automatizarea desfășurării protocolului probei de efort, în contextul general al monitorizării probei, am conceput schema bloc din fig. 3.77 care se află în curs de finalizare. Astfel conform schemei bloc, controlul parametrilor mecanici ai covorului rulant (înclinație și viteză), este efectuat de un sistem cu microcontroler, aflat în dialog cu sistemul IBM-PC prin intermediul portului serial COM2. Calculatorul va furniza la fiecare interval important de timp noii parametrii pe care microcontrolerul îi va aduce și menține la valorile setate prin protocol. Reglarea vitezei covorului rulant se face prin controlul în buclă închisă a turației motorului de curent continuu alimentat printr-un redresor comandat de la rețeaua de 220 V. Utilizarea unui regulator pentru viteză se impune datorită frecărilor greu de controlat ale covorului datorită greutății diferite a pacienților. În schimb înclinația covorului rulant se realizează în buclă deschisă printr-un mecanism cu clichet acționat de un electromagnet și reductor mecanic, care duce la înălțarea unui capăt al covorului rulant.

3.6.2 Creare prototip ST prin mediere cu supraeșantionare.

Legat de analiza segmentului ST, a existat permanent idea sporiri gradului de precizie al parametrilor care-l descriu. În acest sens am elaborat un procedeu original de creare a unui prototip pentru segmentul ST, pe care să se efectueze apoi măsurătorile pentru raport.

Ținând cont de gradul de afectare cu artefacte de mișcare a traseului ECG, este riscant a se lua în considerare parametrii de la un un singur segment ST prelevat, soluția reprezentând-o medierea câtorva dintre acestea. Numărul celor mediate nu trebuie să fie prea mare, pentru a nu include în mediere și evoluția morfologiei pe termen mai lung a segmentului ST. Am constatat practic, că un număr de 8 este suficient pentru aceasta. Medierea nu se face numai pentru segmentul propriu-zis ci pentru un interval mai mare începând cu complexul QRS prelevat.

Ideea medierii cu supraeșantionare, are la bază observația că undele selectate nu sunt eșantionate în aceleași momente de timp, referitoare la momentele de referință ale debutului lor fiziologic, ci eșantionarea poate "cădea" oriunde în limita duratei T_e , raportate la acele momente de referință. Acest hazard se reduce dacă frecvența de eșantionare este mare, dar principal el rămâne. Situația este exemplificată în fig. 3.78, unde se constată că același eveniment poate fi reprezentat de eșantioane diferite. La frecvența de 200 Hz, acest lucru este evident ținând cont că unui complex QRS ($\cong 100$ ms) îi revin doar 20 de eșantioane.

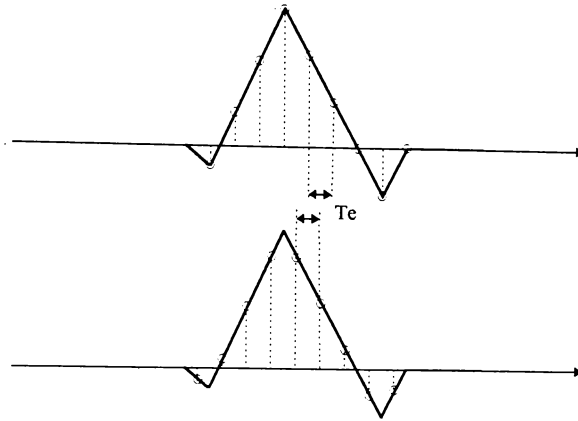


fig. 3.78. Hazardul momentelor de eșantionare.

Prin medierea clasică a eșantioanelor din două complexe QRS succesive, prelevate în două subsecvențe, se propagă o eroare datorată deci decalajului existent în realitate între undele conținute în cele două subsecvențe, fiind afectată morfologia reală a undelor mai ales în zonele cu variații mari (vîrfurile undelor).

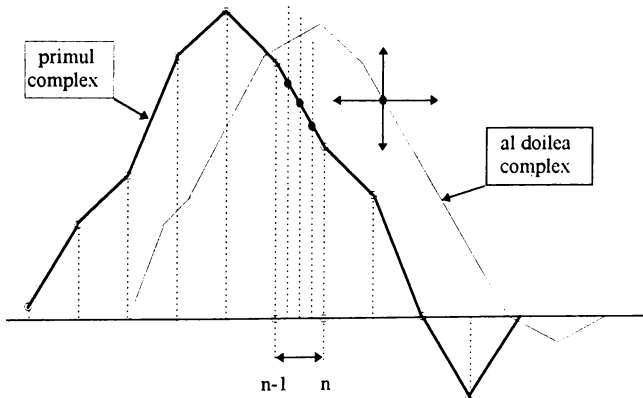


fig. 3.79. Determinarea poziției optime pentru mediere.

Prin medierea propusă se reduce acest inconvenient, undele din cele două subsecvențe, fiind mai bine "suprapuse". Principiul metodei este ilustrat în fig. 3.79. și presupune următoarele etape:

- determinarea primului complex (L eșantioane);

- prelevarea primei secvențe Y1, care să încadreze primul complex cu M eșantioane la stânga și la dreapta (o secvență de lungime L+2M);
- o interpolare liniară între fiecare două valori ale eșantioanelor consecutive. În acest fel, primul traseu, de lungime L+2M, este aproximat printr-o linie poligonală.
- divizarea fiecărui interval Te în k subintervale (se "supraeșantionează" teoretic traseul ECG). În fiecare din aceste noi puncte intermediare formate se determină valoarea pe care eșantioanele teoretice ar trebui să le aibă conform interpolării făcute. Se crează în acest fel o nouă secvență Y1*, având dimensiunea de k·(L+2m) elemente.
- selectarea secvenței Y2, corespunzătoare complexului QRS următor, de lungime L.
- "deplasarea" celei de-a doua secvențe față de prima, cu M poziții la stânga și la dreapta, față de poziția centrală a complexului.
- În fiecare punct, **inclusiv cele noi create**, se calculează diferența medie pătratică conform relației 6.1:

$$d_j = \frac{\sum_{i=1}^L (Y1^*_{(i \cdot k + j)} - Y2_{(i)})^2}{L} \quad (3.91)$$

- În relația 3.91 diferența d_j depinde de poziția relativă a celor două secvențe, exprimată prin indicele j ∈ [-k·M ; +k·M].
- Se determină minimumul valorilor din tabloul d_j, și se reține valoarea indicelui corespunzător, j_{min}.
- Dacă valorile din acest tablou sunt mai mari decât o valoare minimă impusă, atunci se trage concluzia că secvența a doua este aberantă și nu se va lua în calcul.
- Se trece în continuare la poziționarea exactă a celui de-al doilea complex, pe ordonată. Pentru aceasta se va căuta minimumul valorii date de relația 3.92, în funcție de x. Limitele variabilei reale x se stabilesc prin încercări. De pasul și limitele la care ajunge x va depinde rapiditatea algoritmului de căutare.

$$d_{j_{min}, x} = \frac{\sum_{i=1}^L (Y1^*_{(i \cdot k + j_{min})} - (Y2_{(i)} - X))^2}{L} \quad (3.92)$$

- După determinarea minimumului, se vor înlocui în tabloul Y1* valorile care anterior fuseseră calculate prin interpolare cu valorile secvenței Y2, plus ecartul x, determinat anterior.

$$Y2_{(i)} \rightarrow Y1^*_{(i \cdot k + j_{min})} \quad (3.93)$$

- Dacă poziția j_{min} a fost deja "ocupată", atunci fie că se face o mediere cu valorile existente, fie că se ignoră noile valori.

- Procesul de mediere ia sfârșit când toate cele k poziții intermediare se ocupă cu valori din secvențe reale, sau când s-au epuizat numărul fixat de complexe de mediat.

3.6.3. Analiza aritmiilor.

Programul de analiză a aritmiilor a fost dezvoltat inițial pentru o monitorizare continuă în unități de terapie intensivă, tabelul 3.6 prezentând o clasificare a câtorva aritmii mai importante în ordinea priorității lor (Tompkins and Webster 1981). Deși prin specificul ei proba de efort se desfășoară sub supravegherea medicului specialist, analiza aritmiilor pe durata probei reprezintă un element util și care se dorește a fi dezvoltat.

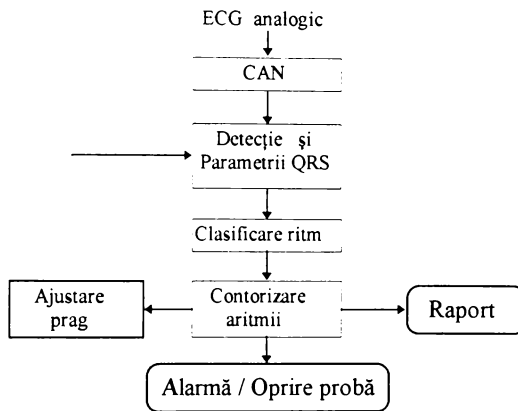


fig 3.80. Algoritm pentru analiza aritmiilor.

Figura 3.80 prezintă organigrama după care a fost dezvoltat algoritmul pentru analiza aritmiilor. La baza sa se află un detector QRS care trebuie să fie dintre cele mai performante. Pentru clasificarea aritmiilor, la fiecare complex QRS se determină lățimea sa și intervalul R-R. Menționez că pentru proba de efort, în algoritmul în timp real a fost implementată în paralel cu detecția și cuantificarea morfologiei segmentului ST.

Elementul specific al analizei de ritm îl constituie în cadrul probei de efort, atingerea ritmului limită, moment în care se va lua decizia opririi testului. Acest fapt survine pe fondul unui cord ischemic, fiind interpretat ca atare dar în același timp reprezintă un factor de evitare a riscului suprasolicitării inimii.

$$R-R_{\min} = k \cdot \frac{60}{215 - 0,66 \cdot \text{virsta}} \quad (3.94)$$

Coeficientul k se ia în general de 1,1 ÷ 1,2 în funcție de aprecierea de către medic a stării generale a pacientului.

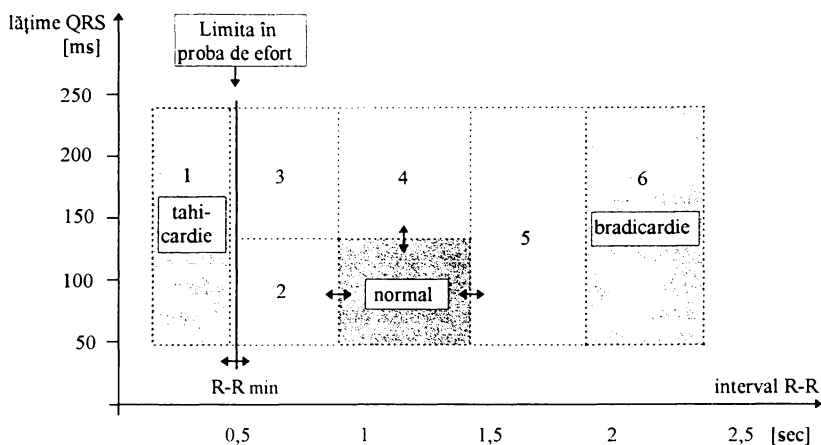


fig. 3.81. Principiu de clasificare aritmii.

Nr	Denumire	Descriere	Caracter
1	Asistolie	Nu există activitate ventriculară 2-3 sec.	Dramatic
2	Fibrilație ventriculară	Activitate ventriculară neordonată mai mul de 3 sec	Dramatic
3	Bloc sino-atrial	Lipsa activității atriale și ventriculare	Dramatic
4	Bradicardie limită	ritm < 40 / min (R-R > 1,5 s)	Dramatic
5	Tahicardie limită	ritm > 120 / min (R-R < 0,5 s)	Dramatic
6	R peste T	CVP cu suprapunerea unei R peste T	Alarmant
7	Bătăi atriale premature (BAP)	Contrații ectopice cu originea în atrii (extrasistolă atrială)	Alarmant
8	Contrații ventriculare premature (CVP)	Contrații ectopice cu originea în ventricule (extrasistolă ventriculară)	Alarmant
9	Bigeminism	CVP în alternanță cu câte o bătaie normală	Alarmant
10	Trigeminism	CVP în alternanță cu câte două bătai normale	Alarmant
11	Ritm limită de efort	Atingerea R-R minim - calculat, pe durata probei de efort.	Oprire proba

Tabelul 3.6. Câteva aritmii în ordinea importanței clinice.

Clasificarea aritmiilor se face în conformitate cu zonele din fig. 3.81. În principiu, (Tompkins 1988), fiecare nou complex detectat poate fi clasificat în una din următoarele categorii: normal, prematur, întârziat sau aberant iar din punct de vedere al lăţimii, în normale sau anormale. Limitele zonelor din fig. 3.81 sunt calculate în procente din valorile limitelor considerate a delimita zona funcţionării normale. Şi limitele zonei considerate normale se pot actualiza în funcţie de pacient, sau la acelaşi pacient limitele se pot adapta pe durata probei.

Cu excepţia băţailor considerate normale, toate celelalte vor fi contorzitate în zonele în care "cad" după clasificare. Pe lângă aceasta se va lua în considerare caracterul secvenţei băţailor dintr-o zonă: singulare, duble, triple, bigemene, trigemene, etc, astfel încât să poată constitui criteriu de alarmare, conform cu cele prezentate principal în tabelul 3.6.

3.6.4. Aprecierea clinică a rezultatelor testului de efort.

Toate rezultatele obţinute în urma testului de efort, sunt în general considerate ca un element în plus în tabloul clinic al pacientului, medicul fiind cel care în final le integrează în complexul de date ce duc la stabilirea diagnosticului.

Rezultatele imediate ale testului de efort sunt reprezentate de raportul testului, furnizat imediat după terminarea probei. Analiza segmentului ST în timpul probei de efort are ca specific faptul că semnalul este afectat preponderent de artefacte de mişcare. Acest fapt face ca la alegerea soluţiei de cuantificare a stării segmentului ST să optez pentru o variantă cu metode integrale.

El trebuie să cuprindă evoluţia în timp a tuturor elementelor de interes ale traseului ECG şi în special parametrilor necesari pentru evaluarea stării de ischemie a cordului. Pentru acestea vor trebui reliefate:

- graficul evoluţiei în timp a denivelării segmentului ST
- reducerea intervalului P-R după debutul probei.
- reducerea undei R. Este în general agreată ideea că unda R se reduce ca şi amplitudine deşi acest lucru are şi cauze obiective cum ar fi modificările respiratorii. Reducerea pare să fie mai accentuată la pacienţii sănătoşi decât la coronarieni. În general este asociată şi cu reducerea undei T.

Raportul cuprinde, pe lângă datele referitoare la evoluţia segmentului ST la intervalele fixate de protocol şi o serie de date furnizate de un program existent anterior acestuia şi care înglobează datele referitoare la pacient, având rolul unui "bilet de ieşire din spital".

Din acest punct de vedere, testul este considerat:

- **pozitiv**, dacă s-a identificat o ischemie a cordului pe baza analizei datelor obiective din timpul desfăşurării testului;
- **negativ**, dacă nu s-a diagnosticat o ischemie.

Pe lângă aceste rezultate imediate mai există aşanumitele "rezultate în timp" ale testului de efort, care se referă la confirmarea în timp şi pe alte căi a rezultatelor imediate. Astfel în urma diagnosticării prin alte metode considerate de mare siguranţă a stării cordului (angiografia sau testări cu substanţe de marcaj nucleare), rezultatele testului se pot dovedi

- **adevărate**, dacă rezultatul testului este confirmat;
- **fals pozitive**, dacă în realitate nu există o ischemie prezisă de test.

- **fals negative**, dacă în realitate există o ischemie nedovedită de test.

Referitor la aceste rezultate din urmă, în literatura aferentă domeniului se folosesc frecvent termeni legați de eficacitatea și gradul de încredere oferite de testul de efort. Acești termeni se definesc ca și mărimi statistice și se determină prin testarea unor loturi de bolnavi, cărora li se confirmă sau infirmă diagnosticul prin alte metode. Se definesc procentual astfel:

$$\text{sensibilitatea} = \frac{\text{adevarat pozitive}}{\text{adevarat pozitive} + \text{fals negative}} \cdot 100$$

$$\text{specificitatea} = \frac{\text{adevarat negative}}{\text{fals pozitive} + \text{adevarat negative}} \cdot 100$$

$$\text{valoare de predictie} = \frac{\text{adevarat pozitive}}{\text{adevarat pozitive} + \text{fals pozitive}} \cdot 100$$

Aceste valori se vor determina în timp, prin urmărire statistică a loturilor de bolnavi, urmând ca din analiza lor să se stabilească care parametri ai probei de efort trebuie actualizați spre a obține rezultate superioare.

Capitolul 4

CONCLUZII

4.1 Sinteză

În cadrul tezei de doctorat cu titlul "**Contribuții la prelucrarea numerică a electrocardiografei în proba de efort**" sunt prezentate cercetările care au contribuit la realizarea efectivă a unui sistem complet pentru monitorizarea probei de efort. Teza este structurată în patru capitole și anume:

1. Prezentarea domeniului
2. Stadiul actual al cercetărilor
3. Realizări și contribuții personale
4. Concluzii.

În organizarea conținutului tezei am făcut efortul de a contura bine părțile de cercetare propriu-zisă și rezultate practice. Toate sarcinile de cercetare impuse de obiectivele tezei, au fost abordate sistematic prin:

- trecerea în revistă a stării actuale din domeniu;
- prezentarea aparatului matematic cu unele soluții originale;
- implementare;
- evidențierea rezultatelor.

Situată la intersecția mai multor domenii majore de preocupări științifice: medicina, informatica, electronica, lucrarea reprezintă o contribuție la implementarea unor metode moderne de prelucrare numerică a semnalelor pe tehnică de calcul accesibilă (IBM-PC) pentru rezolvarea unor probleme concrete solicitate de necesități reale ale asistenței medicale din zilele noastre, precum monitorizarea probei de efort. Cheia realizării obiectivelor tezei, consider a fi utilizarea metodelor moderne de prelucrare numerică a semnalelor în scopul asigurării unui nivel ridicat actului medical.

Concretizarea cercetărilor într-un domeniu interdisciplinar, prin realizarea sistemului de monitorizare automată în cadrul Laboratorului de Investigații prin Proba de Efort la Spitalul Județean Sibiu, conferă un plus de valoare demersurilor de elaborare și finalizare a tezei.

Nu în ultimul rând, lucrarea poate fi privită ca o încercare de soluționare a unor probleme specifice unui context dat. În condițiile în care există producători de marcă în domeniu, argumentele care au stat la baza demarării și finalizării tezei au fost în principal trei:

- Costul economic prohibitiv la nivelul posibilităților economice ale unei unități sanitare în actuala etapă de trecere spre o economie de piață, impune utilizarea de mijloace materiale accesibile ca preț de cost.
- Valorificarea realizărilor anterioare în domeniu ale echipei de medici și nu în ultimul rând a dotării deja existente (ca de exemplu covorul rulant realizat într-o etapă premergătoare și cercetările de informatică medicală, cu referire la proba de efort).
- Formarea unui centru de preocupări în domeniul procesării semnalelor biomedicale, în beneficiul Facultății de Medicină și al Facultății de Inginerie ale Universității "Lucian Blaga" din Sibiu, pe de o parte dar mai ales în beneficiul serviciilor medicale ale Spitalului Județean, în special secția de Cardiologie, pe de altă parte.

Lucrarea este realizată în spiritul unor principii și a unor tendințe generale actuale bine definite în domeniul monitorizării și procesării semnalelor biomedicale. Rezolvarea temei a presupus o documentare specială și a obligat la parcurgerea unei bibliografii vaste.

4.2 Contribuții

În capitolul 1 intitulat "**Prezentarea domeniului**" este abordată într-o manieră proprie de organizare a materialului, problematica stadiului actual al monitorizării inteligente în cardiologie. Sunt stabilite totodată și principalele sarcini ce revin cercetărilor, în spiritul principiilor și al tendințelor generale actuale bine definite în domeniul monitorizării și procesării semnalelor biomedicale, argumentate cu referințe bibliografice.

