13

ECG Analysis Systems

Willis J. Tompkins

This chapter covers the techniques for analysis and interpretation of the 12-lead ECG. Then it discusses ST-level analysis that is used in cardiac stress test systems. Finally, there is a summary of the hardware and software design of a portable ECG arrhythmia monitor.

13.1 ECG INTERPRETATION

Computer interpretation of the 12-lead ECG uses algorithms to determine whether a patient is normal or abnormal. It also provides written description of any abnormalities discovered.

13.1.1 Historical review of ECG interpretation by computer

ECG interpretation techniques were initially developed and used on mainframe computers in the early 1960s (Pordy et al., 1968). In those days, mainframe computers centrally located in computing centers performed the ECG analysis and interpretation. The ECGs were transmitted to the computer from remote hospital sites using a specially designed ECG acquisition cart that could be rolled to the patient's bedside. The cart had three ECG amplifiers, so three leads were acquired simultaneously and transmitted over the voice-grade telephone network using a three-channel analog FM modem. The interpretation program running in the mainframe computer consisted of several hundred thousand lines of FORTRAN code.

As technology evolved, minicomputers located within hospitals took over the role of the remote mainframes. The ECG acquisition carts began to include embedded microprocessors in order to facilitate ECG capture. Also, since the interpretation algorithms had increased failure rates if the ECG was noisy, the microprocessors increased the signal-to-noise ratio by performing digital signal preprocessing algorithms to remove baseline drift and to attenuate powerline interference.

Ultimately the ECG interpretation programs were incorporated within the bedside carts themselves, so that the complete process of acquisition, processing, and interpretation could be done at the patient's bedside without transmitting any data to a remote computer. This technology has now evolved into stand-alone microprocessor-based interpretive ECG machines that can be battery powered and small enough to fit in a briefcase.

The early ECG carts had three built-in ECG amplifiers and transmitted 2.5-second epochs of three simultaneous channels. In order to acquire all 12 leads, they sequenced through four groups of three leads each, requiring 10 seconds to send a complete record. Thus, the four acquired three-lead sets represented four different time segments of the patient's cardiac activity. Since a 2.5-second interval only includes two or three heartbeats, the early algorithms had difficulty in deducing abnormalities called arrhythmias in which several heartbeats may be involved in a rhythm disturbance. In order to improve arrhythmia analysis, three additional leads, typically the VCG leads, were recorded for a longer period of six seconds and added to the acquired data set (Bonner and Schwetman, 1968).

The modern microprocessor-based interpretive machines include eight ECG amplifiers so that they can simultaneously sample and store eight leads—I, II, and V1–V6. They then synthesize the four redundant leads—III, aVR, aVL, and aVF (see Chapter 2). These machines include enough memory to store all the leads for a 10-second interval at a clinical sampling rate of 500 sps.

13.1.2 Interpretation of the 12-lead ECG

ECG interpretation starts with feature extraction, which has two parts as shown in Figure 13.1. The goals of this process are (1) waveform recognition to identify the waves in the ECG including the P and T waves and the QRS complex, and (2) measurement to quantify a set of amplitudes and time durations that is to be used to drive the interpretation process. Since the computer cannot analyze the ECG waveform image directly like the human eye-brain system, we must provide a relevant set of numbers on which it can operate.

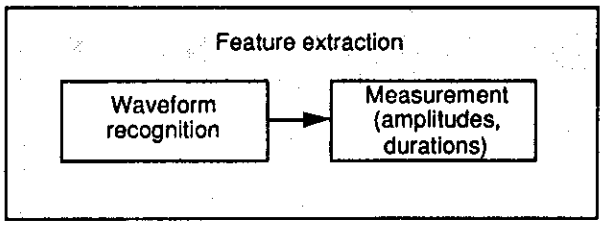

Figure 13.1 ECG feature extraction.

The first step in waveform recognition is to identify all the beats using a QRS detection algorithm (as in Chapter 12). Second, the similar beats in each channel are time-aligned and an average (or median) beat is produced for each of the 12 leads (see Chapter 9). These 12 average beats are analyzed to identify additional waves and other features of the ECG, and a set of measurements is then made and assembled into a matrix. These measurements are analyzed by subsequent processes.

Figure 13.2 is an ECG of a normal male patient acquired using an interpretive ECG machine. Although eight seconds of each lead are stored in the machine, only 2.5 seconds of each lead are printed on the paper copy as a summary. Figure 13.3 shows the internal measurement matrix that the ECG machine found for this patient. The amplitudes are listed in μV and the durations in ms. For example, in lead I, the R-wave amplitude (RA) is 1140 μV or 1.14 mV, and the R-wave duration (RD) is 71 ms.

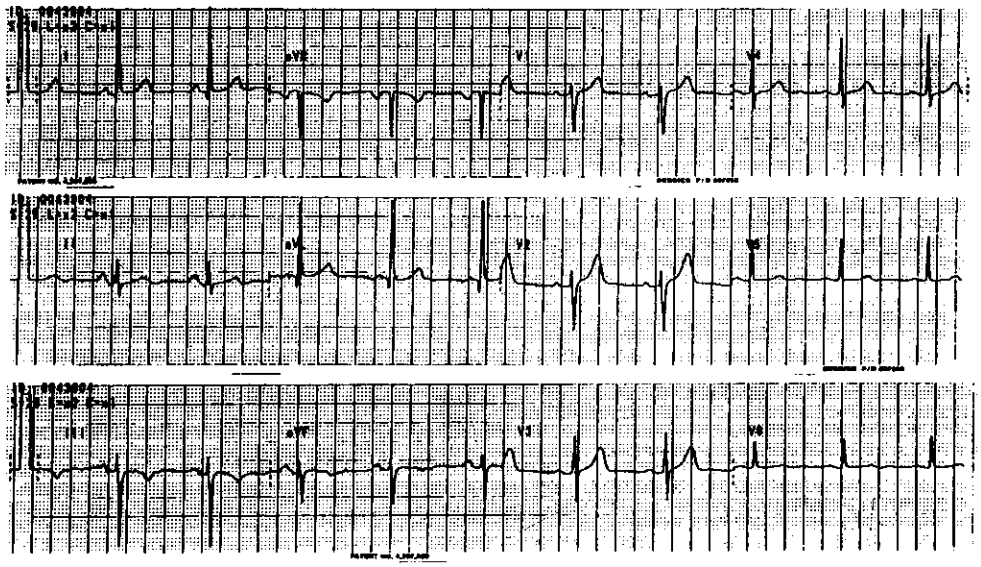

Figure 13.2 The 12-lead ECG of a normal male patient. Calibration pulses on the left side designate 1 mV. The recording speed is 25 mm/s. Each minor division is 1 mm, so the major divisions are 5 mm. Thus in lead I, the R-wave amplitude is about 1.1 mV and the time between beats is almost 1 s (i.e., heart rate is about 60 bpm). Recording was made on an Elite interpretive ECG machine (Siemens Burdick, Inc.).

