

AFIT/GE/BE/78-27

LOCATION OF DISASSOCIATED P WAVE
IN AN ELECTROCARDIOGRAM

THESIS.

AFIT/GE/BE/78-27 Charles E. Hightower
Capt. USAF

JUN 4 1979

Approved for public release; distribution unlimited

012225

Approved for public release; distribution unlimited

Preface

The work described in this thesis accomplishes two tasks. First, two electrocardiograms (ECG) of a heart progressing from near normal to P wave disassociation were recorded. This furnished the data to test an algorithm designed to identify P waves. The algorithm relied on the P wave being associated with the R wave only during the first two heart beats. After this a routine of "stepping" the distance between P waves (P-P interval) down the ECG was used. After each "step" a search was made for the peak of the P wave.

I would like to thank those who have contributed their time and guidance in this research. I give special thanks to my thesis advisor, Dr. Mathew Kabrisky, who proposed the study and gave continued guidance and support. Edwin L. Stanley, M.D., and Staff at Cox Heart Institute gave valuable guidance. R. Karl Kardenat, M.S., and his research team prepared the dogs for the operation, performed the operations, and operated their test equipment. Shawnee Klein assisted in the editing.

I also wish to thank my wife, Cecilia, and son, John, for their support and giving up my time with them.

Charles E. Hightower

Table of Contents

	Page
Preface	ii
List of Figures	v
List of Tables	viii
Abstract	ix
I. Introduction	1
Background	1
Purpose	4
Assumptions	5
Approach	6
II. Data Collection	7
General Preparations	7
Blockage Technique	13
Toxicity Technique	16
Analog to Digital Conversion	16
III. Computer Program	18
Background	18
Initializing the Program	20
Reading the Data	20
Initial P-P Interval	21
Remaining P-P Intervals	24
Output	25
Continuing the Main Program	27
IV. Results	30
Data Collection	30
Computer Program	32
V. Conclusion and Recommendation	39
Conclusion	39
Recommendation	40
Bibliography	42
Appendixes	
A. Table of Real Time ECG Monitors	43

	Page
B. Block Diagram of Equipment Used to Record ECG	46
C. Data on Dogs	48
D. Program to Convert Data Into Eight Place Data Field	50
E. Computer Program to Find P Wave	53
Vita	85

List of Figures

Figure	Page
1. Longitudinal cross section of a heart showing S-A Node, A-V Node, A-V Bundle, and Purkinje fibers	2
2. Normal ECG Wave form	2
3. ECG's showing the three degrees of block (a) first degree block, (b) second degree block, (c) third degree block or disassociated P wave	3
4. Cathode ray tube monitor showed the wave shapes from the internal heart lead, external chest lead, and blood pressure	9
5. The tip of the internal heart lead was a small piece of solder on the end of wire in a catheter sleeve	10
6. The complete internal lead and catheter system is shown	11
7. The placement of the internal lead was monitored using a fluoroscope x-ray. The faint thin line is the internal lead. The round spot with a dark line coming from it is the external chest lead	11
8. The catheter system used in the blockage technique was 19.7 inches long	14
9. This ECG is from a heart progressing from near normal (a) to disassociated P wave (b).	18
10. A wave shape being analyzed by subroutine MAX to find the maximum points. The wave shapes begin with a positive slope (a), changes to flat (b and c), and changes to negative (d)	23
11. An analyzed ECG having a constant P-P interval then a changed interval	26
12. How the P-P interval is transferred to another data block	28
13. (a) Dog 2's ECG has a negative P wave. (b) Dog 3's ECG has a positive P wave. (c) Dog 3's ECG has a data drop out occurring at a 1.6 Hz rate	31

Figure	Page
14. The violence around the P wave disassociation experienced on Dog 3's ECG	32
15. The computer program picked out the P wave in the first few points in this data block . .	33
16. The computer program picked out the peak of the P wave just before the spike associated with a data drop out	34
17. The computer program identified the remains of a P wave after a data drop out, but this caused a slightly long P-P interval. By the third P wave in the data block the peak of the P wave had been reidentified	35
18. Data block 43 has strang data	36
19. This is the combination of data blocks 74 and 75. Data block 74 is presented up to the broken line	38
20. Block diagram of the equipment used to record ECGs	47
21. Program used to change data format	51
22. List of Control cards used to execute the program listed in Figure 21	51
23. This part of the computer program sets parameters	54
24. The code starting the interpreting loop . . .	56
25. Subroutine READ reads a data block of 512 points into memory	56
26. Parameters to find the maximum point on the wave forms are set	58
27. Subroutine MAX finds the peaks of the wave forms	60
28. The code identifying the initial P waves and initial P-P interval	62
29. Parameters to find the remaining P waves are set	63

Figure	Page
30. How the main program finds the P-P interval .	65
31. Subroutine GREATER returns the location of the P wave	66
32. Output is basically a call to subroutine MARK	68
33. Subroutine MARK draws a graph of the data . .	70
34. Subroutine AXIS prepares the CRT screen to plot the data	71
35. Subroutine AXIS is continued	72
36. Subroutine PMARK marks P waves in the output .	74
37. The main program is ended	74

List of Tables

Table	Page
1. Real Time ECG Monitors	44
2. Data on Dogs	49

Abstract

Heart failure was induced in one dog by a blockage of the anterior descending branch of the coronary ostium. Heart failure was induced in another dog by overdose of digitalis. The electrocardiogram (ECG) from each dog was recorded on an analog magnetic tape using an AMPEX Model FR1300 recorder. An algorithm to locate and identify the P wave using the periodicity characteristic of the P wave was tested against this data. The P waves in the first two heart beats are identified by their proximity to the R wave. The distance between these wave shapes (P-P interval) is "stepped" down the ECG. At each "step" a search is initiated for the peak of the P wave. If a peak is not found within a set number of data points, the last P-P interval is retained. The accuracy of the algorithm is excellent until an extreme abnormal wave shape interferes with the ECG around the area of the P wave. When this happens, the algorithm has a tendency to stop identifying P waves. The results from the computer program failures show that more thorough knowledge of the relationships between the P wave and other wave shapes on the ECG need to be known.

LOCATION OF DISASSOCIATED P WAVE IN AN ELECTROCARDIOGRAM

I. Introduction

Background

When a patient is admitted to a Cardiac Care Unit (CCU), his electrocardiogram (ECG) is monitored at different time intervals depending upon the severity of the diagnosed heart problem. Until recently, this monitoring was done by nurses in the CCU. Computer programs which can do the monitoring are now available. These programs are used to inform and prepare the nursing staff for changes toward a worsening patient condition.

An ECG is an analog recording of electrical potentials from the heart as it depolarizes and repolarizes. The impulse to contract begins in the S-A node (See Figure 1), and travels around the atrium, or upper chamber of the heart, to the A-V node. The A-V node is the only conduction point for this impulse to the ventricles, or lower chambers of the heart. At this node, the impulse is delayed and then conducted down the A-V bundle to the Purkinje fibers, which transmit the impulse to the remainder of the heart.

The different wave shapes on an ECG correspond to the depolarization or repolarization of the atrium and ventricles. The QRS complex is the depolarization of the ventricles prior to contraction. The T wave is the repolarization of the ventricles. The wave form depicting the repolarization of

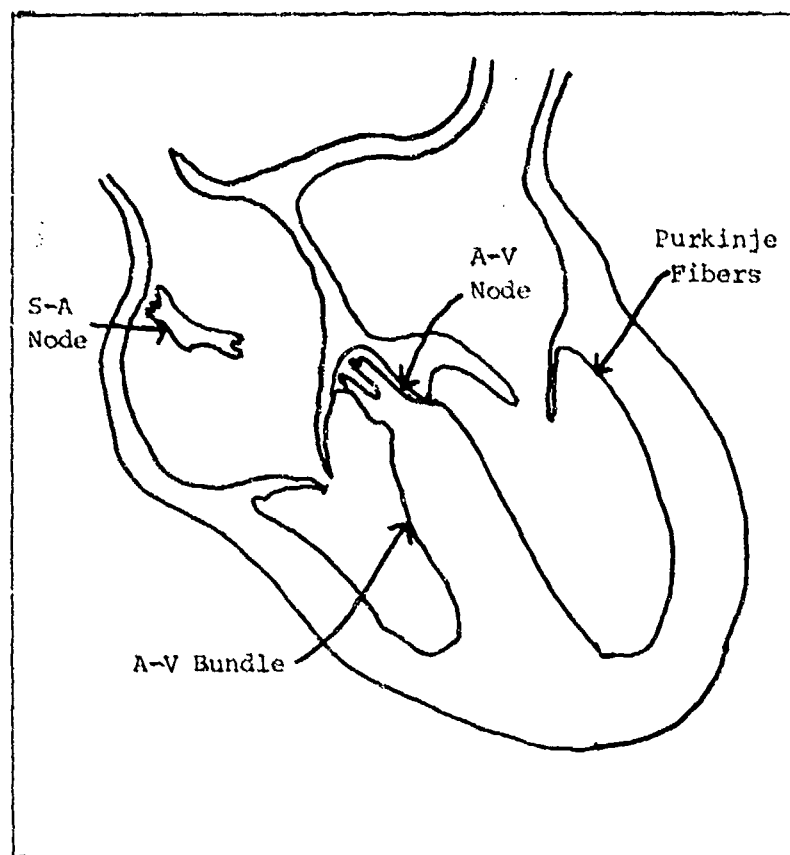


Fig. 1. Longitudinal cross section of a heart showing S-A Node, A-V Node, A-V Bundle, and Purkinje fibers.

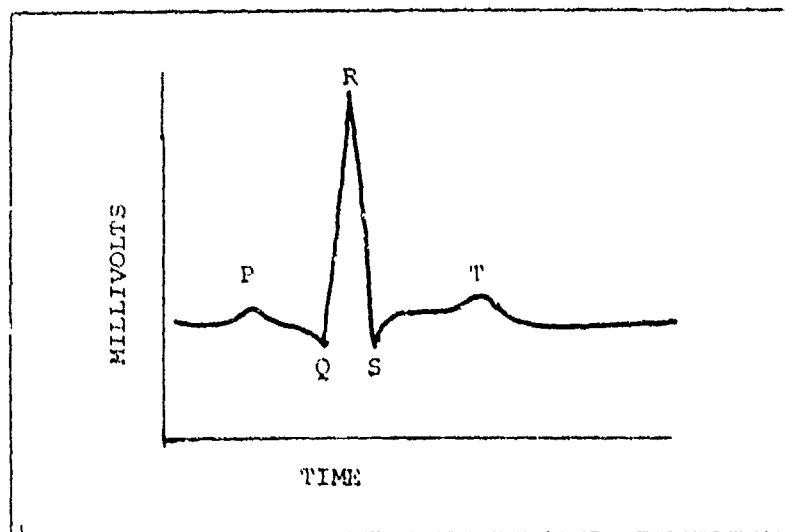


Fig. 2. Normal ECG wave form.

the atria is hidden by the QRS complex. In a normal ECG, as illustrated in Figure 2, the P-R segment is a relatively fixed interval. Because the A-V node is the only conduction path for the contraction impulse to the ventricle, the P-R segment will lengthen if this node starts delaying the signal more than normal or if the node becomes completely blocked. When this interval, usually about 0.16 seconds, lengthens to a value above 0.20 seconds, the patient has a first degree block (see Figure 3(a)). When the length increases to 0.25 to 0.45 seconds, the A-V node is sufficiently blocked to completely stop an occasional impulse. This will cause the ECG to have a P wave with no following QRS complex. When this occurs, the patient is said to have a second degree block (see Figure 3(b)). A third degree block occurs when the A-V node is completely blocked, and the atrium and ventricles are beating at two different rates.

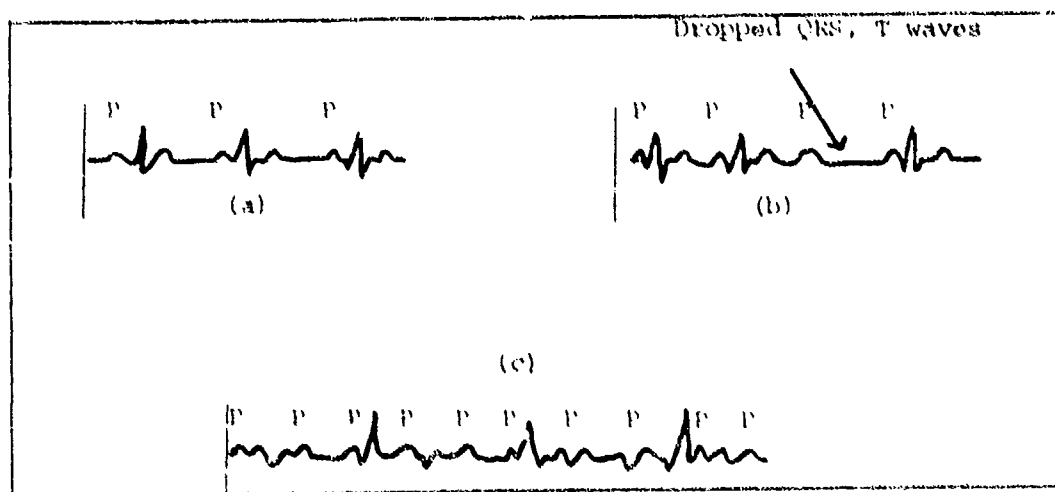


Fig. 3. ECG's showing the three degrees of block (a) first degree block, (b) second degree block (c) third degree block or disassociated P wave. (Ref 4:215)

This is termed a disassociated P wave or heart blockage (see Figure 3(c)), depending on other factors (Ref 4:215).

At the present time there are computer programs designed to warn of many cardiac problems (Ref 3, 6, 9). Appendix A is a table of published information on real-time rhythm monitors. However, there is no published system to detect P waves (Ref 2:293).

Two previous students at the Air Force Institute of Technology, Flick (Ref 1) and Pairett (Ref 7), developed programs to detect the P wave, but both algorithms failed to find the disassociated P wave. Pairett assumed that the P wave would be directly in front of the QRS complex. His program achieved 96% accuracy in finding the P wave as long as the P wave was associated. Flick did not make the P wave association assumption. He describes two programs in his thesis. The first program selected a set of waves for a feature selection process. This program also obtained frequency domain data from the P wave, T wave, and other sections of the ECG. The second program located the P wave using frequency domain and rhythm analysis. Flick's program had a success rate of 86%.

Purpose

This thesis has a two-fold purpose. The first purpose is to collect ECG data from hearts that lie on the spectrum from nearly normal to cardiac failure. Cardiac failure is defined in Guyton's Textbook of Medical Physiology (Ref 4:

332) as "failure of the heart to pump blood adequately."

The second purpose of this thesis is to develop a computer program to find the disassociated P wave. This program would require minimal intervention from an operator and would not be based on identification of the P wave or its location with respect to other ECG wave shapes.

Assumptions

The ECG for a heart proceeding from near-normal to death was taken from a dog. This was done with the knowledge that a dog's heart produces an ECG similar to a human's. It was assumed that the dog's increased heart rate compared to a man's, would not hide any essential data. Cardiac failure was induced by blockage and an overdose of drugs on the assumption that an ECG produced by these two techniques would be representative of ECG's recorded in a CCU.

The computer program is based on the following assumptions:

1. The heart-beat rate does not change faster than 0.01 seconds over a period of three seconds.
2. This program is a subroutine for a larger ECG monitoring program. and the main program and this program can be made compatible.
3. There is not any important information in the ECG wave forms above 100 Hertz.
4. When this program is initiated the patient is in good enough condition to produce two consecutive

heart beats without PVC's and the patient does not have a third degree blockage.

Approach

To gather the data, two techniques were used. In the first technique the dog was anesthetized and a catheter was inserted in an artery feeding the heart muscle. This procedure produced a blockage type ECG. In the second technique the dog was also anesthetized, but an overdose of Cedilanid-D, digitalis, was given. This caused a different type of ECG. The ECG's were digitized and prepared for computer analysis.

The computer program to analyze the data finds the P, QRS, and T waves in the first two heart beats. From this identification, the P-P interval, the distance between P waves, is found. This P-P interval is then "stepped" on through the program. An interval around the expected P wave is tested for a peak value. Then another test is initiated to see if the area contains a PVC or some other abnormality that would produce an erroneous answer. Finally, a graph is drawn of the wave shapes and the identified P wave.

II. Data Collection

General Preparations

There was a strong need to have a collection of electrocardiograms (ECG) proceeding from near normal heart beat patterns to cardiac failure. This collection would contain all three stages of A-V blockage, Premature Ventricular Contractions (PVC), and as many other heart beat abnormalities as possible. This data would serve as a data library for any work done in the study of ECGs.

The ECG library at Cox Heart Institute had been searched, but a suitable collection of ECGs was not found. Therefore, the collection had to be generated. A dog's heart beat is similar to a human's except it beats at a faster rate. Hence, a cardiac failure could be induced into a dog and his ECG recorded on magnetic tape for future processing, storage, and analysis. Appendix B shows a block diagram of the equipment used to record the ECG.

The two dogs used in this experiment were obtained through Wright State Medical School research channels. Appendix C is a table showing what procedures were done to each dog prior to its arrival at Cox Heart Institute. One dog weighed 34.1 pounds and the other weighed 39.2 pounds. Both were male German Sheperds.

Prior to the experiment the dogs were washed and the intestinal tract emptied. Each dog was anesthetized with Sodium Pentothal, and shaved on the right hind leg and left

side of the neck before being brought into the experiment room. The dosage of Sodium Pentothal depended on the weight of the dog. Too much would kill the dog and too little would not anesthetize him sufficiently. Throughout the experiment, more Sodium Pentothal was administered if necessary.

The recording was done on an AMPEX portable analog recorder using one inch magnetic tape. Only three channels were used. Channel one recorded the ECG from a chest lead. Channel two recorded the electrical signal from a lead placed in an artery beside the heart muscle. Channel three was used as an editing signal for data retrieval.

The chest lead location was chosen to give maximum P wave amplitude and to conform as closely as possible to the positioning of ECG leads on a human in a Cardiac Care Unit (CCU). On one dog the maximum P wave amplitude was negative. This would serve to test any monitoring program against a negative P wave, since this form is used in CCUs under certain circumstances. The chest lead wave shape was monitored on a cathode ray tube (CRT) throughout the experiment (see Figure 4).

The research team led by Mr. Karl Kardenat from Wright State Medical School at Cox Heart Institute performed all of the surgery in this thesis. This surgery and all dog preparations used Cox Heart Institute facilities.