În capitolul 2 denumit "**Stadiul actual al cercetărilor**", revendic ca și contribuție personală modul de organizare al studiului bibliografic, impus de tematica tezei, cu obiectivul precis de a evidenția stadiul actual al cercetărilor și realizărilor în principalele domenii care vor concura la realizarea sistemului de monitorizare a probei de efort. Acestea sunt trecute pe rând în revistă în subcapitole distincte și anume: Achiziția de date ECG; Detecția QRS; Compresia de date ECG iar în ultima parte a capitolului este prezentată o sinteză a principalelor soluții din literatura de specialitate vizând cuantificarea morfologiei segmentului ST.

Capitolul 3 intitulat "**Realizări și contribuții personale**" este cel mai consistent capitol al tezei, aici fiind prezentate realizările personale și în special soluțiile originale și elementele de progres.

În subcapitolul "*Arhitectura propusă pentru cercetare*", este prezentată arhitectura considerată optimă pentru un sistem complex de monitorizare în cadrul secției Cardiologie, în acord cu principiile consacrate în domeniu iar includerea probei de efort, consider a fi un element original.

Subcapitolul "*Sistemul de achiziție de date*" începe cu precizarea că sistemul de achiziție a reprezentat în ansamblul tezei un mijloc de atingere a obiectivelor tezei, nu un scop în sine. Sunt prezentate în prima parte principiile proiectării sistemului de achiziție de date ECG. Este prezentat apoi sistemul achiziție de date utilizat

pentru monitorizare (anexa 1), bazat pe un calculator IBM-PC, cu referiri la programul de gestionare a lui. Ca realizări personale menționez:

- stabilirea unei relații originale pentru calculul numărului minim de biți necesar cuantificării, pentru semnale ECG
- evidența originală a exploatării bugetului de timp între achiziție, procesare și transfer de date în modul de lucru cu redirectarea intreruperii INT8 a unui calculator IBM-PC;

Tot în acest subcapitol este soluționată și problema afișării pe monitorul PC a traseului ECG, fiind prezentate soluțiile autorului pentru:

- un algoritm de compresie eficient și rapid pentru afișaj;
- reprezentarea grafică cu "bară glisantă".

Subcapitolul "*Implementarea detectoarelor QRS*" este cel mai vast și este consacrat procedeele de prelucrare numerică a semnalului ECG utilizate în teză, vizînd preprocesarea (filtrarea), detecția și identificarea undelor. Sunt propuse și implementate soluțiile cele mai avantajoase pentru procesarea în timp real a semnalului ECG în contextul monitorizării probei de efort, după care sunt prezentate rezultatele, cu o bună susținere printr-un material grafic original. În acest context sunt de evidențiat:

- algoritm și program original CAR_FREQ.PAS (anexa 2) pentru studiul și trasarea caracteristicilor de frecvență ale algoritmilor de procesare numerică a ECG. El permite trasarea caracteristicilor de frecvență atât pentru algoritmi liniari (recursivi sau nerecursivi), cât și pentru cei neliniari, ajutînd în acest fel la analiza eficientă și rapidă a comportării algoritmilor în domeniul frecvență;
- un program original, QRS.PAS (anexa 3) pentru studiul "off line" al eficienței algoritmilor de procesare ai traseului ECG. Programul reprezintă un instrument eficace pentru cercetare și pentru proiectarea analizei automate a semnalului ECG. Programul conține mai multe module, selectabile din meniul general:
 1. studiul algoritmilor consacrați de obținere a filtrelor numerice, cu ajutorul transformatei Fourier, Fourier discretă, Z și Biliniară. Structura filtrului și valorile coeficienților se realizează automat impunînd caracteristica de frecvență dorită, în mod specific fiecărui tip de filtru. Imediat, eficiența filtrului poate fi probată printr-o analiză temporală pe semnale din baza de date;
 2. studiul algoritmilor specifici de procesare a traseului ECG, cu posibilitatea introducerii facile în programul sursă, spre studiu, de noi algoritmi;
 3. vizualizarea separată sau suprapusă a câte două trasee ECG din baza proprie de date, inclusiv cele procesate, cu posibilități multiple de măsurare, zoom și deplasare;
 4. generarea matematică de semnale (sinusoidale, treaptă, impuls, zgomot alb, etc.) în scopul studiului algoritmilor;
 5. generarea de noi "semnale ECG" pentru studiul algoritmilor, semnale realizate prin adăugarea cu ponderea dorită la semnalele ECG reale a unor semnale generate matematic;

- program original FFT.PAS pentru determinarea spectrului de frecvență al semnalelor ECG înainte și după procesare. Este realizat pe baza Transformatei Fourier Rapide, cu posibilități facile de evidențiere a efectului momentului de debut al secvenței selectate, a lungimii secvenței și a tipului de ferestruire asupra spectrului rezultat;
- două funcții de ferestruire originale, extrem de eficiente pentru studiul semnalelor cu un spectru larg de frecvență;
- un algoritm liniar original de corelare a traseului ECG cu funcții șablon preluate din semnal, care realizează simultan și eliminarea derivatei liniei izoelectrice (FTS);
- o prezentare a unor algoritmi neliniari originali pentru detecția QRS în vederea procesării ECG în timp real;
 1. algoritm și program original pentru detecția QRS bazat pe analiza minimelor și maximelor locale;
 2. algoritm original și program pentru procesarea ECG în vederea identificării complexului QRS, bazat pe reaxarea semnalului, cu ajutorul "înfășurătoarei" inferioare a semnalului;
- Ambii algoritmi sunt implementați ca module în programul general QRS.PAS, secțiunea algoritmi neliniari;
- soluție originală pentru cuantificarea eficienței preprocesării ECG în vederea identificării complexului QRS, etapă esențială a oricărei procesări a traseului ECG;
- o raportare originală a rezultatelor procesării ECG cu ajutorul filtrelor nerecursive clasice (Fourier) comparativ cu rezultatele procesării cu metode specifice (șablon S5 + FTS).

În subcapitolul "*Algoritmi de compresie*" pe lângă soluțiile clasice sunt studiate și implementate soluții de compresie care exploatează în plus specificul semnalului ECG. Sintetic remarc următoarele contribuții:

- studiul și implementarea unor algoritmi de compresie utilizând ca metode: codarea aritmetică, predicția liniară cu Entropie Maximă, "evantai";
- studii și soluții originale de optimizare a algoritmilor de compresie în cazul compresiei de date ECG pentru stabilirea celor care se pretează cel mai bine la procesarea în timp real;
- studiu și program original pentru eficientizarea predicției liniare la codarea semnalelor ECG.

Următorul subcapitol, "*Filtrarea semnalelor codate*", conține o soluție originală de sporire a vitezei și eficienței procesării semnalului ECG. Ideea este aceea a utilizării unor algoritmi de filtrare care să opereze pe semnale codate în vederea compresiei. Principalele elemente prezentate sunt:

- un studiu original asupra filtrării semnalelor codate prin predicție liniară.
- un algoritm original de filtrare a semnalelor codate Delta.
- programe originale DELTA.PAS și _DELTA.PAS (anexa 2), incluse în TRAIT.PAS, pentru studiul performanțelor codării /decodării Delta a semnalelor ECG precum și pentru filtrarea semnalelor codate delta.

Subcapitolul intitulat "*Aplicațiile clinice ale rezultatelor*", prezintă realizările globale ale cercetărilor, în acord cu cerințele medicale, fiind cel care însumează

practic marea majoritate a demersurilor făcute pe parcursul elaborării tezei. Punctate, realizările sunt următoarele:

- program original 0_PRINC.PAS (anexa 4), pentru monitorizarea în timp real a probei de efort, conținând soluții originale pentru :
 1. crearea sau actualizarea fișei pacientului;
 2. posibilitatea derulării mai multor protocoale în funcție de indicația medicului;
 3. interfațare ergonomică pentru operator, cu posibilitatea personalizării, înainte de startul probei, a modului de procesare ECG, care să țină seama de particularitățile achiziției la fiecare pacient în parte. În acest scop poate fi selectat tipul de preprocesare (filtrare) cel mai adecvat și pragul optim pentru detecția QRS;
 4. achiziția în timp real a trei canale de date, ce pot fi programate independent;
 5. crearea bazei de date pentru fiecare pacient, prin selecția la intervalele de timp dictate de protocol a secvențelor de semnal reprezentative, și a altora considerate de medic, ca fiind de interes;
 6. rezultate globale ale probei de efort materializate prin grafice ale evoluției în timp a principalilor parametri urmăriți, pe parcursul probei;
 7. soluție originală pentru medierea complexelor QRS prin supraeșantionarea secvențelor SABLON.PAS (anexa 4), în vederea creșterii preciziei măsurătorilor.
- posibilitatea de a introduce în programul final criterii noi de evaluare și cuantificare a parametrilor testului de efort, în vederea dezvoltării de noi cercetări;
- transferul rezultatelor obținute, altor programe de prelucrare statistică;

Din punct de vedere al graficii înglobate, teza cuprinde un număr de 104 figuri și grafice, din care marea majoritate sunt originale fiind rezultatul soluțiilor propuse sau concretizarea unor rezultate.

Lista bibliografică cuprinde 164 de titluri, acoperind aria domeniilor solicitate de obiectivele tezei, iar ca situate de timp ea acoperă un interval larg și în același timp actual, între anii 1978 - 1996. În lista bibliografică sunt citate și 18 lucrări din domeniu la care realizatorul tezei de față este autor sau coautor. Din acestea la 8 este prim-autor, iar la 10 este coautor.

În partea de *Anexe* sunt prezentate materiale pentru ilustrarea în detaliu a unor părți ale lucrării și anume:

Anexa1 - schema electrică a plăcii de achiziție utilizate;

Anexa2 - program pentru trasarea caracteristicilor de frecvență ale algoritmilor de procesare ECG intitulat CAR_FREQ.PAS și părțile esențiale ale procedurilor de filtrare a semnalelor codate DELTA.PAS și DELTA.PAS;

Anexa3 - program de analiză și procesare of line al semnalelor QRS PAS, cu ilustrarea unit-ului CARDINAL.PAS de filtrare prin metoda Fourier.

Anexa4 - program de monitorizare a probei de efort; 0_PRINC.PAS și programul de mediere cu supraeșantionare SABLON.PAS

4.3 Direcții de dezvoltare a cercetărilor.

Actul de cercetare științifică nu trebuie să aibe momente de sfârșit ci doar puncte care să marcheze anumite etape. Descoperirea de noi probleme este poate tot atât de valoroasă ca și rezolvarea lor. Diversitatea problemelor descoperite pe parcursul elaborării tezei prin munca în echipă interdisciplinară, confirmă convingerea că lucrarea de față poate constitui un punct de plecare în dezvoltarea unor noi cercetări pe termen lung, atât în domeniul procesării numerice a semnalelor, cât și în domeniul medical. În ideea creșterii calității actului medical iată câteva obiective pentru continuarea cercetărilor din cadrul acestei teze:

- dezvoltarea metodelor de evaluare cantitativă a rezultatelor preprocesărilor ECG, în vederea identificării QRS;
- studiul proprietăților spectrale ale algoritmilor de predicție;
- perfecționarea în continuare a modulelor de program pentru afișarea și dialogul "prietenos" al operatorului cu calculatorul, o primă etapă fiind transcrierea programului de monitorizare pentru a rula sub Windows;
- utilizarea de sisteme de calcul performante și diversificarea modulelor de program, pentru achiziția și prelucrarea simultană a 12 derivații;
- dezvoltarea de noi algoritmi de detecție și identificare a undelor traseului ECG, în condițiile specifice ale artefactelor de mișcare, cu precădere a unor algoritmi recursivi;
- dezvoltarea bazei proprii de date cu acoperirea unui domeniu vast de patologii;
- dezvoltarea unui sistem de achiziții de date bazat pe ideea exploatării numerice a unor procesării analogice (derivări, treceri prin zero, valori efective) cu posibilitatea controlării prin soft a parametrilor procesării analogice. În vederea procesării în timp real soluția "transferării" de sarcini părții analogice, va spori performanțele sistemului;
- dezvoltarea de cercetări în timp pentru evidențierea rezultatelor medicale ale probei de efort, în scopul completării experienței clinice a cardiologilor.

BIBLIOGRAFIE

1. Abenstein J.P., Tompkins W.J., (1982) "*A new data-reduction algorithm for real-time ECG analysis*", IEEE-Trans, BME vol 29, Nr 1, 43-48.
2. Adrian T., Stefanelli M.C., D'Alvano F. (1987) "*Data compression via exponential quantisation of the Walsh spectrum*", Journal of Clinical Engineering , Vol 12, Nr 5, 373-378.
3. Ahlstrom M.L., Tompkins W. (1983) "*Automated High Speed Analysis of Holter Tapes with Microcomputers*", IEEE-Trans, BME-30, 651-657
4. Ahlstrom M.L., Tompkins W. (1985) "*Digital Filters for Real-Time ECG Signal Processing Using Microprocessors*", IEEE-Trans, BME -32 nr 9, 708-713
5. Ahmed N., Rao K. R. (1975) *Ortogonal transforms for digital signal procesing*, Springer Verlag.
6. Ahmed N., Milne P.J., Harris S.G.(1975) "*ECG data compression via orthogonal transforms*", IEEE-Trans, BME, vol 22, Nr 6, 484-487
7. Akselrod S., Norymberg M., Peled I., Karabelnik E (1987) "*Computerised Analysis of ST Segment Changes in Ambulatory Electro-cardiograms*", Medical and Biological Engineering and Comuting, vol 25 sept, 513-519
8. van Alste J. A., van Eck W., Herman O. E., (1986), "*ECG baseline wander reduction using linear phase filters*", Computers and Biomedical Research , vol. 19, 417-427.
9. Arthur R.M., Kovesh N.G., Ambos H.D., Cain M.E.(1995) "*Phase and Group-Delay Characteristics of Signal Averaged Electrocardiograms from Patients with Ventricular Tachycardia*", IEEE-Trans, BME vol 42, nr.1, 29-34
10. Aziz P.M., Sorensen H., Van der Spiegel J. (1996) "*An overview of Sigma-Delta Converters*" , IEEE Signal Processing Magazine, ian, 1996,
11. Balochi R., Macerata A, Marchesi C, Taddei A (1989) "*Report on Existing Signal Processing Methods*", KISS-AIM report # 890901 Pisa-Italy
12. Barros A.K., Yoshizawa M., Yasuda Y. (1995) "*Filtering noncorelated noise in Impedance cardiology*", IEEE-Trans, BME vol 42, Nr 3, 324-327
13. Belforte G., De Mori R., Ferraris F. (1979) "*A contribution to the automatic processing of electrocardiograms using syntactic methods*", IEEE -Trans, BME-26, Nr 3, 125-136.

14. Berbari E.J., Lander P. (1992) "*The State of the Art in High Resolution Electrocardiography*", IEEE Trans BME, No.5, 488-501.
15. Bertrand M., Guardo R., (1977) "*Microprocessor applications for numerical ECG encoding and transmission*", IEEE Proceedings, Vol. 65, Nr 5, 714-722.
16. Bessette F., Nguyen L. (1989) "*Automated electrocardiogram analysis: state of the art*", Med. Inform., Vol 14, 43-51
17. Birman K. (1983) "*Rule - Based Learning for More Accurate ECG Analysis*", IEEE -Trans, PAMI-4, 369-380
18. Blanchard S.M., Barr R.C. (1985) "*Comparison of methods for adaptive sampling of cardiac electrograms and electrocardiograms*" , Med. and Biol. Eng. and Comput., vol 23, 377-380.
19. Bohs L., Barr R.C. (1988) "*Real-time adaptive sampling with the Fan algorithm*", Med. and Biol. Eng. and Comput., vol 26, 565-573.
20. Brydon J. (1976) "*Automatic Monitoring of Cardiac Arrhythmias*", IEEE Medical Electronics Monographs 18-22, 27-41.
21. Carrault G., Thoraval L., Senhadji L., Bellanger J.J., Coatrieux J.L, Balocchi R., Macerata A., Marchesi C. (1990) "*Knowledge Based Interactive Signal Monitoring System; Final Report on Advanced Signal Processing Methods and Algorithms*", KISS-AIM Report#1060, Rennes- France, July.
22. Carrault G., Senhadji L., Bellanger J.J., Le Pichon J. (1990) "*ECG Segmentation by Means of Change Detection Theory*", Proceedings of the XII IEEE-EMBS Annual Conferences, Philadelphia-USA, 817-818.
23. Cetin A.E., Koymen H., Aydin M.C. (1993) "*Multichannel ECG Data Compression by Multirate Signal Processing and Transform Domain Coding Techniques*", IEEE Trans BME, Vol. 40, No. 5, 495-499.
24. Chang A.A., Chang W. (1992) "*Hilbert Transform and Fourier Descriptors to ECG Signal Analysis*", IEEE Trans. BME, 512-528.
25. Chen S.W., Clarkson P.M., Fan Q., (1996) "*A robust sequential detection algorithm for cardiac arrhythmio classification*", IEEE-Trans, BME vol 43, nr 11, 1120- 1125
26. Cheng Q., Lee H., Thakor N. (1987) "*ECG Waveform Analysis by Significant Point Extraction: Pattern Matching*", Computers and Biomedical Research, Vol.20, 428-442.
27. Chu C.H.H., Delp E.J. (1989) "*Impulse Noise Suppression and background normalisation of electrocardiogram signals using morphological operators*" , IEEE Trans, BME, Vol. 36, No. 2, 262-273.
28. Chung E.K. (1983) *Exercice Electrocardiography: Practical Approach (Second Edition)* ,Williamms & Wikins, Baltimore / London, 1983.

29. Ciocloada Gh., (1981) "*Contribuții privind prelucrarea potențialelor biologice pentru analiza automată a tulburărilor de ritm cardiac*" Teza de doctorat IPTV Timișoara.
30. Clark K., McLear P., Kortas R., Mead C., Thomas Jr. L. (1981) "*ARGUS/2H Detection of ST-Segments Changes in Ambulatory ECG Recordings*", Computers in Cardiology, IEEE-press, 27-31.
31. Cohen A. (1988) "*Biomedical Signal Processing*", CRC Press Inc., 1988
32. Collins S., Arzbaecher R. (1981) "*An Efficient Algorithm for Waveform Analysis Using the Correlation Coefficient*", Computers and Biomedical Research, Vol.14, 381-389.
33. Cook A.M., Webster J.G. (editors) (1990) *Clinical Engineering*, Prentice-Haal, Englewood Cliffs, New Jersey
34. Cortelazzo G.M. (1987) "*On Efficient FIR Interpolation Via Signal Processing*", Digital Signal Processing, Cappellini V., Constantinides A.G. (eds), 27-31.
35. Cotoi S. "*Testul de efort în evaluarea cardiovasculară*", 1988, Tratat de medicină internă, voll, sub redacția R.Păun; Ed Medicală București.
36. Cox J.R., Nolle F.M., Fozard H.A., Olivier G.C. (1968) "*AZTEC : a preprocessing program for real-time ECG rythm analysis*", IEEE-Trans, BME-33, No.15, 128-129.
37. Craelius W., Restivo M., El_Sherif N. (1986) "*Criteria for Optimal Averaging of Cardiac Signals*", IEEE-Trans, BME-33, No.10, 957-966.
38. Crow R., Campbell S., Prinea R. (1978) "*Accurate Automatic Measurement of ST-Segments response in the exercise electrocardiogram*", Computers and Biomedical Research, Vol.11, 243-256.
39. Cupal J., Reese D. (1988) "*A Multichannel Digital Biolemetry System with Direct Interface to a Personal Computer*", Journal of Clinical Engineering, vol 13, 127-132
40. De Lucia F., Passariello G., (1990) "*Microcomputer based Coronary Care Unit central station*", Journal of Clinical Engineering, vol. 15 Nr.5, 45-51.
41. Dobbs S., Schmitt Neil., Ozemek H. (1984) "*QRS detection by template matching using real-time correlation on a microcomputer*", Journal of Clinical Engineering, Vo.19, No.3, 197-212.
42. Dragu I. -*Prelucrarea numerică a semnalelor discrete*, Ed. Militară, 1985 .
43. Dudea C. *Atlas de Electrocardiografie Clinică*.Ed. Medicală 1988, București.
44. Dumpala S., Reddy N., Sarna S. (1982) "*An algorithm for the detection of peaksin biological signals*", Computer Program in Biomedicine, Vol.14, 249-256.