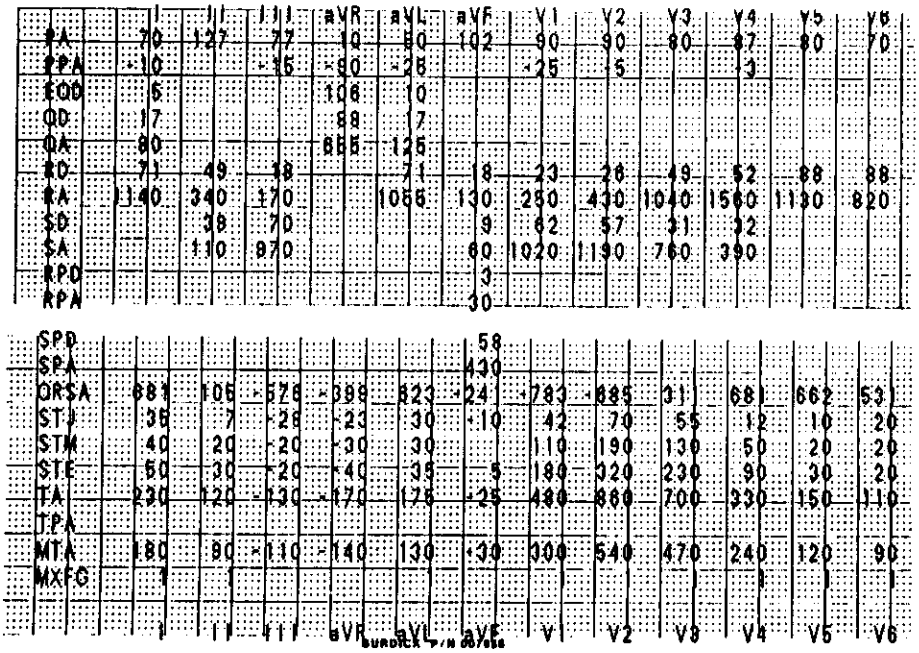

Figure 13.3 Measurement matrix produced by an Elite interpretive ECG machine (Siemens Burdick, Inc.). Amplitudes are in μV and durations in ms.

There are two basic approaches for computerized interpretation of the ECG. The one used in modern commercial instrumentation is based on decision logic (Pordy et al., 1968; Macfarlane et al., 1971). A computer program mimics the human expert's decision process using a rule-based expert system. The second approach views ECG interpretation as a pattern classification problem and applies a multivariate statistical pattern recognition method to solve it (Klingeman and Pipberger, 1967).

Figure 13.4 shows the complete procedure for interpretation of the ECG. The feature extraction process produces a set of numbers called the measurement matrix. These numbers are the inputs to a decision logic or statistical process that drives an interpretation process which assigns words to describe the condition of the patient.

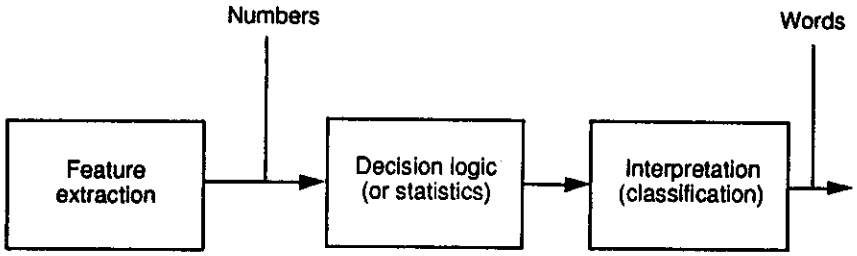

Figure 13.4 The steps in ECG interpretation.

The decision logic approach is based on a set of rules that operate on the measurement matrix derived from the ECG. The rules are assembled in a computer program as a large set of logical IF-THEN statements. For example, a typical decision rule may have the following format (Bonner and Schwetman, 1968):

```
Rule 0021:      IF
                (1) QRS ≥ .11 sec. on any two limb leads AND
                (2) Sd. ≥ .04 sec. on lead I or aVL  AND
                (3) terminal R present lead V1

                THEN
                (a) QRS .11 seconds;  AND
                (b) terminal QRS rightward and anterior; AND
                (c) incomplete right bundle branch block
```

The rules are usually developed based on knowledge from human experts. The pathway through the set of IF-THEN statements ultimately leads to one or more interpretive statements that are printed in the final report. Unfortunately, it is well known that a group of cardiologists typically interpret the same set of ECGs with less than 80 percent agreement. In fact, if the same ECGs are presented to one cardiologist at different times, the physician typically has less than 80 percent agreement with his/her previous readings. Thus, a decision logic program is only as good as the physician or group of physicians who participate in developing the knowledge base.

One advantage of the decision logic approach is that its results and the decision process can easily be followed by a human expert. However, since its decision rules are elicited indirectly from human experts rather than from the data, it is likely that such a system will never be improved enough to outperform human experts. Unlike human experts, the rule-based classifier is unable to make use of the waveforms directly. Thus, its capability is further limited to looking at numbers that are extracted from the waveforms that may include some measurement error. Also, with such an approach, it is very difficult to make minor adjustments

to one or few rules so that it can be customized to a particular group of patients. Even slightly changing one rule may lead to modification of many different pathways through the logic statements.

For the multivariate statistical pattern recognition approach to ECG interpretation, each decision is made directly from the data; hence this approach is largely free from human influence. Decisions are made based on the probabilities of numbers in the ECG measurement matrix being within certain statistical ranges based on the known probabilities of these numbers for a large set of patients. Since this technique is dependent directly on the data and not on the knowledge of human experts, it is theoretically possible to develop an interpretive system that could perform better than the best physician.

