

ОТПЕЧАТВАНЕ НА ЕЛЕКТРОКАРДИОГРАМИ С МИКРОТОЧКОВ ТЕРМОПРИНТЕР
 ст.н.с. I ст. Иван Доцински д.т.н., Медицински факултет - София
 1993

Първото съобщение за запис на електрокардиограма (ЕКГ) с микроточков термопринтер (МТ) е през 1985 [1]. Сега този начин на представяне на данните е широко разпространен в съвременните електрокардиографи, поради неоспоримите си предимства.

Всички МТ имат топлинна глава и мотор за движение на хартията. Използуваният МТ е 480Z-N GOTEKBA с еднобайтова паралелна организация на прехвърляне на данните в 832-битов преместващ регистър на топлинната глава (ПР).

Софтуерните проблеми по управление на МТ се състоят в:

- подходящо разположение на многоканалните (напр. 3-канални) записи по вертикала, дефинирана от броя на точките на главата;
- създаване на втора стойност към дискрета от АЦП, ако са необходими свързващи точки по вертикала между два съседни дискрета;
- сравнение между симултанни двойки стойности (дискрет и създадена) в каналите и обединяването им в случай на припокриване;
- формиране на графична линия по време на прехвърляне на данни в ПР, като на мястото и вътре в двойките стойности се въвежда 1 (черни точки), а навсякъде другаде 0 (бели точки).

Общоприетият начин за въвеждане на свързващи точки е показан на фиг. 1. Дискретите са отбелязани с *, вторите стойности - с #, а разстоянието между тях - с х.

Следващата фигура представя възможни комбинации от триканални двойки стойности. Припокриването е белег за пресичане на графики.

*	*	1	1	0				
х#	х	1	1	0				
хх	#	*	1	1	0			
#х	х	х	_ 1	_ 1	_ 0			
* х	*	х*	#	0	1	0		
х х	х	#х	х*	1	0	1	1	
х *	х	х	хх	0	0	0	0	
х #	х	х	*х*	3	0	0	1	
*	х х	#	#	х#	а	0	0	0
* *** #	х ***	#	#	#	я	0	0	1
* х *	*	#	#	#	т	0	0	0
*	#	*	#	#	_ 0	_ 0	_ 0	_ 0

фиг. 1

фиг. 2

фиг. 3

Разработеният алгоритъм за формиране на графична линия използва предварително определени размери на черни и бели ивици. Данните се въвеждат в ПР отдолу нагоре, като старшият бит заема най-ниската позиция в байта. Процедурата е много бърза при цели байтове 00000000 или 11111111 (лявата част на фиг. 3), но за съжаление се срещат и смесени байтове (другите части на фигурата).

Алгоритъмът се състои от следните стъпки:

1. На 8-те входа на ПР се подава 00000000. Дължината на първата бяла ивица се поставя в първия от двата използвани регистъра, след което се преминава в 3.

2. Генерира се сигнал за въвеждане на данните. На 8-те входа на ПР се подава 00000000.

3. От първия регистър се изважда 8. Ако флагът за пренос е чист, се генерира сигнал за въвеждане на данните и действието се повтаря. Когато флагът се вдигне се преминава в 4.

4. Числото в допълнителен до две код, което се съдържа в първия регистър, се прехвърля във втория.

5. Първият регистър се зарежда със следващата черна ивица. Когато няма повече ивици се преминава в 12.

6. Данните на входа на ПР се преместват наляво, след което младшият бит се установява в 1. Първият регистър се намалява с 1, а вторият се увеличава с 1. Това се повтаря, докато единият от тях се нулира. Ако това е първият, се преминава в 10. В другия случай се преминава в 7.

7. Генерира се сигнал за въвеждане на данните. На 8-те входа на ПР се подава 11111111.

8. От първия регистър се изважда 8. Ако флагът за пренос е чист, се генерира сигнал за въвеждане на данните и действието се повтаря. Когато флагът се вдигне се преминава в 9.