45. Escalona O., Passariello G., Mora F., (1986) "*An algorithm for microprocessor based on QRS detection*", Journal of Clinical Engineering, Vol.11, No.3, 213-219.
46. Escalona O. J., Mitchell R. H., Balderson D.E. (1993) "*Fast and reliable QRS alignment technique for high-frequency analysis of signal-averaged ECG*", Med. and Biol.Eng.Comput, 31, s137-S146.
47. Forlini F. J. Cohn K., Langston M. F. (1975) "*ST segment isolation and quantification as a means of improving diagnostic accuracy in treadmill stress testing*". American Heart Journal, 90, p483.
48. Fraden J., Newman M.R. (1980) "*QRS detection*", Medical and Biological Engineering and Computing, Vol 18, 125-132.
49. Friedman H., (1985) *Diagnostic electrocardiography and vectorcardiography*, Mc Graw-Hill Company, New York-USA.
50. Friesen G., Jannett T., Jadallah M., Jates S., Quint S., Nagle T. (1990) "*A comparison of the noise sensitivity of nine QRS detection algorithms*", IEEE-Trans, BME-37, No.1, 85-98.
51. Froelicher V., Wolthuis R., Fisher J., Uhl G. (1981) "*Variations in normal electrocardiographic response to treadmill testing*", American Journal of Cardiology, Vol.47, 1161-1167.
52. Furth B., Perez A. (1988) "*An adaptive real-time ECG compression algorithm with variable threshold*", IEEE Trans BME, Vol. 35, No. 6, 89-493.
53. Furno G., Tompkins W. (1983) "*A learning filter for removing noise interference*", IEEE-Trans, BME-30, No.4, 234-235.
54. Giakoumakis E., Papaconstantinou G., Skordalakis E. (1987) "*Rule-based Systems and pattern recognition*", Pattern Recognition Letters, Vol.5, No.4, 267-272.
55. Giallorenzo M., Cohen J., Mora F., Passariello G., Lara L. (1988) "*Ambulatory monitoring device using the Fan method as data compression algorithm*", Med. and Biol. Eng. and Comput., vol 26, 439-443.
56. Gîrlașu St. - *Prelucrarea în timp real a semnalelor*, Ed. Scrisul Românesc, 1978 .
57. Gîrlașu St. - *Introducere în analiza spectrală și de corelație*, Ed. Facla, 1982 .
58. Goovaerts H., Ros H., van den Akker T.J., Schneider H. (1976) "*A digital QRS detector based on the principle of countour limiting*", IEEE-Trans, BME-23, No.2, 154-160.
59. Gradwohl J., Pottala E., Horton M., Bailey J. (1989) "*Comparison of two methods for removing baseline wander in the ECG*", Computers in Cardiology, IEEE-Press, 493-496.

60. Gritzali F. (1988) "*Towards a generalised scheme for QRS detection in ECG Waveforms*", Signal Processing, Vol.15, 183-192.
61. Gritzali F., Frangakis G., Papaconstantinou G. (1989) "*Detection of the P and T waves in an ECG*", Computers and Biomedical Research, Vol.22, 83-91.
62. Hamilton P.S., Tompkins W.J. (1991) "*Compression of the ambulatory ECG by average beat subtraction and residual, differencing*", IEEE Transaction on Biomedical Engineering, Vol. 38, No. 3, 253-259.
63. Hamilton P.S., Tompkins W.J. (1986) "*Quantitative investigation of QRS detection rules using the MIT/BIH arrhythmia database*", IEEE Transaction on Biomedical Engineering, Vol. BME-33, No. 12, 1157-1165.
64. Holsinger W., Kempner K., Miller M. (1971) "*A QRS preprocessor based on digital differentiation*", IEEE-Trans, BME-18, No.3, 212-216.
65. Horowitz S.L. (1977) "*Peak recognition in waveform*" in Syntactic Pattern Recognition and Application, Fu K.S. (ed.), Prentice-Hall, Englewood Cliffs, New Jersey-USA, 31049.
66. Horne C. J., Zhang K. J., Propst J. (1984) "*ST segment evaluation by discrete cosine and Fourier transforms*", Computers in Cardiology, IEEE Computers Society, 265-268.
67. Huhta J., Webster J. (1973) "*60-Hz interference in electrocardiography*", IEEE-Trans, BME-20, 91-101.
68. Hsiung H., Cheng C., Jan G. (1989) "*A real-time QRS detection and delineation algorithm based on the inflection angle of the ECG waveform*", Proceedings of the XI IEEE-EMBS Conference, Seattle-USA, 138-139.
69. Jalaaliddine S., Hutchens C. (1987) "*Ambulatory ECG wave detection for automated analysis: A review*", Instrument society of America Transaction, Vol.26, No.4, 33-43.
70. Jalaaliddine S., Hutchens C., Strattan R., Coberly W., (1990) "*ECG Data Compression techniques; a unified approach*" IEEE-Trans, BME-37, nr 4, 329-343
71. Jane R., Rix H. (1991) "*Alignment methods for averaging of high-resolution cardiac signals: a comparative study of performance*", IEEE Trans BME, 38, No.6, 571-579
72. Jurca T., Stoiciu D. *Instrumentație și măsurare*, Editura de Vest, 1996.
73. Kao T, Yu B.C., Chen C. (1989) "*ECG data compression for digital transmission on telephone lines*", Journal of Clinical Engineering, Vol 14, nr 5, 431-438.

74. Labonski P., Hediger T., Passamante A. (1987) "*Non-linear Autoregressive Processing of Noisy Signals*", Digital Signal Processing, Cappellini V., Constantinides A.G. (eds), 200-204.
75. Lage S.G., Gutierrez M.A., Nicoletis M.A.L., Furuie S. (1988) "*A bedside computerized system for monitoring and processing biological signals in intensive care units*", Computers in Cardiology, IEEE- Press 561-564
76. Laguna P., Jane R., Meste O., Poon P.W., Caminal P., Rix H., Thakor N.V. (1992) "*Adaptive Filter for Event-Related Bioelectric Signals Using an Impulse Correlated Reference Input : Comparison with Signals Averaging Techniques*", IEEE Trans, BME, Vol. 39, No. 10, 1032-1043.
77. Laguna P., Thakor N., Caminal P., Jane R., (1990) "*Low-pass differentiators for biological signals with known spectra: applications to ECG signal processing*", IEEE-Trans, BME-37, No.11, 420425.
78. Lachiver G., Eichner J., Bessette F., Seufert W. (1987) "*An algorithm for ECG data compression using spline function*", Computers in Cardiology, IEEE press, 575-578.
79. Lamberti C., Coccia P. (1989) "*ECG data compression for ambulatory device*", Computers in Cardiology, IEEE press, 171-174
80. Lee H., Cheng Q., Thakor N. (1987) "*ECG waveform analysis by significant point extraction: data reduction*", Computers and Biomedical Research, Vol.20, 410-427.
81. Levcov C., Michov G., Ivanov R., Daskalov I. (1984) "*Subtraction of 50-Hz interference from the electrocardiogram*", Medical and Biological Engineering and Computing, Vol.22, 371-373.
82. Ligtenberg A., Kunt M. (1983) "*A robust-digital QRS detection algorithm for arrhythmia monitoring*", Computer and Biomedical Research, Vol.16, 273-286.
83. Lin K., Chang W. (1987) "*ECG signal analysis by linear predictive method*", Proceedings of the IX IEEE-EMBS Annual Conference, Boston-USA, 557-558.
84. Lin K., Chang W.H. (1989) "*QRS feature extraction using linear prediction*", IEEE Trans, BME, Vol. 36, No. 10, 1050-1055.
85. Lynn P. A., (1971), "*Recursive digital filters for biological signals*", Med. and Biol.Eng. 9, 37-43
86. McClellan J. (1988) "*Parametric signal modeling*", in Advanced Topics in signal rocessing, Lim J. and Openheim A. (Eds), Prentice-Hall, New Jersey-USA.
87. McHenry P. L., Stowe D. E., Lancaster M. C. (1968) "*Computer quantitation of the ST-segment response during maximal treadmill exercise*". Circulation, 38, 691-695,

88. McManus C.D., Teppner U., Dietrich N. (1985) "*Estimation and removal of baseline drift in the electrocardiogram*", Computers and Biomedical Research, vol 18, 1-9.
89. Manițu I., Mișu P.I., Bera L.(1995) "*ECG Monitoring in Efforts Tests*", Proceeding of the National Conference On Preventive Cardiology, may, 1995, Sibiu.
90. Mark R.G., Moody G.B. (1988) "*Automated arrhythmia analysis*", Encyclopedia of medical devices and instrumentation, Webster J. (ed), John Wiley and Sons, New York-Usa, 120-130.
91. Merri M., Farden D.C., Mottley J.G. (1990) "*Sampling frequency of the electrocardiogram for spectral analysis of the heart rate variability*", IEEE Transaction on Biomedical Engineering, Vol. 37, No. 1, 99-105.
92. Mișu P.I., Breazu M.(1992) "*About an IBM-PC coupled ADC solution*" Acta universitatis Cibiensis. Vol. X, 1992, Technical Series, Electronics.
93. Mișu P.I., Raica M.(1992) "*Programierbare Logikbausteine. PAL und GAL in der Praxis*" Acta universitatis Cibiensis. Vol. X, Technical Series, Electronics.
94. Mișu P.I., Breazu M.,(1995) "*Filtre numerique pour une voie d'aquisition de signal ECG*", Acta universitatis Cibiensis. Vol.XIV, 1995, Technical Series, Electronics.
95. Mișu P.I. Toma E., Breazu M.(1992) "*Program for Digital Filters Design*" Acta universitatis Cibiensis. Vol. X, Technical Series, Electronics.
96. Mișu P.I. (1993), "*Procesarea numerică a semnalului ECG*" Referat 1, în cadrul pregătirii pentru doctorat; decembrie 1993
97. Mișu P.I. (1994), "*Monitorizarea în Unități de Terapie Intensivă*" Referat 2, în cadrul pregătirii pentru doctorat; decembrie 1994
98. Mișu P.I., Bera L.(1995) "*Study in QRS Algorithms Implementation*", Proceeding of the National Conference on Preventive Cardiology, may, 1995, Sibiu
99. Mișu P.I.(1995), "*Data Aquisition and ECG Signal Processing*", Proceeding of the National Conference on Preventive Cardiology, may , 1995, Sibiu.
100. Moody E.B.,Tzanakou E.M., Chokroverty S. (1989) "*An adaptive approach to spectral analysis of pattern-reversal visual evoked potentials*", IEEE Trans. BME, Vol 36, No. 4, 439-447.
101. Moody G., Mark R., Olson W., Peterson S., Schluter P., Walters J. (1979) "*Real-time ambulatory arrhythmia analysis with a microcomputer*", Computers in Cardiology, IEEE-Press, 57-62.

102. Mora F.A., Passariello G., Carrault G., Le Pichon J.P. (1993) "*Intelligent Patient Monitoring and Management Systems : A Review*", IEEE-Engineering in Medicine and Biology, dec., 23-33.
103. Murthy I., Rangaraj M. (1979) "*New concepts for PCV detection*", IEEE-Trans, Bme-26, No.7, 409-415.
104. Nygard M., Sornmo L. (1982) "*A QRS delineation algorithm with low sensitivity to noise and morphology changes*", Computers in Cardiology, IEEE-Press, 347-350.
105. Nygard M., Sornmo L. (1983) "*Delineation of the QRS complex using the envelope of the ECG*", Medical and Biological Engineering and Computing, Vol.21, 538-547.
106. Okada M. (1979) "*A digital filter for the QRS complex detection*", IEEE-Trans, BME-26, 700-703.
107. Pahlm O., Sornmo L. (1984) "*Software QRS detection in ambulatory monitoring; A review*", Medical and Biological Engineering and Computing, Vol 22, 289-297.
108. Pahlm O., Sornmo L. (1987) "*Data processing of exercise ECG's*", IEEE-Trans, BME-34, No. 2, 158-165.
109. Pan J., Tompkins W. (1985) "*Real-time QRS detection algorithm*", IEEE-Trans, BME-32, No.3, 230-236.
110. Papakonstantinou G., Gritzali F. (1981) "*Syntactic filtering of ECG waveforms*", Computers and Biomedical Research, Vol 14, 158-167.
111. Papakonstantinou G., Skordalakis E., Gritzali F. (1986) "*An attribute grammar for QRS detection*", Pattern Recognition, 19, 197-303
112. Patrick S., Hamilton, (1996). "*A Comparison of Adaptive and Nonadaptive Filters for Reduction of Power Line Interference in the ECG*", IEEE TBME, 43, No.1, 105-108,.
113. Perriyalwar S., Marble A.E., Nugent S.T., Swingler D.N. (1990) "*An adaptive "Wiener" filter for estimating the time-derivative of the left ventricular pressure signal*", IEEE Trans. BME, Vol 37, No. 4, 417-425.
114. Ping Z.L., Lin Z. (1992) "*Preliminary evaluation in clinical applications of frequency spectrum electrocardiogram*", IEEE Trans. BME, 544-576.
115. Policec A., Gligor T.D., Ciocloada Gh.-*Electronică medicală*, Ed. Dacia, 1983
116. Policec A., Mișu P.I., Voiniciuc C. "*QRS Detection Algorithm in Real Time*", Acta universitatis Cibiniensis. Vol. XV, 1996, Technical Series, Electronics.
117. Pop E. ș.a. - *Metode în prelucrare numerică a semnalelor*, Ed. Facla, 1989 .

118. Press W.H., Flannery B. P., Teukolsky S. A., Vetterling W. T. *Numerical Recipes in PASCAL*. Cambridge University Press, 1990.
119. Principe J.C., Smith J.R. (1986) "*Design and Implementation of Linear Phase FIR Filters for Biological Signal Processing*", IEEE Transaction on Biomedical Engineering, Vol. BME-33, 550-559.
120. Radu O. - *Filtre numerice*, E.T., 1979 .
121. Rappaport S., Gillick G., Moody G., Mark R. (1982) "*QRS morphology classification: quantitative evaluation of different strategies*", Computers in Cardiology, IEEE-press, 33-38.
122. Reddy B.R.S., Murty A.S.N. (1986) "*ECG data compression using Fourier descriptors*", IEEE-Trans, BME, vol 33, 438-434
123. Ritter J.A., Thomas L.J., Ripley K.L. (1981) "*A micro computer system for clinical arrhythmia monitoring*", Computers in Cardiology, IEEE-press, 79-84
124. Rompelman O., Ross H.H., (1986) "*Coherent averaging technique: a tutorial review*" J. Biomed. Eng., 58, 610-643.
125. Ruiz R., Hernandez C., Mira J. (1984) "*Method for mapping cardiac arrhythmia in real time using microprocessor based systems*" Medical and Biol. Eng. and Computing, vol 22, 160-167
126. Ruttimann U.E., Pipberger H.V. (1979) "*Compression of the ECG by prediction or interpolation and entropy encoding*", IEEE-Trans, BME- 26, Nr 11, 613-623.
127. Schmitt J. M., Al-Fadel, H. (1989) "*Design of medical instrumentation for application in developing countries*", Journal of Clinical Engineering. Vol. 14, 299-306
128. Shaw G.R., Savard P. (1995) "*On the detection of QRS variation in the ECG*", IEEE-Trans, BME, vol 42, Nr 7, 736-741.
129. Siregar P., Coatrieux J.L., Mabo P. (1993) "*How Can Deep Knowledge Be Used in CCU Monitoring*", IEEE Engineering in Medicine and Biology, 92-99.
130. Siregar P., di Renzo M. (1990) *Final Report on Knowledge Representation and Reasoning Methods*. KISS, AIM Program, A1006-1.2.C.F.R.-900930
131. Skordalakis E. (1986) "*Recognition of the shape of the ST-segment in the ECG waveform*", IEEE-Trans, BME33, No.10, 972-974.
132. Skordalakis E., Trahanias P. (1987) "*Primitive pattern selection and extraction in ECG waveform*", International Conference on Pattern Recognition (1986), IEEE-computer society, 380-382.
133. Skordalakis E., Papaconstantinou G. (1984) "*Toward an attribute grammar for description of ECG waveform*", International Conference on Pattern Recognition(1984), IEEE-computer society, 1383-1385.

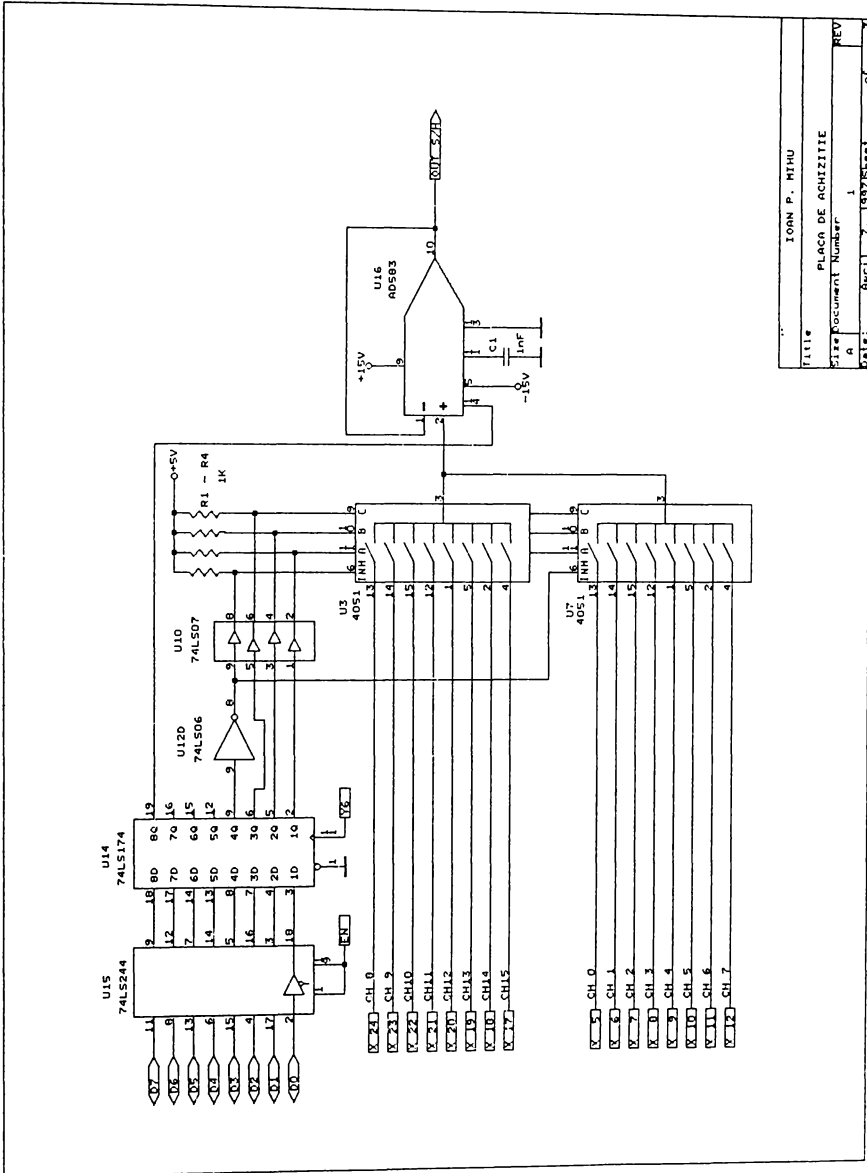
134. Smith N.C., Platt J.S. (1989) "*An ECG compression algorithm for full disclosure in a solid state real-time Holter monitor*", Computers in Cardiology, IEEE press, 569-572.
135. Sornmo L., Pahlm O. (1984) "*Evaluation of noise immunity for algorithms in exercise ECG analysis*", Computers in Cardiology, IEEE-press, 349-352.
136. Sornmo L., Pahlm O. (1985) "*Adaptive QRS detection :a study of performance*", IEEE Trans. BME, Vol BME-32, No. 6, 392-401.
137. Stearns S. (1988) "*Fundamentals of adaptive signal processing*", in Advanced topics in signal processing, Lim J., Oppenheim A. (Eds.), Prentice-Hall, New Jersey-USA.
138. Sukuvaara T., Sydanmaa M., Niemen H., Heikela Koski E.M.J.(1993) "*Object Oriented Implementation of an Architecture for Patient Monitoring*", IEEE Engineering in Medicine and Biology, Nr 12, 69-81 .
139. Taylor K. (1979) "*A wide dinamic range R-wave trigger*", Journal of Clinical Engineering, vol 4, Nr. 2, 131-134.
140. Thakor N., Webster J., Tompkins W. (1984) "*Estimation of QRS complex power for desing of a QRS filter*", IEEE-Trans, BME-31, No.12, 770-778.
141. Thakor N.V., Zhu Y.S. (1991) "*Applications of adaptive filtering to ECG analysis : noise cancellation and arrhithmia detection*", IEEE Trans. BME, Vol 38, No. 8, 785-793.
142. Thull B., Popp H., Rau G. (1992) "*Man-Machine Interaction In Critical Care Settings*", IEEE Engineering in Medicine and Biology, Nr 12, 42-49.
143. Toma L. , "*Sisteme de achiziție și prelucrare numerică a semnalelor*", Editura de Vest, 1996, Timișoara.
144. Tompkins W., Webster J., (editors)(1981) *Design of microcomputer based medical instrumentation*, Prentice-Hall, Englewood Cliffs, New Jersey-USA.
145. Tompkins W. (1988), "*Ambulatory monitoring*", Encyclopedia of medical devices and instrumentation, Webster J. (ed), Lohn Wiley and Sons, New York, 20-28.
146. Trahanias P., Skordalakis E., Papaconstantinou G. (1989) "*A syntactic method for the classification of the QRS patterns*", Pattern Recognition Letters, Vol. 9, No. 1, 13-18.
147. Trahanias P. (1993) "*An approach to QRS complex detection using mathematical morphology*", IEEE Trans. BME, Vol 40, No. 2, 201-205.
148. Tremblay G., LeBlanc A.R.. "*Near-Optimal Signal Preprocessor for Positive Cardiac Arrhithmia Identification*", IEEE Trans. BME, Vol BME-32, No. 2, February 1985, 141-151.