However, unlike the decision logic approach, which can produce an explanation of how the decision is reached, there is no logic to follow in this approach, so it is not possible to present to a human expert how the algorithm made its final interpretation. This is the major reason that this technique has not been adopted in commercial instrumentation.

In clinical practice, physicians overread and correct computerized ECG interpretive reports. If similar waveforms are analyzed subsequently, the computer software makes the same diagnostic error over and over. Although it is desirable for an ECG interpretation system to "learn" from its mistakes, there is no current commercial system that improves its performance by analyzing its errors.

Figure 13.5 shows the final summary provided to the clinician by an interpretive ECG machine for the ECG of Figure 13.2. The machine has classified this patient as "Normal" with normal "Sinus rhythm."

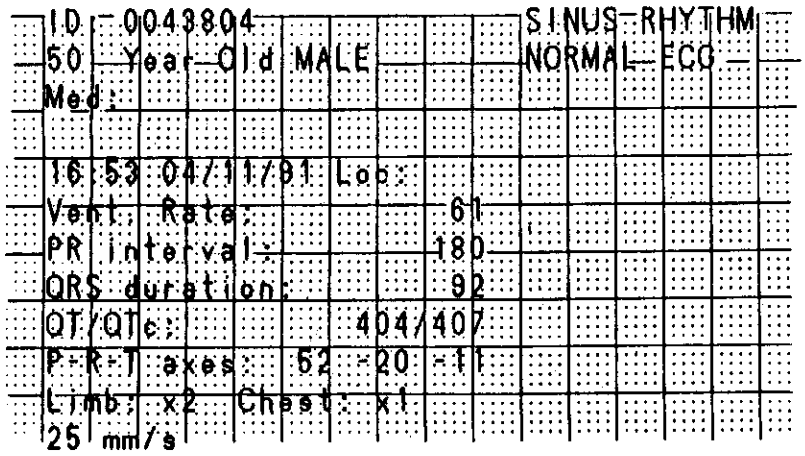

Figure 13.5 Summary and interpretation (upper right corner) produced by an Elite interpretive ECG machine (Siemens Burdick, Inc.).

13.2 ST-SEGMENT ANALYZER

The ST-segment represents the period of the ECG just after depolarization, the QRS complex, and just before repolarization, the T wave. Changes in the ST-segment of the ECG may indicate that there is a deficiency in the blood supply to the heart muscle. Thus, it is important to be able to make measurements of the ST-segment. This section describes a microprocessor-based device for analyzing the ST-segment (Weisner et al., 1982).

Figure 13.6 shows an ECG with several features marked. The analysis begins by detecting the QRS waveform. Any efficient technique can be implemented to do this. The R wave peak is then established by searching the interval corresponding to 60 ms before and after the QRS detection mark, for a point of maximal value. The Q wave is the first inflection point prior to the R wave. This inflection point is recognized by a change in the sign of slope, zero slope, or a significant change in slope. The three-point difference derivative method is used to calculate the slope. If the ECG signal is noisy, a low-pass digital filter is applied to smooth the data before calculating the slope.

The isoelectric line of the ECG must be located and measured. This is done by searching between the P and Q waves for a 30-ms interval of near-zero slope. In order to determine the QRS duration, the S point is located as the first inflection point after the R wave using the same strategy as for the Q wave. Measurements of the QRS duration, R-peak magnitude relative to the isoelectric line, and the RR interval are then obtained.

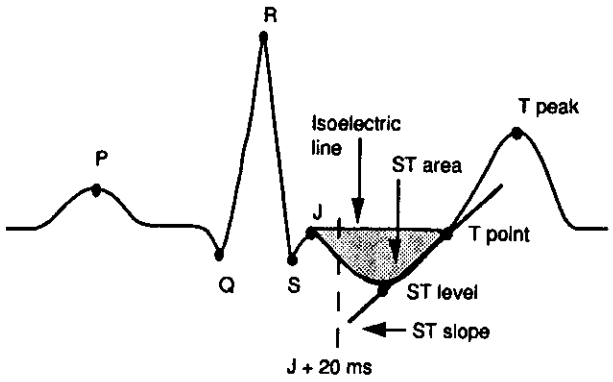

Figure 13.6 ECG measurements made by the ST-segment analyzer. The relevant points of the ECG, J, T, ST level etc. are indicated. The window in which the ST level is searched for is defined by J + 20 ms and the T point.

The J point is the first inflection point after the S point, or may be the S point itself in certain ECG waveforms. The onset of the T wave, defined as the T point, is found by first locating the T-wave peak which is the maximal absolute value, relative to the isoelectric line, between J + 80 ms and R + 400 ms. The onset of the T wave, the T point, is then found by looking for a 35-ms period on the R side of the T wave, which has values within one sample unit of each other. The T point is among the most difficult features to identify. If this point is not detected, it is assumed to be J + 120 ms.

Having identified various ECG features, ST-segment measurements are made using a windowed search method. Two boundaries, the J + 20 ms and the T point, define the window limits. The point of maximal depression or elevation in the window is then identified. ST-segment levels can be expressed as the absolute change relative to the isoelectric line.

In addition to the ST-segment level, several other parameters are calculated. The ST slope is defined as the amplitude difference between the ST-segment point and the T point divided by the corresponding time interval. The ST area is calculated by summing all sample values between the J and T points after subtracting the isoelectric-line value from each point. An ST index is calculated as the sum of the ST-segment level and one-tenth of the ST slope.

13.3 PORTABLE ARRHYTHMIA MONITOR

There is a great deal of interest these days in home monitoring of patients, particularly due to cost considerations. If the same diagnostic information can be obtained from an ambulatory patient as can be found in the hospital, it is clearly more cost effective to do the monitoring in the home. Technological evolution has led to a high-performance computing capacity that is manifested in such devices as compact, lap-sized versions of the personal computer. Such battery-powered systems provide us with the ability to do computational tasks in the home or elsewhere that were previously possible only with larger, nonportable, line-powered computers.