9. Числото в допълнителен до две код, което се съдържа в първия регистър, се прехвърля във втория.

10. Първият регистър се зарежда със следващата бяла ивица. Когато няма повече ивици се преминава в 12.

11. Данните на входа на ПР се преместват наляво (младшият бит става 0). Първият регистър се намалява с 1, а вторият се увеличава с 1. Това се повтаря, докато единият от тях се нулира. Ако това е първият, се преминава в 5. В другия случай се преминава в 2.

12. Край на алгоритъма.

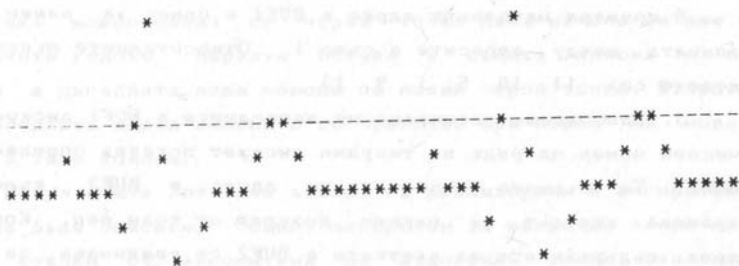
Алгоритъмът възприема целия байт в една от следващите

конфигурации 00000000 1xxxxxxx и 11111111 0xxxxxxx като смесен.

Разработен бе и друг алгоритъм, който използва таблици, съдържащи образи на смесени байтове. Той е сложен, но е по-бърз само при байтове, като този в средната част на фиг. 3, поради което не е показан тук. Инак основание да се смята, че оптималното решение е комбинация от двата алгоритъма с различни клонове за отделните видове смесени байтове.

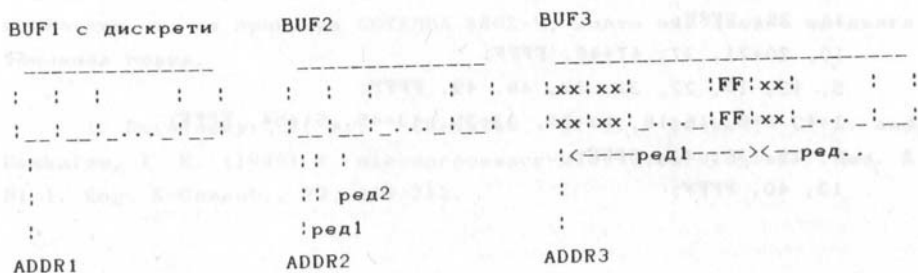
Естественият начин за печатане на ЕКГ е по оста на движение на хартията, но един напречен запис извън реално време може да представя симултанно приети сигнали като 12 канална информация.

Тривиалният подход за формиране на графична линия, напр. от дискретите на реда, отбелязан с тирета на фиг. 4 се състои в: 1) определяне на нивото на реда; 2) сравнение на амплитудите на всички дискрети в отвеждането с това ниво; 3) въвеждане на 1 в местата, където има съвпадение и 0 навсякъде другаде. За съжаление последователната проверка е ужасно бавна, а създаването на свързващи точки (непоказани на фигурата) е допълнителна трудност.



фиг. 4

По-надолу са изложени принципите на подход и алгоритми, които позволяват изключително бърз напречен печат.



фиг. 5

1. Еднопасово преброяване на дискретите и получаване на частични суми за всеки ред. Двубайтовите дискрети са в буфера BUF1 с начален адрес ADDR1 (фиг. 5). Сумите се поставят последователно в двубайтовия буфер BUF2 с адрес ADDR2 за първия най-горен ред. Ефективните адреси на сумите на редовете се получават, като към базовия адрес ADDR2 се добавя номера на реда, който се дефинира с разликата между максималната амплитуда на дискретите (най-горния ред) и амплитудата на текущия дискрет.

На фиг. 4 сумите на редовете са: 2, 0, 0, 6, 8, 30, 4, 2.