149. Udupa L.K., Murty I.S.N. (1980) "Syntactic approach to ECG rhythm analysis", IEEE Trans. BME, Vol BME-27, No. 7, 370-375.
150. Usui S., Amidror I. (1982) "Digital low-pass differentiation for biological signal processing", IEEE-Trans, BME-29, No.10, 686-693.
151. Van den Akker T.J., Ros H., Koeleman A., Dekker C. (1982), "An on-line method for reliable detection of waveforms and subsequent estimation of events in physiological signals", Computers and Biomedical Research, 405-417.
152. Voiniciuc C., Mihu P.I., "Modelling of QRS Complex" Proceeding of the 5th International Conference on Optimization of Electric and Electronic Equipments, Braşov, may15-17 1996
153. Voiniciuc C., Mihu P.I., "Bipolar Electrodes Potential Measurements" Proceeding of the International Conference on Optimisation of Electric and Electronic Equipments, Braşov, may15-17 1996
154. Voiniciuc C., Mihu P.I., "Contribution to QRS model", Proceeding of the International Conference on Applied and Theoretical Electricity, Craiova juin 6-7, 1996, pg.205-210.
155. Voiniciuc C., Mihu P.I., Duţă M. "Variants bipolar electrodes", Proceeding of the International Conference on Applied and Theoretical Electricity, Craiova juin 6-7, 1996, pg.211-216.
156. Voiniciuc C., Mihu P.I. Creţu M., Coman V., "ECG data acquisition board " Acta universitatis Cibiniensis. Vol. XV, 1996, Technical Series, Electronics.
157. Voiniciuc C., Mihu P.I. Creţu M., Niţă L., "ECG Acquisition and numerical processing " Acta universitatis Cibiniensis. Vol. XV, 1996, Technical Series, Electronics.
158. Voiniciuc C., Mihu P.I., Coman V., Verdeanu R. "Sistem de achiziție a semnalelor bioelectrice", Sesiunea Națională de Comunicări Științifice-Academia Aviației și Apărării Antiaeriene " , H. Coandă", Braşov, 21-22 noe. 1996.
159. Voiniciuc C., Creţu M., Mihu P.I., Voiniciuc T. "Prelucrarea numerică a ECG prin metode software", Sesiunea Națională de Comunicări Științifice-Academia Aviației și Apărării Antiaeriene", H. Coandă", Braşov, 21-22 noe. 1996.
160. Weisner S., Tompkins W., Tompkins Bonnie(1982) "A compact, microprocessor-based ECG ST-segment analyzer for operating room", IEEE-Trans, BME-29, No. 9, 642-649.
161. Winter B., Webster J.G. (1983) "Driven right leg circuit design", IEEE-Trans, BME-30, No. 1, 62-66.
162. Witten I.H., Neal R., Cleary J. (1987) "Arithmetic coding for data compression", Communication of the ACM, vol 30, Nr 6, 520-540.

163. Womble E., Halliday J.S., Mittre S.K. (1977) "*Data compression for storing and transmitting ECG's / VCG's*", IEEE Proceedings, Vol 65, Nr5, 702-706
164. Yu B., Liu C., Lee M., Chen C., Chiang B. (1985) "*A non linear digital filter for cardiac QRS complex detection*", Journal of Clinical Engineering, Vol.10, Nr.3, 193-201.

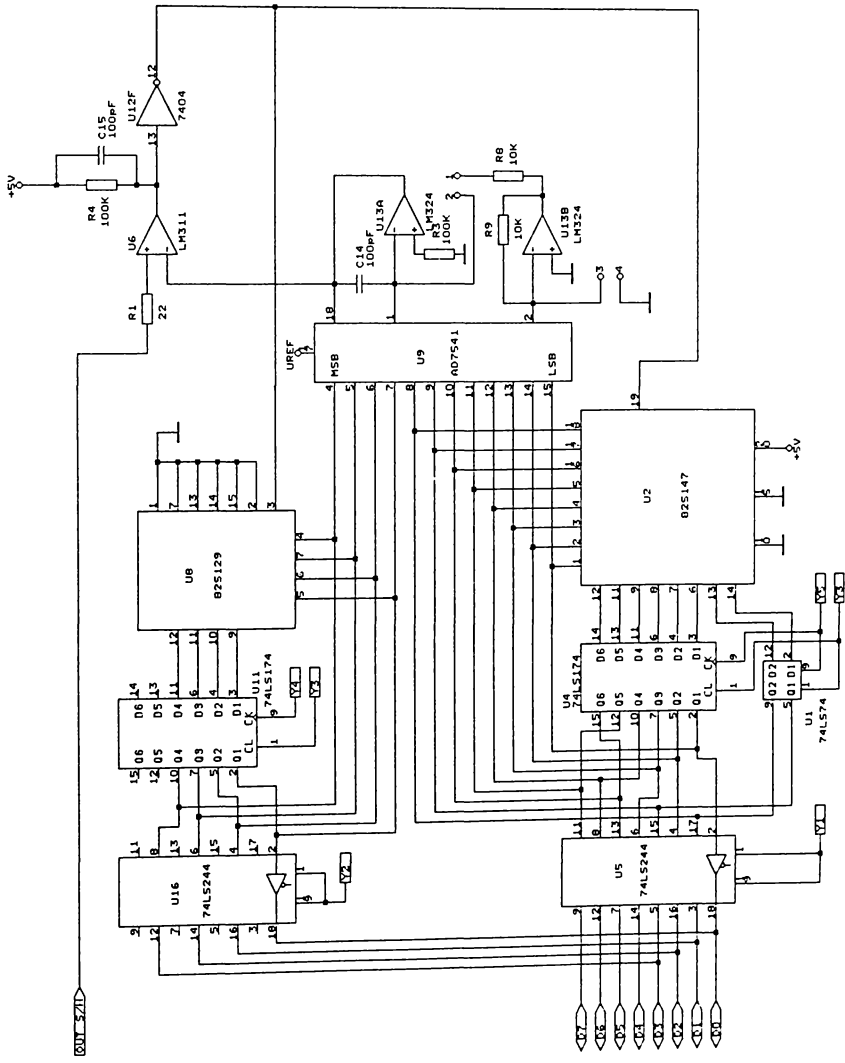
ANEXA A1

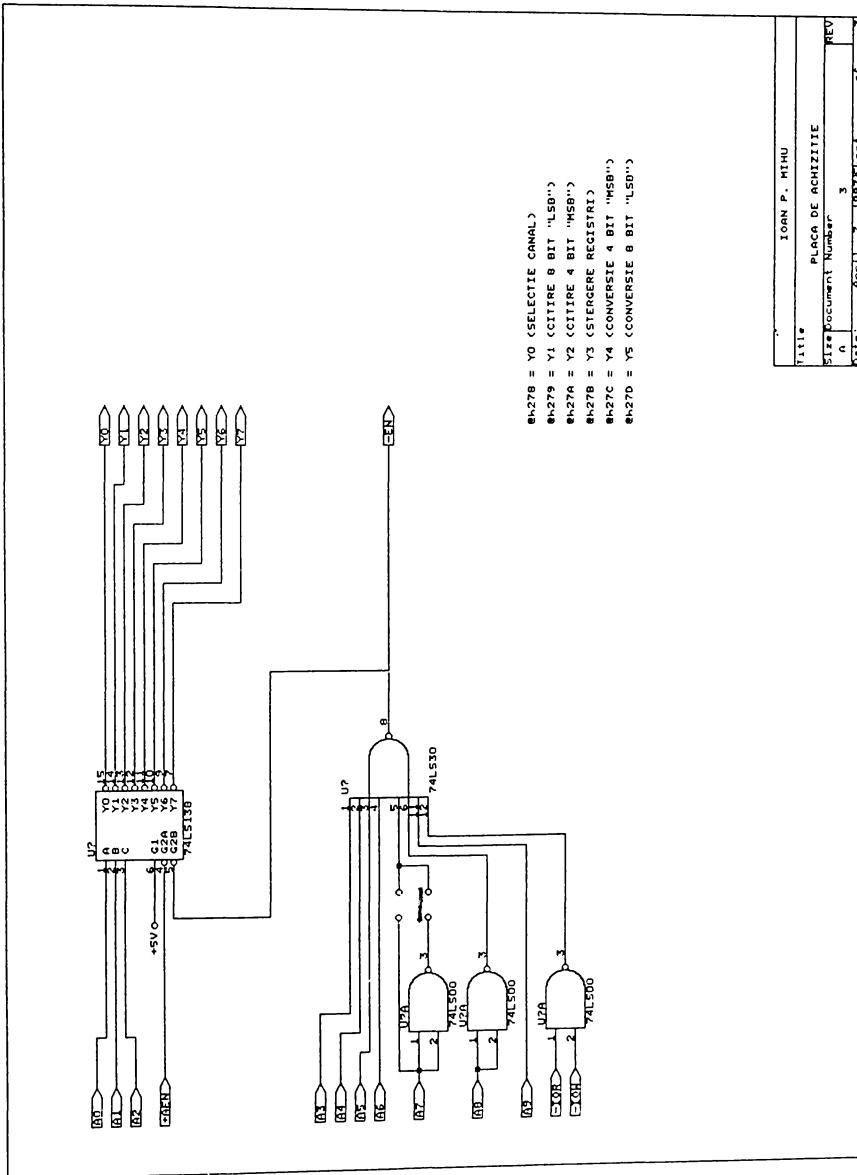
SCHEMA ELECTRICĂ A PLĂCII DE ACHIZIȚIE



IOHN P. MIHU	
PLACA DE ACHIZITIE	
Titlu	REV
Numar Document	1
Revizii	0
DATA	17.11.1997
Page	of

TITLU	IOAN P. MIHU
PLACA DE ACHEZITIE	
Serie Document Number	2
Date	April 7, 1997/Strat





IOAN P. MINU	
PLACA DE ACHEZITIE	
Title	
Size	Document Number 3
Date	Apr 17, 1997 Filesk
REV	OF

ANEXA A2

**PROGRAM
PENTRU TRASAREA CARACTERISTICILOR DE FRECVENȚĂ
ALE UNOR ALGORITMI DE PROCESARE ECG**

```

program CAR_FREQV;
uses Dos,crt,Graph,Depot,Afis2fis;

CONST
  NdatMin=-180; {necesar ca tabloul sa inceapa "inainte" de zero pt prelucrarile recursive.}
  Ndatmax=1100; {nr de esantioane ale fisierelor de date}
  Nfis=40;      {nr maxim de fisiere de date (.DAT si .TRT), ce pot fi cautate}
  Nmin=-40; Nmax=40;
  fond =6;
  fondHoriz=9;
  fondVert=9;
  fondSel=4;

TYPE
  Viewporttype =record gX0,gY0,gX1,gY1:integer;clip:boolean; end;
  vectcauta    =array[1..Nfis] of string;
  vect         =array[1..Ndatmax] of real;
  mesaj        =string[12];
  t_infodate   =array[1..10] of mesaj;
  date         =array[NdatMin..NdatMax] of real;

VAR
  infodate      :t_infodate;
  infofis      :file of t_infodate;
  gX0, gX1, gY0, gY1,gX,gY, {coordonate dreptunghi.}
  nr_elem      :integer; {numarul de elem. ale fisierului}
  point_mes,point_fis,point_origx ,
  zmare,zmic,
  N,i,j,k,Nfisgas :integer;
  Y ,Ymin,Ymax,Ydatmin,Ydatmax.
  Ytrtmin,ytrtmax,
  a,b,r,s ,ln10   :real;
  nume_fis ,
  n_fis,          {numele fisierului de date de afisat}
  numecauta,
  nume_fisdat,nume_fistrt,prenume :string[20];
  datefis       :SearchRec;
  trat.wind,clip,ciclu :boolean;
  titre,titlu    :^vectcauta;
  Y1mes,Y2mes,
  Ymes          :^vect;
  car           :char;
  fer1,fer2,fer3,fer4,
  fer5,fer6,fer : ViewPortType;

```

```

fis1                :file of real;  { fisierul de amplitudine de afisat}
fis2                :file of real;  { fisierul de faza de afisat}
Imax,  {cite esantioane din "sin" vor fi trecute prin filtru}
Kmax    :integer;    {numarul de puncte al car. de frec.}
ImaxYneg,ImaxYpoz,ImaxY,
ImaxXneg,ImaxXpoz,ImaxX ,
Mod_Yneg,Mod_Ypoz,
Mod_Xneg,Mod_Xpoz,
dY                :real;
Xdate,  {intrare filtru}
Ydate  {iesire filtru}    :^date;
Mod_X,
Mod_Y,  { tablou pentru amplitudine }
Defaz   { tablou pentru faza }           :date;
Y2,z2, Y1,z1  {var. intermediare pt filtre }    :^date;
}-----}
procedure SETFER;
begin
fer1.gX0:=1;fer1.gY0:=10;fer1.gX1:=639;fer1.gY1:=240;fer1.clip:=true;
fer2.gX0:=1;fer2.gY0:=265;fer2.gX1:=638;fer2.gY1:=435;fer2.clip:=true;
fer3.gX0:=1;fer3.gY0:=220;fer3.gX1:=638;fer3.gY1:=390;fer3.clip:=true;
fer4.gX0:=1;fer4.gY0:=220;fer4.gX1:=638;fer4.gY1:=390;fer4.clip:=true;
fer5.gX0:=55;   fer5.gY0:=10;   fer5.gX1:=630;   fer5.gY1:=180;
fer6.gX0:=55;   fer6.gY0:=220;   fer6.gX1:=630;   fer6.gY1:=390;
end;
}-----}
procedure SETFER1(fer1,fer2:ViewPortType;var gX0,gY0,gX1,gY1:integer;var clip:boolean);
begin
gX0:=fer1.gX0;gY0:=fer1.gY0;gX1:=fer1.gX1;gY1:=fer1.gY1;clip:=fer1.clip;
SetColor(12);
end;
}-----}
procedure SETFER2(fer1,fer2:ViewPortType; var gX0,gY0,gX1,gY1:integer;var clip:boolean);
begin
gX0:=fer2.gX0;gY0:=fer2.gY0;gX1:=fer2.gX1;gY1:=fer2.gY1;clip:=fer2.clip;
SetColor(12);
end;
}-----}
procedure DesCadre(var gX0,gY0,gX1,gY1:integer);
begin
  SetLineStyle(DottedLn,0,NormWidth);
  SetColor(3);
  Rectangle(gX0,gY0,gX1,gY1+2);
end;
}-----}
procedure INTRODPARAM1(fer1,fer2:viewPortType;gX0,gY0,gX1,gY1:integer );
var j :integer;
    TempStr:string[7];
begin
  SetViewPort(0,0,GetMaxX,GetMaxY,true);
  clearViewPort;
  SETFER;
  SETFER1(fer1,fer2,gX0,gY0,gX1,gY1,clip);
  DESCADRE(gX0,gY0,gX1,gY1);    {cadre pentru fereastra "I" }
  OutTextxy(10,20,' Acest program va permite reprezentarea grafica a caracteristicilor de ');
  OutTextxy(10,40,' frecventa si faza ale filtrului numeric pe care l-ati integrat in programul ');
  OutTextxy(10,60,' sursa din fisierul CAR_FREQ.PAS ');
  OutTextxy(10,80,' Constructia graficelor se face punct cu punct in felul urmatoar: se trece ');