The lead inside the body was made by putting a small piece of solder on the end of a thin strand of insulated wire (see Figure 5). This wire was then put into a catheter

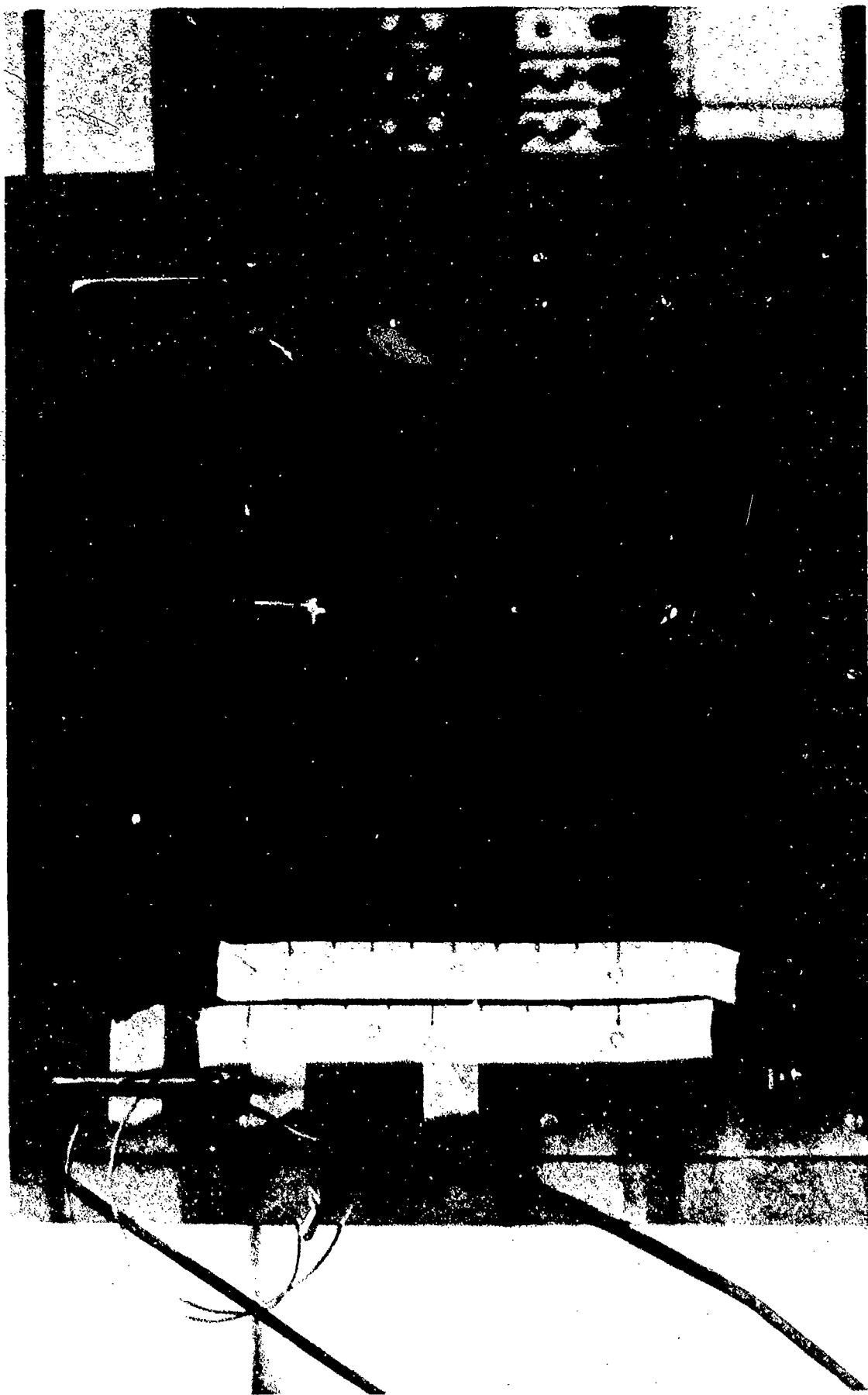


Fig. 4. Cathode ray tube monitor showed the wave shapes from the internal heart lead, external chest lead, and blood pressure.



Fig. 5. The tip of the internal heart lead was a small piece of solder on the end of wire in a catheter sleeve.

sleeve to help direct it down the artery (see Figure 6). Then a longitudinal incision was made over the carotid artery and the artery loosened from the surrounding tissue. Next, a small cut was made in the artery, and the artery clamped above the cut. The tip of the lead was inserted and guided down the artery using a fluoroscope x-ray to see its placement (see Figure 7). The arteries cannot be seen on the fluoroscope screen, therefore a technique of remote directing was used. The signal from this lead was displayed on the CRT monitor (see Figure 4). The lead's placement was adjusted to

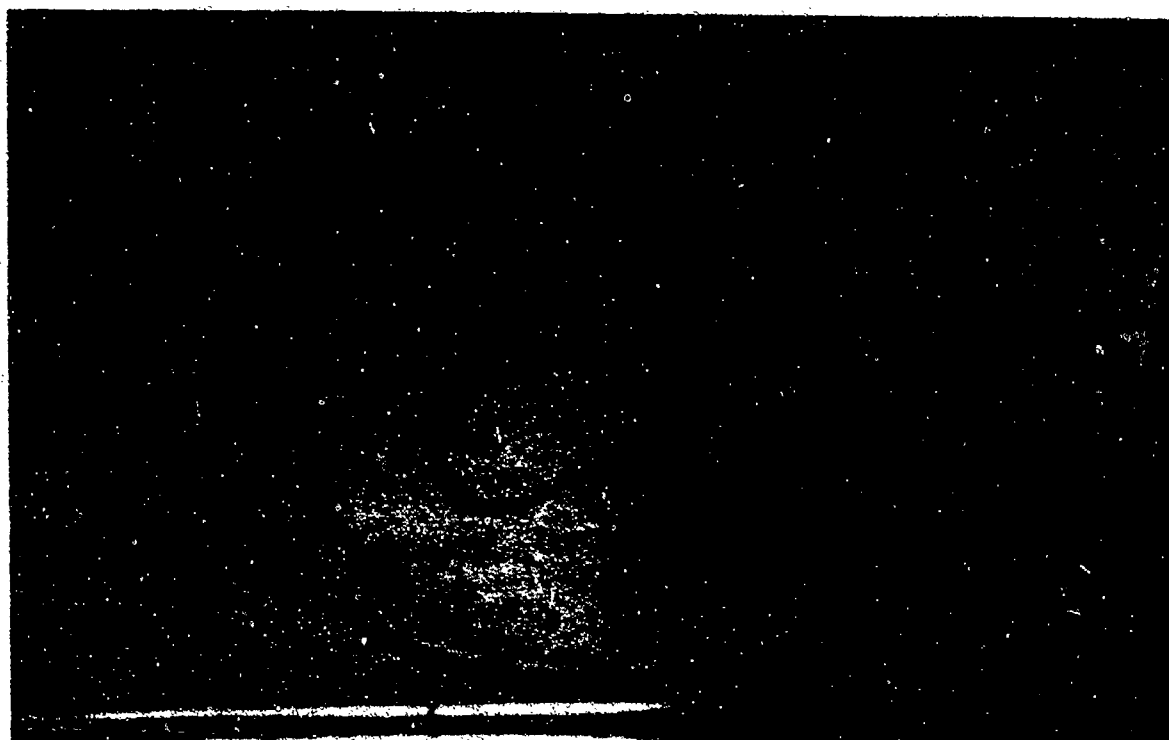


Fig. 6. The complete internal lead and catheter system is shown.

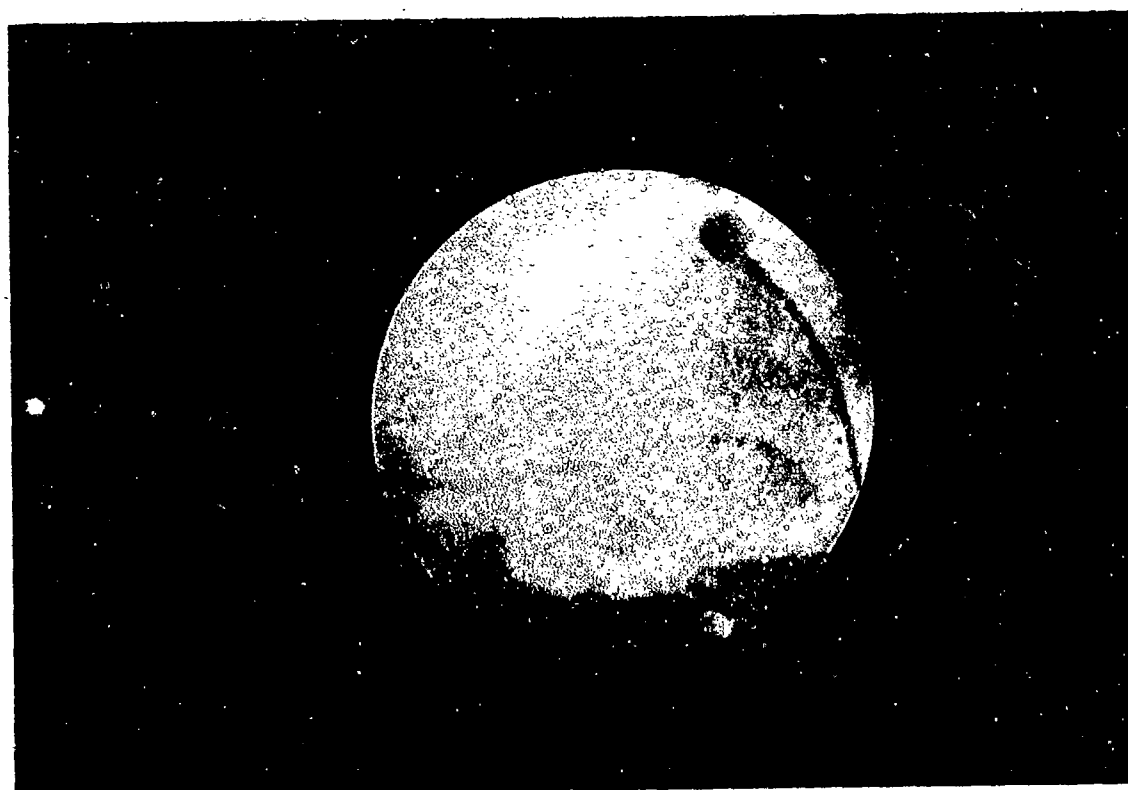


Fig. 7. The placement of the internal lead was monitored using a fluoroscope x-ray. The faint thin line is the internal lead. The round spot with a dark line coming from it is the external chest lead.

give a maximum P wave spike. Once this lead was in place, the artery was clamped around the lead wire to minimize blood loss. The wave form from this lead gave an amplified P wave compared to that seen on an ECG. In one case the wave form corresponding to the P wave was higher than the remaining wave shapes on the ECG. This lead would help point out the P waves on the ECG.

The third channel was an editing channel where a one volt signal on the tape would indicate a wave form to be digitized in the other data channels. This voltage came from a direct current power supply through a double-pole single throw switch. The input was monitored by a digital voltmeter at the recorder input.

Also displayed on the CRT was the systemic blood pressure (SBP) (see Figure 4). This wave form was collected by a transducer actuated by fluid pressure from the blood in an artery in the right hind leg. This wave shape was very important in monitoring the condition of the dog during the experiment.

With all the preparation done, the dog was ready for the experiment. Two means of inducing cardiac failure were used: one on each of the two dogs. The first means was through blockage of the artery feeding the A-V node section of the heart muscle. The second method was through an overdose of a toxic drug.

Blockage Technique

The blockage technique used a catheter with a coil of copper wire placed at the end of the inner catheter. The entire catheter system is made up of an outer radiopaque (Formacath) polyethylene catheter with an outer diameter of approximately 0.110 inches and a length of about 11.8 inches. The vinyl inner catheter had an outer diameter of approximately 0.065 inches and was 19.7 inches long. The back end of the inner catheter had a three-way stopcock attached. The outer catheter had an adapter connected to its back end that would allow the inner catheter to move with a slight amount of friction. The front end of the catheter was bent at a right angle to the coronary ostium plane, after the catheter was put at the root of the aorta. The inner catheter was then placed through the lumen of the outer catheter (see Figure 8).

The blockage device was a solid copper wire 0.02 inches in diameter that was bent to form a helical shape. The length of the helical wire was approximately 0.04 inches. The length of the coil helped determine the clotting time for a formation around the wire and, thus, the time to cardiac failure. From experiments performed by R. Karl Kordenat, this time could vary from an hour to several days (Ref 5:363). The coil is placed on the front end of the inner catheter. The coil should have some resistance from slipping off but still be loose enough for the outer catheter to push it off.

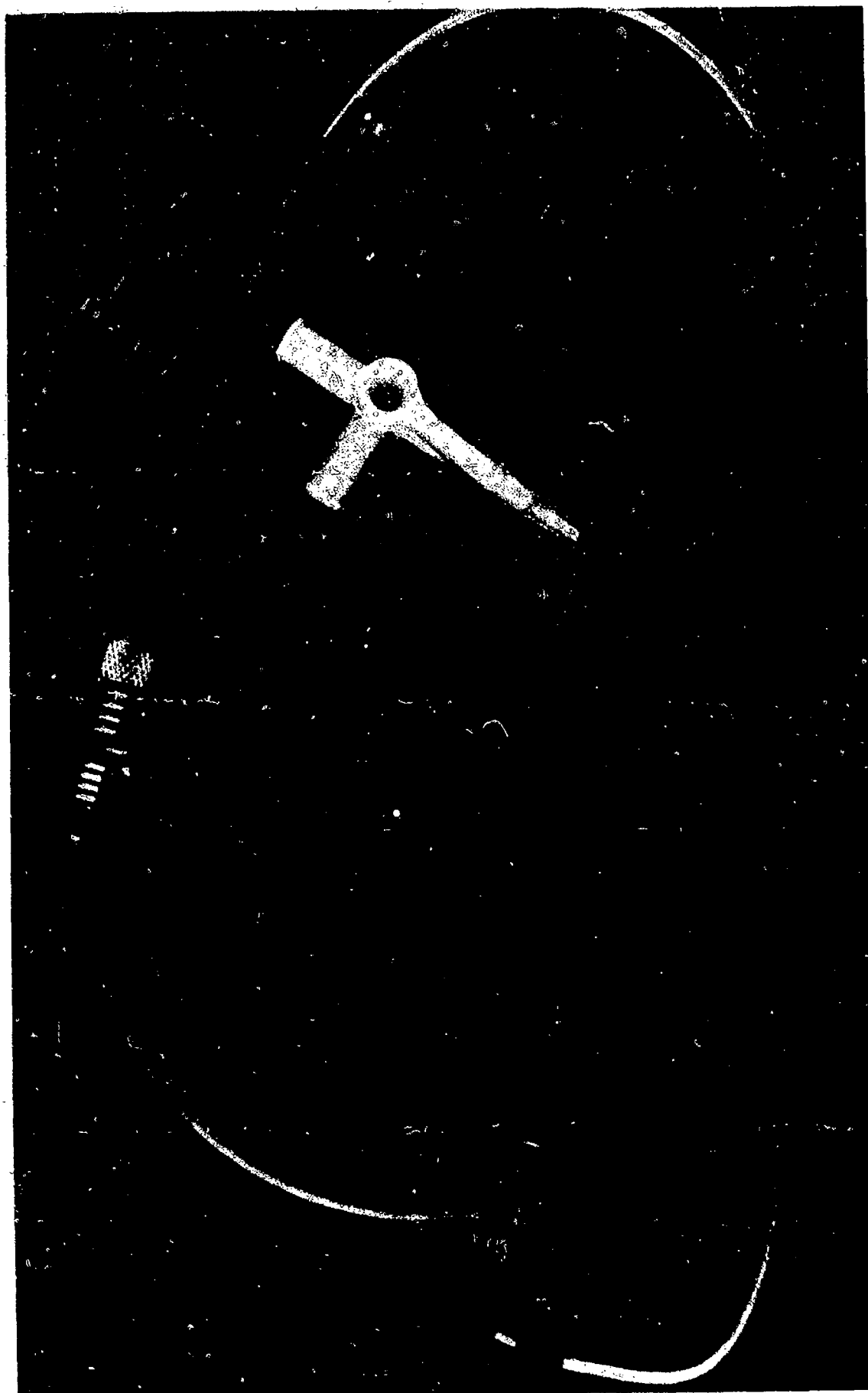


Fig. 6. The catheter system used in the blockage technique was 19.7 inches long.

Once the catheter system is prepared, it is placed in the same opening in the carotid artery as the internal ECG lead wire. The elasticity of the artery wall is sufficient and the ECG wire small enough to make this possible. Once again, under fluoroscopic control, the catheter is snaked through the artery system to the tip of the coronary ostium. The catheter is then directed into the anterior descending branch of the coronary ostium. A contrast media (Hypoque 75) is periodically injected through the inner catheter to help show the artery locations.

Once the tip of the inner catheter is in the desired position, the outer catheter is pushed forward, forcing the coil off the inner catheter. The coil is now free in the artery and can be "directed" to the desired position. Like the size of the wire coil, the position of the coil in the artery with respect to the branches off the artery is important for determining the time until cardiac failure. The position is also important in determining what wave forms the heart will produce prior to failure (Ref 5:363). For example, if placed too high in the anterior descending branch P wave disassociation has a reduced chance of happening.

Once the coil was in place, the heart gave good data for 55 minutes before sinus rhythm was lost and could not be regained. PVC's were observed about 30 minutes after the coil was released. Xylocaine and Pronestyl were administered to help the heart recover. A few minutes later Sodium Bicarbonate was also given to help. These drugs reduced the

occurrence of the PVC's and the heart began beating more in sinus rhythm. About 13 minutes later the PVC's returned and drugs had no effect on them.

Toxicity Technique

For this procedure the dog was prepared as described above. However, instead of inserting a helical coil, Ceditanid-D, digitalis, was injected into the bloodstream. The weight of the dog determined the dosage. An injection of 0.8 mg was made every ten minutes until a depressed S-T segment on the ECG was observed. This was observed 15 minutes after the first injection. Following the initial depressed S-T segment, three more injections were made at approximately one hour after initial injection. No drugs were given to prolong the heart action because of their reaction with the digitalis.

Analog to Digital Conversion

The data on the analog tape had to be put into a format which the digital computer would accept. To accomplish this, the analog tape was played back on a Bell and Howell VR3700B tape-deck set on the intermediate band width at a speed of 7-1/2 inches per second. This output was sent to a Comcore 5000 analog computer. This computer multiplied the incoming signal amplitude by 50 and then passed the signal through a capacitor to remove the base line drift. The signal was then passed through a 100 Hertz (Hz) low pass filter. This filter reduced signals above 100 Hz at a rate of 3 dB per octave.