This increase in computing capability, with its concurrent decrease in size and power consumption, has led to the possibility of designing *intelligent* biomedical instrumentation that is small and light enough to be worn by an ambulatory patient (Tompkins, 1978; Webster, 1978; Webster et al., 1978; Tompkins et al., 1979; Tompkins, 1980; Tompkins, 1981a; Tompkins, 1981b; Tompkins, 1982; Tompkins, 1983). In addition, instrumentation can now be designed that was not contemplated previously, because microcomputer technology had not yet evolved far enough to support such applications (Sahakian et al., 1978; Tompkins et al., 1980; Weisner et al., 1982a; Weisner et al., 1982b; Chen and Tompkins, 1982; Tompkins et al., 1983). Portable instrumentation will ultimately perform monitoring functions previously done only within the confines of the hospital.

13.3.1 Holter recording

An initial goal is to replace the functions of the Holter tape recorder, the current device of choice for determining if an ambulatory patient has a potential cardiac problem. An optimal replacement device would be a microprocessor-based, portable arrhythmia monitor with processing algorithms similar to those now found in monitoring systems used in the cardiac care unit of today's hospital (Abenstein and Thakor, 1981; Thakor et al., 1984a).

Figure 13.7 shows the Holter approach, which is to record the ECG of a patient for a full day on magnetic tape. This recording and its subsequent return to a laboratory for playback and analysis restricts the timely reporting of suspected arrhythmias to the physician. The results of a Holter recording session are typically not known by the physician for several days.

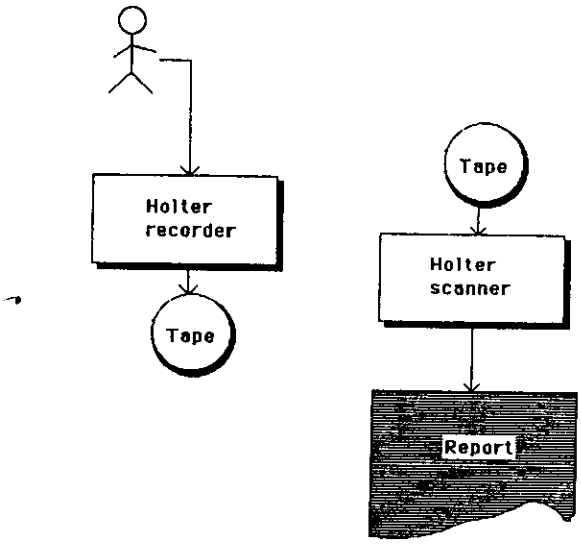

Figure 13.7 The contemporary Holter recording technique records the two-channel ECG for 24 hours on analog magnetic tape. A technician uses a Holter scanner to analyze the tape at 60 or 120 times real time and produces a final report.

What is called Holter monitoring is inappropriately named, as there is no monitoring in the normal context of the word. Monitoring typically provides continuous, real-time information about the state of the patient, as in intensive care monitoring.

The Holter recording approach to ambulatory ECG acquisition is a limited, low-technology technique. Although some improvements have been made in the small

tape recorder itself, it is an electromechanical device with limited capability of improvement. Most of the technology changes have occurred in the central station Holter scanning equipment. However, there is still a great deal of manual intervention required to reduce the data captured to a useful form. This approach is labor intensive, and so it will remain for the foreseeable future.

13.3.2 Portable arrhythmia monitor hardware design

The *intelligent* portable arrhythmia monitor of Figure 13.8 will capture the ECG during suspected abnormal periods, and immediately send selected temporal epochs back to a central hospital site through the voice-grade telephone network.

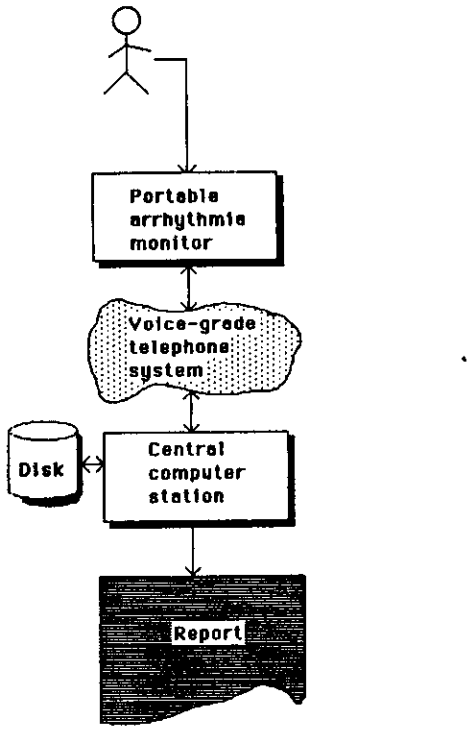

Figure 13.8 Portable arrhythmia monitor. This microprocessor-based device analyzes the ECG in real time and communicates captured data to a central host computer through the voice-grade telephone line.

This approach should provide a significant diagnostic edge to the cardiologist, who will be able to make judgments and institute therapeutic interventions in a much more timely fashion than is possible today. In addition, the clinician will be able to monitor the results of the therapy and modify it as needed, another factor not possible to do in a timely fashion with the tape recorder approach.

The hardware design of a portable arrhythmia monitor is quite straightforward, dependent only on the battery-operable, large-scale-integrated circuit components available in the marketplace. The primary semiconductor technology available for battery-operated designs is CMOS.

The hardware generations, however, evolve rapidly with the progressive improvements in semiconductor technology. For example, our initial portable arrhythmia monitor designs in 1977 were based on the COSMAC microprocessor (RCA CDP1802), a primitive central processing unit by today's standards. Modern commercial devices use the 80C86 microprocessor (Intel), a CMOS chip that is from the family of parts that make up the models of the IBM PC and compatibles.

Figure 13.9 shows a block diagram of one of our prototype monitors. In addition to the microprocessor, a portable arrhythmia monitor requires analog and digital support electronics. Analog amplifiers do the front-end ECG amplification and signal conditioning. An analog-to-digital (A/D) converter integrated circuit changes the analog ECG to the digital signal representation needed by the microprocessor.