```

```

OutTextxy(10,100,' un numar suficient de mare de esantioane Imax (pt. stabilirea unui regim
per- ');
OutTextxy(10,120,' manent) dintr-un semnal sinusoidal (SIN) de amplitudine unitara. Din
ultimele ');
OutTextxy(10,140,' doua esantioane de la iesirea filtrului se reconstituie matematic
amplitudinea ');
OutTextxy(10,160,' si faza semnalului sinusoidal ce apare la iesirea filtrului. ');
OutTextxy(10,180,' Se creste apoi frecventa semnalului SIN de la intrare cu dF si procesul se
reia ');
OutTextxy(10,200,' pina la epuizarea intrgului domeniu de frecventa ales. ');
OutTextxy(10,220,' Se poate alege pentru grafic o axa de frecventa liniara sau logaritmica. ');
OutTextxy(10,300,' Urmatoarele date ale construirii graficului dF.Fmin.Fmax.Fe se pot
modifica ');
OutTextxy(10,320,' doar in programul sursa. ');
OutTextxy(10,340,' Celelalte date necesare reprezentarilor grafice se vor introduce ');
OutTextxy(10,360,' conform instructiunilor din caseta urmatoare. ');
OutTextxy(10,420,' Tastati ENTER pentru a continua ');
readln;
SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);
setviewport(gX0+1,gY0+1,gX1-1,gY1-1,clip);
clearViewPort;
SetViewPort(0,0,GetMaxX,GetMaxY,true);
DESCADRE(gX0,gY0,gX1,gY1); {cadrul pentru fereastra "1" }
setlinestyle(solidln,0,normwidth);
SetColor(6);
OutTextxy(1,280,' Fmin=dF=0.05Hz; Fmax= 200 Hz ');
OutTextxy(1,295,' Frecventa de esantionare Fe= 200 Hz');
Fe:=200;
OutTextxy(1,310,' Ecartul intre punctele graficului dF= 0.05 Hz);
dF:=0.01;
OutTextxy(1,325,'Introduceti nr de esant din SIN "trecute" Imax= ');
OutTextxy(1,340,'(trebuie sa fie mai mare decat ordinul filtrului)');
INTRNUMGRAPH(386,325,1.1,r);
Imax:=round(r);
SetViewPort(0,0,gX1,gY1,true);{pregatesc afisarea filtrelor}
SetViewPort(0,0,GetMaxX,GetMaxY,true);
SETFER1(fer1,fer2,gX0,gY0,gX1,gY1,clip);
clearviewport;
gY1:=gY1+120;
DESCADRE(gX0,gY0,gX1,gY1); {cadrul pentru fereastra "1" }
OutTextxy(10,20,'FTS 1: ydate^[i]:=Xdate^[i]-Xdate^[i-1]; ');
OutTextxy(10,35,' 2: ydate^[i]:=Xdate^[i]-Xdate^[i-2];');
OutTextxy(10,50,' 3: ydate^[i]:=(-Xdate^[i-3]-2*Xdate^[i-2]+2*Xdate^[i-1]+Xdate^[i])/8. ');
OutTextxy(10,65,' 4: ydate^[i]:=(Xdate^[i-3]-8*Xdate^[i-2]+8*Xdate^[i-1]-Xdate^[i])/12. ');
OutTextxy(10,80,' 5: ydate^[i]:=(-3*Xdate^[i-6]-2*Xdate^[i-5] -Xdate^[i-4] );
+Xdate^[i-2]+2*Xdate^[i-1]+3*Xdate^[i])/12. ');
OutTextxy(10,110,' 6: ydate^[i]:=(22*Xdate^[i-6]-67*Xdate^[i-5]-58*Xdate^[i-4] );
+58*Xdate^[i-2]+67*Xdate^[i-1]-22*Xdate^[i])/252. ');
OutTextxy(10,125,' 7: ydate^[i]:=(Xdate^[i-3]+Xdate^[i-2]+Xdate^[i-1]+Xdate^[i])/4. ');
OutTextxy(10,155,' 8: ydate^[i]:=(Xdate^[i-2]+2*Xdate^[i-1]+Xdate^[i])/4. ');
OutTextxy(10,170,' 9: ydate^[i]:=2*ydate^[i-1]-ydate^[i-2]+Xdate^[i]-2*Xdate^[i-
6]+Xdate^[i-12]; ');
OutTextxy(10,185,' 10: ydate^[i]:=ydate^[i-256]+Xdate^[i]-Xdate^[i-1];');
OutTextxy(10,350,' xx: ...un nou filtru proiectat se va introduce aici in progr sursa ');
SetColor(6);
OutTextxy(1,370,' Introduceti nr. filtrului Nr_filtru= ');
OutTextxy(1,385,' ( Este nr.filtrului din programul sursa ');
INTRNUMGRAPH(386,370,1.1,r);

```

```

Nr_filtru:=round(r);
OutTextxy(1,410,'          Introduceti tipul de grafic : ');
OutTextxy(1,425,'          (natural = "0" , logaritmic = "1") ');
INTRNUMGRAPH(386,410,1,1,r);
clearViewPort;
OutTextxy(1,320,'          Asteptati sfirsitul calculelor ! ');
if r=0 then begin
    tip_grafic:=true;
    Fmax:=120;end
else begin
    tip_grafic:=false;
    Fmax:=990;end;
end;
{-----}
procedure Calc(i:integer);
var p,m:integer;
begin
xdate^[i]:=sin(2*pi*i*f / Fe);
case nr_filtru of
{***** FTS*****}
1:  ydate^[i]:=-Xdate^[i]+2*Xdate^[i-4]-Xdate^[i-8]; {3.42 din teza}
2:  ydate^[i]:=Xdate^[i]-4*Xdate^[i-4]+6*Xdate^[i-8]-4*Xdate^[i-12]+Xdate^[i-16];
    {3.44 din teza}
3:  ydate^[i]:=(-Xdate^[i-3]-2*Xdate^[i-2]+2*Xdate^[i-1]+Xdate^[i])/8;
4:  ydate^[i]:=(Xdate^[i-3]-8*Xdate^[i-2]+8*Xdate^[i-1]-Xdate^[i])/12;
5:  ydate^[i]:=(-3*Xdate^[i-6]-2*Xdate^[i-5] -Xdate^[i-4]
    +Xdate^[i-2]+2*Xdate^[i-1]+3*Xdate^[i])/12;
6:  ydate^[i]:=(22*Xdate^[i-6]-67*Xdate^[i-5]-58*Xdate^[i-4]
    +58*Xdate^[i-2]+67*Xdate^[i-1]-22*Xdate^[i])/252;
{***** FTJ *****}
7:  ydate^[i]:=(Xdate^[i-2]+2*Xdate^[i-1]+Xdate^[i])/4;
8:  ydate^[i]:=(Xdate^[i-3]+Xdate^[i-2]+Xdate^[i-1]+Xdate^[i])/4;
9:  ydate^[i]:=2*ydate^[i-1]-ydate^[i-2]+Xdate^[i]-2*Xdate^[i-6]+Xdate^[i-12];
10: ydate^[i]:=ydate^[i-1]+Xdate^[i]-2*Xdate^[i-20]+Xdate^[i-40];
11: begin
    y1^[i]:=2*Y1^[i-1]-Y1^[i-2]+Xdate^[i]-2*Xdate^[i-6]+Xdate^[i-12];
    ydate^[i]:=(Xdate^[i]-Y1^[i])/256;
    end;
100: begin
    {***** Aici se va insera noul algoritm de filtru*****}
    end;
end;
end;
{-----}
BEGIN
clrscr;
IdentInitGraph;
SETFER;
SETFER1(fer1,fer2,gX0,gY0,gX1,gY1,clip);
DESCADRE(gX0,gY0,gX1,gY1); {cadrul pentru fereastr "1" }
outtextxy(10,100,'measaj1');
SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);
DESCADRE(gX0,gY0,gX1,gY1); {cadrul pentru fereastr "1" }
INTRODPARAM1(fer1,fer2,gX0,gY0,gX1,gY1 );
f:=dF;
if (MaxAvail < 6*SizeOf(date))
    then writeln("NOT ENOUGH MEMORY")
    else begin GetMem(Xdate,SizeOf(date)); GetMem(Ydate,SizeOf(date));

```

```

        GetMem(Y1,SizeOf(date)); GetMem(Y2,SizeOf(date)); GetMem(Z1,SizeOf(date));
        GetMem(Z2,SizeOf(date)); end;
for i:=Ndatmin to Ndatmax do begin
  Xdate^[i]:=0; Ydate^[i]:=0; Y1^[i]:=0; Y2^[i]:=0; Z1^[i]:=0; Z2^[i]:=0; end;
ln10:=ln(10);
k:=0;
c:=Ngarfmax*ln10/ln(1000/dF);{const pt ca extremitatea graficului sa fie 1000Hz}
repeat
  i:=0;
  repeat
    Calc(i); inc(i);
  until i=Imax;
  a:=2*pi*(i-1)*f/Fe;
  b:=2*pi*(i-2)*f/Fe;
  if abs (Ydate^[i-2])<0.00000000001 then
    begin {pt a evita impartirea cu zero}
      Mod_Y[k]:=Mod_Y[k-1];
      Defaz[k]:=Defaz[k-1]; end
    else begin
      r:=Ydate^[i-1] / Ydate^[i-2];
      if abs(r*cos(b)-cos(a))<0.00000000001 then s:=1
      else S:=sin(a + ArcTan((sin(a)-r*sin(b))/(r*cos(b)-cos(a)))));
      if abs(s)<0.000000001 then
        Mod_Y[k]:=Mod_Y[k-1] {pt evitarea impartirii cu nr ff mici !!!}
        else begin
          Mod_Y[k]:=abs(Ydate^[i-1]) /abs( s);
          if tip_grafic=false then Mod_Y[k]:=20*ln(Mod_Y[k])/ln10; end;
          Defaz[k]:=ArcTan((sin(a)-r*sin(b))/(r*cos(b)-cos(a)));
        end;
      inc(k);
      if tip_grafic then begin f:= k*0.208; end
      else f:=dF*exp(k*ln10/c);
until f>Fmax;
FreeMem(Xdate,SizeOf(date)); FreeMem(Ydate,SizeOf(date)); FreeMem(Y1,SizeOf(date));
FreeMem(Y2,SizeOf(date)); FreeMem(Z1,SizeOf(date)); FreeMem(Z2,SizeOf(date));
Kmax:=k;
k:=-1;
n_fis:='Ampli.dat';
assign (fis1,"+n_fis);
rewrite(fis1); {deschiderea fisierului de citit}
n_fis:='Faza.dat';
assign (fis2,"+n_fis);
rewrite(fis2); {deschiderea fisierului de citit}
repeat
  k:=k+1;
  write(fis1,Mod_y[k]);
  write(fis2,Defaz[k]);
until k=Kmax;
close(fis1); close(fis2);
proc _51; {procedura de afisare a celor doua semnale "depuze" in doua fisiere *.DAT sau *.TRT}
CloseGraph;
END.

```

unit DEPOT; (unit in care se afla depuse procedurile frecvent apelate in programele de analiza ECG si monitorizare a probei de efort. Marea majoritate a acestor proceduri servesc graficii si modalitatilor de afisare folosite de aceste programe. Consider de interes pentru a fi prezentata procedura FILTRU_QRS, in care sunt implementati algoritmi experimentati pentru detectia QRS)

INTERFACE

uses DOS,CRT,GRAPH;

CONST

```
Ndatmax =1100; {nr de esantioane ale fisierelor de date }
Nfis     =40;   {nr maxim de fisiere de date (.DAT si .TRT) din directorul curent ,ce pot
              fi cautate }
Nmin     =-40;  Nmax  =40;   fond     =6;
fondHoriz=9;   fondVert =9;   fondSel  =4;
Nmax_ord=20;{nr max posibil al ordinului functiei de transfer}
```

TYPE

```
tip_ab    =array[0..Nmax_ord,0..2] of real;
tip_k     =array[0..Nmax_ord] of real;
tip_COEF  =array[0..Nmax_ord] of real;
tip_num_fis=string[4];
Viewporttype =record gX0,gY0,gX1,gY1:integer; clip :boolean; end;
vectcauta  =array[0..Nfis] of string;
vect      =array[0..Ndatmax] of real;
vectcoef   =array[Nmin..Nmax] of real;
mesaj     =string[12];
t_infodate =array[1..10] of mesaj;
```

VAR

```
hp,hpw,hptemp,Hdat   : vectcoef;
infodate             : t_infodate;
infofis              : file of t_infodate;
gX0, gX1, gY0, gY1, gX,gY : integer; {le "rectangle" de cadrillage pour gr.}
nr_elem              : integer; {numarul de elem. ale fisierului}
point_mes,point_fis,point_orig ,
zmare,zmic,
N,i,j,k,Nfisgas      : integer;
Qn.{ }
tip_aprox, {variabila "comuna" cu un alt unit care o are declarata in INTERFACE}
tip_filtru, {variabila "comuna" cu un alt unit care o are declarata in INTERFACE}
s,tip,Kmax ,q,qq,kf,Kmes {utilizati in UNIT "BILIN" si "Z"}
                    :integer;
Y, {valoare curenta la citire sau scriere fisiere REAL}
Ydatmin,Ydatmax, {utilizate la scalarea valorilor din fisiere *.DAT }
Ytrtmin,ytrtmax, { " " " *.TRT }
Ymin,Ymax, { " " " pt ambele }
Fe, {frecv. de esantionare cu care s-a "cules" semnalul aflat in fisiere *.DAT}
Fmax,
deltaf,
Hmes,Fimes,
Fp, {f. limita de trecere a benzii }
Fa, { o a doua frecventa din afara benzii la care se va impune o atenuare Amin}
Amin, {atenuarea impusa la Fa}
Epsilon, {necesar la formula de calculul Cebisev}
Riple, {"salturile" in banda de trecere la fct. Cebisev}
Ntemp,RE,IM : real;
nume_fis ,n_fis,numecauta,
nume_fisdat,nume_fistrt,
prenume : string[20];
```



```

datefis          : SearchRec;
trat.wind.clip.ciclu.cadru : boolean;
titre,titlu     : ^vectcauta;
Y1mes,Y2mes,Ymes : ^vect;
car             : char;
fer1,fer2,fer3,fer4,
fer5,fer6,fer   : ViewPortType;
fis             : file of real;

```

```

procedure IDENTINITGRAPH;
procedure INTRNUMGRAPH(gX0,gY0,gX1,gY1:integer;var NUM:real);
procedure INTRstrGRAPH(gX0,gY0,gX1,gY1:integer;var SIR:string);
procedure LISTAFIS(numecauta:string;i:integer;var titlu:vectcauta;var Nfisgas:integer);
procedure FILTRU_QRS(tip_filtru:integer;nume_fis:string;var tip_filtru_afis:string);
procedure INTRODPARAM_BILIN_si_Z(fer1,fer2:viewPortType;gX0,gY0,gX1,gY1:integer);
procedure AFITEXT;
procedure AflaExtreme(n_fis:string;var nr_elem:integer;var Ymin,Ymax:real);
procedure SETFER(var fer1,fer2:viewPortType);
procedure SETFER1(fer1,fer2:ViewPortType;var gX0,gY0,gX1,gY1:integer;var clip:boolean);
procedure SETFER2(fer1,fer2:ViewPortType;var gX0,gY0,gX1,gY1:integer;var clip:boolean);
procedure DesCadre(gX0,gY0,gX1,gY1:integer);
procedure SELNK(var i:integer);
procedure SELECTIE_FEREASTRA1(var i:integer);

```

IMPLEMENTATION

```

{-----}
procedure FILTRU_QRS(tip_filtru:integer;nume_fis:string;var tip_filtru_afis:string);
CONST
    N1min = -170; {coordonata de inceput}
    NmaxQRS = 10; {nr de intervale RR memorate}
TYPE
    vectdat = array[N1min..NDATmax] of real;
    vecttrt = array[N1min..NDATmax] of real;
    vectZOEL = array[1..NmaxQRS,1..2] of integer;
    vectZOEL2 = array[1..NmaxQRS] of real;
VAR
    Sablon :array[0..20] of real;
    Y2,z2, Y1,z1 :vectdat;
    Ydat :vectdat; {!! Vectdat are o lungime mai mare,de la -N1min la 0, pentru ca in
                    calculul primelor valori Ytrt (incepind cu Ytrt(0)),sa fieasigurate valori
                    nule de calcul Ydat(-1),Ydat(-2)... Astfel se inlatura efectul " tranzitoriu"
                    necunoscut de la inceputul creeri fisierului fistrt}
    Ytrt :vecttrt; { si vect Ytrt trebuie sa fie extins pt. filtrele recursive}
    Yreper :vectZOEL;
    Panta :vectZOEL2;
    Ytemp,Ytemp_1,
    x,dx,nivel,
    panta_limita {pentru "infasuratoare inferioara"}
    prag,dY :real;
    m,n,i,j,k,p,q,
    Irep2,Irep1,
    Iret, {coordonata ultimului minim retinut}
    dl,rr :integer;
    fisdat,fistrt :file of real; {fisdat-fisier de date existent}
    numefdat,
    numefirt :string[30]; {fistrt- " ce va fi creat prin procesarea datelor din fisdat.}
    flag_QRS,
    flag_refacere :boolean;

```

```

BEGIN
  for i:=N1min to Ndatmax do begin Ydat[i]:=0; Y1[i]:=0; y2[i]:=0; z1[i]:=0; z2[i]:=0; end;
  nume_fis:=copy(nume_fis,1,pos('.',nume_fis)-1);
  assign(infofis,"+copy(nume_fis,1,4)+'_QRS.INF");
  rewrite(infofis);
  for i:=1 to 10 do infodate[i]:=' ';
  infodate[1]:={nume_fis}'PREPROCESARE/INF';
  infodate[3]:='Ndat='+1100'{tempstr1};
  infodate[4]:='Fe='+200'{tempstr2};
  infodate[8]:=nume_fis;
  infodate[9]:=' ';
  infodate[10]:=' ';
  write(infofis,infodate);
  close(infofis);
  numefdat:="+nume_fis+'.DAT';
  numeftrt:="+copy(nume_fis,1,4)+'_QRS.TRT';
  assign(fisdat,numefdat);
  assign(fistrt,numeftrt);
  reset(fisdat);
  rewrite(fistrt);
  i:=1; {indicele citirii din fisierul de date}
  for p:=1 to NmaxQRS do for q:=1 to 2 do Yreper[p,q]:=0; { constante pt izoelectrica}
  for p:=1 to NmaxQRS do panta[p]:=0; rr:=0; j:=1;
  p:=0; {nr curent al unui esantion din cadrul unui complex la refacere}
  dl:=13 {13}; {diferenta dintre punctul fiducial si momentul sesizarii QRS}
  flag_QRS:=false; flag_refacere:=false;
  nivel:=50; {variabila de la prag...}
  Ytemp_1:=0; {lungimea folosita in relatia de recursivitate}
  Panta_limita:=10; { constante pt infasuratoare infer}
  lret:=0;
  {***** pentru filtrare cu sablon creat din semnal *****}
  if tip_filttru=46 then {se creaza initial sablonul}
  begin
    q:=12;{lungimea sablonului}
    x:=0;
    m:=0;
    repeat
      read(fisdat,Ydat[i]);
      Z1[i]:=- ( Ydat[i]+Ydat[i-1])/2-(Ydat[i-1]+Ydat[i-2])*1/4+(Ydat[i-2]+Ydat[i-3])*0+
        (Ydat[i-3]+Ydat[i-4])*1/4+(Ydat[i-4]+Ydat[i-5])*1/2+(Ydat[i-5]+Ydat[i-6])*1/4+
        (Ydat[i-6]+Ydat[i-7])*0-(Ydat[i-7]+Ydat[i-8])*1/4;
      if Z1[i]>x then x:=Z1[i];
      if i>100 then begin
        if Z1[i-10]>(x*0.8) then
          begin
            for k:=0 to q do Sablon[k]:=Ydat[i-20+k];
            inc(m);
          end;
        end;
        inc(i);
      until (eof(fisdat))or(m=1);
      seek(fisdat,0);{pregatim din nou fisierul pentru a fi citit de la cap}
      {urmeaza "normarea" sablonului}
      for k:=0 to q do Sablon[k]:= Sablon[k]/x;
      x:=0;
      for k:=0 to q do x:=x+ Sablon[k];
      for k:=0 to q do Sablon[k]:=Sablon[k]-x/(q+1);