This step was designed to remove most of the noise in the signal. At this point the signal path was split.

One signal went to a strip chart that gave a continuous readout of the signal. The second signal went to the Xerox Sigma 7 computer. This computer digitized the signal. The computer sampled the analog data every two milliseconds using 16 bits per sample. The bit precision exceeded the recommended digitization rate of 500 samples per second using 9 bits per sample set by the Committee of Electrocardiography of the American Heart Association (Ref 8). The number of data points per block of data (512) was chosen to fill the Control Data Corporation, CDC 6600 computer's buffer.

The digitized signal was then recorded on an on-line digital tape system. The digitized signal was also routed back to the strip chart through a digital to analog converter. This produced a read-out on the strip chart of both the continuous data and the sampled data.

An existing program was used to convert the data from Sigma 7's 32 bit word to CDC 6600's 16 bit word. The output of this conversion was written onto a magnetic tape. A program was then written to change the data on this tape to an eight digit field with an accuracy of two decimal places. This program is in Appendix C.

III. Computer Program

Background

The previous approaches used to solve the P wave identification problem used a function that transformed the wave shape into a different domain (Ref 1, 2). After comparing the wave shape in Figure 9(a) to that in Figure 9(b), there seems to be no resemblance between them. Therefore, comparing the different domain characteristics would lead to false identification.

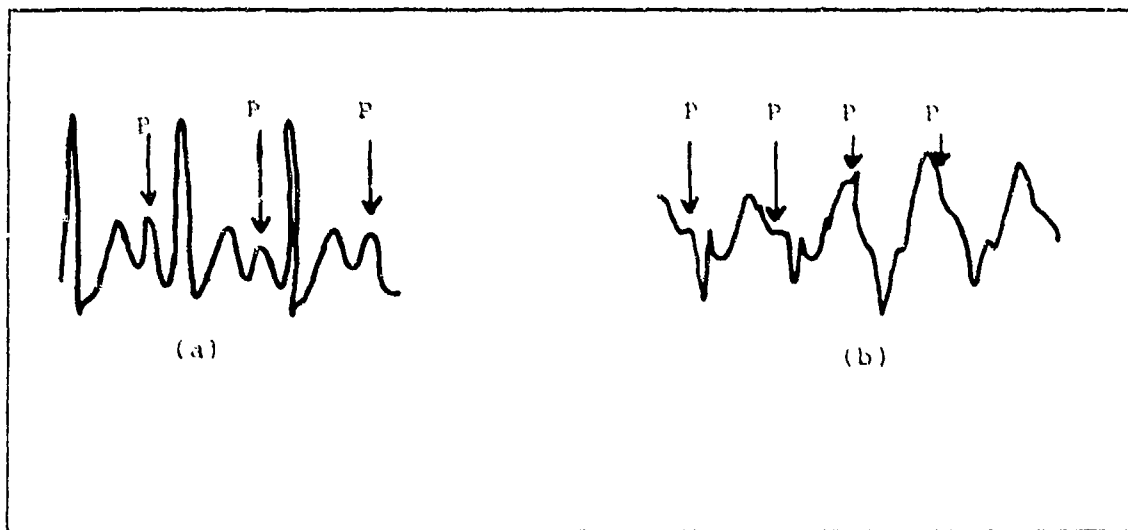


Fig. 9. This ECG is from a heart progressing from near normal (a) to disassociated P wave (b).

One important characteristic of the P wave is its periodicity. The distance between P waves, P-P interval, at P wave disassociation had changed 0.07 second from the P-P interval in a normal heart. The cardiologist typically uses this characteristic to identify the P waves. He finds the R wave and then looks for a peak above the noise about 0.16

seconds before the R wave and calls this peak a P wave. This procedure is repeated for the next heart beat. One point of a pair of dividers is placed on the peak of each P wave, giving the P-P interval.

With this interval known, the physician moves the dividers down the ECG looking for wave form peaks around each point. If the peaks become consistently different from where the divider point is, the heart rate has become faster (smaller P-P interval); or become slower (longer P-P interval). Then the interval represented by the dividers must be adjusted.

When the ECG contains a disassociated P wave, the interval on the dividers is virtually the only way to identify the P wave. The difficulty is that during disassociation, there are usually many other wave forms present; such as, Premature Ventricular Contractions (PVC), which cover or, at best, make the P wave less apparent (see Figure 9(b)). Frequently the P wave will not be apparent for several beats because it is "covered up" by wave shapes having higher amplitudes. In this case the dividers are just moved down the ECG as normal until a P-P interval can be identified. Then the distance on the dividers is adjusted, and this distance is moved back up the ECG looking for P waves.

This technique depends heavily on having at least two associated P waves before disassociation occurs, or the task becomes significantly harder. Even when having these two P waves present, a long period of time can be required to

find these P waves. This time requirement complicates the real-time analysis of ECG's.

Initializing the Program

This program is patterned after the approach used above. The program's list of variables, a copy of the program, and explanation of the program by sections is in Appendix E. The main points and unique elements of the program will now be covered.

The program is designed to require a minimum amount of interaction with the operator. However, the operator does have to enter the number of peaks in the ECG. This will come from looking at the patient's ECG strip. If the patient's heart beat has joined the P wave of the next beat, there will be three peaks. If the T and P have joined, there will be two peaks. The operator will also need to enter whether the P wave is positive or negative. This information comes from the patient's ECG. Whether the P wave is positive or negative, depends on which ECG lead placement scheme is used (Ref 4:195). If the P wave is positive, a zero (0) is entered; if the P wave is negative, a one (1) is entered. The operator will also enter which block of data he first wants to see, and how often he wants to see the output.

Reading the Data

Next the program calls the subroutine READ to read the digitized wave shape values in blocks of 512 data points. After the data is read into memory, a matrix of integer

data values is formed. This matrix is formed by ignoring all the values below a whole number. A rounding off process is not used because the exact maximum amplitude value is not important. If the P waves are positive, all negative data points are set equal to zero. If the P waves are negative, the absolute value of each data point is found. Then a threshold value below the peak value of the P wave is found, and all data below this threshold is set equal to zero. These two processes are designed to eliminate any noise spikes which would be identified as maximum values in the first program task.

Initial P-P Interval

After the data has been initially processed, a decision is made whether this is the first data block or not. If this is not the first data block, a P-P interval has been found and control is transferred further down the program to process the data. If this is the first data block, the first two P waves must be found and then the P-P interval determined.

To find the first two P waves, a subroutine MAX is called that identifies the maximum values in the first two heart beats. The basic idea is that the slope of the line, amplitude difference between two adjacent points, is found and compared to the last slope. The three conditions are possible: slopes are positive (upward); negative (downward); and zero (flat). All changes in slope are labeled as follows: a change to a

positive slope is labeled one (1); a change to a negative slope is labeled zero (0); and a zero slope does not change the slope label. The maximum point is the point before the slope label changes from one to zero.

Figure 10 is an example of how this subroutine works. A value of one (1) is initially assigned to the last slope variable, LSIGN. The amplitude difference between the last data point and the present data point is then found. If this difference is positive, the present slope variable, JSIGN, is assigned a one. The process continues with the remaining data points as long as the slope is positive. If the slope changes to zero, as in Figure 10 (b) and (c), the values of LSIGN and JSIGN remain the same. When the slope changes to negative (see Figure 10(d), JSIGN changes to zero. Since this is different from LSIGN (which is 1), a maximum point is defined as the last point where JSIGN and LSIGN equalled one (1), (MAX 1 in Figure 10). The subroutine continues until all the data in the data search window, as defined in other parts of program, has been searched for maximum points. If more than one maximum point is found (see Figure 10), a check is initiated to see how close the maximum points are. If they are within a set interval (3 data points in this program), the maximum value of these points is found and called the maximum point.

After the maximum points are found, the main program identifies the R wave in the first heart beat by finding the maximum point of all the maximum values returned.

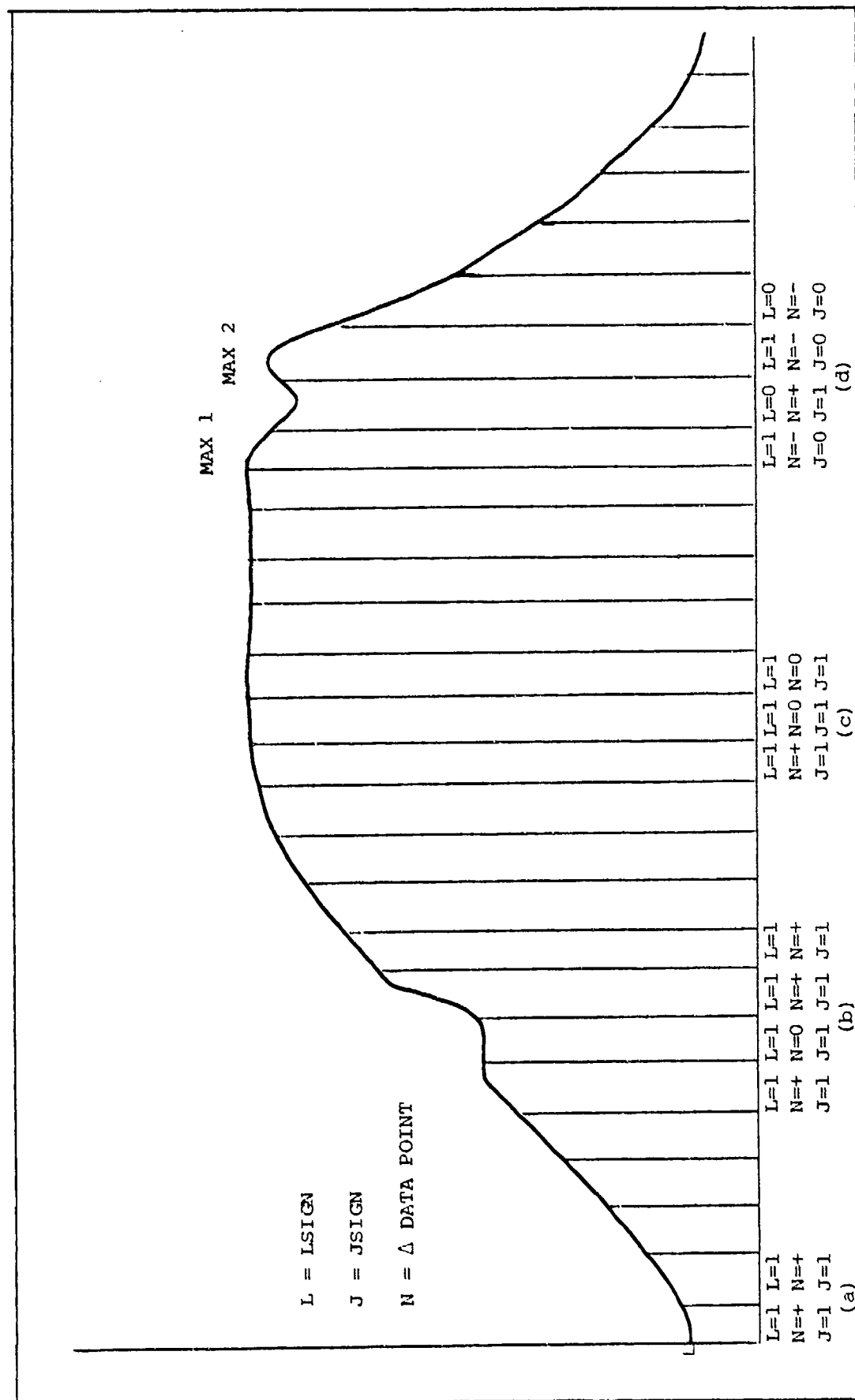


Fig. 10. A wave shape being analyzed by subroutine MAX to find the maximum points. The wave shapes begin with a positive slope (a), changes to flat (b and c), and changes to negative (d).

The maximum point before the R wave peak is identified as the P wave. If the R wave is the first wave form, the last wave form in the heart beat is called the P wave. Once the wave forms in the first heart beat are found, the process is repeated for the second heart beat.

Once both P waves are identified, the distance between them is the number of data points found and is called the P-P interval. Since the sample rate is 500 samples per second, this P-P interval could be multiplied by .002 to find the interval in time.

Remaining P-P Intervals

With the initial P-P interval found, the subroutine GREATER is used to find the remaining P waves. Subroutine GREATER defines certain parameters for subroutine MAX to use to find the next P wave. These parameters define where to start looking for the next P wave and the data search window. Initially, this window is set to ± 2 data points from the expected P wave location. If a peak is not found, the data window is expanded by two (2) data points in both directions, until the window reaches a maximum value of 30 data points. The search for peak values is accomplished by subroutine MAX. If more than one maximum value is returned, the largest value is used. If no peak is found, the P wave is assumed to be where it was expected. This assumes the heart rate will not change significantly until a P wave is found.

The location of the P wave, or where it should have been (if a peak was not found), is transferred to the main program.

The value transferred from subroutine GREATER is used to find the new P-P interval. If the P wave was not at the expected location, a different P-P interval is found and placed in the P-P interval matrix. If the P wave was at the expected location or peak was not found, the P-P interval does not change from the previous one; and that value is placed into the P-P interval matrix. Figure 11 is an example of the interval remaining the same and then changing.

Output

The output of the program is accomplished by a call to subroutine MARK, which in turns calls subroutine AXIS and PMARK. Subroutines were used to facilitate changing methods of output. This particular program uses a Textronix 4014 terminal for CRT graphing with a Textronix 4631 copier attached to the terminal for paper copies of the CRT display.

Subroutine MARK initializes or erases the screen; calls AXIS; plots a data block; sets parameters for PMARK; and prints the value of the P-P intervals in that data block. AXIS is designed to draw a frame around the output presented. Then the X and Y axes are drawn and divided by tic marks. This subroutine also labels the major tic marks and both axes. After all of this is drawn, control is returned to subroutine MARK.

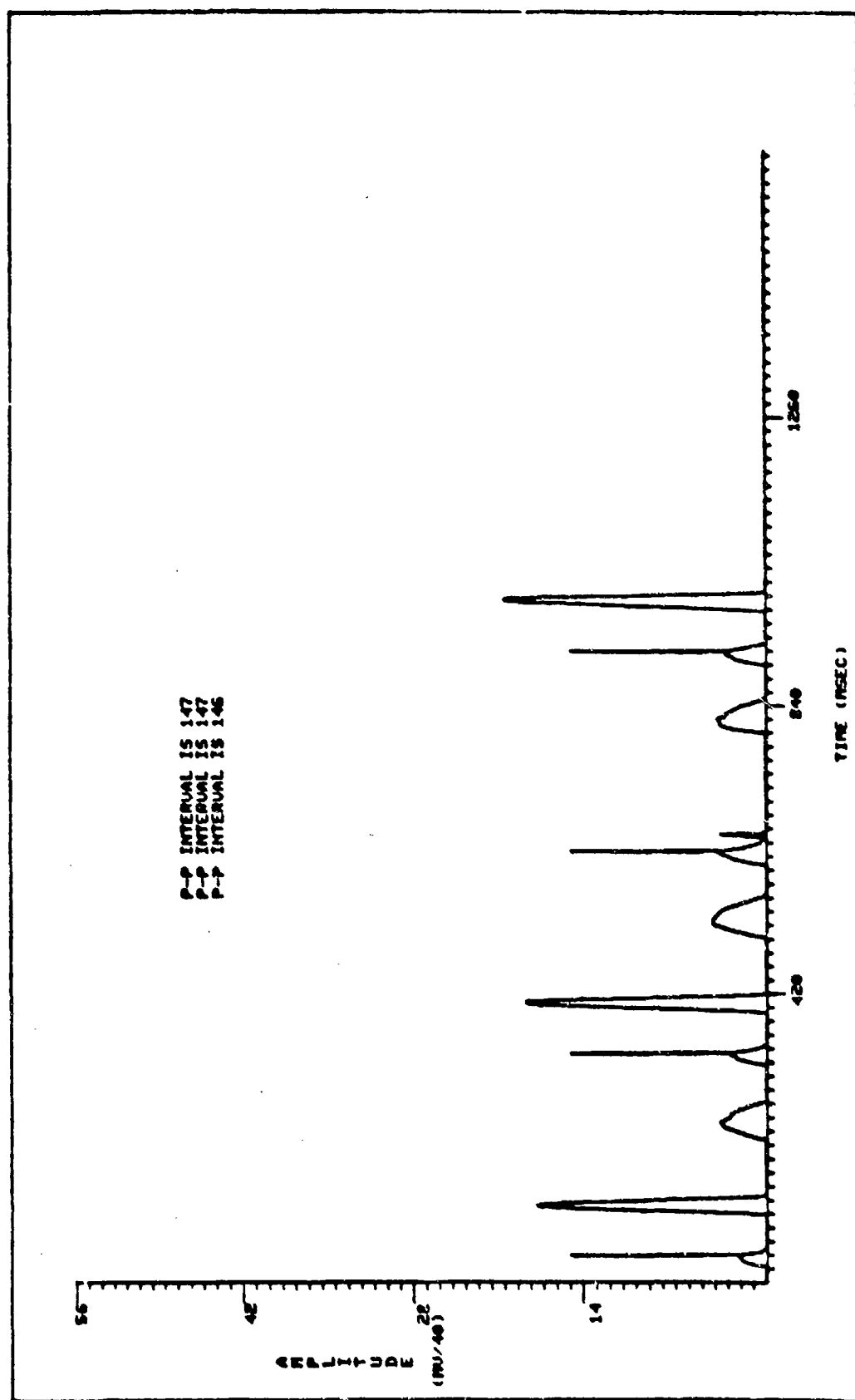


Fig. 11. An analyzed ECG having a constant P-P interval then a changed interval.

MARK now plots the data and checks to see if a plotted data point corresponds to a peak of an identified P wave. If the data point does correspond, parameters for PMARK are set, and PMARK is called.

PMARK draws a straight vertical line from the identified P wave peak. This line is used to easily identify the wave forms selected by the program. Once the P waves have been marked, control is again returned to subroutine MARK, which now prints out the P-P intervals in the block of data being shown on the screen. After this, the program is halted so that the graph can be studied or copied. When the program is started again, the screen is prepared for the next plot.