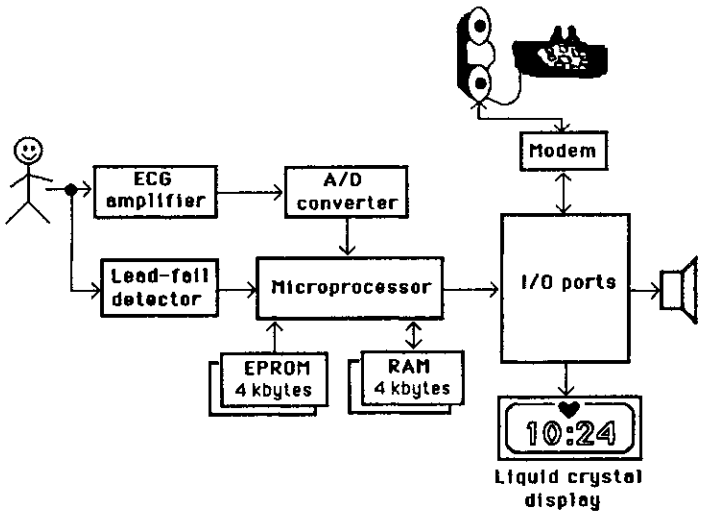

Figure 13.9 Block diagram of portable arrhythmia monitor.

ROM memory holds the program that directs the performance of all the functions of the instrument, and RAM memory stores the captured ECG signal. Input/output (I/O) ports interface audio and visual displays and switch interactions in the device. A modem circuit provides for communication with a remote computer so that captured ECG signals can be transmitted back to a central site (Thakor et al., 1982).

All these technologies are changing rapidly, so we must constantly consider improving the hardware design. Newer designs permit devices with (1) fewer components, (2) greater reliability, (3) smaller size and weight, (4) less power consumption, and (5) greater computational power (thereby permitting more complex signal processing and interpretive algorithms). Thus, there is a technological force driving us to continually improve the hardware design. In industry, an engineering compromise must come into play at some point, and the design must be frozen to produce a viable product. However, we are in a university environment, and so can afford to indulge ourselves by continual iteration of the design.

13.3.3 Portable arrhythmia monitor software design

Unfortunately, the frequent hardware changes lead to equally frequent software redesign. The software for an instrument is very hardware dependent. Each new microprocessor has its own unique machine language. Thus, we end up rewriting the same programs for different processors, and thereby waste considerable programming time. This software problem has led us to explore higher-level languages that are transportable from one kind of microprocessor to another.

We have settled on the C language, conceived at Bell Laboratories, as the most nearly ideal language now available for this type of real-time instrumentation application. Its primary advantages are (1) a programming level low enough to achieve the requisite machine control, and (2) transportability from one type of microprocessor to another. Providing that we follow a few software design rules, a real-time program written in the C language for one type of microprocessor can be easily reconfigured to run on a different one. Although it is not perfect, the C language considerably reduces the programming time necessary to rewrite the software when changing microprocessors. We then use our programming time to concentrate on improving the algorithms.

For a portable arrhythmia monitor, the two major software design tasks are (1) QRS detection and (2) arrhythmia analysis (Abenstein, 1978; Mueller, 1978). The QRS detection must be nearly perfect, otherwise the arrhythmia analysis algorithms will be fooled too often by false reports of beats that are not really there (i.e., false positives) or lack of reporting of beats that are missed (i.e., false negatives). An additional software design task important in these devices with limited memory is a data reduction algorithm (Tompkins and Abenstein, 1979; Abenstein and Tompkins, 1982).

QRS detection algorithm

Included in the various techniques that are used to implement a QRS detector are linear digital filters, nonlinear transformations, decision processes, and template matching (Thakor, 1978; Thakor et al., 1980; Ahlstrom and Tompkins, 1981; Furno and Tompkins, 1982; Thakor et al., 1983; Thakor et al., 1984b; Tompkins and Pan, 1985). Typically two or more of these techniques are combined together in a detector algorithm.

The most common approach in contemporary commercial ECG instrumentation is based on template matching. A model of the normal QRS complex, called a template, is extracted from the ECG during a learning period on a particular patient. This template is compared with the subsequent incoming real-time ECG to look for a possible match, using a mathematical criterion for goodness of fit. A close enough match to the template represents a detected QRS complex. If a waveform comes along that does not match but is a suspected abnormal QRS complex, it is treated as a separate template, and future suspected QRS complexes are compared with it. We have elected not to use this technique for the detection process, since it requires considerable memory for saving the templates (depending on how many you use; at least one system permits 40) and significant computational power for matching the templates to the real-time signal.

Instead, we use an algorithm based entirely on digital filters. Section 12.5 summarizes the QRS detection algorithm that we have developed for this application.

Arrhythmia analysis

From the QRS detector, the QRS duration and the RR intervals are determined. The ECG signal is then classified based on the QRS duration and the RR interval. Figure 13.10 is a conceptual drawing of an arrhythmia analysis algorithm based on the two parameters, RR interval and QRS duration (Ahlstrom and Tompkins, 1983). In this two-parameter mapping, we establish a region called *normal* by permitting the algorithm to first learn on a set of eight QRS complexes defined by a clinician as having *normal* rhythm and morphology for the specific patient. This learning process establishes the initial center of the *normal* region in the two-dimensional mapping space.

Boundaries of all the other regions in the map, except for region "0", are computed as percentages of the location of the center of the *normal* region. Region "0" has fixed boundaries based on physiological limits. Any point mapped into region "0" is considered to be noise because it falls outside what we normally expect to be the physiological limits of the smallest possible RR interval or QRS duration.

An abnormality such as tachycardia causes clusters of beats to fall in region "1" which represents very short RR intervals. Bradycardia beats fall in region "6". Typically, abnormalities must be classified by considering sequences of beats. For example, a premature ventricular contraction with a full compensatory pause would be characterized by a short RR interval coupled with a long QRS duration,

followed by a long RR interval coupled with a normal QRS duration. This would be manifested as a sequence of two points on the map, the first in region "3" and the second in region "5". Thus, arrhythmia analysis consists of analyzing the ways in which the beats fall onto the mapping space.