```

```

i:=1;
end;
repeat
  read(fisdai,Ydat[i]);
{*****Derivata 2 "patru puncte"*****}
  if tip_filtru=1 then begin tip_filtru_afis:='Derivata 2 "patru puncte" ';
    Ytemp:=-Ydat[i]+2*Ydat[i-4]-Ydat[i-8];{filtru QRS_1 ,acoperitor
      la care se identifica unda R cu 4 esantioane
      la stinga si 3 esantioane la dreapta ,la
      o esantioane de 200 es/sec}
    end;
{*****filtru tip deriv de ord 3 "patru puncte"*****}
  if tip_filtru=2 then begin tip_filtru_afis:='Derivata 3 "patru puncte" ';
    Ytemp:=Ydat[i]-4*Ydat[i-4]+6*Ydat[i-8]-4*Ydat[i-12]+Ydat[i-16];
    end;
    {*** CORELATII ***}

{*****filtru tip corelatie cu un "triunghi". sablon simplu SS1*****}
  if tip_filtru=30 then begin tip_filtru_afis:=' Sablon SS1 ' ;
    Ytemp:=Ydat[i-1]*1/4+Ydat[i-2]*2*1/4+Ydat[i-3]*3*1/4+
      Ydat[i-4]*4*1/4+Ydat[i-5]*3*1/4+Ydat[i-6]*2*1/4+Ydat[i-7]*1*1/4;
    end;
{*****filtru tip corelatie cu un "alt triunghi" sablon simplu SS2*****}
  if tip_filtru=31 then begin tip_filtru_afis:=' Sablon SS2 ' ;
    Ytemp:= -Ydat[i]*1/2+Ydat[i-1]*0 +Ydat[i-2]*1/2+
      Ydat[i-3]*1+Ydat[i-4]*1/2+Ydat[i-5]*0-Ydat[i-6]*1/2;
    end;
{*****filtru tip corelatie cu derivare pe sablon simplu SS1*****}
  if tip_filtru=32 then begin tip_filtru_afis:=' SS1+FTS ' ;
    Ytemp:=Ydat[i-1]*1*1/4+Ydat[i-2]*2*1/4+Ydat[i-3]*3*1/4+
      Ydat[i-4]*4*1/4+Ydat[i-5]*3*1/4+Ydat[i-6]*2*1/4+Ydat[i-7]*1*1/4-
      (Ydat[i-2]*1*1/4+Ydat[i-3]*2*1/4+Ydat[i-4]*3*1/4+
      Ydat[i-5]*4*1/4+Ydat[i-6]*3*1/4+Ydat[i-7]*2*1/4+Ydat[i-8]*1*1/4);
    end;
{*****filtru tip corelatie cu derivare pe sablon simplu SS1*****}
  if tip_filtru=33 then begin tip_filtru_afis:=' SS2+FTS ' ;
    Ytemp:= -Ydat[i]*1/2+Ydat[i-1]*0 +Ydat[i-2]*1/2+
      Ydat[i-3]*1+Ydat[i-4]*1/2+Ydat[i-5]*0-Ydat[i-6]*1/2
      -(-Ydat[i-1]*1/2+Ydat[i-2]*0 +Ydat[i-3]*1/2+
      Ydat[i-4]*1+Ydat[i-5]*1/2+Ydat[i-6]*0-Ydat[i-7]*1/2);
    end;
    {*** Corelatii plus FTJ ***}

{*****filtru tip corelatie cu un triungi aflat si sub axa*****}
  if tip_filtru=41 then begin tip_filtru_afis:=' Sablon 1+FTS ' ;
    Ytemp:= -Ydat[i]- Ydat[i-1]*1/2+Ydat[i-2]*0 +Ydat[i-3]*1/2+
      Ydat[i-4]*1+Ydat[i-5]*1/2+Ydat[i-6]*0-Ydat[i-7]*1/2;
    end;
{*****filtru tip corelatie cu un alt triungi aflat si sub axa*****}
  if tip_filtru=42 then begin tip_filtru_afis:=' Sablon 2+FTS ' ;
    Ytemp:= -Ydat[i]*1/2- Ydat[i-1]*1/2+Ydat[i-2]*0+Ydat[i-3]*1/2+
      Ydat[i-4]*1+Ydat[i-5]*1/2+Ydat[i-6]*0-Ydat[i-7]*1/2- Ydat[i-8]*1/2;
    end;
{*****filtru tip corelatie cu un alt triungi aflat si sub axa*****}
  if tip_filtru=43 then begin tip_filtru_afis:=' Sablon 3+FTS ' ;
    Ytemp:= -Ydat[i]*1/2- Ydat[i-1]*1/4+Ydat[i-2]*0+Ydat[i-3]*1/4+
      Ydat[i-4]*1/2+Ydat[i-5]*3/4+Ydat[i-6]*1/2+Ydat[i-7]*1/4- Ydat[i-8]*0
      -Ydat[i-9]*1/4-Ydat[i-10]*1/2-(Ydat[i-11]+Ydat[i-12]+Ydat[i-13])*1/4;

```

```

end;
{*****filtru tip corelatie cu un alt triungi aflat si sub axa*****}
if tip_filtru=44 then begin tip_filtru_afis:=' Sablon 4+FTS ';
Ytemp:=- Ydat[i]*1/2- Ydat[i-1]*1/4+Ydat[i-2]*0+Ydat[i-3]*1/4+
Ydat[i-4]*1/2+Ydat[i-5]*3/4+Ydat[i-6]*1/2+Ydat[i-7]*1/4- Ydat[i-8]*0
-Ydat[i-9]*1/4-Ydat[i-10]*1/2
-(Ydat[i-11]+Ydat[i-12]+Ydat[i-13]+Ydat[i-14]+Ydat[i-15]+Ydat[i-16])*1/8;
end;
{*****filtru tip corelatie plus FTS *****}
if tip_filtru=45 then begin tip_filtru_afis:=' Sablon 5+FTS+FTJ ';
Ytemp:=- ( Ydat[i]+Ydat[i-1])/2-(Ydat[i-1]+Ydat[i-2])*1/4+(Ydat[i-2]+Ydat[i-3])*0+
(Ydat[i-3]+Ydat[i-4])*1/4+(Ydat[i-4]+Ydat[i-5])*1/2+(Ydat[i-5]+Ydat[i-6])*1/4+
(Ydat[i-6]+Ydat[i-7])*0-(Ydat[i-7]+Ydat[i-8])*1/4{- Ydat[i-8]; end;
{***Sablon din SEMNAL***}

{*****filtru corelatie cu sablon "din semnal" plus FTS *****}
if tip_filtru=46 then begin tip_filtru_afis:=' Sablon "din semnal" ';
Ytemp:=0;
for k:=0 to q do Ytemp:=Ytemp+Ydat[i-k]*Sablon[q-k]; end;
{*** FILTRE LUNGIME / ENERGIE ***}

{*****filtru tip "lungime_8"*****}
if tip_filtru=5 then begin tip_filtru_afis:=' Lungime " 8 " ';
Ytemp:=Ytemp_1+SQRT((Ydat[i]-Ydat[i-1])*(Ydat[i]-Ydat[i-1]))-
SQRT((Ydat[i-9]-Ydat[i-8])*(Ydat[i-9]-Ydat[i-8]));
Ytemp_1:=Ytemp; end;
{*****filtru tip "lungime_6"*****}
if tip_filtru=6 then begin tip_filtru_afis:=' Lungime " 6 " ';
Ytemp:=Ytemp_1+SQRT((Ydat[i]-Ydat[i-1])*(Ydat[i]-Ydat[i-1]))-
SQRT((Ydat[i-7]-Ydat[i-6])*(Ydat[i-7]-Ydat[i-6]));
Ytemp_1:=Ytemp; end;
{*****filtru tip "energie_8"*****}
if tip_filtru=7 then begin tip_filtru_afis:=' Energie "8" ';
Ytemp:=Ytemp_1+((Ydat[i]-Ydat[i-1])*(Ydat[i]-Ydat[i-1]))-
((Ydat[i-9]-Ydat[i-8])*(Ydat[i-9]-Ydat[i-8]));
Ytemp_1:=Ytemp; end;
{*****filtru tip "energie_6"*****}
if tip_filtru=8 then begin tip_filtru_afis:=' Energie "6" ';
Ytemp:=Ytemp_1+((Ydat[i]-Ydat[i-1])*(Ydat[i]-Ydat[i-1]))-
((Ydat[i-7]-Ydat[i-6])*(Ydat[i-7]-Ydat[i-6]));
Ytemp_1:=Ytemp; end;
{*** ALTE FILTRE LINIARE ***}

{*****filtru tip "triunghiular"*****}
if tip_filtru=9 then begin tip_filtru_afis:=' Triunghiular ';
Y1[i]:=Y1[i-1]+(Ydat[i]-Ydat[i-4])/4;
z1[i]:=z1[i-1]+(Y1[i]-Y1[i-4])/4;
y2[i]:=y2[i-1]+(Ydat[i]-Ydat[i-160])/160;
z2[i]:=z2[i-1]+(y2[i]-y2[i-160])/160;
Ytemp:=(z1[i-156]-z2[i]); end;
{*** FILTRE NELINIARE ***}

{*****REAXARE dupa infasuratoarea inferioara*****}
if tip_filtru=12 then begin tip_filtru_afis:=' Axare infasuratoare infer. ';
{se face initial o filtrare- corelatie S5 + FTS}
Z1[i]:=- ( Ydat[i]+Ydat[i-1])/2-(Ydat[i-1]+Ydat[i-2])*1/4+(Ydat[i-2]+Ydat[i-3])*0+
(Ydat[i-3]+Ydat[i-4])*1/4+(Ydat[i-4]+Ydat[i-5])*1/2+(Ydat[i-5]+Ydat[i-6])*1/4+
(Ydat[i-6]+Ydat[i-7])*0-(Ydat[i-7]+Ydat[i-8])*1/4;

```

```

{
    Z1[i]:=Ydat[i];} {fara filtrare prealabila}
{urmeaya axarea}
x:=(Z1[i]-Z1[Iret])/(I-Iret); {x =panta dintre punctul curent si ultimul minim retinut}
    {Iret=coordonata ultimului minim retinut}
if x<panta_limita then
begin
    if (Z1[i]>Z1[i-1])and((i-1) > Iret) then
        begin
            for k:=Iret to (i-1) do
                Z2[k]:=Z1[k]-Z1[Iret]-(Z1[i-1]-Z1[Iret])*(k-Iret)/(i-1-Iret);
                Iret:=i-1;
            end;
        end;
    Ytemp:=Z2[i-10];
end;

{*****sfirsitul selectiei filtrelor*****}
    write(fistrt,Ytemp);
    i:=i+1;
until (eof(fisdat))or(i=Ndatmax);
close(fisdat);
close(fistrt);
END;
{***** {of Filtru_QRS}*****}

```

ANEXA A3

**PROGRAM
PENTRU DE ANALIZĂ ȘI PROCESARE "OFF LINE"
A SEMNALELOR ECG**

```

PROGRAM QRS;
USES crt,Dos,Graph,F_discr,Cardinal,Bilin,Transf_Z,
    Afi2supr,Afis2fis,FFT,SUM_fis,copy_ECG.id_QRS,praguri,
    Objects, Views, Drivers, Menus, App;
TYPE
    PMyBackground = ^TMyBackground;
    TMyBackground = object(TBackground)
        constructor Init(var Bounds: TRect);
        procedure Draw; virtual;end;
    PMyDesktop = ^TMyDesktop;
    TMyDesktop = object(TDesktop)
        procedure InitBackground; virtual;end;
    HApp = object (TApplication)
        procedure InitMenuBar; virtual;
        procedure HandleEvent(var Event: TEvent); virtual;
        procedure InitStatusLine; virtual;
        function GetPalette: PPalette; virtual;
        procedure InitDesktop; virtual;end;
CONST
    Ndatmax = 1100; cmSin = 120; cmsinsin = 121; cmFisFis = 110;
    cmCopyA = 111; cmCopyB = 211; cmFFT = 112; cmFDiscr = 113;
    cmCardinal = 114; cmBilin = 115; cmTransfZ = 116; cmAfi2Fis = 118;
    cmAfiSupr = 119; cmQRS1 = 201; cmQRS2 = 202;
    cmQRS30 = 2030; cmQRS31 = 2031; cmQRS32 = 2032; cmQRS33 = 2033;
    cmQRS41 = 2041; cmQRS42 = 2042; cmQRS43 = 2043; cmQRS44 = 2044;
    cmQRS45 = 2045; cmQRS46 = 2046;
    cmQRS55 = 205; cmQRS56 = 206; cmQRS57 = 207; cmQRS58 = 208;
    cmQRS59 = 209; cmQRS10 = 210; cmQRS12 = 212; cmPRAG1 = 360;
VAR
    Buffer : Array[0..20000] of Word;
    procedure Activate;
    var T : TEvent;
    begin
        T.What := evCommand;
        T.Command := cmMenu;
        T.InfoPtr := nil;
        Application^.PutEvent(T);
    end;
    {-----}
    constructor TMyBackground.Init(var Bounds: TRect);
    begin
        inherited Init(Bounds, ',');
    end;

```

```

-----
procedure TMyBackground.Draw;
var DrawBuffer: TDrawBuffer;
begin
  inherited Draw;
  MoveStr(DrawBuffer, '                                ', 31);
  WriteLine(Size.X DIV 2 - 18, Size.Y DIV 2 - 2, 36, 8, DrawBuffer);
  MoveStr(DrawBuffer, '                                Q R S ', 31);
  WriteLine(Size.X DIV 2 - 18, Size.Y DIV 2 + 0, 36, 1, DrawBuffer);
  MoveStr(DrawBuffer, '                                Ioan P. MIHU ', 31);
  WriteLine(Size.X DIV 2 - 18, Size.Y DIV 2 + 3, 36, 1, DrawBuffer);
  MoveStr(DrawBuffer, '                                Universitatea Sibiu', 31);
  WriteLine(Size.X DIV 2 - 17, Size.Y DIV 2 + 4, 34, 1, DrawBuffer);
end;
-----
procedure TMyDesktop.InitBackground;
var R: TRect;
begin
  GetExtent(R);
  Background := New(PMyBackground, Init(R));
end;
-----
procedure HApp.InitDesktop;
var R: TRect;
begin
  GetExtent(R);
  R.Grow(0, -1);
  Desktop := New(PMyDesktop, Init(R));
end;
-----
function HApp.GetPalette: PPalette;
const
  CAppColor = #S79#S70#S78#S74#S4F#S48#S4E#S17#S1F#S1A#S31#S31#S1E#S71#S1F +
    #S37#S3F#S3A#S13#S13#S3E#S21#S3F#S70#S7F#S7A#S13#S13#S70#S7F#S7E +
    #S70#S7F#S7A#S13#S13#S70#S70#S7F#S7E#S20#S2B#S2F#S78#S2E#S70#S30 +
    #S3F#S3E#S1F#S2F#S1A#S20#S72#S31#S31#S30#S2F#S3E#S31#S13#S38#S00 +
    #S17#S1F#S1A#S71#S71#S1E#S17#S1F#S1E#S20#S2B#S2F#S78#S2E#S10#S30 +
    #S3F#S3E#S70#S2F#S7A#S20#S12#S31#S31#S30#S2F#S3E#S31#S13#S38#S00 +
    #S7F#S7E#S1F#S2F#S1A#S20#S32#S31#S71#S70#S2F#S7E#S71#S13#S38#S00;
  CHelpColor = #S07#S7F#S7A#S71#S71#S70#S7E#S0E;
  CNewColor = CAppColor{ + CHelpColor};
  CNewBlackWhite = CAppBlackWhite{ + CHelpBlackWhite};
  CNewMonochrome = CAppMonochrome{ + CHelpMonochrome};
  P: array[apColor..apMonochrome] of string[Length(CNewColor)] =
    (CNewColor, CNewBlackWhite, CNewMonochrome);
begin
  GetPalette := @P[AppPalette];
end;
-----
procedure HApp.InitMenuBar;
var R: TRect;
begin
  GetExtent(R);
  R.B.Y := R.A.Y + 1;
  MenuBar := new(PMenuBar, Init(R, NewMenu(
    NewSubMenu('S~e~mnale', hcNoContext, NewMenu(
      NewItem(' Sinus', "", kbNoKey, cmSin, hcNoContext,
        NewItem(' Sinus+Sinus', "", kbNoKey, cmSinSin, hcNoContext,

```

```

NewLine(
NewItem('Sumare 2 semnale', ", kbNoKey, cmFisFis, hcNoContext,
NewLine(
NewItem(' A:\*.DAT', ", kbNoKey, cmCopyA, hcNoContext,
NewItem(' B:\*.DAT', ", kbNoKey, cmCopyB, hcNoContext,
NewLine(
NewItem(' E~x-it', ", kbNoKey, cmQuit, hcNoContext, nil)))))))))
NewSubMenu('~O~sciloscop', hcNoContext, NewMenu(
NewItem('Semnale Separate', ", kbNoKey, cmAfi2Fis, hcNoContext,
NewItem('Semnale Suprapuse', ", kbNoKey, cmAfiSupr, hcNoContext,
NewLine(
NewItem('E~x-it', 'Alt+X', kbAltX, cmQuit, hcNoContext, nil))))),
NewSubMenu('~S~pectru ', hcNoContext, NewMenu(
NewItem(' FF~T~', ", kbNoKey, cmfft, hcNoContext,
NewLine(
NewItem(' altele...', ", kbNoKey, cmfft, hcNoContext, nil))),
NewSubMenu('Tr.~L~INIARE', hcNoContext, NewMenu(
NewItem('Transf Fourier', ", kbNoKey, cmCardinal, hcNoContext,
NewItem('Fourier Discreta', ", kbNoKey, cmFdiscr, hcNoContext,
NewLine(
NewItem('Transf BILINIARA', ", kbNoKey, cmBilin, hcNoContext,
NewItem('Transf Z', ", kbNoKey, cmTransfZ, hcNoContext, nil)))))))))
NewSubMenu('Tr.SPE~C~IFICE', hcNoContext, NewMenu(
NewSubMenu(' LINIARE ', hcNoContext, NewMenu(
NewSubMenu(' Derivate ', hcNoContext, NewMenu(
NewItem('ord2,"4puncte" ', ", kbNoKey, cmQRS1, hcNoContext,
NewItem('ord3,"4puncte" ', ", kbNoKey, cmQRS2, hcNoContext,
NewLine(
NewItem('E~x-it', 'Alt+X', kbAltX, cmQuit, hcNoContext, nil))))),
NewLine(
NewSubMenu(' Corelatie ', hcNoContext, NewMenu(
NewSubMenu(' Corelatie SIMPLA', hcNoContext, NewMenu(
NewItem('Corelatie1 ', ", kbNoKey, cmQRS30, hcNoContext,
NewItem('Corelatie2 ', ", kbNoKey, cmQRS31, hcNoContext,
NewItem('Corelatie1+FTS ', ", kbNoKey, cmQRS32, hcNoContext,
NewItem('Corelatie2+FTS ', ", kbNoKey, cmQRS33, hcNoContext, nil))))),
NewSubMenu(' Corelatie + FTS', hcNoContext, NewMenu(
NewItem('Sablon 1+FTS ', ", kbNoKey, cmQRS41, hcNoContext,
NewItem('Sablon 2+FTS ', ", kbNoKey, cmQRS42, hcNoContext,
NewItem('Sablon 3+FTS ', ", kbNoKey, cmQRS43, hcNoContext,
NewItem('Sablon 4+FTS ', ", kbNoKey, cmQRS44, hcNoContext,
NewItem('Sablon 5+FTS', ", kbNoKey, cmQRS45, hcNoContext,
NewItem('Sablon "din semnal"', ", kbNoKey, cmQRS46, hcNoContext,
NewLine(
NewItem('E~x-it', 'Alt+X', kbAltX, cmQuit, hcNoContext, nil)))))))))
NewLine(
NewSubMenu('Transformata LUNGIME ', hcNoContext, NewMenu(
NewItem('lungime"8" ', ", kbNoKey, cmQRS5, hcNoContext,
NewItem('lungime"6"', ", kbNoKey, cmQRS6, hcNoContext,
NewLine(
NewItem('E~x-it', 'Alt+X', kbAltX, cmQuit, hcNoContext, nil))))),
NewLine(
NewSubMenu('Transformata Energie ', hcNoContext, NewMenu(
NewItem('Energie"8" ', ", kbNoKey, cmQRS7, hcNoContext,
NewItem('Energie"6" ', ", kbNoKey, cmQRS8, hcNoContext,
NewLine(
NewItem('E~x-it', 'Alt+X', kbAltX, cmQuit, hcNoContext, nil))))),
NewLine(