Continuing the Main Program

After the output has been generated, control is returned to the main program. Now the operator is given an opportunity to exit the program, or have another block of data read into memory. If the operator elects to continue the program, another block of data is read. A system to transfer the area of the next expected P wave is needed to be devised. Since there are almost 1.5 million data points to be processed, the data has to be analyzed in block segments. This requires a system to carry forward into the next data block, a part of the P-P interval not used in the previous segment. For example (refer to Figure 12 for assistance), if the P-P interval is 291 data points, but only 100 data points are left in this data block, then the first 191 data points must be ignored in the next data block

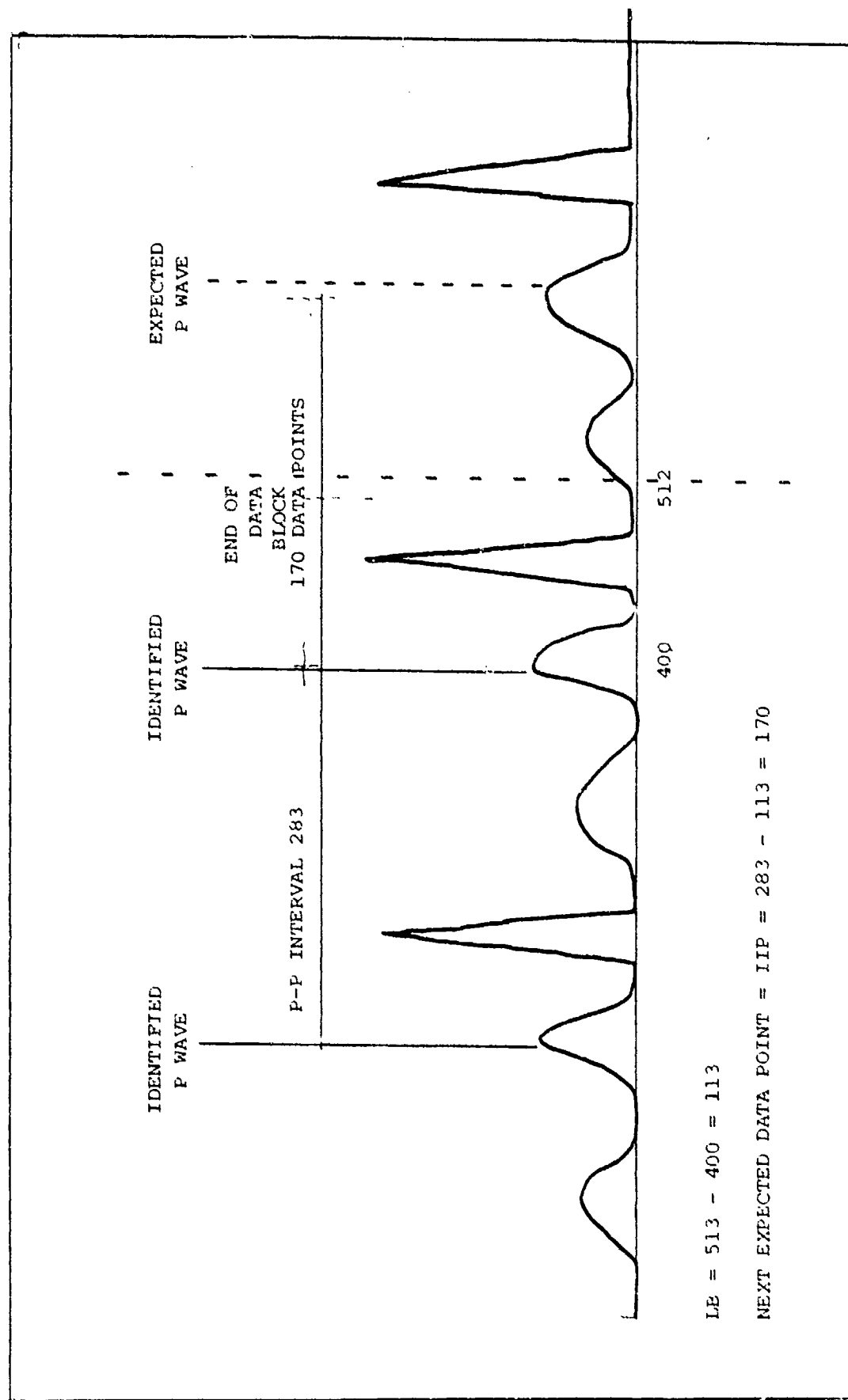


Fig. 12. How the P-P interval is transferred to another data block.

to get the data window around the calculated P wave. This skipping routine consists of finding the number of data points left in the first data block ($LB = 512 - \text{maximum point}$), and then subtracting this from the known P-P interval. Once the skipping routine is finished, the next problem is how to calculate the next P-P interval. This interval is calculated by adding LB to the number of data points corresponding to the identified P wave. Since this last addition routine can only be performed when going from one data block to another, a decision step must be included as to whether to use this step or not. The last addition routine is used only when the difference between the P wave data point is less than the last calculated P-P interval. Therefore, the decision step is based on whether the location value is less than the expected P-P interval.

IV. Results

Data Collection

Data were collected from three dogs. The blockage technique was used on two dogs: Dog 1 and Dog 2. The toxicity technique was used on Dog 3. The electrocardiograms (ECG) from these dogs were recorded on separate reels of analog tape for future use.

The ECG's from Dog 2 and Dog 3 were analyzed by a cardiologist and disassociated P waves identified. The ECG from Dog 2 had a negative P wave because the AVL lead was used as the exterior lead. This provided a data base to test the computer program against negative P waves. Figure 13(a) is a sample of this ECG.

The ECG from Dog 3 had a positive P wave (see Figure 13(b)), but there were abnormal wave shapes at 1.6 Hz throughout the ECG. These wave shapes were apparently caused by the Frequency Modulated (FM) analog recorder being over modulated by a strong signal going into it. This accidental loss of signal proved to be valuable in order to test the computer program against abnormal wave forms in an ECG. A patient in a Cardiac Care Unit frequently produces artificial wave forms different from a normal heart. A computer program to find a P wave must take this into account. Figure 13(c) is an example of Dog 3's ECG with these wave forms.

The wave forms on an ECG of a heart dying from blockage is significantly different from that of a heart dying from

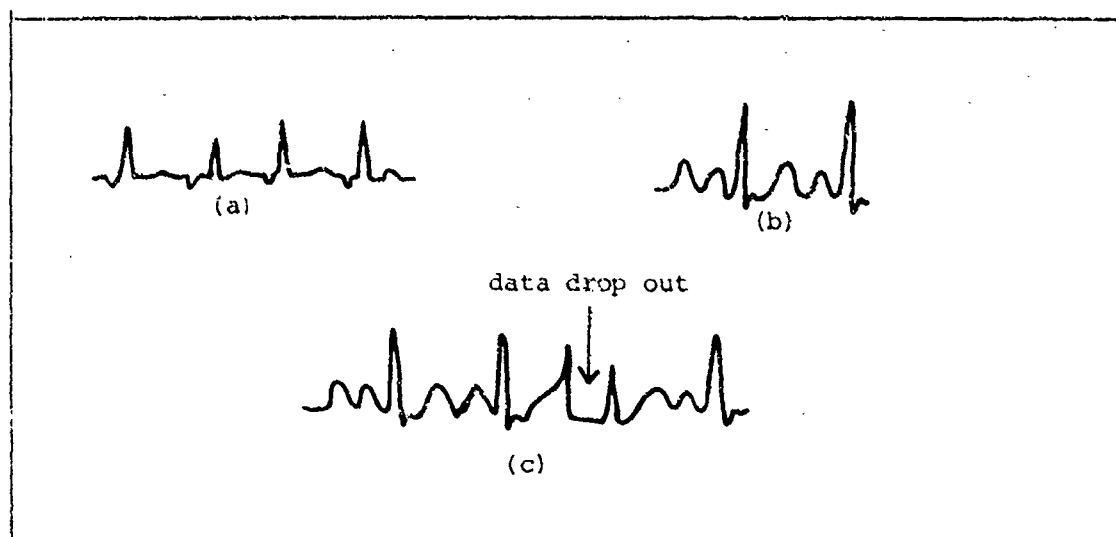


Fig. 13. (a) Dog 2's ECG has a negative P wave.
 (b) Dog 3's ECG has a positive P wave.
 (c) Dog 3's ECG has a data drop out
 occurring at a 1.6 Hz rate.

toxicity. The ECG from a blocked heart has wave forms with a higher amplitude peaks and longer duration. The ECG from the toxic heart has lower peaks and is more rhythmic. In general, a heart dying from a blockage produces a more "violent" ECG. This ECG was harder for the cardiologist to read.

To read this ECG the cardiologist used the internal lead to help find the P wave. Generally, this internal lead provided a good wave shape to help find the P wave. On occasion, near the end of the ECG and the death of the dog, the P wave, even in this lead, could not be found.

The violence around the P wave disassociation area on an ECG is shown in Figure 14.

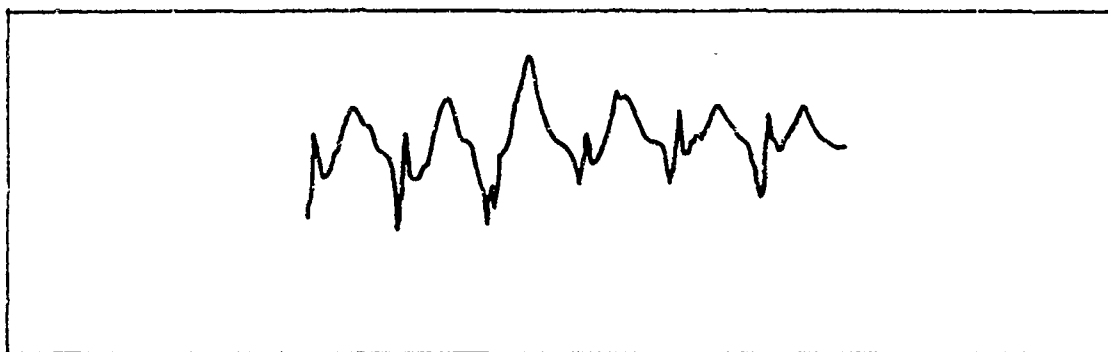


Fig. 14. The violence around the P wave disassociation experienced on Dog 3's ECG.

Computer Program

The computer program uses the last P-P interval to find the area of the next P wave. This demands that the data be handled sequentially without any breaks, once the P-P interval is initially calculated. The computer program processed the data with an output every 50th data block after the first block of data. The program showed a promising ability to pick out P waves in many instances (see Figures 15 and 16) and correct itself at other times (see Figure 17).

The computer program was stopped when the first wrong wave form was detected. To find why the program failed, the program was restarted with an output of every 10th data block and finally every data block until the cause was isolated. This procedure was accomplished twice during the first 50 data blocks, and again during the second 50 data blocks.

The first problem was in data block 43 (see Figure 18). The program picked out the first P wave in the block, and

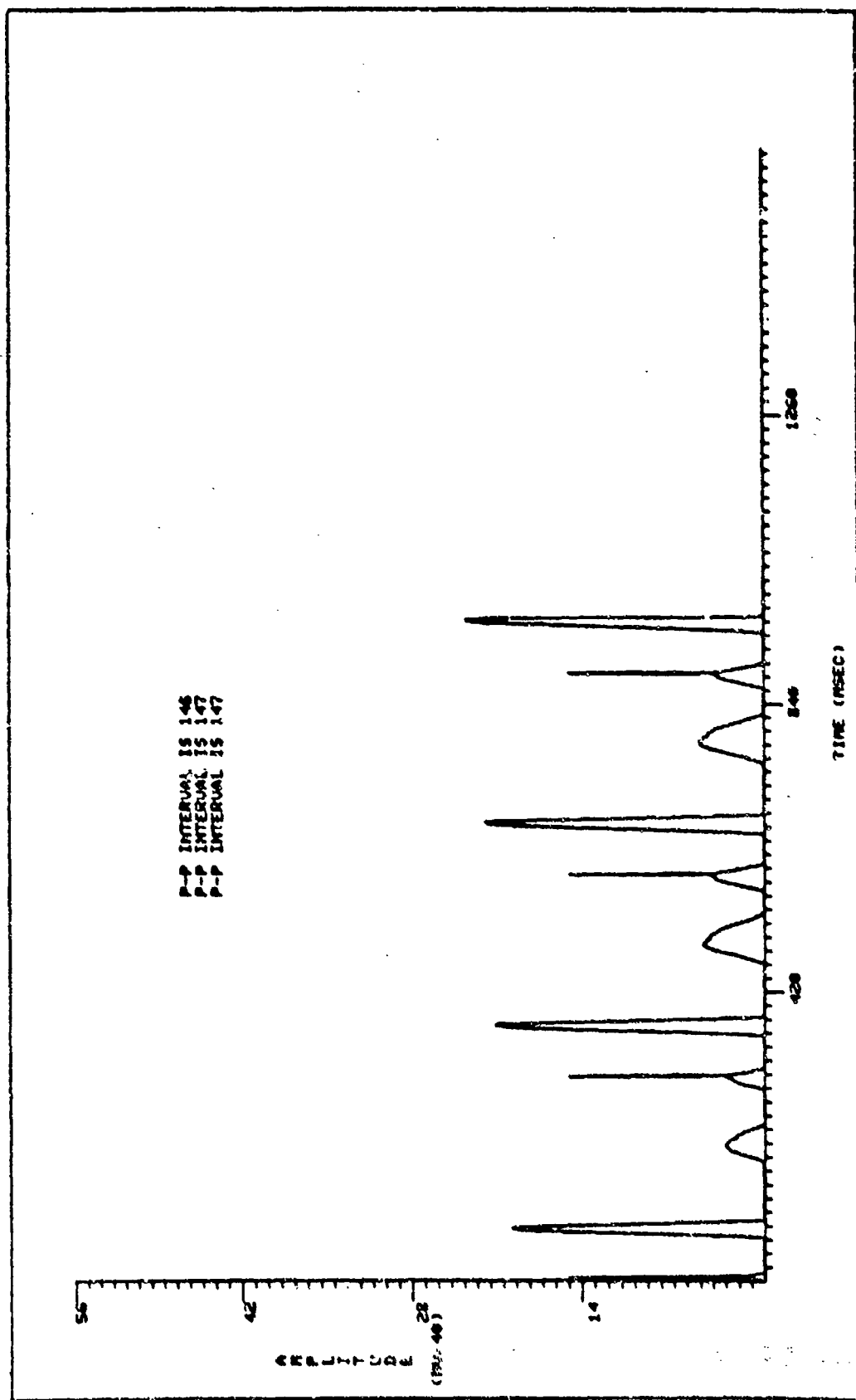


Fig. 15. The computer program picked out the P wave in the first few points in this data block.

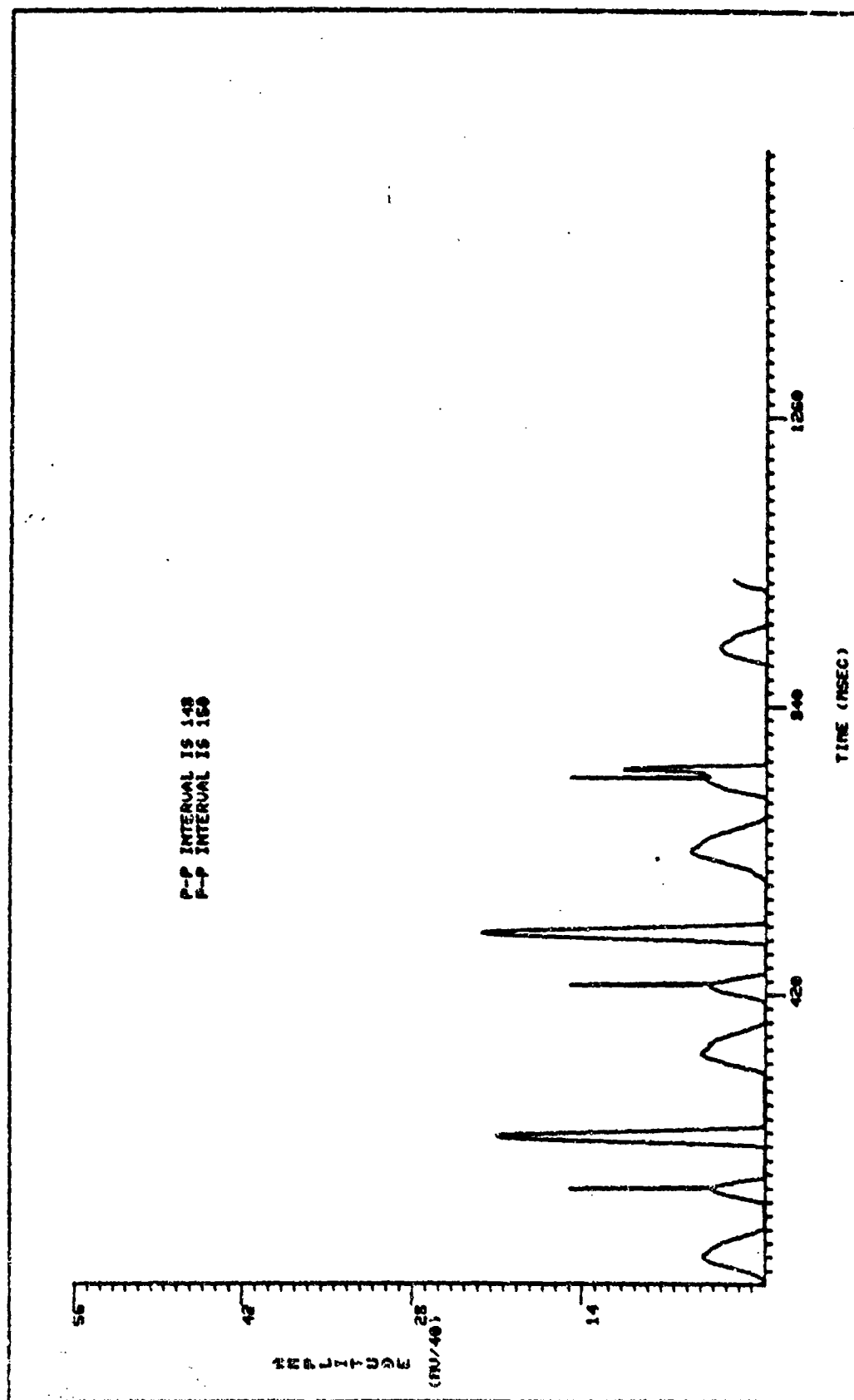


Fig. 16. The computer program picked out the peak of the P wave just before the spike associated with a data drop out.