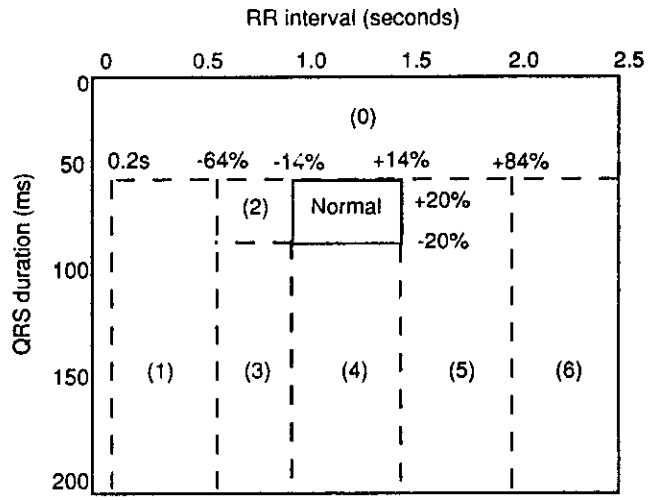

Figure 13.10 Arrhythmia analysis algorithm based on mapping the RR interval and QRS duration into two-dimensional space.

The center of the *normal* region is continuously updated, based on the average RR interval of the eight most-recent beats classified as normal. This approach permits the normal region to move in the two-dimensional space with normal changes in heart rate that occur with exercise and other physiological changes. The boundaries of other regions are modified beat-by-beat, since they are based on the location of the *normal* region. Thus, this algorithm adapts to normal changes in heart rate.

The classification of the waveforms can be made by noting the regions in which successive beats fall. Figure 13.11 lists some of the algorithms to detect different arrhythmias. The technique described is an efficient method for extracting RR interval and QRS duration information from an ECG signal. Based on the acquired information, different arrhythmias can be classified.

Normal:	If a beat falls in the normal box.
Asystole:	No R wave for more than 1.72 s; less than 35 beats/min
Dropped:	A long RR interval; beat falls in Region 6.
R-on-T:	A beat falls in Region 2.
Compensated PVC:	A beat in Region 3, followed by another in Region 5.
Uncompensated PVC:	A beat in Region 3, followed by another in the normal region.
Couplet:	Two consecutive beats in Region 3 followed by a beat in the normal region, or in Region 5.
Paroxysmal Bradycardia:	If there are at least three consecutive points in Region 5
Tachycardia:	Average RR interval is less than 120 beats/min
Fusion:	A beat with a wide QRS duration; falls in Region 4.
Escape:	A beat with a delayed QRS complex; falls in Region 5.
Rejected:	A beat that has an RR interval of 200 ms or less, or QRS duration of 60 ms or less.

Figure 13.11 Classification of beats in the ECG signal based on QRS duration and RR interval.

13.3.4 The future of portable arrhythmia monitoring

One reason that is difficult to displace the old Holter technology with a modern high-technology approach is that the former is a full-disclosure technique that is inherently resistant to change. That is, even though the typical physician looks only at the final report and almost never looks at all the complete 24-hour ECG signal recorded, it is implicit that a skilled technician has analyzed the ECG. Also, the physician has the ultimate security blanket in that the data is there should it ever be necessary to go back through it again.

A second reason that the microprocessor-based, real-time approach has not affected the Holter market as yet is that the diagnostic algorithms are not fully perfected. The portable monitor must be able to do many of the tasks that are now being done in the coronary care unit. It must be able to detect reliably QRS complexes, perhaps better than hospital-based systems, since false judgments will cause unnecessary data to be stored in its limited memory. If problems occur, the portable device must perform self-diagnosis and make suggestions to the patient as to how to cure the problem. There is no technician in the ambulatory environment to correct problems when things go wrong.

Before physicians will rush to accept such a device, it must be clearly proven in clinical trials that it can capture the important clinical information at least as well as and at lower cost than the current Holter recording approach. Such a clinical demonstration is indeed difficult since there is no golden standard for proving

performance. The MIT/BIH and AHA ECG databases serve only as preliminary evaluators.

Our current device does not attempt to make an interpretation of the patient's ECG. Instead it is designed to separate out suspicious waveforms from those that are considered normal for an individual patient. It then transmits the suspicious data by telephone to a clinician for a diagnostic judgment. Thus, it is a type of screening device. This approach as a first design step is simpler than a comprehensive ICU approach where critical clinical decisions must be made rapidly.

No portable arrhythmia monitor (or real-time Holter monitor as it is sometimes called) has yet been successful in the medical instrumentation marketplace. Part of the reason for this is that real-time QRS detection and arrhythmia analysis algorithms are not yet good enough for this application. However, there is steady progress in improvement of these algorithms, as microprocessors are being used more and more in bedside monitoring systems.

There is no doubt that Holter recording will be displaced, at some time, by microprocessor-based portable monitors. When that time comes, it will lead to lower diagnostic costs, greater device reliability, better clinical research capabilities, and continually evolving performance. The utility of these devices will evolve with the technology, just as the early four-function calculator has evolved into the lap-sized, high-performance portable computer of today.

13.4 REFERENCES

Abenstein, J. P. 1978. Algorithms for real-time ambulatory ECG monitoring. *Biomed. Sci. Instrum.*, **14**: 73–79.

Abenstein, J. P. and Thakor, N. V. 1981. A detailed design example—ambulatory ECG monitoring. In Tompkins, W. J. and Webster, J. G. (eds) *Design of Microcomputer-Based Medical Instrumentation*. Englewood Cliffs, NJ: Prentice Hall.

Abenstein, J. P. and Tompkins, W. J. 1982. A new data reduction algorithm for real-time ECG analysis. *IEEE Trans Biomed Eng.*, **BME-29**: 43–48.

Ahlstrom, M. L. and Tompkins, W. J. 1981. An inexpensive microprocessor system for high speed QRS width detection. *Proc. 1st Annual IEEE Compmed. Conf.*, 81–83.

Ahlstrom, M. L. and Tompkins, W. J. 1983. Automated high-speed analysis of Holter tapes with microcomputers. *IEEE Trans. Biomed. Eng.*, **BME-30**: 651–57.

Bonner, R. E. and Schwetman, H. D. 1968. Computer diagnosis of the electrocardiogram II. *Comp. Biomed. Res.*, **1**: 366.

Furno, G. S. and Tompkins, W. J. 1982. QRS detection using automata theory in a battery-powered microprocessor system. *IEEE Frontiers of Eng. in Health Care*, **4**: 155–58.

Klingeman, J., and Pipberger, H. V. 1967. Computer classification of electrocardiograms. *Comp. Biomed. Res.*, **1**: 1.