```



```

NewSubMenu('Triunghi ', hcNoContext, NewMenu(
  NewItem('Triunghi1 ', "", kbNoKey, cmQRS9, hcNoContext, nil), nil))))),
  NewLine(
NewSubMenu(' NELINIARE ', hcNoContext, NewMenu(
NewSubMenu('Linie izoelectrica', hcNoContext, NewMenu(
  NewItem('Varianta 1 ', "", kbNoKey, cmQRS10, hcNoContext,
  NewLine(
  NewItem('E~x~it', 'Alt+X', kbAltX, cmQuit, hcNoContext, nil))),
NewSubMenu('Axare dupa Infasuratoare inf.', hcNoContext, NewMenu(
  NewItem('Varianta 1 ', "", kbNoKey, cmQRS12, hcNoContext, nil)),
NewSubMenu('Median', hcNoContext, NewMenu(
  NewItem('Varianta 1 ', "", kbNoKey, cmQRS10, hcNoContext, nil)),
  NewLine(
  NewItem('E~x~it', 'Alt+X', kbAltX, cmQuit, hcNoContext, nil))))), nil))),
NewSubMenu('Ident.~Q~RS', hcNoContext, NewMenu(
NewSubMenu('Pt. transf. specifice LINIARE ', hcNoContext, NewMenu(
  NewItem('Praguri ADAPTATIVE ', "", kbNoKey, cmPRAG1, hcNoContext,
  NewItem('Praguri NEADAPTATIVE', "", kbNoKey, cmPRAG1, hcNoContext,
  NewLine(
  NewItem('E~x~it', 'Alt+X', kbAltX, cmQuit, hcNoContext, nil))))),
  NewLine(
  NewItem(' .....', "", kbNoKey, cmQuit, hcNoContext,
  NewLine(
  NewItem('E~x~it', 'Alt+X', kbAltX, cmQuit, hcNoContext, nil))))))),
NewSubMenu('~A~larma', hcNoContext, NewMenu(
  NewItem(' Aritmie 1 ', "", kbNoKey, cmQRS1, hcNoContext,
  NewItem(' Aritmie 2 ', "", kbNoKey, cmQRS1, hcNoContext,
  NewItem('.....', "", kbNoKey, cmQRS1, hcNoContext,
  NewLine(
  NewItem('E~x~it', 'Alt+X', kbAltX, cmQuit, hcNoContext, nil))))), nil))))),
end;
{-----}
procedure HApp.InitStatusLine:
var R : TRect;
begin
  GetExtent(R);
  R.A.Y := R.B.Y - 1;
  StatusLine := new(PStatusLine, Init(R, NewStatusDef(0, 2000,
  NewStatusKey('~F10~ Meniu', kbF10, cmMenu,
  NewStatusKey('~Alt-X~ Exit', kbAltX, cmQuit, nil)), nil));
end;
{-----}
procedure HApp.HandleEvent:
begin
  inherited HandleEvent(Event);
  if Event.What = evCommand then
  case Event.Command of
    cmSin: {proc_11};
    cmSinSin: {proc_12};
    cmFisFis: proc_14;
    cmCopyA: proc_21;
    cmCopyB: proc_22;
  end;
end;

```

```

cmFFT:      begin proc_31; activate;end;
cmFDiscr:   begin proc_41;activate;end;
cmCardinal : begin proc_42;activate;end;
cmBilin:    begin proc_43;activate;end;
cmTransfZ : begin proc_44;activate;end;
cmQRS1:     begin tip_filtru:= 1; proc_45;activate;end;
cmQRS2:     begin tip_filtru:= 2; proc_45;activate;end;
cmQRS30:    begin tip_filtru:= 30; proc_45;activate;end;
cmQRS31:    begin tip_filtru:= 31; proc_45;activate;end;
cmQRS32:    begin tip_filtru:= 32; proc_45;activate;end;
cmQRS33:    begin tip_filtru:= 33; proc_45;activate;end;
cmQRS41:    begin tip_filtru:= 41; proc_45;activate;end;
cmQRS42:    begin tip_filtru:= 42; proc_45;activate;end;
cmQRS43:    begin tip_filtru:= 43; proc_45;activate;end;
cmQRS44:    begin tip_filtru:= 44; proc_45;activate;end;
cmQRS45:    begin tip_filtru:= 45; proc_45;activate;end;
cmQRS46:    begin tip_filtru:= 46; proc_45; activate;end;
cmQRS5:     begin tip_filtru:= 5; proc_45; activate;end;
cmQRS6:     begin tip_filtru:= 6; proc_45; activate;end;
cmQRS7:     begin tip_filtru:= 7; proc_45; activate;end;
cmQRS8:     begin tip_filtru:= 8; proc_45; activate;end;
cmQRS9:     begin tip_filtru:= 9; proc_45; activate;end;
cmQRS10:    begin tip_filtru:= 10; proc_45; activate;end;
cmQRS12:    begin tip_filtru:= 12; proc_45; activate;end;
cmAfi2Fis : begin proc_51; Activate;end;
cmAfiSupr : beginproc_52; Activate;end;
cmPRAG1 :   begin proc_70; Activate; end;

end;
desktop^.draw;
menubar^.draw;
statusline^.draw;
end;
{-----{inceput program principal QRS}-----}
VAR
  MApp : HApp;
BEGIN
  MApp.Init;
  MApp.Run;
  MApp.Done;
END.
UNIT CARDINAL; {Filtrare cu ajutorul transformatei Fourier}

INTERFACE
uses DOS,CRT,GRAPH,DEPOT.F_discr.desene;
procedure PROC_42;

IMPLEMENTATION
const
  dreptunghi:array[1..4] of PointType = ((x:5;y:5),(x:630;y:5), (x:630;y:165),(x:5;y:165));
var
  F,Ft,Fi,Fs{,Fe.Fmax,deltaf,} :real;
{-----}
procedure INTRODPARAM(fer1,fer2:viewPortType:gX0,gY0,gX1,gY1:integer );
var j :integer;
    Ntemp:real;
    TempStr:string[7];
begin
  SetViewPort(0,0,GetMaxX,GetMaxY,true);

```

```

SETFER(fer1,fer2);
DESCADRE(1,415,160,460); {cadrul pt fer. mica de afisaj Fe ,Ft,Fi,Fs }
DESCADRE(170,415.240,460); {cadrul pt fer. mica de afisaj N }
SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);
setviewport(gX0+1,gY0+1,gX1-1,gY1-1,clip);
clearViewport;
SetViewport(0,0,GetMaxX,GetMaxY,true);
SetColor(6);
OutTextxy(1,225,' Introduceti: ');
OutTextxy(1,240,' 1. frecventa de esantionare .....Fe = Hz');
INTRNUMGRAPH(+50,240,1,1,Fe);
str(Round(Fe):0,TempStr);
OutTextxy(20,420,' Fe(Hz) = '+TempStr);
if tip in [1,2] then begin
  OutTextxy(1,260,' 2. frecventa de trecere = frecventa limita');
  if tip=1 then OutTextxy(1,270,' superioara a benzii de trecere (-3dB) ');
  if tip=2 then OutTextxy(1,270,' inferioara a benzii de trecere (-3dB) ');
  OutTextxy(1,280,' a filtrului. ');
  OutTextxy(1,290,' Atentie! Ft<Fe/2 .....Ft = Hz');
INTRNUMGRAPH(+50,290,1,1,Ft);
str(Round(Ft):0,TempStr);
OutTextxy(20,430,' Ft(Hz) = '+TempStr);
end;
if tip in [3,4] then begin
  OutTextxy(1,260,' 2. frecventa inferioara a benzii de trecere ');
  OutTextxy(1,270,' a filtrului trece banda (FTB).....Fi = Hz ');
  INTRNUMGRAPH(+450,270,1,1,Fi);
  str(Round(Fi):0,TempStr);
  OutTextxy(20,430,' Fi(Hz) = '+TempStr);
  OutTextxy(1,290,' 3. frecventa superioara a benzii de trecere ');
  OutTextxy(1,300,' a filtrului trece banda (FTB).....Fs = Hz ');
  INTRNUMGRAPH(+450,300,1,1,Fs);
  str(Round(Fs):0,TempStr);
  OutTextxy(20,440,' Fs(Hz) = '+TempStr);
end;
if tip in [1,2] then OutTextxy(1,320,' 3. ordinul filtrului = numarul ultimelor N ');
if tip in [3,4] then OutTextxy(1,320,' 4. ordinul filtrului = numarul ultimelor N ');
OutTextxy(1,330,' esantioane asupra carora se va opera in ');
OutTextxy(1,340,' timpul filtrarii, respectiv numarul ');
OutTextxy(1,350,' coeficientilor filtrului (0<N<50).....N = ');
INTRNUMGRAPH(+450,350,1,1,Ntemp); N:=Round(Ntemp);
str(N:0,TempStr);
OutTextxy(178,420,' Ordin');
OutTextxy(178,428,' filtru:');
OutTextxy(178,444,' N = '+TempStr);
OutTextxy(1,375,' Tastati ENTER pentru a continua ! '); Readln;
end;
}
procedure SELECTIEI(var s:integer);
const dreptunghi1:array[1..4] of PointType = ((x:150;y:50),(x:250;y:50),
(x:250;y:110),(x:150;y:110));
dreptunghi2:array[1..4] of PointType = ((x:2;y:20),(x:101;y:20),
(x:101;y:40),(x:2;y:40));
}
procedure deswin_horiz(s:N01 :integer);
begin
  SETFER(fer1,fer2);
  SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);

```

```

SetViewPort(gX0+1,gY0+1,gX1-1,gY1+1,True);
ClearViewPort;
SetColor(2);
SetfillStyle(SolidFill,6);
SetLineStyle(DottedLn,0,NormWidth);
FillPoly(4,dreptunghi);
OutTextXY(1,10,'Alegeti tipul filtrului folosind '
+char(24)+' sau '+char(25));
OutTextXY(1,150,'apoi validati cu ENTER ');
SetfillStyle(SolidFill,6);
SetLineStyle(DottedLn,0,NormWidth);
FillPoly(4,dreptunghi1);
end;
Function TITLU(s:integer):string;
begin
  case s of
    0 : titlu := ' ';
    1 : titlu := ' Trece Jos';
    2 : titlu := ' Trece Sus';
    3 : titlu := 'Trece Banda';
    4 : titlu := 'Opre. Banda';
    5 : titlu := ''; {lista ferestrelor poate continua }
  end;
end;
BEGIN
s:=1;
deswin_horiz(s,4{N01});
repeat
  SetViewPort(150+1,270+1,250-1,330+1,True);
  SetfillStyle(SolidFill,4);
  SetLineStyle(DottedLn,0,NormWidth);
  FillPoly(4,dreptunghi2);
  SetColor(2);
  OutTextXY(10,7,TITLU(s-1));
  OutTextXY(10,27,TITLU(s));
  OutTextXY(10,47,TITLU(s+1));
  q:=s;
  selnk( s );
  if ord(car) in [72,80] then begin
    SetColor(6);
    OutTextXY(10,7,TITLU(q-1));
    OutTextXY(10,27,TITLU(q));
    OutTextXY(10,47,TITLU(q+1));
  end;
until ord(car)=13;
END;
{-----!!!!-----}
procedure CALCULhp(N,tip:integer;Ft,Fe:real ;var hp:vectcoef);
if not(wind) then begin
for k:=0 to (2*(N-1)) do begin
  case tip of {in functie de tipul de filtru selectat, se va calcula "vectorul"de coeficienti hp}
    1 : begin
      if k=(N-1) then hp[0]:=2*Ft/Fe {func (sinx)/x are limita "1"pt x=0}
      else hp[-(N-1)+k]:=2*(sin(Pi*Ft*(-(N-1)+k)/Fe))/(Pi*(-(N-1)+k)); {FTJ}
      end;
    2 : begin
      if k=(N-1) then hp[0]:=1-(2*Ft/Fe)
      else hp[-(N-1)+k]:=2*(sin(Pi*(-(N-1)+k)/2)-sin(Pi*Ft*(-(N-1)+k)/Fe))/

```

```

        (Pi*(-(N-1)+k));          {FTS}
    end;
3 : begin
    if k=(N-1) then hp[0]:=2*(Fs-Fi)/Fe
    else hp[-(N-1)+k]:=2*(sin(Pi*Fs*(-(N-1)+k)/Fe)-sin(Pi*Fi*(-(N-1)+k)/Fe))/
        (Pi*(-(N-1)+k));          {FTB}
    end;
4 : begin
    if k=(N-1) then hp[0]:=1-(2*(Fs-Fi)/Fe)
    else hp[-(N-1)+k]:=2*(sin(Pi*(-(N-1)+k)/2)-sin(Pi*Fs*(-(N-1)+k)/Fe)+
        sin(Pi*Fi*(-(N-1)+k)/Fe))/(Pi*(-(N-1)+k));          {FOB}
    end;
end;
end;
end;
end;
{-----}
procedure CONV1(kf,N,q,gX0,gY0,gX1,gY1:integer;deltaf,Fe:real;
    var gX,gY:integer;var Hmes:real);
var Temp:real;j:integer;
begin
    Temp:=0;j:=0;
    repeat
        Temp:=Temp+hptemp[-(N-1)+2*j]*cos(-(N-1)+2*j)*kf*deltaf/Fe;
        j:=j+1;
    until j=N;
    if Temp<0 then Temp:=-Temp;
    Hmes:=Temp;
    gX:=10+Round(((gX1-gX0)/Fmax)*(deltaf/pi)*kf);
    gY:=gY1-gY0-10-Round(120*Hmes)
end;
{-----}
procedure CONV2(kf,N,q,gX0,gY0,gX1,gY1:integer;deltaf,Fe:real;
    var gX,gY:integer;var Fimes:real);
begin
    Fimes:=((N-1)*kf*deltaf/Fe/pi)/10;
    gX:=10+Round(((gX1-gX0)/Fmax)*(deltaf/pi)*kf);
    gY:=(gY1-gY0) div 2-Round((gY1-gY0)*fimes/pi);
end;
{-----}
procedure deswin_horiz(s,N0 :integer);
begin
    SETFER(fer1,fer2);
    SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);
    SetViewPort(gX0+1,gY0+1,gX1-1,gY1+1,True);
    ClearViewPort;
    Setcolor(2);
    SetfillStyle(SolidFill,6);
    SetLineStyle(DottedLn,0,NormWidth);
    FillPoly(4,dreptunghi);
    OutTextXY(1,10,'                Alegeti tipul ferestrei dorite cu : '+char(24)+' sau '+char(25));
    OutTextXY(1,150,'                Validati cu ENTER                ');
    SetfillStyle(SolidFill,6);
    SetLineStyle(DottedLn,0,NormWidth);
    FillPoly(4,dreptunghi 1);
end;
Function TITLU(s:integer):string;
begin

```

```

case s of
  0 : titlu := '          ';
  1 : titlu := ' HANNING  ';
  2 : titlu := ' HAMMING  ';
  3 : titlu := 'BLACKMAN';
  4 : titlu := 'BARTELETT';
end;
end;
BEGIN
s:=1;
deswin_horiz(s,N0);
repeat
  SetViewPort(250+1,270+1,350-1,330+1,True);
  SetfillStyle(SolidFill,4);
  SetLineStyle(DottedLn,0,NormWidth);
  FillPoly(4,dreptunghi2);
  SetColor(2);
  OutTextXY(10,7,TITLU(s-1));
  OutTextXY(10,27,TITLU(s));
  OutTextXY(10,47,TITLU(s+1));
  q:=s;
  selnk( s );
  SetColor(6);
  OutTextXY(10,7,TITLU(q-1));
  OutTextXY(10,27,TITLU(q));
  OutTextXY(10,47,TITLU(q+1));
until ord(car)=13;
END;
begin {corpul principal al procedurii FEREASTRA}
  SELECTIE_FEREASTRA(s);
  case s of 1: {se implementeaza HANNING}
    for k:=0 to (2*(N-1)) do
      hpw[-(N-1)+k]:=hp[-(N-1)+k]*(1+cos(pi*(-(N-1)+k)/(N-1)))/2;
    2: {se implementeaza HAMMING}
    for k:=0 to (2*(N-1)) do
      hpw[-(N-1)+k]:=hp[-(N-1)+k]*(0.54+0.46*cos(pi*(-(N-1)+k)/(N-1)));
    3: {se implementeaza BLACKMAN}
    for k:=0 to (2*(N-1)) do hpw[-(N-1)+k]:=
      hp[-(N-1)+k]*(0.42+0.5*cos(pi*(-(N-1)+k)/(N-1))+0.08*cos(2*pi*(-(N-1)+k)/(N-1)));
    4: {se implementeaza . BARTELET}
    begin
      k:=0;
      repeat
        hpw[-(N-1)+k]:=hp[-(N-1)+k]*(1+(-(N-1)+k)/(N-1));
        inc(k);
      until -(N-1)+k>0 ;
      repeat
        hpw[-(N-1)+k]:=hp[-(N-1)+k]*(1-(-(N-1)+k)/(N-1));
        inc(k);
      until k>2*(N-1);
    end;
  end;
end;
{*****Procedura principala a UNIT-ului*****}
procedure PROC_42;
var temp1:integer;
    tempstr1,tempstr2:string[7];
BEGIN

```

```

if MaxAvail < SizeOf(vectcauta) then writeln('NOT ENOUGH MEMORY')
else begin
    GetMem(titlu,SizeOf(vectcauta)); end;          {Alocare dinamica de mem in HEAP}
N:=1; wind:=FALSE; Kmes:=1; deltax:=5; q:=1;
for k:=Nmin to Nmax do hp[k]:=0;          {se initializeaza toti coef la zero}
for k:=Nmin to Nmax do hptemp[k]:=0;      {se initializeaza toti coef la zero}
IDENTINITGRAPH;
SETFER(fer1,fer2);
SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);
DesCadre( gX0,gY0,gX1,gY1);
SETFER1(fer1,fer2,gX0,gY0,gX1,gY1,clip);
DesCadre( gX0,gY0,gX1,gY1);
SetColor(6);
    OutTextxy(10,24,' Acest meniu va permite proiectarea unui filtru numeric pornind de la
carac- ');
    OutTextxy(10,36,' teristica de frecventa dorita si verificarea eficientei sale in etapele: ');
    OutTextxy(10,48,' 1.Alegerea tipului de filtru dorit a fi realizat: FTJ,FTS,FTB sau FOB ');
    OutTextxy(10,60,' 2.Introducerea datelor necesare viitoarei constructii a caracteristicii de');
    OutTextxy(10,72,' frecventa:frecventa de esantionare(Fe), ordinul filtrului (N), frecvente');
    OutTextxy(10,84,' le impuse benzii de trecere, conform tipului de filtru ales (Ft,Fi,Fs). ');
    OutTextxy(10,96,' 3.Calculul automat al coeficientilor folosind metoda');
    OutTextxy(10,108,' -----TRANSFORMATA FOURIER -----');
    OutTextxy(10,120,' 4.Pentru verificarea eficientei filtrului se poate trece prin filtrul rea-');
    OutTextxy(10,132,' lizat un "semnal" continut in fisiere cu extensia *.DAT ; Rezultatul ');
    OutTextxy(10,144,' filtrarii se va depune intr-un fisier cu extensia *.TRT ; Vizualizarea ');
    OutTextxy(10,156,' si evaluarea rezultatelor se va face in meniul "Osciloscop". ');
    OutTextxy(10,168,' 5.La dorinta se pot vizualiza coeficientii si structura filtrului. ');
    OutTextxy(10,280,' Pentru selectarea optiunilor precum si pentru introducerea datelor ');
    OutTextxy(10,292,' urmati indicatiile din ferestrele ce vor urma ! ');
    OutTextxy(10,360,' Tastati ENTER pentru a continua '); readln;
SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);
SetViewPort(gX0+1,gY0+1,gX1-1,gY1+1,True); ClearViewPort;
SETFER(fer1,fer2);
SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);
DesCadre( gX0,gY0,gX1,gY1);
SetViewPort(gX0+1,gY0+1,gX1-1,gY1+1,True); ClearViewPort;
repeat
repeat
if not(wind) then begin
SELECTIE1(tip);
INTRODPARAM(fer1,fer2,gX0,gY0,gX1,gY1);
CALCULhp(N.tip,Ft,Fe,hp);
end;
repeat
SetViewPort(0,0,GetMaxX,GetMaxY,true);
REPREZGRAPH;
until ord(car)=13;
SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);
SetViewPort(gX0+1,gY0+1,gX1-1,gY1+1,True);
ClearViewPort;
SetfillStyle(SolidFill,6);
SetLineStyle(DottedLn,0,NormWidth);
FillPoly(4,dreptunghi);
setcolor(2);
OutTextXY(1,30,' Doriti sa corectati filtrul prin metoda "FEREASTRA" ? "D"a sau "N"u ');
OutTextXY(1,50,' (prin aceasta metoda se pot inlatura "oscilatiile" amplificarii ');
OutTextXY(1,60,' atit in banda de trecere cit si in banda de atenuare = efect Gibbs). ');
repeat