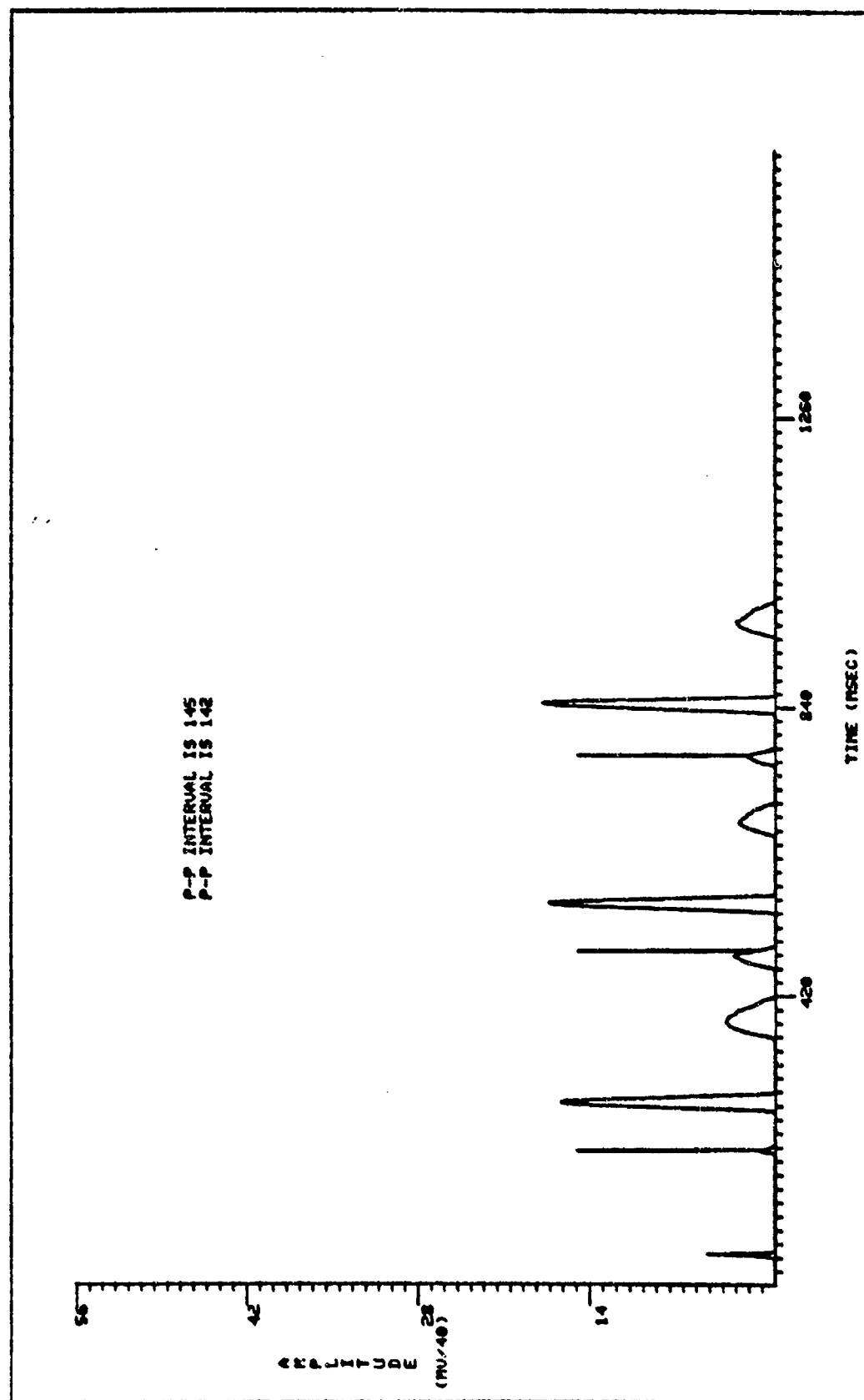


Fig. 17. The computer program identified the remains of a P wave after a data drop out, but this caused a slightly long P-P interval. By the third P wave in the data block the peak of the P wave had been reidentified.

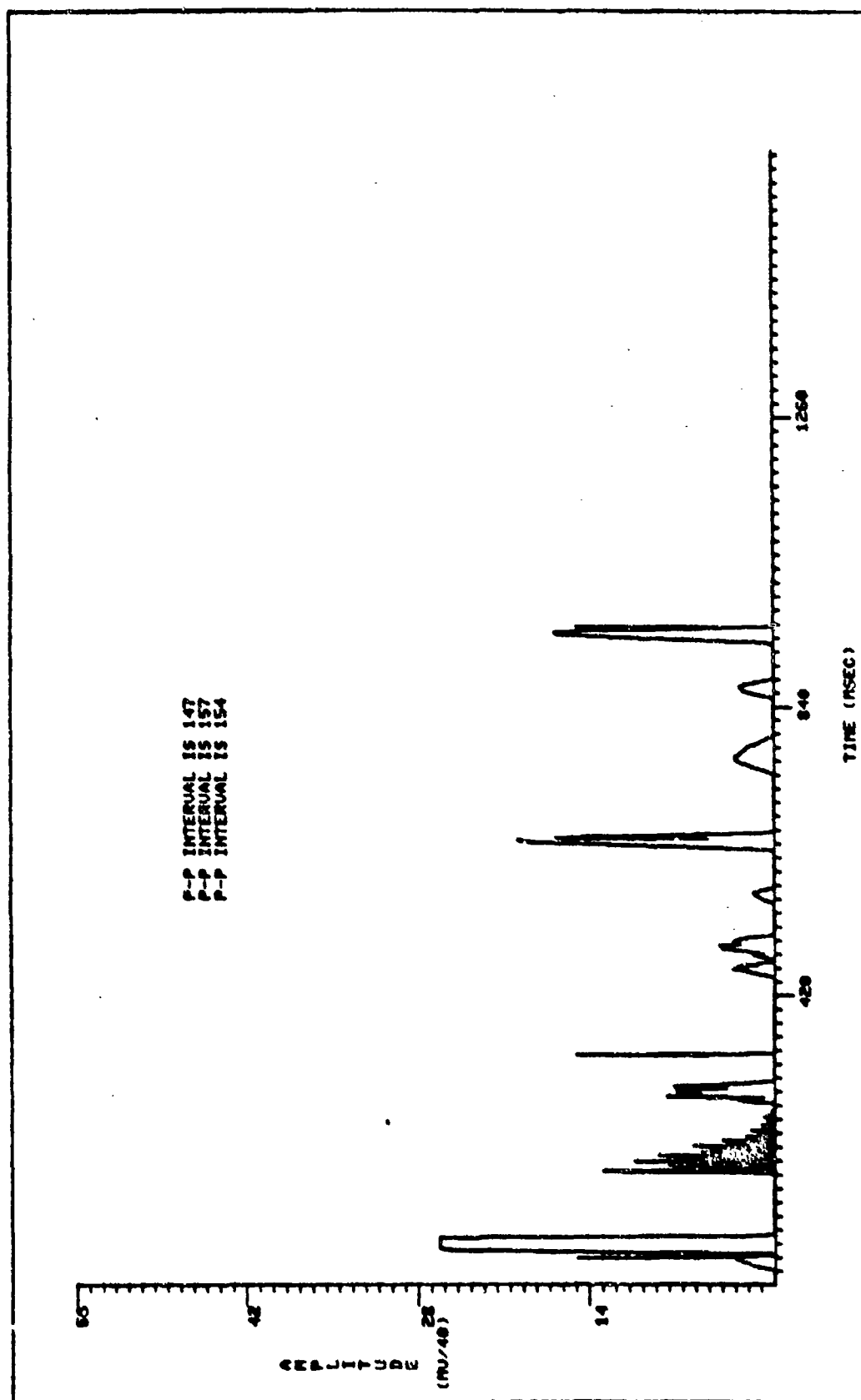


Fig. 18. Data block 43 has strang data.

then the data produced wave shapes that the program could not handle. After that data, the program became an R wave detector. The reason for the strange data wave shapes is unknown but may reflect a singular error in the data recording scheme.

The second problem was in data blocks 74 and 75. Through data block 74 (the graph up to the broken line in Figure 19), the program picked out the correct P waves. In data block 75 the first P wave was identified but only part of the second P wave was present because of the 1.6 Hz data loss problem described previously. The program identified this small amount of data as the P wave. When the P-P interval was transferred to the second wave shape, the peak of the P wave was outside of the expanded data search window. Therefore, the program assumed the P wave was not there and the last P-P interval was retained. With the next data block, the program became a T wave detector.

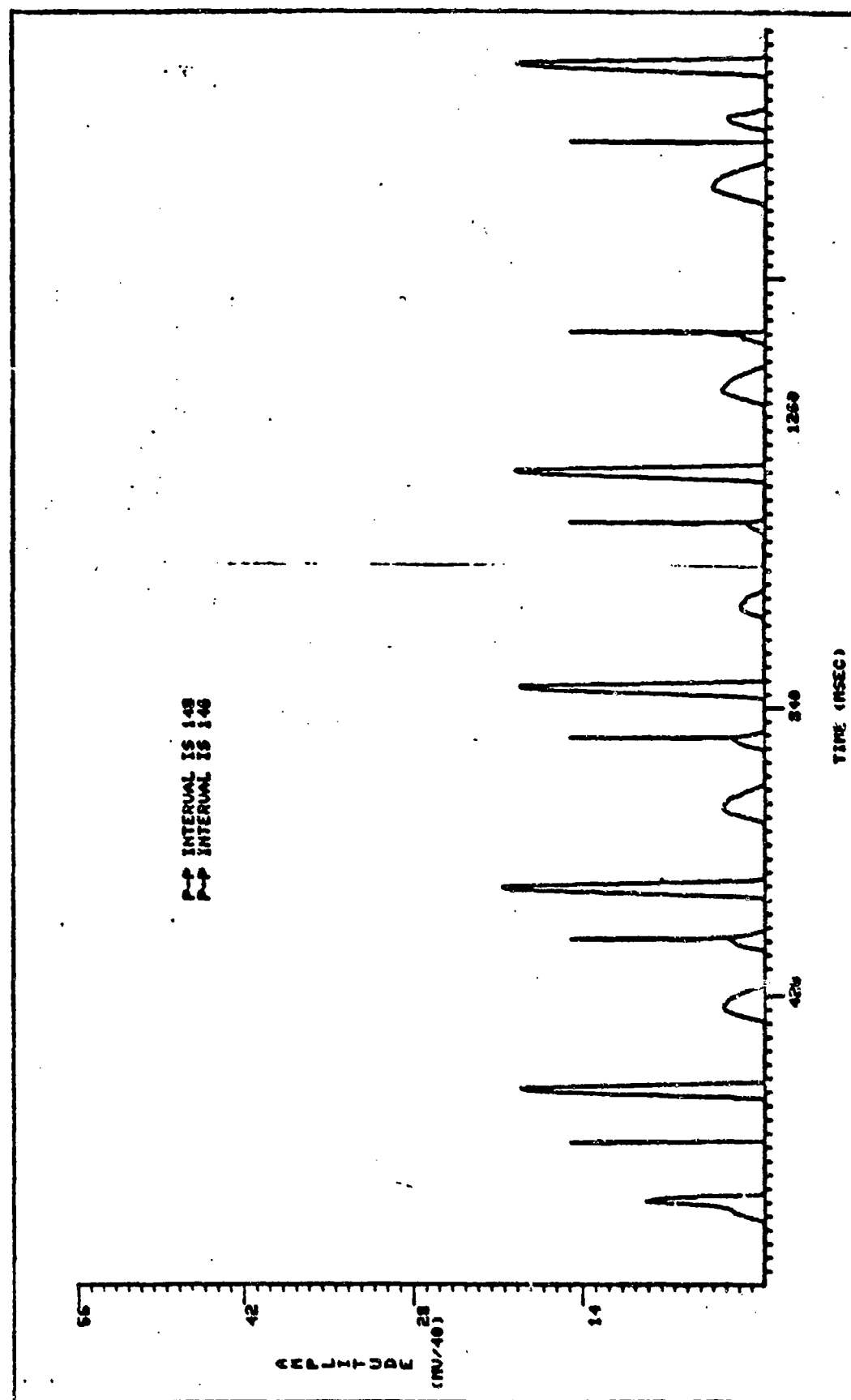


Fig. 19. This is the combination of data blocks 74 and 75. Data block 74 is presented up to the broken line.

V. Conclusion and Recommendation

Conclusion

Two main conclusions concern the precision of the data and the success of the computer program. First, the sample rate, 1000 samples per second, was too high. This produced redundant data. The data rate of 500 samples per second was more manageable and met the needs of the program. The analyzed data also did not need to be sampled to a 0.01 millivolt accuracy. The program is designed to find the P wave, not its absolute peak value. A relative relationship between wave form values was sufficient. The 0.01 millivolt accuracy caused problems by inducing unwanted noise into the data which had to be taken out. A flat peak was caused by the ignoring of fractional millivolt variations and the sampling routine. However, since the graphing pen, or CRT beam, has a finite width, an absolute peak is not desired, this flat peak is acceptable.

The computer program demonstrated that an adjustable data search window is needed. When the search window was set too large (30 data points), whole heart beats were skipped, particularly in the data close to disassociation. This was determined by human analysis of the data using the computer program logic, given that the first two P waves were identified correctly. The problem in this area of the ECG seems to be the increased heart rate. This compresses the wave shapes, putting the T and R waves within the large search window. With these wave shapes inside the window,

the program picked the highest peak (the R wave) as the P wave. With the data window small (5 data points), the P wave was easily missed during the early part of the data. This was because the wave forms were not compressed and large. The 0.07 second change in the P-P interval cited previously represents 35 data points, which is a significant change when wave forms become close to each other. The solution was an adjustable data search window that started at 5 data points and expanded to a maximum of 30 data points if a peak was not previously identified.

The computer program failures could not be corrected because only one piece of information, periodicity of the P wave, was used to identify the P wave. Many other factors such as present wave shape, relationship to the R wave, and historical shape of the P wave are used by a cardiologist to identify the P wave. These factors will be especially important in identifying disassociated P waves. While this program failed to identify P waves at human accuracy rates, it developed another useful approach that can be added to existing techniques to find the P wave.

Recommendation

The first thing to be studied is the outline of a thorough set of relationships between the P wave and the other wave shapes on an ECG. At the present these relationships are not well defined. The data produced by this thesis can be useful in defining these relationships. However, care must be

exercised in applying findings based on this data because there are small, but important, differences between the dog and the human ECG.

After these relationships are defined, they should be thoroughly tested against human data. This will probably require certain constants and some rules to be adjusted. Once these rules are thoroughly tested, a computer program can be written to implement the rules.

If this program produces a sufficiently high probability of being correct, a digital hardware package could be designed to implement the program. This hardware should be in a small package using the latest in solid state technology to analyze ECG's on a real time basis. The small size would facilitate each patient in a CCU having his own monitor, without adding appreciably to the equipment around his bed.

Bibliography

1. Flick, Charles A. Computer Identification of P Waves in Electrocardiograms. Unpublished thesis. Wright-Patterson Air Force Base, Ohio: Air Force Institute of Technology, December 1977.
2. Fozzard, Harry and Plato, Kinias. "Computers for Recognition and Management of Arrhythmias," Medical Clinics of North America, 60: 291-298 (March 1976).
3. Godfrey, J. C., et al. "Cardiac Arrhythmia Monitoring System," Proceedings Fall 1974 Computer Society Conference. 173-175. New York, New York: The Institute of Electrical and Electronic Engineers, Inc., 1974.
4. Guyton, Arthur C. Textbook of Medical Physiology (Fifth Edition). Philadelphia: W. B. Saunders Company, 1971.
5. Kardenat, R. Karl, Paul Kezdi, and Edwin L. Stanley. "A New Catheter Technique for Producing Experimental Coronary Thrombosis and Selective Coronary Visualization," American Heart Journal, 83: 360-364 (March 1972).
6. Oliver, G. C., et al. "Detection of Premature Ventricular Contractions with a Clinical System for Monitoring Electrocardiographic Rhythms," Computers in Biomedical Research, 4: 523-541 (August 1971).
7. Pairett, Charles E. Computer Analysis of Electrocardiograms. Unpublished thesis. Wright-Patterson Air Force Base, Ohio: Air Force Institute of Technology, May 1974.
8. Report of Committee on Electrocardiography, American Heart Association, "Recommendations for Standardization of Leads and of Specifications for Instruments in Electrocardiography and Vectorcardiography," Circulation, 35: 583-602 (July 1969).
9. Swenne, C. A., et al. "A Computerized, Interactive Coronary Care Unit Monitoring System," IEEE Transactions on Biomedical Engineering, BME-24: 63-67 (January 1977).