Macfarlane, P. W., Lorimer, A. R., and Lowrie, T. D. V. 1971. 3 and 12 lead electrocardiogram interpretation by computer. A comparison in 1093 patients. *Br. Heart J.*, **33**: 226.

Mueller, W. C. 1978. Arrhythmia detection program for an ambulatory ECG monitor. *Biomed. Sci. Instrum.*, **14**: 81–85.

Pan, J. and Tompkins, W. J. 1985. A real-time QRS detection algorithm. *IEEE Trans. Biomed. Eng.*, **BME-32**(3): 230–36.

Pordy, L., Jaffe,H., Chesky, K. et al. 1968. Computer diagnosis of electrocardiograms, IV, a computer program for contour analysis with clinical results of rhythm and contour interpretation. *Comp. Biomed. Res.*, **1**: 408–33.
Thakor, N. V. 1978. Reliable R-wave detection from ambulatory subjects. *Biomed. Sci. Instrum.*, **14**: 67–72.
Thakor, N. V., Webster, J. G., and Tompkins, W. J. 1980. Optimal QRS filter. *Proc. IEEE Conf. on Frontiers of Eng. in Health Care*, **2**: 190–95.
Thakor, N. V., Webster, J. G., and Tompkins, W. J. 1982. A battery-powered digital modem for telephone transmission of ECG data. *IEEE Trans. Biomed. Eng.*, **BME-29**: 355–59.
Thakor, N. V., Webster, J. G., and Tompkins, W. J. 1983. Optimal QRS detector. *Med. & Biol. Eng. & Comput.*, **21**: 343–50.
Thakor, N. V., Webster, J. G., and Tompkins, W. J. 1984a. Design, implementation, and evaluation of a microcomputer-based portable arrhythmia monitor. *Med. & Biol. Eng. & Comput.*, **22**: 151–59.
Thakor, N. V., Webster, J. G., and Tompkins, W. J. 1984b. Estimation of QRS complex power spectra for design of a QRS filter. *IEEE Trans. Biomed. Eng.* **BME-31**: 702–706.
Tompkins, W. J. 1978. A portable microcomputer-based system for biomedical applications. *Biomed. Sci. Instrum.*, **14**: 61–66.
Tompkins, W. J. 1980. Modular design of microcomputer-based medical instruments. *Med. Instrum.*, **14**: 315–18.
Tompkins, W. J. 1981a. Portable microcomputer-based instrumentation. In Eden, M. S. and Eden, M. (eds) *Microcomputers in Patient Care*, Park Ridge, NJ: Noyes Medical Publications, 174–81.
Tompkins, W. J. 1981b. Role of microprocessors in ambulatory monitoring. *Proc. AAMI*, 99.
Tompkins, W. J. 1982. Trends in ambulatory electrocardiography. *IEEE Frontiers of Eng. in Health Care*, **4**: 201–04.
Tompkins, W. J. 1983. Arrhythmia detection and capture from ambulatory outpatients using microprocessors. *Proc. AAMI*, 122.
Tompkins, W. J. and Abenstein, J. P. 1979. CORTES—A data reduction algorithm for electrocardiography. *Proc. AAMI*, 277.
Tompkins, W. J., Tompkins, B. M., and Weisner, S. J. 1983. Microprocessor-based device for real-time ECG processing in the operating room. *Proc. AAMI*, 122.
Tompkins, W. J., Webster, J. G., Sahakian, A. V., Thakor, N. V., and Mueller, W. C. 1979. Long-term, portable ECG arrhythmia monitoring. *Proc. AAMI*, 278.
Webster, J. G. 1978. An intelligent monitor for ambulatory ECGs. *Biomed. Sci. Instrum.*, **14**: 55–60.
Webster, J. G., Tompkins, W. J., Thakor, N. V., Abenstein, J. P., and Mueller, W. C. 1978. A portable, microcomputer-based ECG arrhythmia monitor. *Proc. 31st ACEMB*, 60.
Weisner, S. J., Tompkins, W. J., and Tompkins, B. M. 1982a. A compact, microprocessor-based ECG ST-segment monitor for the operating room. *IEEE Trans. Biomed. Eng.*, **BME-29**: 642–49.
Weisner, S. J., Tompkins, W. J., and Tompkins, B. M. 1982b. Microprocessor-based, portable anesthesiology ST-segment analyzer. *Proc. Northeast Bioeng. Conf.*, 222–26.

13.5 STUDY QUESTIONS

13.1 In the modern version of the portable arrhythmia monitor, the arrhythmia analysis is based on mapping what two variables into two-dimensional space?
13.2 Current real-time QRS detection algorithms developed at the UW can correctly detect approximately what percentage of the QRS complexes in a standard 24-hour database?

13.3 In arrhythmia analysis, the RR interval and QRS duration for each beat are mapped into a two-dimensional space. How is the location of the center of the box marked *Normal* established?

13.4 Which of the following best describe the portable arrhythmia monitor developed at UW: (a) is a distributed processing approach, (b) selects important signals and stores them on magnetic tape for subsequent playback to a central computer over the telephone, (c) stores RR intervals and QRS durations in its memory so that a 24-hour trend plot can be made for these variables, (d) uses ST-segment levels as part of the arrhythmia analysis algorithm, (e) saves 30 16-second ECG segments in its memory, (f) transmits over the telephone using a separate modem that fits in a shirt pocket, (g) currently uses an CMOS 8088 microprocessor but will be updated soon, (h) always stores the ECG segment that preceded an alarm, (i) is being designed as a replacement for a Holter recorder, (j) uses the new medical satellite network to send its data to the central computer by telemetry, (k) has a built-in accelerometer for monitoring the patient's activity level, (l) includes 256 kbytes of RAM to store ECG signals, (m) uses two features extracted from the ECG in the arrhythmia analysis, (n) does near-optimal QRS detection so it will be produced commercially by a company early next year, (o) saves all the sampled two-channel ECG data for 24 hours, (p) stores the single ECG segment that caused an alarm, (q) analyzes the 12-lead ECG.

13.5 Which of the following best describe the portable arrhythmia monitor developed by Hewlett Packard: (a) a CMOS Z80 is the microprocessor, (b) the signal is transmitted to the central PC over the telephone, (c) sampled ECG waveforms are saved in RAM memory, (d) a 12-lead ECG is analyzed.