```

```

car:=upcase(readkey);
if ord(car)=68 then wind:=TRUE;
IF ord(car)=78 then wind:=FALSE;
until (ord(car)=78)or(ord(car)=68);
if wind then begin      {aici voi insera proc ""FEREASTRA"}
FEREASTRA(N,hp,hpw);
for k:=Nmin to Nmax do hp[k]:=hpw[k];
end;
until not(wind);      {aici se incheie ciclul "repeat" utilizat pt}
{refacerea caracteristicii in cazul in care s-a optat pentru "FEREASTRA"}
OutTextXY(1,90,'      Doriti sa refaceti filtrul realizat ? "D" a sau "N"u');
repeat
car:=upcase(readkey);
if ord(car)=78 then trat:=TRUE;
IF ord(car)=68 then begin {se va reface filtrul}
trat:=FALSE;{mai trebuie pus ceva pt a sterge vechile date introduse}
SetViewPort(0,0,GetMaxX,GetMaxY,true);
SetColor(0);
SetLineStyle(DottedLn,0,NormWidth);
SetfillStyle(SolidFill,0);
FillPoly(4,dreptunghi_refac);
SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);
DesCadre( gX0.gY0.gX1.gY1);
end;
until (ord(car)=78)or(ord(car)=68);
until trat : {aici se incheie primul ciclu "repeat" ce contine: introd.date,calc.coef.vizualizare
H(jf), refacere."ferestre",refacere}
OutTextxy(1,120,'      Doriti sa treceti prin filtrul realizat un "semnal" ? "D" a sau "N"u');
repeat
car:=upcase(readkey);
until (ord(car)=78)or(ord(car)=68);
IF ord(car)=68 then
begin
for k:=0 to Nfis do TITLU^[k]:=' ';
numecauta:='*.DAT';      { se va vizualiza intii fis *.DAT }
i:=1;
LISTAFIS(numecauta,i,TITLU^,Nfisgas);      {se face lista fis. *.DAT }
numecauta:='*.TRT';      { se va cauta in continuare fis *.TRT }
i:=Nfisgas+1;
LISTAFIS(numecauta,i,TITLU^,Nfisgas);      {se adauga lista fis. *.TRT }
nume_fis:='';
i:=1;
SELECTIE_FEREASTRA1(i,nume_fis);      {se sel. fis dorit }
for k:=1 to 10 do infodate[k]:=' ';
SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);
SetViewPort(gX0+1,gY0+1,gX1-1,gY1+1,True);
SetfillStyle(SolidFill,6);
SetLineStyle(DottedLn,0,NormWidth);
mesaj_prenume;
INTRstrGRAPH(520,320,600,330.prenume);
SETFER2(fer1,fer2,gX0,gY0,gX1,gY1,clip);
SetViewPort(gX0+1,gY0+1,gX1-1,gY1+1,True);
SetColor(2);
OutTextXY(200,140,'      Asteptati sfirsitul calculelor !!! ');
Calcul_Fis_TRT(N);
Calcul_INFO_FIS(nume_fis.prenume.infodate);
end; {sfirsit pt cazul in care s-a trecut un semnal prin filtru}
FillPoly(4,dreptunghi);

```



```
outtextxy(1,15,' Doriti sa afisati coeficientii si structura filtrului realizat ');
outtextxy(1,30,' "D"a sau "N"u ');
repeat
  car:=upcase(readkey);
until (ord(car)=78)or(ord(car)=68);
IF ord(car)=68 then
  begin
  SetViewPort(0,0,GetMaxX,GetMaxY,true);
  FillPoly(4,dreptunghi4);
  FillPoly(4,dreptunghi5);
  desen4; {Procedura desenarea structurii filtrului in dreptunghi5! (cel mic)}
  str(N:0,TempStr1);
  outtextxy(1,190,' Coeficientii filtrului de ordin '+tempstr1+' realizat sunt: ');
  for j:=0 to N-1 do
    begin
      str(j:0,TempStr1);
      if not(wind) then str(Hp[j]:2:4,TempStr2)
        else str(Hpw[j]:2:4,TempStr2);
      OutTextxy(10+125*(j-5*Round(-0.49+0.2*j)),210+20*Round(-0.49+0.2*j),
        'C('+tempstr1+')='+tempstr2);
    end;
  outtextxy(1,460,' Tastati ESC pentru a parasi procedura ');
  repeat
    car:=readkey;
  until ord(car)=27;
  end;
FreeMem(titlu,SizeOf(vectcauta));
closegraph;
end;
END.
```

ANEXA A4

**PROCEDURĂ
DE MEDIERE CU SUPRAEȘANTIONARE
A SEMNALELOR ECG**

**DIN PROGRAMUL DE MONITORIZARE
A PROBEI DE EFORT**

USES dos.graph.crt.afi2sablon.dat_2.dat;

CONST

Factor_de_multipl = 4; {nr de puncte intermediare ce se vor plasa intre doua
esntioane ale sablonului, pentru a genera "SABLONUL_PRECIS"}
Dimens_epsilon_X = 100; {dimensiunea tabloului de erori}
Dimens_epsilon_Y = 100; {dimensiunea tabloului de erori pe Y}
Margine_EPS = 5; {cu cit se reduce fereastra de calcul pt EPS fata de
dimens initiala a Sabl_perfect. }
Dy = 1; {cuanta de deplasare pe verticala a Sablonului_real trebuie
sa fie comparabila cu pragul zgom de cuantizare la achizitie}
Nr_dupa_minim = 5; { cit sa continuam cautarea dupa identificarea unui minim }

TYPE

TipSablon_precis = array[-100+0..100+Factor_de_multipl*Nr_esant_in_sablon] of real;
Tip_Flaguri_Sablon_precis=array[-100..100+Factor_de_multipl*Nr_esant_in_sablon]
of boolean; { -100..+100 pentru a permite glisarea sablonului peste Sablon_precis }

VAR

test ,
sens : boolean; {atasata sensului de cautare: true = dreapta; false = stinga}
testi,
ind_shift, {dif dintre poz. initiala a Sabl_perf si pozitia la un moment dat a
Sabl_real deplasat pe Y}
ind_shift_min, {se retine valoarea ecartului unde este minima EPSILON_Y}
minim_minimorum, {cea mai mica valoare din Min_Epsilon}
Ind_Epsilon, {indice pt cautare in tabloul Epsilon}
treceri_prin_min, {la cautare de cite ori se trece prin minim}
Ind_sablon_precis, {pentru cautat in SABLONUL_PRECIS}
ecart, {dif dintre poz. initiala a Sabl_perf si pozitia la un moment dat
pt. Sabl_real deplasat}
ecart_min, {se retine valoarea ecartului unde este minima EPSILON}
refac_cautarea, {pentru a reface cautarea pe X dupa cea de pa Y}
Nr_sablon_copiat : integer;
Shift_Y, { valoarea cu cit trebuie depl. pe Y Sablonul real}
D, {Diferenta intre doua Epsilon[i]}
maxim_Epsilon :real;
Epsilon_Y : array[-Dimens_epsilon_Y..Dimens_epsilon_Y] of real;
Epsilon_X : array[-Dimens_epsilon_X..Dimens_epsilon_X] of real;
Min_Epsilon : array[1..Nr_de_sabloane] of real; {se retine minimul din
tabloul Epsilon}
Flaguri_sablon_precis: Tip_Flaguri_Sablon_precis; {flaguri atasate fiecarui punct pt a sti
daca el a fost creat prin aproximare cu o functie matematica}

```

                                (FALSE) sau provine de la valori reale (TRUE).}
    Sablon_precis      : TipSablon_precis;
    f_Sablon_precis   : file of real;  { file of TipSablon_precis}
    tempstr           : string[30];
}-----}
procedure INITIALIZEZ;
VAR
    i :integer;
begin
    Nr_esant_in_sablon:=_Nr_esant_in_sablon;
    Nr_de_sabloane:=_Nr_de_sabloane;
    Lungime_inreg_ECG:=_Lungime_inreg_ECG;
    assign(fINF,'0_cine-i.0__');
    reset(fINF);
    read(fINF, nume_de_transmis);
    close(fINF);
    if nume_de_transmis="" then nume_de_transmis:='0_____';
    i:=0;
    repeat
        str(i,tempstr);
        assign(f_Sabloane,'date'+copy(nume_de_transmis,1,3)+'_s_'+tempstr+'.dat');
        reset(f_Sabloane);
        if i=0 then begin read(f_Sabloane, Sablon_0min);           end;
        if i=3 then begin read(f_Sabloane, Sablon_3min);           end;
        if i=6 then begin read(f_Sabloane, Sablon_6min);           end;
        if i=9 then begin read(f_Sabloane, Sablon_9min);           end;
        if i=12 then begin read(f_Sabloane, Sablon_12min);          end;
        if i=15 then begin read(f_Sabloane, Sablon_15min);          end;
        close(f_Sabloane);
        i:=i+3;
    until i>15;
    Ind_sablon_precis:=0; sens      :=true; Nr_sablon:=1; maxim_Epsilon :=0;
    ECART      :=0; Shift_Y      :=0; refac_cautarea:=0; Nr_sablon_copiat:=1;
    for i:=1 to Nr_de_sabloane do Min_Epsilon[i]:=100000;
    for i:=0 to Factor_de_multipl*Nr_esant_in_sablon do
        begin Sablon_precis[i]:=0; Flaguri_sablon_precis[i]:=false; end;
    for i:=0 to Nr_esant_in_sablon do
        begin
            Sablon_precis[i*Factor_de_multipl]:=sablon_0min[Nr_sablon_copiat,i];
            Flaguri_sablon_precis[i*Factor_de_multipl]:=true;
        end;
    for i:=Margine_EPS to Nr_esant_in_sablon-Margine_EPS do
        begin
            maxim_Epsilon :=maxim_Epsilon + (Sablon_precis[i*Factor_de_multipl+4]-
                Sablon_precis[i*Factor_de_multipl])*(Sablon_precis[i*Factor_de_multipl+4]-
                Sablon_precis[i*Factor_de_multipl]);
        end;
    maxim_Epsilon :=maxim_Epsilon /(Nr_esant_in_sablon-2*Margine_EPS);
    for i:=-Dimens_epsilon_X to Dimens_epsilon_X do Epsilon_X[i]:=100000;
    for i:=-Dimens_epsilon_Y to Dimens_epsilon_Y do Epsilon_Y[i]:=100000;
end;
}-----}
procedure CREAZA_PUNCTE_INTERMEDIARE;
VAR
    i1,                {valoarea indicelui inferior pentru o interpolare}
    i2,                {valoarea indicelui superior pentru o interpolare}
    sesiz_trecere,     {pentru a sesiza daca se gasesc date consecutive (false)}
    i                  : integer;

```

```

consec      :boolean; {stabileste daca avem doua flag_sablon_perf TRUE consec}
begin
sesiz_trecere:=0; i:=1;
Ind_sablon_precis:=0;
repeat
  i1:=Ind_sablon_precis; consec:=True;
  repeat
    repeat
      inc(Ind_sablon_precis);
      if ((Ind_sablon_precis>=Factor_de_multipl*Nr_esant_in_sablon-1)
        and (sesiz_trecere = 0)) then
        OutTextXY(1,10,'          NU mai sunt locuri libere !!! ');
if Flaguri_sablon_precis[Ind_sablon_precis]=false then
  begin consec:=false;inc(sesiz_trecere);end;
  until Flaguri_sablon_precis[Ind_sablon_precis];
  if consec then i1:=Ind_sablon_precis;
until not(consec) or (Ind_sablon_precis>=Factor_de_multipl*Nr_esant_in_sablon);
i2:=Ind_sablon_precis; i:=1;
if i2 > i1 then
  repeat
    Sablon_precis[i1+i]:=Sablon_precis[i1]+i*(Sablon_precis[i2]-Sablon_precis[i1])/(i2-i1);
    inc(i);
  until i=i2-i1;
until Ind_sablon_precis=Factor_de_multipl*Nr_esant_in_sablon;
assign(f_sablon_precis,'date\' +copy(numede_transmis.1,3)+'_s___trt');
rewrite(f_sablon_precis);
for i:=0 to Factor_de_multipl*_Nr_esant_in_sablon do
  begin
    x:=Sablon_precis[i];
    write(f_sablon_precis,x);
  end;
close(f_sablon_precis);
assign(f_sablon_precis,'date\' +copy(numede_transmis.1,3)+'_4s___trt');
rewrite(f_sablon_precis);
for i:=0 to Factor_de_multipl*_Nr_esant_in_sablon do
  begin
    x:=(Sablon_precis[i]+Sablon_precis[i+1]+Sablon_precis[i+2]+Sablon_precis[i+3])/4;
    write(f_sablon_precis,x);
  end;
close(f_sablon_precis);
assign(f_sablon_precis,'date\' +copy(numede_transmis.1,3)+'_4s___trt');
rewrite(f_sablon_precis);
for i:=0 to Factor_de_multipl*_Nr_esant_in_sablon do
  begin
    x:=(Sablon_precis[i]+Sablon_precis[i+1]+Sablon_precis[i+2]+Sablon_precis[i+3]
      +Sablon_precis[i+4]+Sablon_precis[i+5]+Sablon_precis[i+6]+Sablon_precis[i+7])/8;
    write(f_sablon_precis,x);
  end;
close(f_sablon_precis);
end;
}-----}
function EPS:real;
{Se deplaseaza "Sablonul" real. Sumarea trebuie sa se produca deci numai pe interiorul
"Sablon_perfect" intersectat cu "Sablonul". Aceasta intersectie trebuie sa contina in
mod obligatoriu complexul QRS. Dar acest lucru face sa avem numar diferiti de
termeni la calculul EPS. Scade o data cu depl spre dreapta . Trebuie deci sa incepem
de la inceput pe o fereastră mai mica, egala dacă se poate chiar cu dimens
"Sablon_perfect"- dimens maxima a deplasării "Sablonul" real la stinga si la dreapta }

```

```

VAR
  i      :integer;
  x      :real;
begin
  x:=0;
  for i:=Margine_EPS to Nr_esant_in_sablon-Margine_EPS do begin
  x:=x+
    (Sablon_precis[i*Factor_de_multipl+ecart]-(sablon_0min[Nr_sablon,i]+Shift_Y))*
    (Sablon_precis[i*Factor_de_multipl+ecart]-(sablon_0min[Nr_sablon,i]+Shift_Y));
  end;
  x:=x/(Nr_esant_in_sablon-2*Margine_EPS);
  EPS:=x;
end;
{-----}
procedure CAUTA_DUPA_X;
VAR i      :integer;
begin
  i:=0; treceri_prin_min :=0;
  repeat
    Epsilon_X[ecart]:=EPS;
    D:= Epsilon_X[ecart] - Epsilon_X[ecart-1];
    if sens then inc(ecart)
      else begin dec(ecart);D:=-D;end;
    if ecart>Margine_EPS*Factor_de_multipl
      then begin ecart:=Margine_EPS*Factor_de_multipl;sens:=not(sens);end;
    if ecart>Margine_EPS*Factor_de_multipl
      then begin ecart:=Margine_EPS*Factor_de_multipl;sens:=not(sens);end;
    if ecart<-(Margine_EPS*Factor_de_multipl)
      then begin ecart:=-(Margine_EPS*Factor_de_multipl);sens:=not(sens);end;
    if D>0 then
      begin
        inc(i);
        if i>=Nr_dupa_minim then begin i:=0; sens:=not(sens);end;
      end
      else i:=0;
    if ecart_min=ecart then inc(treceri_prin_min);
    if Epsilon_X[ecart]<min_epsilon[Nr_sablon] then
      begin min_epsilon[Nr_sablon]:=Epsilon_X[ecart];   ecart_min:=ecart; end;
  until treceri_prin_min >=3;
  ecart:=ecart_min;  {pregatim cautarea pe verticala}
  sens:=true;
  treceri_prin_min:=0;
end;
{-----}
procedure CAUTA_DUPA_Y;
VAR
  i      :integer;
begin
  i:=0;
  repeat
    Epsilon_Y[ind_shift]:=EPS;
    D:= Epsilon_Y[ind_shift] - Epsilon_Y[ind_shift-1];
    if ind_shift_min=ind_shift then inc(treceri_prin_min);
    if Epsilon_Y[ind_shift]<min_epsilon[Nr_sablon] then begin
      min_epsilon[Nr_sablon]:=Epsilon_Y[ind_shift];
      ind_shift_min:=ind_shift; end;
    if sens then inc(ind_shift) else begin dec(ind_shift);D:=-D;end;
  SHIFT_Y:=ind_shift*D;

```

```

if ind_shift>50 then begin ind_shift:=50;sens:=not(sens);end;
if ind_shift<-50 then begin ind_shift:=-50;sens:=not(sens);end;
if D>0 then
  begin
    inc(i);
    if i>=5 then begin i:=0; sens:=not(sens);end;
  end
  else i:=0;
until treceri_prin_min >=4;
ecart:=ecart_min; {pregatim cautarea pe verticala}
sens:=true;
treceri_prin_min:=0;
SHIFT_Y:=ind_shift_min*Dy;
end;
{-----}
function SABLON_ABERANT:boolean;
VAR
  i :integer;
begin
  if Min_Epsilon[Nr_sablon]>2*maxim_epsilon then
    begin
      Sablon_aberant:=true;
      sound(500);delay(500);nosound;
    end;
end;
{-----}
procedure REFAC_SABLON_PERFECT;
  {desi fereastra de calcul a erorii a fost mai mica vom pune toate valorile
  sablonului real in cel "perfect", mai putin ecartul la unul din capete}
VAR
  i :integer;
begin
  for i:=0 to Nr_esant_in_sablon do
    begin {mai putin ecartul la capatul din dreapta}
      if i*Factor_de_multipl+ecart< Factor_de_multipl*Nr_esant_in_sablon then
        begin
          if Flaguri_sablon_precis[i*Factor_de_multipl+ecart] then
            begin
              {deci daca a "picat" peste unul deja inlocuit sau existent dintr-un sablon
              real avem ca solutii:
              1. sa nu-l mai schimbam;
              2. sa facem o medie cu cel precedent
              3. sa-l inlocuim cu ultimele valori. Am ales-o pe prima.}
              Sablon_precis[i*Factor_de_multipl+ecart]:=
              (Sablon_precis[i*Factor_de_multipl+ecart]+ shift_Y+sablon_0min[Nr_sablon.i])/2;
            end
          else
            begin {daca nu a fost o valoare reala se va inlocui si se vor seta flagurile}
              Sablon_precis[i*Factor_de_multipl+ecart]:=shift_Y+sablon_0min[Nr_sablon.i].
              Flaguri_sablon_precis[i*Factor_de_multipl+ecart]:=true;
            end;
          end;
        end;
      end;
    end;
end;
BEGIN {of program SABLON}
  IdentInitGraph;
  INITIALIZEZ;

```

```
CREAZA_PUNCTE_INTERMEDIARE;
if Nr_sablon_copiat=Nr_sablon then inc(Nr_sablon);
repeat
  repeat
    CAUTA_DUPA_X;
    CAUTA_DUPA_Y;
    test:=Sablon_aberant;
    if ((Nr_sablon>=2) or not(Sablon_aberant)) then
      begin
        REFAC_SABLON_PERFECT;
        Shift_Y:=0;
        CREAZA_PUNCTE_INTERMEDIARE;
        proc_52;{afisare}
      end;
    inc(refac_cautarea);
  until refac_cautarea>=2;
  inc(Nr_sablon);
until Nr_sablon=Nr_de_sabloane;
OutTextXY(1,10,'                S-au epuizat sabloanele !!! ');
OutTextXY(1,180,'                Iesiti cu ENTER          ');
readln;
CloseGraph;
END.
```