APPENDIX A
TABLE OF REAL TIME ECG MONITORS

Table 1. Real Time ECG Monitors (Ref 2:294-295)

System	Sample Rate Per Second	Algorithm	Variables	Evaluation
EXPERIMENTAL				
Washington University	500	Feature extraction	Signal amplitude	PVC detection 78% False positive 0.4% beats
University of Chicago	500	Feature extraction	Signal amplitude Aztec aperture	PVC detection 90% False positive 0.07% beats QRS duration
Worcester Polytechnical Institute	200	Stored normal, cross correlation	Signal amplitude QRS thresholds	PVC detection 80% False positive 0.04% beats
Edinburgh (MedPhysics)	-	Stored normal, area difference	Signal amplitude	Arrhythmia events 99% False positive alarms about 15% of true positives
Edinburgh (Surgery)	-	Feature extraction, probability	None	Arrhythmia events 70% False positive alarms about 17% of true positive
Rotterdam	250	RR interval	Signal amplitude	PVC detection 85%
Stockholm	100	Stored normal, cross correlation	Signal amplitude	PVC detection 84% False positive 0.45% beats
Utah	200	Feature extraction	Signal amplitude	Unreported
Stanford	125	Feature extraction	Signal amplitude	Unreported
Dusseldorf	-	Threshold analog	None	Unreported

Continued

Table 1--continued

System	Sample Rate Per Second	Algorithm	Variables	Evaluation
COMMERCIAL				
Mennen Greatbatch	500	Feature extraction	Signal amplitude	PVC detection 80% False positive 0.4% beats
Could	250	Stored normal, cross correlation	Signal amplitude Various thresholds	Abnormal detection 99% False positive 0.1% beats
Hewlett-Packard	125	Feature extraction	Signal amplitude	PVC detection 88% False positive 0.12% beats
Electronics for Medicine	200	Stored normal, cross correlation	Signal amplitude Prematurity and shape PVC	Unreported
American Optical Corporation	240	Feature extraction, standard normals	Signal amplitude PVC criteria	PVC detection 80%

APPENDIX B

BLOCK DIAGRAM OF EQUIPMENT USED TO RECORD ECC

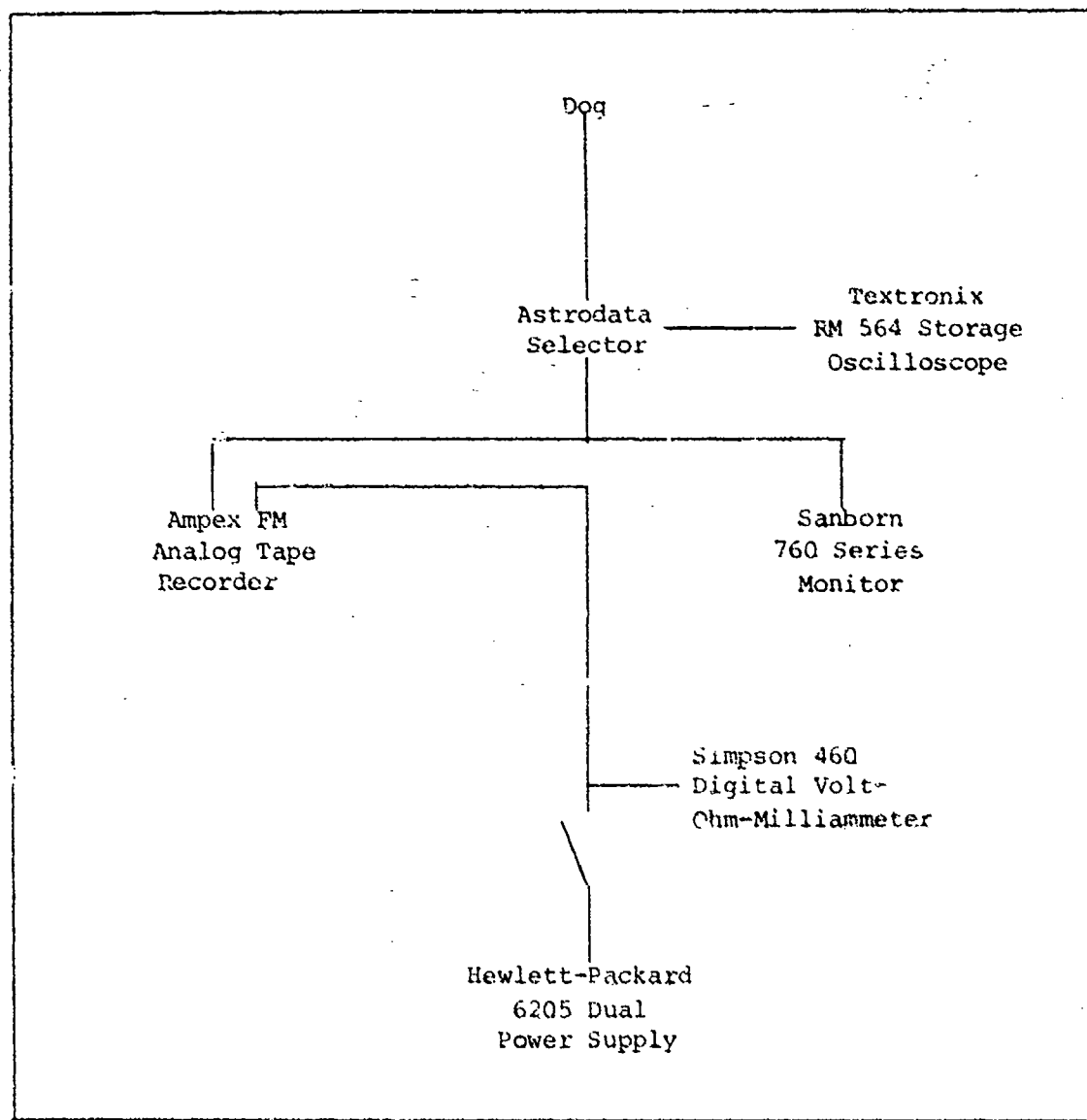


Fig. 20. Block diagram of the equipment used to record ECGs.

APPENDIX C
DATA ON DOGS

Table 2. Data on Dogs

	Dog 1	Dog 2
Weight	34.1 lbs.	39.2 lbs.
Breed	German Shepard	German Shepard
Animal Procedures:		
Dipped for Ectoparasites	Yes	Yes
DHL Vaccine	Yes	Yes
Rabies Vaccine	Yes	Yes
Voice Biopsy	Yes	Yes
Treated for Ear Mites	Yes	Yes
Microfilaria Exam	Negative	Negative

APPENDIX D
PROGRAM TO CONVERT DATA INTO
EIGHT PLACE DATA FIELD

```

PROGRAM COMP (INPUT,OUTPUT,TAPE7,TAPE2)
DIMENSION DATA (512)
DO 3 I=1,2786
  READ(7) (DATA(J),J=1,512)
  IF(EOF(7).NE.0) GO TO 900
  PRINT(2,13) (DATA(K),K=1,512)
13FORMAT(10F8.2)
3CONTINUE
900STOP
END

```

Fig. 21. Program used to change data format.

Figure 21 shows the computer program used to change the data into a format usable in the P wave identification program after the original data was put into CDC 6600 word length. The program reads the data in an unformatted form, changes the format to floating point with a field length of eight characters including two decimal places. This new format is then written onto a drum storage unit for storage. This program can be expanded to punch the data onto cards, print the data onto paper, or use magnetic tape for storage.

```

Job card.
FTN.
REQUEST, TAPE2,*PF.
LABEL,TAPE7,R,L=1fn1,VSU=vsu,NORING.
LGO.
CATALOG, TAPE2, pfn1,RP=999.
EXIT,S,
REQUEST, TAPE2,*Q.
ROUTE,TAPE2,DC=PR,TID=id,FID=1fn2,ST=sys.

```

Fig. 22. List of control cards used to execute the program listed in Figure 21.

Figure 22 shows the control cards necessary to compile the program, mount the magnetic tape containing the data converted to CDC 6600 format, request permanent file space for the changed data, execute the program, and catalog the changed data. A series of control cards (starting with EXIT,S.) is also listed that routes the output to the user's intercom terminal if the program should be terminated before it is finished. The variables used in these control cards are as follow:

- lfnl - The file name written of the magnetic tape.
- vsu - The number assigned to the tape to be read.
- pfnl - The permanent file name assigned to the new data.
- id - The intercom identification letters assigned to the operator.
- lfn2 - The local file name assigned to the routed file.
- sys - The system designator receiving the new file.

APPENDIX E
COMPUTER PROGRAM TO FIND P WAVE

In this appendix the program to find P waves will be presented and discussed in detail. The first part will be a detailed discussion of the program in short sections. The final part will list the entire program.

Program Discussion

The computer program will be presented in sections as listed in the controlling program. Since most calculations are accomplished in the subroutines, these subroutines will be discussed when they are first called. Subsequent calls to these subroutines will reference the figure number.

Setting Parameters.--The first few lines of the program (see Figure 23) identify the program, establish storage

```

PROGRAM FIND(TAPE2,INPUT,OUTPUT,PLOT)
C*****
C      THIS IS MAIN PROGRAM TO FIND P WAVES IN AN ECG
C*****
COMMON IDATA(512),N1,LK,IP(3000),N/GREAT/IIP,ND/MARK/IBAUD,J
1,PP(2793)
COMMON /MAX/ISTEP,IMAX(6),IPP(6),KK,NB,NC,M
INTEGER PP
REWIND 2
C*****      READ IN THE IBAUD VALUE FOR INITT
C      TAKE THE LISTED BAUD RATE AND DIVIDE BY 10
WRITE*,32HTHE BAUD RATE DIVIDED BY 10 IS
READ*,IBAUD
C*****      SET UP READ AND ANALYSIS LOOP
WRITE*,24HTHE NUMBER OF PEAKS IS
READ*,N
WRITE*,53HTYPE 0 - IF P WAVE IS POSITIVE; TYPE 1 - IF P WAVE
1IS NEGATIVE
READ*,N1
WRITE*,36HWHICH BLOCK OF DATA FIRST GRAPHED?
READ*,M1
WRITE*,44HHOW MANY BLOCKS OF DATA BETWEEN EACH GRAPH?
READ*,M2

```

Fig. 23. This part of the computer program sets parameters.

areas, form integer variables, rewind the data storage device, and ask for initial parameters. In this program, TAPE 2 specified in the program identification statement, is used as data input. The output of the program is the use of a plotting routine as specified in this statement. The specification of common storage was used because of the multiple use of subroutines, and the large use of common variables is both the calling and called routines. The P-P interval matrix (PP) was specified as an integer because all data values are in integer form. The use of REWIND 2 command facilitates using the program many times during a single session.

The interactive part of the program sets the initial parameters. The variable IBAUD is used to set the correct baud rate of the CRT displays if used to graph the output. Variable N is the number of peaks in one heart beat on the ECG and is used to decide where the P wave is in relation to the R wave. Variable N1 will define whether the P wave is positive or negative. This is entered from the ECG data. N1 is used to decide which routine in subroutine READ is used to put the data into an integer form. The variable MA specifies the first data block to be graphed. Variable MB specifies the number of data blocks between graphing outputs. These last two variables are used to skip to the data of interest and to control how often the data is graphed. This is particularly important when computer time is tightly

controlled, as well as allowing specific pieces of data output to be found without all the data being outputed.

Read Data.--Figure 24 shows the computer code that starts the data interpreting loop and reads the data into memory. Loop counter K1 specifies which data block is being read, analyzed and possibly outputed.

```

DO 5 K1=1,2786
C***** READ DATA
CALL READ

```

Fig. 24. The code starting the interpreting loop.

The line CALL READ shifts control to subroutine READ (see Figure 25).

```

C*****
C      SUBROUTINE READ - READ IN DATA, GET DATA INTO
C      POSITIVE P WAVE FORM, AND CONVERT TO INTIGER FORMAT
C*****
      SUBROUTINE READ
      COMMON IDATA(512),N1,LK,IP(3000),N
      DIMENSION DATA(512)
      READ(2,10) (DATA(I),I=1,512)
10     FORMAT (10F8.2)
      IF(EOF(2).NE.0.)GO TO 9
      IF(N1.E2.1)GO TO 13
C*****      FOR POSITIVE P WAVE
      DO 1 I=1,512
      IF(DATA(I).LT.0.)DATA(I)=0.
1     IDATA(I)=DATA(I)
      GO TO 9
C*****      FOR NEGATIVE P WAVE
13     DO 11 I=1,512
      IDATA(I)=DATA(I)
      IDATA(I)=IABS(IDATA(I))
      IF(IDATA(I).LE.10)IDATA(I)=0
11     CONTINUE
      N=2
9      RETURN
      END

```

Fig. 25. Subroutine READ reads a data block of 512 points into memory.

Subroutine READ used variable N1 and generates the integer data matrix IDATA and could modify the value of N. The floating point data matrix DATA is generated and used in this subroutine only. The data is stored in an eight character field which contains two decimal places; therefore, when it is initially read into memory, a floating point column matrix (1 x 512) must be used. A test for an end of file mark is made. If an end of file is encountered, the subroutine is exited. Next, the variable N1 is used to decide which set of routines will put the data into an integer form. If the P wave is positive ($N1 = 0$), the values in the DATA matrix are compared, one at a time, to zero. If the data are greater than, or equal to zero, the value is unchanged. If the data are negative, the value is set equal to zero. This can be done because all the information needed to find the P wave has a positive value. After this is done, the integer data matrix, IDATA, is formed by ignoring the decimal point in the DATA matrix. Since the relative amplitude of the data is the only amplitude information needed, this can be done without losing information. If all of the 512 datum points have been handled, control is returned to the main program.

If the P wave is negative, the data are handled differently. First, the IDATA matrix is formed by ignoring the decimal point in the DATA matrix. Then the negative datum points are made positive by finding the absolute value of each datum point. Each of these values are compared to a

threshold (10 in this case) that is between the peak of the P wave and the peak of the noise or Q wave, whichever is higher. Every value at or below the threshold is set equal to zero. After all the data in IDATA have been compared, the value of N is set at two, since the T wave usually has a smaller amplitude in this data than the P wave, and thus was nearly always below threshold. Then control was returned to the main program.

Initial Maximum Points.--After the data have been read, a decision is made on which data block is being analyzed (see Figure 26). If the data block is the first one, then the

```

C*****      IF NOT FIRST BLOCK OF DATA FIND REMAINING P WAVES
      IF(K1.GE.2)GO TO 102
C*****
C      INITIALLY FIND P-P INTERVAL
C*****
      NB=2
      NC=500
      KK=2*N
C*****      FIND THE MAXIMUM POINTS
      CALL MAX

```

Fig. 26. Parameters to find the maximum point on the wave forms are set.

program does not skip the next three sequences of steps. The first datum point in the search window (NB) is set equal to two (2); then the maximum datum point in the search window (NC) is set equal to 500. Next, the number of peaks to be found (KK) is set. With these parameters established, sub-

routine MAX is called to find the peaks in the data wave forms (see Figure 27).

Subroutine MAX finds the maximum point of a wave form using a change in slope procedure. A detailed explanation of the theory used by this subroutine can be found in Chapter 3. This subroutine uses variables NB, NC, and KK, and data matrix IDATA to find the wave form peaks and returns variable M and variable matrices IMAX and IPP. Initially, LSIGN (the direction of the last slope) is set equal to one (1); and M (the number of maximum points found) is set equal to zero (0). The size of the data search window is set from NB to NC and the loop searching the data for the maximum value is started. The difference in amplitude between the present datum point and the last datum point found (NA). If NA is positive, the present slope variable (JSIGN) is set to one (1), LSIGN is set equal to JSIGN, and the loop is continued. If NA is negative, JSIGN is set equal to zero (0); and a comparison is made between JSIGN and LSIGN. If they are equal, the loop is continued. If the difference between LSIGN and JSIGN (LSIGN-JSIGN) is less than or equal to zero, the loop is continued. If the difference is positive, the slope has turned negative and a maximum point has been found. In this case M is incremented by one (1) and the amplitude of the last datum point is entered into the maximum amplitude matrix (IMAX). The number of the datum point maximum is entered into the datum point maximum matrix (IPP). If this is the first maximum point, LSIGN is

```

C*****
C      SUBROUTINE MAX - FIND MAXIMUM POINT USING A CHANGE
C      IN SLOPE PROCEDURE
C*****
      SUBROUTINE MAX
      COMMON IDATA(512),N1,LK,IP(3000),N
      COMMON /MAX/ISTEP,IMAX(6),IPP(6),KK,NB,NC,M
      LSIGN=1
      M=0
      DO 13 I=NB,NC
C*****      FIND THE DIFFERENCE IN AMPLITUDE BETWEEN LAST AND
C      PRESENT DATA POINTS - SLOPE
      NA=IDATA(I)-IDATA(I-1)
C*****      IF SLOPE IS POSITIVE
      IF(NA.GT.0)GO TO 2
C*****      IF SLOPE IS NEGATIVE
      IF(NA.LT.0)GO TO 4
C*****      IF SLOPE IS ZERO
      GO TO 13
C*****      FOR POSITIVE SLOPE
      2      JSIGN=1
      GO TO 3
C*****      FOR NEGATIVE SLOPE
      4      JSIGN=0
C*****      IF SLOPE DOES NOT CHANGE, OR IF SLOPE GOES FROM
C      NEGATIVE TO POSITIVE: CONTINUE ROUTINE
      IF(JSIGN.EQ.LSIGN)GO TO 3
      IF((LSIGN-JSIGN).LE.0)GO TO 3
C*****      IF SLOPE CHANGE FROM POSITIVE TO NEGATIVE, SET
C      MAXIMUM POINT
      M=M+1
      IMAX(M)=IDATA(I-1)
      IPP(M)=I-1
C*****      IF FIRST MAXIMUM FOUND,CONTINUE PROGRAM
      IF(M.EQ.1)GO TO 3
C*****      IF ANOTHER MAXIMUM FOUND WITHIN A REGION, FIND
C      MAXIMUM OF TWO VALUES
      IF((IPP(M)-IPP(M-1)).GT.3)GO TO 7
      IF(IMAX(M).GE.IMAX(M-1))5,6
      5      IMAX(M-1)=IMAX(M)
      IPP(M-1)=IPP(M)
      6      M=M-1
C*****      IF NUMBER OF MAXIMUM POINTS EQUAL TH NUMBER NEEDED,
C      RETURN
      7      IF(M.EQ.KK)GO TO 15
C*****      SET PREVIOUS SLOPE INDICATOR
      3      LSIGN=JSIGN
      13     CONTINUE
      15     RETURN
      END

```

Fig. 27. Subroutine MAX finds the peaks of the wave forms.

set equal to JSIGN and the loop is continued. If M is greater than one, the difference between the present and last maximum datum point number is found. If this difference is greater than three (3) (an empirically determined value), a comparison between M and KK is made. If they are equal, the subroutine is exited. If they are not equal, LSIGN is set equal to JSIGN and the loop is continued. If the difference between maximum data point numbers is less than or equal to three, the maximum amplitude of the two maximum points is found and set into the IMAX matrix at position M-1, and the number of that maximum point is put into the IPP matrix at position M-1. Next, M is reset to M-1 and a comparison between M and KK is made as described above. If NA is equal to zero (0), the values of LSIGN and JSIGN are unchanged and the loop continues. When all of the data points between NB and NC inclusive have been examined or M equals KK, control is returned to the calling routine.