2. Разпределение на паметта на двубайтовия буфер BUF3 за съхраняване на адресите на дискретите, подредени по редове със сканиране отляво надясно и отгоре надолу. При напречния запис адресите на дискретите съвпадат с посоката на формиране на графичната линия и ще се използват за изчисляване на черни и бели ивици. Първият адрес на първия ред се поставя в клетката, определена от началния адрес на BUF3. Местото на клетката с първия адрес на втория ред се получава чрез добавяне на сумата на дискретите в първия ред, увеличена с 2 за двубайтова стоп-марка и т.н. Началните адреси на редовете се зареждат в BUF2 на мястото на използваните вече техни суми.

В примера началният адрес в BUF1 е приет за равен на 1, а разликата между адресите е също 1. Относителните първи адреси на редовете са: 11, 10, 5, 1, 9, 13.

3. Еднопасово сканиране на записаните в BUF1 дискрети. Всеки изчислен номер на реда на текущия дискрет показва определена клетка в BUF2. Тя съдържа първоначално адрес в BUF3, където ще се съхранява адресът на първия дискрет от този ред. Когато той се открие, съдържанието на клетката в BUF2 се увеличава, за да обнови адреса на клетката, където ще се постави адреса на следващия дискрет от същия ред. Сортирането на дискретите на BUF1 завършва с стоп-марката FFFF в края на всички редове.

В примера адресите на дискретите във всеки ред са:

11, 38, FFFF;

10, 20+21, 37, 47+48, FFFF;

5, 12, 19, 22, 32, 39, 46, 49, FFFF;

1+4, 6+8, 16+18, 23+31, 33+35, 43+45, 51+54, FFFF;

9, 15, 36, 42, FFFF;

13, 40, FFFF;

4. Изчисляване на дължината на лентите. Дължината на черната лента е равна на броя на монотонно нарастващи адреси на дискрети. Тя може да бъде и 1. Дължината на бялата ивица се получава, като се намали с 1 разликата между адресите на два съседни дискрета. Въвежда се допълнителна бяла ивица, когато началният или крайният адрес в един ред не съвпада с началния, съответно крайния адрес на дискретите в BUF1.

Дължините на лентите от примера са представени по-долу, като черните са подчертани. Относителният краен адрес е 54.

10, 1, 26, 1, 15;

9, 1, 9, 2, 15, 1, 9, 2, 5;

4, 1, 6, 1, 4, 1, 2, 1, 9, 1, 6, 1, 6, 1, 2, 1, 4;

4, 1, 3, 7, 3, 4, 2, 1, 3, 7, 3, 5, 4;

8, 1, 5, 1, 10, 1, 5, 1, 11;

12, 1, 26, 1, 13;

Два реда, вторият и третият, не са представени, поради липса на свързващи точки.

Анализът на фиг. 1 от гледна точка на напречен запис показва, че всяка новопоявила се черна точка дава начало на две следи в по-долните редове. Първата остава в същата колона и изчезва, когато в съседната лява колона се появи черна точка. Втората следа е в съседната дясна колона и се пренахва при поява на нова черна точка в тази колона.

Резултатите от този анализ са реализирани в алгоритъм, който няма да бъде описван. Общият алгоритъм за напречен запис използва някои стъпки от алгоритъма за надлъжния запис, отнасящи се до прехвърлянето на черни и бели точки в ПР.

Програмите са написани на Assembler за процесор 6809 със стандартна тактова честота 1 MHz. На следващите фигури са показани надлъжен и напречен запис на имитирана ЕКГ. Резултатите са приложени и за принтера GOTE MBA 880Z-N, който има двойно по-дълга топлинна глава.

I. Dotsinsky, I. A., Christov, I. I., Levkov, C. L. and Daskalov, I. K. (1985) A microprocessor-electrocardiograph. Med. & Biol. Eng. & Comput., 23, 209-212.