Finding Initial P-P Interval.--This part of the computer program identifies the P waves in the first two heart beats and finds the initial P-P interval (see Figure 28). The P wave counter (KB) is set to zero (0) first. Next, the DO LOOP that identifies the different wave forms is initiated. This loop will be repeated twice, once for each of two heart beats. The DO LOOP counter is incremented by the number of wave peaks (N). Therefore, for three peaks K will equal one (1) and four (4); for two peaks K will equal one (1) and three (3), but the next increment is above four so it will

```

      KB=0
      DO 7 K=1,4,N
C*****      FIND R AND THEN P WAVES
      KB=KB+1
      IF(N.EQ.2)GO TO 108
      IF(IMAX(K).LE.IMAX(K+1))109,110
109  IF(IMAX(K+1).LE.IMAX(K+2))111,112
111  IP(KB)=IPP(K+1)
      GO TO 7
110  IF(IMAX(K).LE.IMAX(K+2))114,115
114  GO TO 111
115  IP(KB)=IPP(K+2)
      GO TO 7
112  IP(KB)=IPP(K)
      GO TO 7
108  IF(IMAX(K).LE.IMAX(K+1))116,117
116  GO TO 112
117  IP(KB)=IPP(K+1)
7     CONTINUE
C*****      FIND P-P INTERVAL
      PP(1)=IP(2)-IP(1)

```

Fig. 28. The code identifying the initial P waves and initial P-P interval.

not repeat the loop. The first step is to increment KB by one, then decide if N equals two. If N equals two, then the maximum of only two wave shapes is to be found. Therefore, control is shifted further down the program. The next 13 steps are just a series of comparison between maximum points to find the highest peak. This peak is then labeled the R wave, and the wave just before it is labeled the P wave. If the R wave is the first wave form, the last wave form in the heart beat is labeled the P wave. The amplitudes of the P and T waves are never compared because these will vary depending on ECG lead placement. The data point number that corresponds to the P wave is entered into the P wave identity matrix IP. After the P waves in the first two heart beats

have been identified, the wave form identifying loop is exited. The initial P-P interval is found by subtracting IP (2) from IP (1). This is then entered into the P-P interval matrix PP which gives the P-P interval in the number of data points between P waves. This number can be multiplied by 0.002 to find the time in seconds between P waves.

Parameters to Find Remaining P Waves.--The first five equations set the parameters to find the remaining P waves in the first data block (see Figure 29). The next four

```

C*****      SET PARAMETERS TO FIND REMAINING P WAVES IN
C              FIRST BLOCK OF DATA
              J=1
              LK=1
              KNPT=6
              ISTEP=IP(2)
C*****      SET PARAMETERS TO FIND P WAVES IN REMAINING BLOCKS
C              OF DATA
              GO TO 119
102  IIP=PP(LK)-(513-ISTEP)
              ISTEP=1
              KNPT=LK+5
              J=LK+1

```

Fig. 29. Parameters to find the remaining P waves are set.

equations modify almost all these parameters so the P waves in subsequent data blocks can be found. The variable J is now the P wave counter for output purposes. The variable LK serve as a counter on the P-P interval as well as the initial parameter on the next DO LOOP. Variable KNPT serves as the terminal parameter on the next DO LOOP. Both of these will be changing during execution of the program, so they

must be variables. The number of the datum point being examined is controlled by ISTEP. The first series of commands in this section equal the last entry in the IP matrix. With this set of commands completed, control is skipped further down the program.

If this is not the first data block being examined, the second set of equations is used to set the needed parameters. The first equation in this set is used to find the expected position of the first P wave in the new data block. Variable IIP stands for the distance to the expected P wave. The distance in this case is the number of datum points left in the P-P interval, PP(), after subtracting the number of points in the last data block since the last P wave, 513-ISTEP. This number (IIP) is used in a later subroutine as the center of a search area to find the P wave peak. ISTEP is reset to one (1) so the first data point will be considered. KNPT is always greater than LK by five so the next DO LOOP will not be exited before all the P waves are identified. J is incremented to the last identified PP interval plus one (1).

Find Remaining P Waves.--This section of the program is a DO LOOP that finds the P wave by calling the subroutine GREATER and then calculating the P-P interval (see Figure 30). The parameters for the DO LOOP are LK and KNPT which are described above. The call to subroutine GREATER will return the location of the P wave.

```

C*****      FIND P WAVES
119 DO 8 KN=LK,KNPT
C*****      CALL GREATER
      CALL GREATER
      IF(ND.GT.512)GO TO 118
      KL=KN+2
      IP(KL)=ISTEP
      LK=KN+1
      IF(ISTEP.LT.PP(LK-1))GO TO 120
      PP(LK)=IP(KL)-IP(LK)
      GO TO 121
120 PP(LK)=IP(KL)+LB
121 IF(ISTEP.GE.512)GO TO 118
      IIP=PP(LK)
8      CONTINUE

```

Fig. 30. How the main program finds the P-P interval.

Subroutine GREATER (see Figure 31) uses variables IIP and ISTEP from the main program, and the matrices IMAX and IPP from subroutine MAX to return the P wave location, ISTEP, to the main program. Variable ND is the expected data point of the P wave peak. NB is the datum point that is the minimum value of the data search window. Variable NA is the initial value of NB and will be used later to set the P wave location if a peak is not found within the search window. The maximum value of the data search window is NC. NE is the initial value of NC. A test is performed to see if the maximum search window value is outside the data block. If it is, the subroutine is exited. The number of peaks to be found, KK, is set at six to insure that the maximum peak in the data search window is found. Next, a call to subroutine MAX (see Figure 27) is made. The information returned from MAX is the matrices IMAX and IPP and variable M. If one or more

```

C*****
C      SUBROUTINE GREATER - FINDS P WAVES
C*****
      SUBROUTINE GREATER
      COMMON /DATA(512),N1,LK,IP(3000),N/GREAT/IIP,ND
      COMMON /MAX/ISTEP,IMAX(6),IPP(6),KK,NB,NC,M
C*****
      SET PARAMETERS
      ND=ISTEP+IIP
      NB=ISTEP+IIP-2
      NA=NB
      NC=NB+5
      NE=NC
      IF(NC.GT.512)GO TO 201
      KK=6
C*****
      CALL MAX
206  CALL MAX
      IF(M.LE.1)GO TO 203
C*****
      IF MORE THAN ONE MAX POINT, FIND HIGHEST MAX
      DO 204 KA=1,M
      IF(IMAX(1).GE.IMAX(KA))GO TO 204
      IMAX(1)=IMAX(KA)
      IPP(1)=IPP(KA)
204  CONTINUE
C*****
      DECIDE IF P WAVE IS PRESENT
203  IF(M.EQ.1)GO TO 202
      NB=NB-2
      NC=NC+2
      IF((NC-NB).GE.30)GO TO 205
      IF(NC.LE.12)GO TO 207
      NC=NE+1
      IF(NC.EQ.512)GO TO 205
207  GO TO 206
C*****
      IF NOT A MAX POINT, KEEP SAME P-P INTERVAL
205  IB=NA+2
      ISTEP=IB
      GO TO 201
C*****
      IF MAX POINT, SET DATA INDEX
202  ISTEP=IPP(1)
201  RETURN
      END

```

Fig. 31. Subroutine GREATER returns the location of the P wave.

peaks have been found, M will return with a non-zero number. A decision is made if M is less than or greater than one. If M is greater than one, the amplitude of the peaks are

compared in a DO LOOP to find the maximum peak. The number of this datum point is then identified as the location of the P wave. If M is one or zero, a decision is made which of the two values it is. If M is zero, then a peak was not found and the data search window is enlarged by four data points (an empirically determined value). NB is decreased by two data points while NC is increased by two data points. If the data search window (NC-NB) is greater than or equal to 30, the maximum size, then a P wave is assumed not to be present. In this case the position of the expected P wave, IB, is found; ISTEP is set equal to IB; and the subroutine is exited. If the window is not greater than 30, a check to see if NC is within the data block is made. If NC is in the data block, the enlarged data search window is examined for a peak as described above. If NC is outside the data block, its value is reset to NE plus one. Again a check is made to see if the newest NC is inside the data block. If NC is equal to the data block maximum point, 512, the P wave peak is assumed to be at the data block maximum as described above. If NC does not equal 512, the search is initiated again to find the P wave peak as described above. If M equals one, ISTEP is set equal to the value in IPP and control is returned to subroutine GREATER.

When control is returned to GREATER, a decision is made whether ND is outside the data block. If it is, then the DO LOOP is exited. If ND is in the data block, the P wave location counter, KL, is calculated. Next the matrix IP is

set equal to the value of ISTEP found in GREATER. Next LK is incremented. Now a comparison between ISTEP and the last value in the PP matrix is made. If ISTEP is less than this value, a new data block has been examined for a P wave and a different procedure must be used to find the P-P interval. This procedure is to add the location of the present P wave peak to the number of data points between the last P wave and the end of the last data block (LB). If ISTEP is not less than the last value in PP, the new value for PP is found by subtracting the present IP value from the last IP value. Independent of the ISTEP value in relation to the last PP value, a check is made to see if ISTEP is outside the data block. If it is greater than or equal to 512, the DO LOOP is exited. If ISTEP is less than 512, IPP is set equal to the last PP value, and the DO LOOP is continued.

Output.--Output from the main program is two commands (see Figure 32). The first command is a decision whether

```

C*****      DECIDE WHICH DATA TO GRAPH
118 IF(MA.NE.K1)GO TO 6
C*****      CALL MARK
              CALL MARK

```

Fig. 32. Output is basically a call to subroutine MARK.

output is desired by comparing MA and K1. If MA does not equal K1, then control is shifted further down the program. If MA equals K1, then output is initiated by a call to subroutine MARK.

Subroutine MARK graphs the data, sets parameters for marking P waves, and lists the values of the P-P intervals in this data block (see Figure 33). This subroutine is designed for use on the Textronix CRT terminals using the Plot-10 package of instructions. Calls to subroutines INITT, ERASE, MOVABS, DRWABS, ANMODE, and DCURSR are fully explained in the instruction books for this package. This subroutine can be changed to graph the output on any graphing terminal available. The first command erases the CRT screen the first time there is any output. Then IK is set to 47 which causes the screen to be erased on the next command only in subsequent outputs. The screen is prepared for data by calling subroutine AXIS.

Subroutine AXIS draws a picture frame, a set of axes, and the tic marks and labels the axes and major tic marks (see Figure 34). The graphing subroutines in this subroutine are the same as those in MARK. The two data entries are the Hollerith codes for labeling the Y-axis (LYLET) and major tic marks on the Y-axis (NYTIC). The first seven commands draw a 780 by 983 raster point frame on the screen. The next six commands draw the X and Y axes. The following eight commands draw the minor tic marks on the X-axis. This is followed by a DO LOOP that draws the major tic marks. This organization of commands is next repeated to draw the minor tic marks first, and then the major tic marks on the Y-axis (see Figure 35). After this, the Y-axis is labeled by a DO LOOP that prints one data character in LYLET at a

```

C*****
C      SUBROUTINE MARK - GRAPHS DATA, SET PARAMETERS FOR MARKING
C      P WAVES, WRITES P-P INTERVAL VALUE
C*****
      SUBROUTINE MARK
      COMMON IDATA(512), N1, LK, IP(3000), N/MARK/IPAUD, J, PP(2735)
C*****      INITIALIZE SCREEN
      IF(IK.NE.47) CALL INITT(120)
      IK=47
      CALL ERASE
C*****      CALL AXIS
      CALL AXIS
C*****      PLOT DATA
      IX=100
      IY=150
      CALL MOVABS(IX, IY)
      DO 300 I=1, 512
      NY=IDATA(I)
      CALL DRWABS(IX+I, IY+NY*2)
C*****      DECIDE WHETHER TO MARK P WAVES
      IF(IP(J).NE.I) GO TO 300
      LP=200
      LT=IP(J)+100
C*****      CALL PMARK
      CALL PMARK(LT, LP)
      J=J+1
300  CONTINUE
      IX=15
      IY=600
      CALL MOVABS(IX, IY)
      CALL MOVABS(IX, IY)
      CALL ANMODE
C*****      WRITE VALUE OF P-P INTERVAL IN NUMBER OF DATA
C      POINTS. MULTIPLY VALUE BY .012 FOR TIME BETWEEN
C      P WAVES.
      DO 301 K=J, LK
      WRITE 900, PP(K)
900  FORMAT(50X, 16HP-P INTERVAL IS , I3).
301  CONTINUE
C*****      PAUSE TO STUDY PLOT
      CALL MOVABS(15, 600)
      CALL DCURSR(ICH, IX, IY)
      CALL ERASE
C*****      PREPARE SCREEN FOR REMAINING PROGRAM
      CALL MOVABS(0, 780)
      CALL ANMODE
308  RETURN
      END

```

Fig. 33. Subroutine MARK draws a graph of the data.

```

C*****
C      SUBROUTINE AXIS - DRAW PICTURE FRAME, AXES, TIC MARKS
C      AND WRITE LABELS
C*****
      SUBROUTINE AXIS
C*****
      DATA TO LABEL Y-AXIS
      DATA LYLET/1HA,1HM,1HP,1HL,1HI,1HT,1HU,1HD,1HE,
      11H,7H(4V/40)/
      DATA NYTIC/2H56,2H42,2H28,2H14/
C*****
      DRAW PICTURE FFAME
      IX=15
      IY=75
      CALL MOVABS(IX,IY)
      CALL DRWABS(IX,IY+655)
      CALL DRWABS(IX+1008,IY+655)
      CALL DRWABS(IX+1008,IY)
      CALL DRWABS(IX,IY)
C*****
      DRAW AXIS
      IX=100
      IY=150
      CALL MOVABS(IX,IY)
      CALL DRWABS(IX+823,IY)
      CALL MOVABS(IX,IY)
      CALL DRWABS(IX,IY+530)
C*****
      DRAW X-AXIS TIC MARKS AND LABEL TIC MARKS
      IY=100
      CALL MOVABS(IX,IY)
      DO 1 I=110,923,10
      IY=150
      IX=I
      CALL MOVABS(IX,IY)
      CALL DRWABS(IX,IY-5)
1    CONTINUE
      IY=150
      DO 8 K=110,923,210
      CALL MOVABS(K,IY)
      CALL DRWABS(K,IY-15)
8    CONTINUE

```

Fig. 34. Subroutine AXIS prepares the CRT screen to plot the data.

```

C*****      DRAW Y-AXIS AND LABEL TIC MARKS
      DO 5 I=160,680,10
      IX=100
      IY=I
      CALL MOVARS(IX,IY)
      CALL DRWABS(IX-5,IY)
5      CONTINUE
      DO 9 L=290,680,130
      CALL MOVARS(100,L)
      CALL DRWABS(85,L)
9      CONTINUE
C*****      LABEL Y-AXIS
      CALL MOVABS(40,530)
      CALL MOVARS(40,530)
      CALL ANMODE
      DO 20 I=1,10
      WRITE 900,LYLET(I)
900    FORMAT(5X,A1)
20     CONTINUE
      WRITE 904,LYLET(11)
904    FORMAT(4X,A1)
      WRITE 905
905    FORMAT(5X,A1(/))
C*****      LABEL X-AXIS MAJOR TIC MARKS
      WRITE 901
901    FORMAT(40X,3H420,24X,3H840,24X,4H1260)
      WRITE 906
906    FORMAT(1(/))
C*****      LABEL X-AXIS
      WRITE 902
902    FORMAT(53X,11HTIME (MSEC))
C*****      LABEL Y-AXIS TIC MARKS
      IY=685
      CALL MOVARS(70,IY)
      CALL MOVARS(70,IY)
      CALL ANMODE
      DO 21 I=1,4
      WRITE 903,NYTIC(I)
903    FORMAT(10X,A2,10(/))
21     CONTINUE
      RETURN
      END

```

Fig. 35. Subroutine AXIS is continued.

time for the first 10 entries. Character 11 is printed and finally 21 vertical spaces are skipped to set up for labeling the X-axis major tic marks. These tic marks are labeled, and then some vertical space is skipped to label the X-axis. Finally the Y-axis major tic marks are labeled by a DO LOOP that prints one data character at a time from NYTIC, then skips 10 vertical spaces before printing the next set of characters.

After control is returned from AXIS, subroutine MARK plots the data and marks identified P waves by using a DO LOOP. The X position of the CRT light beam is determined by the DO LOOP control variable. The Y position is determined by multiplying the individual values in IDATA by two (2) and adding this to the initial value of IV. A comparison is made between the next value in the IP matrix and the DO LOOP control variable. If these two are not equal, the DO LOOP is continued. If the two values are equal, a P wave has been identified at this position and needs to be marked. The X position (LT) and Y position (LP) are established and subroutine PMARK is called.

Subroutine PMARK is a short subroutine that simply marks identified P waves (see Figure 36). This subroutine takes the parameters set in MARK and draws a vertical line 100 raster units high to identify the P wave. Then control is returned to MARK.

After the P waves have been marked, subroutine MARK increments the output P wave counter (J) and continues the

```

C*****
C          SUBROUTINE PMARK - MARK IDENTIFIED P WAVES
C*****
      SUBROUTINE PMARK(IX,IY)
      CALL MOVAPS(IX,IY)
      CALL DRWAPS(IX,IY+100)
      RETURN
      FND

```

Fig. 36. Subroutine PMARK marks P waves in the output.

DO LOOP. When the loop has been completed, the CRT beam is moved to a point above the plot and the terminal mode is changed to alphanumeric. Then a DO LOOP is started that prints the P-P interval for those intervals found in that data block. After this loop is completed, the program is halted by a call to DCURSR. This halt is set so the operator can study the output and make copies if needed. A typed zero (0) and RETURN continues the program. Then the light beam is moved to the upper left hand corner, and control is returned to the main program.

Ending the Main Program.--After control is returned from subroutine MARK, MA is incremented by MB (see Figure 37).

```

      MA=MA+MB
6      LB=512-ISTEP
C*****      PAUSE TO STUDY OUTPUT
      WRITE*,"TYPE 1 WHEN READY TO CONTINUE."
      WRITE*,"TYPE 2 IF WANT TO STOP."
      READ*,N2
      IF(N2.EQ.2)GO TO 900
5      CONTINUE
900     STOP
      END

```

Fig. 37. The main program is ended.

Next the value of LB is calculated by subtracting ISTEP from 512. Then a chance for the operator to terminate the program is offered. Directions to the operator are presented. The value entered (N2) is read and a decision using this value is made. If N2 equals two (2), the program goes to STOP. If N2 does not equal two, the DO LOOP controlled by K1 is continued.

Computer Program

```

PROGRAM FIND(TAPE2,INPUT,OUTPUT,PLOT)
C*****
C      THIS IS MAIN PFGOAM TO FIND P WAVES IN AN ECG
C*****
COMMON IDATA(512),N1,LK,IP(3000),N/GREAT/IIP,ND/MARK/I3AUD,J
1,PP(2785)
COMMON /MAX/ISTEP,IMAX(6),IPP(6),KK,NB,NC,M
INTEGER PP
REWIND 2
C*****      READ IN THE I3AUD VALUE FOR INITT
C      TAKE THE LISTED 3AUD RATE AND DIVIDE BY 10
WRITE*,32HTHE 3AUD RATE DIVIDED BY 10 IS
READ*,I3AUD
C*****      SET UP READ AND ANALYSIS LOOP
WRITE*,24HTHE NUMBER OF PEAKS IS
READ*,N
WRITE*,53HTYPE 0 - IF P WAVE IS POSITIVE: TYPE 1 - IF P WAVE
1IS NEGATIVE
READ*,N1
WRITE*,36HWHICH BLOCK OF DATA FIRST GRAPHED?
READ*,M1
WRITE*,44HHOW MANY BLOCKS OF DATA BETWEEN EACH GRAPH?
READ*,M2
DO 5 K1=1,2786
C*****      READ DATA
CALL READ
C*****      IF NOT FIRST BLOCK OF DATA FIND REMAINING P WAVES
IF(K1.GE.2)GO TO 102
C*****
C      INITIALLY FIND P-P INTERVAL
C*****
NB=2
NC=500
KK=2*N
C*****      FIND THE MAXIMUM POINTS
CALL MAX

```

```

KB=0
DO 7 K=1,4,N
C*****
      FIND R AND THEN P WAVES
      KB=KB+1
      IF(N.EQ.2)GO TO 108
      IF(IMAX(K).LE.IMAX(K+1))109,110
      IF(IMAX(K+1).LE.IMAX(K+2))111,112
109  IP(KB)=IPP(K+1)
111  GO TO 7
110  IF(IMAX(K).LE.IMAX(K+2))114,115
114  GO TO 111
115  IP(KB)=IPP(K+2)
      GO TO 7
112  IP(KB)=IPP(K)
      GO TO 7
108  IF(IMAX(K).LE.IMAX(K+1))116,117
116  GO TO 112
117  IP(KB)=IPP(K+1)
7    CONTINUE
C*****      FIND P-P INTERVAL
      PP(1)=IP(2)-IP(1)
C*****      SET PARAMETERS TO FIND REMAINING P WAVES IN
C              FIRST BLOCK OF DATA
      J=1
      LK=1
      KNPT=6
      ISTEP=IP(2)
C*****      SET PARAMETERS TO FIND P WAVES IN REMAINING 3_BLOCKS
C              OF DATA
      GO TO 119
102  IIP=PP(LK)-(513-ISTEP)
      ISTEP=1
      KNPT=LK+5
      J=LK+1
C*****      FIND P WAVES
119  DO 8 KN=LK,KNPT
C*****      CALL GREATER

```

```

000460
000470
000480
000490
000500
000510
000520
000530
000540
000550
000560
000570
000580
000590
000600
000610
000620
000630
000640
000650
000660
000670
000680
000690
000700
000710
000720
000730
000740
000750
000760
000770
000780
000790
000800
000810
000820

```

```

CALL GREATER
IF(ND.GT.512)GO TO 118
KL=KN+2
IP(KL)=ISTEP
LK=KN+1
IF(ISTEP.LT.PP(LK-1))GO TO 120
PP(LK)=IP(KL)-IP(LK)
GO TO 121
120 PP(LK)=IP(KL)+LB
121 IF(ISTEP.GE.512)GO TO 118
IIP=PP(LK)
8 CONTINUE
C***** DECIDE WHICH DATA TO GRAPH
118 IF(MA.NE.K1)GO TO 6
C***** CALL MARK
CALL MARK
MA=MA+M3
6 LB=512-ISTEP
C***** PAUSE TO STUDY OUTPUT
WRITE*,"TYPE 1 WHEN READY TO CONTINUE."
WRITE*,"TYPE 2 IF WANT TO STOP."
READ*,N2
IF(N2.EQ.2)GO TO 900
5 CONTINUE
900 STOP
END
C*****
C SUBROUTINE GREATER - FINDS P WAVES
C*****
C SUBROUTINE GREATER
COMMON IDATA(512),N1,LK,IP(3000),N/GREAT/IIP,ND
COMMON /MAX/ISTEP,IMAX(6),IPP(6),KK,N3,NC,M
C***** SET PARAMETERS
ND=ISTEP+IIP
NB=ISTEP+IIP-2
NA=NB

```

```

000830
000840
000850
000860
000870
000880
000890
000900
000910
000920
000930
000940
000950
000960
000970
000980
000990
001000
001010
001020
001030
001040
001050
001060
001070
001080
001090
001100
001110
001120
001130
001140
001150
001160
001170
001175

```

```

001180
001185
001190
001200
001210
001220
001230
001240
001250
001260
001270
001280
001290
001300
001310
001311
001312
001313
001314
001315
001316
001317
001320
001330
001340
001350
001360
001370
001380
001390
001400
001410
001420
001430
001440
001450

NC=NB+5
NE=NC
IF(NC.GT.512)GO TO 201
KK=6
C*****
206 CALL MAX
      CALL MAX
C*****
IF(M.LE.1)GO TO 203
      IF MORE THAN ONE MAX POINT, FIND HIGHEST MAX
C*****
      DO 204 KA=1,M
      IF(IMAX(1).GE.IMAX(KA))GO TO 204
      IMAX(1)=IMAX(KA)
      IPP(1)=IPP(KA)
204 CONTINUE
C*****
      DECIDE IF P WAVE IS PRESENT
203 IF(M.EQ.1)GO TO 202
      NB=NB+2
      NC=NC+2
      IF((NC-NB).GE.30)GO TO 205
      IF(NC.LE.512)GO TO 207
      NC=NE+1
      IF(NC.EQ.512)GO TO 205
207 GO TO 206
C*****
      IF NOT A MAX POINT, KEEP SAME P-P INTERVAL
205 IB=NA+2
      ISTEP=I3
      GO TO 201
C*****
      IF MAX POINT, SET DATA INDEX
202 ISTEP=IPP(1)
201 RETURN
      END
C*****
SUBROUTINE MARK - GRAPHS DATA, SET PARAMETERS FOR MARKING
      P WAVES, WRITES P-P INTERVAL VALUE
C*****
SUBROUTINE MARK
      COMMON IDATA(512),N1,LK,IP(3000),N/MARK/IBAUD,J,PP(2735)

```

```

C***** INITIALIZE SCREEN
IF(IK.NE.47)CALL INITT(120)
IK=47
CALL ERASE
C***** CALL AXIS
CALL AXIS
C***** PLOT DATA
IX=100
IY=150
CALL MOVABS(IX,IY)
DO 300 I=1,512
NY=IDATA(I)
CALL DRABS(IX+I,IY+NY*2)
C***** DECIDE WHETHER TO MARK P WAVES
IF(IP(J).NE.I)GO TO 300
LP=200
LT=IP(J)+100
C***** CALL PMARK
CALL PMARK(LT,LP)
J=J+1
300 CONTINUE
IX=15
IY=600
CALL MOVABS(IX,IY)
CALL MOVABS(IX,IY)
CALL ANMODE
C***** WRITE VALUE OF P-P INTERVAL IN NUMMER OF DATA
C POINTS. MULTIPLY VALUE BY .002 FOR TIME BETWEEN
C P WAVES.
DO 301 K=J,LK
WRITE 900,PP(K)
900 FORMAT(50X,16HP-P INTERVAL IS ,I3)
301 CONTINUE
C***** PAUSE TO STUDY PLOT
CALL MOVABS(15,600)
CALL OCURR(10,IX,IY)

```

```

001460
001470
001480
001490
001500
001510
001520
001530
001540
001550
001560
001570
001580
001590
001600
001610
001620
001630
001640
001650
001660
001670
001680
001690
001700
001710
001720
001730
001740
001750
001760
001770
001780
001790
001800
001810

```

```

001820
001830
001840
001850
001860
001870
001880
001890
001900
001910
001920
001930
001940
001950
001960
001970
001980
001990
002000
002010
002020
002030
002040
002050
002060
002070
002080
002090
002100
002110
002120
002130
002140
002150
002160
002170

C*****      CALL ERASE
C*****      PREPARE SCREEN FOR REMAINING PROGRAM
      CALL MOVARS(0,760)
      CALL ANMODE
398      RETURN
      END
C*****
C*****      SUBROUTINE PMAFK - MARK IDENTIFIED P WAVES
C*****
      SUBROUTINE PMAFK(IX,IY)
      CALL MOVAPS(IX,IY)
      CALL DRWAPS(IX,IY+100)
      RETURN
      END
C*****
C*****      SUBROUTINE READ - READ IN DATA, GET DATA INFO
C*****      POSITIVE P WAVE FORM, AND CONVERT TO INTEGER FORMAT
C*****
      SUBROUTINE READ
      COMMON IDATA(512),N1,LK,IP(3000),N
      DIMENSION DATA(512)
      READ(2,10) (DATA(I),I=1,512)
      FORMAT (10F8.2)
      IF(EOF(2).NE.0.)GO TO 9
      IF(N1.EQ.1)GO TO 13
C*****      FOR POSITIVE P WAVE
      DO 1 I=1,512
      IF(DATA(I).LT.0.)DATA(I)=0.
      1      IDATA(I)=DATA(I)
      GO TO 9
C*****      FOR NEGATIVE P WAVE
      DO 11 I=1,512
      IDATA(I)=DATA(I)
      IDATA(I)=IARS(IDATA(I))
      IF(IDATA(I).LE.10)IDATA(I)=0
      11      CONTINUE

```

```

002180
002190
002200
002210
002220
002230
002235
002240
002250
002260
002270
002280
002290
002300
002310
002320
002330
002340
002350
002360
002370
002380
002390
002400
002410
002420
002430
002440
002450
002460
002470
002480
002490
002500
002510
002520

N=2
9 RETURN
END
C*****
C SUBROUTINE MAX - FIND MAXIMUM POINT USING A CHANGE
C IN SLOPE PROCEDURE
C*****
SUBROUTINE MAX
COMMON IOATA(512),N1,LK,IP(3000),N
COMMON /MAX/ISTEP,IMAX(6),IPP(6),KK,NR,NC,M
LSIGN=1
M=0
DO 13 I=NR,NC
C***** FIND THE DIFFERENCE IN AMPLITUDE BETWEEN LAST AND
C PRESENT DATA POINTS - SLOPE
NA=IOATA(I)-IOATA(I-1)
C***** IF SLOPE IS POSITIVE
IF (NA.GT.0) GO TO 2
C***** IF SLOPE IS NEGATIVE
IF (NA.LT.0) GO TO 4
C***** IF SLOPE IS ZERO
GO TO 13
C***** FOR POSITIVE SLOPE
2 JSIGN=1
GO TO 3
C***** FOR NEGATIVE SLOPE
4 JSIGN=0
C***** IF SLOPE DOES NOT CHANGE, OR IF SLOPE GOES FROM
C NEGATIVE TO POSITIVE: CONTINUE ROUTINE
IF (JSIGN.EQ.LSIGN) GO TO 3
IF ((LSIGN-JSIGN).LE.0) GO TO 3
C***** IF SLOPE CHANGE FROM POSITIVE TO NEGATIVE, SET
C MAXIMUM POINT
M=M+1
IMAX(M)=IOATA(I-1)
IPP(M)=I-1

```

```

C***** IF FIRST MAXIMUM FOUND,CONTINUE PROGRAM
IF (M.EQ.1)GO TO 3
C***** IF ANOTHER MAXIMUM FOUND WITHIN A REGION, FIND
C      MAXIMUM OF TWO VALUES
IF ((IPP(M)-IPP(M-1)).GT.3)GO TO 7
IF (IMAX(M).GE.IMAX(M-1))5,6
5  IMAX(M-1)=IMAX(M)
6  IPP(M-1)=IPP(M)
M=M-1
C***** IF NUMBER OF MAXIMUM POINTS EQUAL TH NUMBER NEEDED,
C      RETURN
7  IF (M.EQ.KK)GO TO 15
C***** SET PREVIOUS SLOPE INDICATOR
3  LSIGN=JSIGN
13  CONTINUE
15  RETURN
END
C***** SUBROUTINE AXIS - DRAW PICTURE FRAME, AXES, TIC MARKS
C      AND WRITE LABELS
C***** SUBROUTINE AXIS
C***** DATA TO LABEL Y-AXIS
DATA LYLET/1HA,1HM,1HP,1HL,1HI,1HT,1HU,1HD,1HE,
11H,7H(4V/40)/
DATA NYTIC/2H56,2H42,2H28,2H14/
C***** DRAW PICTURE FRAME
IX=15
IY=75
CALL MOVARS(IX,IY)
CALL DRWARS(IX,IY+655)
CALL DRWARS(IX+1008,IY+655)
CALL DRWARS(IX+1008,IY)
CALL DRWARS(IX,IY)
C***** DRAW AXIS
IX=100

```

```

002530
002540
002550
002560
002570
002580
002590
002600
002610
002620
002630
002640
002650
002660
002670
002680
002690
002700
002710
002720
002730
002740
002750
002760
002770
002780
002790
002800
002810
002820
002830
002840
002850
002860
002870
002880

```

```

      IY=150
      CALL MOVABS(IX,IY)
      CALL DRWABS(IX+823,IY)
      CALL MOVABS(IX,IY)
      CALL DRWABS(IX,IY+530)
      C*****
      DRAW X-AXIS TIC MARKS AND LABEL TIC MARKS
      IY=100
      CALL MOVABS(IX,IY)
      DO 1 I=110,923,10
      IY=150
      IX=I
      CALL MOVABS(IX,IY)
      CALL DRWABS(IX,IY-5)
      CONTINUE
      1
      IY=150
      DO 8 K=310,923,210
      CALL MOVABS(K,IY)
      CALL DRWABS(K,IY-15)
      CONTINUE
      8
      C*****
      DRAW Y-AXIS AND LABEL TIC MARKS
      DO 5 I=160,680,10
      IX=100
      IY=I
      CALL MOVABS(IX,IY)
      CALL DRWABS(IX-5,IY)
      CONTINUE
      5
      DO 9 L=290,680,130
      CALL MOVABS(100,L)
      CALL DRWABS(85,L)
      CONTINUE
      9
      C*****
      LABEL Y-AXIS
      CALL MOVABS(40,530)
      CALL MOVABS(40,530)
      CALL ANMODE
      DO 20 I=1,10
      WRITE 900,LYLET(I)

```

```

002890
002900
002910
002920
002930
002940
002950
002960
002970
002980
002990
003000
003010
003020
003030
003040
003050
003060
003070
003080
003090
003100
003110
003120
003130
003140
003150
003160
003170
003180
003190
003200
003210
003220
003230
003240

```

```

900  FORMAT(5X,A1)
20   CONTINUE
    WRITE 904,LYLET(11)
904  FORMAT(4X,A7)
    WRITE 905
905  FORMAT(5X,21(/))
C*****      LABEL X-AXIS MAJOR TIC MARKS
    WRITE 901
901  FORMAT(40X,3H420,24X,3H40,24X,4H1250)
    WRITE 906
906  FORMAT(1(/))
C*****      LABEL X-AXIS
    WRITE 902
902  FORMAT(53X,11HTIME (MSEC))
C*****      LABEL Y-AXIS TIC MARKS
    IY=685
    CALL MOVABS(70,IY)
    CALL MOVABS(70,IY)
    CALL ANMODE
    DO 21 I=1,4
    WRITE 903,NYTIC(I)
903  FORMAT(10X,A2,10(/))
21   CONTINUE
    RETURN
    END

```

```

003250
003260
003270
003280
003290
003300
003310
003320
003330
003340
003350
003360
003370
003380
003390
003400
003410
003420
003430
003440
003450
003460
003470
003480
003490

```

Vita

Captain Charles E. Hightower was born 30 October 1946 in Fort Worth, Texas. He graduated from high school in Fort Worth, Texas, in 1964, and received the degree of Bachelor of Arts in Mathematics from North Texas State University in June 1968. He joined the U. S. Air Force in September, 1968, and was commissioned a Second Lieutenant in the U. S. Air Force after completing Officer Training School in December, 1968. He attended undergraduate Pilot Training at Laredo AFB, Texas in 1969 and completed Electronic Systems Maintenance Course at Keesler AFB, Mississippi, in August, 1970. His next assignment was as Maintenance Officer and Detachment Commander of Detachment 6, 6th Weather Squadron, Ellsworth AFB, South Dakota. In 1972, he was assigned to Detachment 4, 623rd Aircraft Control and Warning Squadron, Okinawa, Japan. After returning to the United States in 1973, he was assigned to the 44th Strategic Missile Wing, Ellsworth AFB, South Dakota, where he served as Combat Crew Commander, Alternate Command Post Commander, Combat Crew Instructor, and Alternate Command Post Instructor. He entered the Engineering Sciences program at Air Force Institute of Technology in August 1976, completing the program in 1977. He subsequently entered the resident graduate electrical engineering program at the same institution in July 1977.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE (When Data Entered)

REPORT DOCUMENTATION PAGE		READ INSTRUCTIONS BEFORE COMPLETING FORM
1. REPORT NUMBER GE/BE/78-27	2. GOVT ACCESSION NO.	3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle) LOCATION OF DISASSOCIATED P WAVE IN AN ELECTROCARDIOGRAM		5. TYPE OF REPORT & PERIOD COVERED MS Thesis
		6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s) Charles E. Hightower Captain		8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS Air Force Institute of Technology (AFIT-EM) Wright-Patterson AFB, Ohio 45433		10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS Aerospace Medical Research Lab/MEB Wright-Patterson AFB OH 45433		12. REPORT DATE December 1978
		13. NUMBER OF PAGES 97
14. MONITORING AGENCY NAME & ADDRESS (if different from Controlling Office)		15. SECURITY CLASS. (of this report) Unclassified
		15a. DECLASSIFICATION/DOWNGRADING SCHEDULE
16. DISTRIBUTION STATEMENT (of this Report) Approved for public release; distribution unlimited		
17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)		
18. SUPPLEMENTARY NOTES Approved for public release; IAW AFR 190-17 Joseph P. Hipps, Major, USAF Director of Information 16 MAY 1979		
19. KEY WORDS (Continue on reverse side if necessary and identify by block number) Electrocardiogram ECG P Wave Computer Program and ECG		
20. ABSTRACT (Continue on reverse side if necessary and identify by block number) Heart failure was induced in one dog by a blockage of the anterior descending branch of the coronary ostium and in another dog by overdose of digitalis. The electrocardiogram (ECG) from each dog was recorded on an analog magnetic tape. An algorithm to locate and identify the P wave using the periodicity characteristic of the P wave was tested against this data. The P waves in the first two heart beats are identified by their proximity to		

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)

Block 20.

(the R wave. The distance between these wave shapes (P-P interval) is "stepped" down the ECG. At each "step" a search is initiated for the peak of the P wave. The accuracy of the algorithm is excellent until an extreme abnormal wave shape, the area of the P wave, is encountered. When this happens, the algorithm may stop identifying P waves. A more thorough knowledge of the relationships between the P wave and other wave shapes needs to be known.

Unclassified

SECURITY CLASSIFICATION OF THIS PAGE(When Data Entered)