13.6 Describe the QRS detection technique that is used most in high-performance commercial arrhythmia monitors such as in the intensive care unit.

13.7 Explain how you would approach the problem of writing software to do 12-lead ECG interpretation by computer so that it would be commercially accepted.

13.8 What are some other techniques of measuring the ST-segment level? Give any advantages or disadvantages as compared to the windowed search method.

14

VLSI in Digital Signal Processing

David J. Beebe

The hardware used to implement the digital techniques discussed previously is he main focus of this chapter. We will first discuss digital signal processors (DSPs) and the functions they must perform. Next, two commercially available DSPs are described. Current high-performance VLSI architectures for signal processing are introduced including parallel processing, bit-serial processing, systolic arrays, and wavefront arrays. A portable ECG and a digital hearing aid are used as examples of VLSI applications in medicine. Finally, the emerging integration of VLSI and biomedical sensors is briefly discussed.

14.1 DIGITAL SIGNAL PROCESSORS

Until about 25 years ago, most signal processing was performed using specialized analog processors. As digital systems became available and digital processing algorithms could be implemented, the digital processing of signals became more widespread. Initially, digital signal processing was performed on general-purpose microprocessors such as the Intel 8088. While this certainly allowed for more sophisticated signal analysis, it was quite slow and was not useful for real-time applications. A more specialized design was needed.

14.1.1 Processor requirements and elements

Digital signal processors are really just specialized microprocessors. Microprocessors are typically built to be used for a wide range of general-purpose applications. In addition, microprocessors normally run large blocks of software, such as operating systems, and usually are not used for real-time computation.

A digital signal processor, on the other hand, is designed to perform a fairly limited number of functions, but at very high speeds. The digital signal processor must be capable of performing the computations necessary to carry out the techniques described in previous chapters. These include transformation to the frequency

domain, averaging, and a variety of filtering techniques. In order to perform these operations, a typical digital signal processor would include the following elements:

1. Control processor
2. Arithmetic processor
3. Data memory
4. Timing control
5. Systems

In early digital signal processing systems, the implementation of the elements shown schematically in Figure 14.1 involved many chips or ICs (integrated circuits). Today all the elements can be realized on a single VLSI chip.

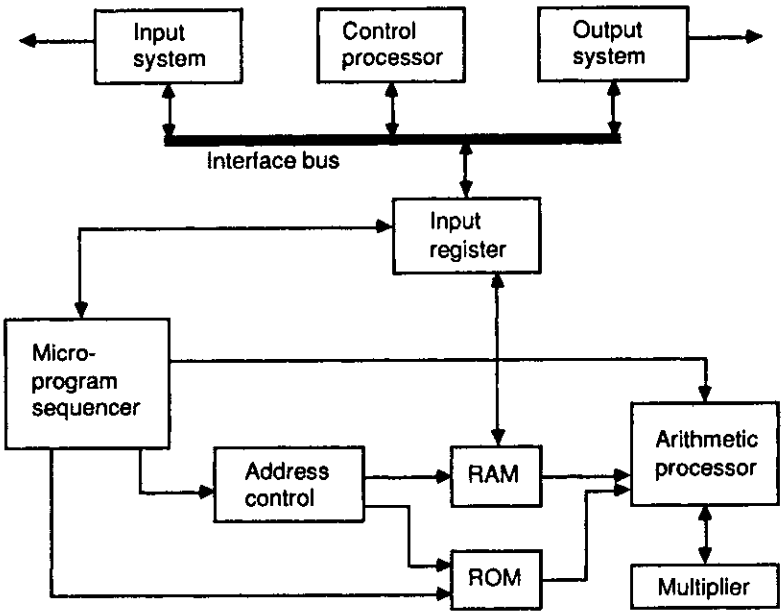

Figure 14.1 A typical digital signal processor modular design.

14.1.2 Single-chip digital signal processors

Some of the earliest attempts to incorporate the elements shown in Figure 14.1 on a single chip took place in the late 1970s. By the early 1980s, several companies including Bell Labs, Texas Instruments, NEC, and Intel had commercially available

single-chip digital signal processors. In the last few years, the use of VLSI have made DSPs easier to use and more affordable.

Comparing the performance of DSPs is not always a straightforward procedure. While MIPS (million instructions per second) or MFlops (million floating-point operations per second) are often used when comparing microprocessor speed, this is not well suited to DSPs. A common benchmark for comparing the performance of DSPs is the multiply and accumulate (MAC) time. The MAC time generally reflects the maximum rate at which instructions involving both multiplication and accumulation can be issued. More meaningful benchmarks would be computations such as FFTs and digital filters. However, comparing these is tricky, because the benchmarks are not always completely described and often do not match those of the competition (Lee, 1988). Figure 14.2 shows the MAC times for some common DSPs.

The following sections describe two DSPs. First we discuss the TMS320 family made by Texas Instruments. This has been the most widely used DSP family. Second we describe the DSP56001 by Motorola. For a rough comparison of speed, consider that a Motorola 68000 microprocessor can handle 270,000 multiplications per second, while the DSP56001 is capable of 10,000,000 multiplications per second (Mo, 1991). That is an increase in speed of 37 times.

Company	Part	Date	MAC (ns)	Bits in mult.
AT&T	DSP1	1979	800	16
Texas Inst.	TMS32010	1982	390	16
Fujitsu	MB8764	1983	100	16
NEC	μPD77220	1986	100	24
Motorola	DSP56001	1987	74	24
AT&T	DSP16A	1988	33	16
Texas Inst.	TMS320C30	1988	60	24
Motorola	DSP96001	1989	75	32

Figure 14.2 Comparison of MAC times for several popular DSPs (adapted from Lee, 1988).

TMS320

TMS320 refers to a family of microprocessors introduced by Texas Instruments in 1982 and designed for application in real-time digital signal processing. The major feature of the TMS320 is a large on-chip array multiplier. In most general-purpose microprocessors, the multiplication is done in microcode. While this provides great versatility, it makes for long MAC times. An on-chip hardware multiplier greatly reduces the MAC times. On a typical microprocessor, approximately 10 percent of the chip area performs arithmetic functions, while on the TMS320, 35 percent of the chip area is dedicated to these functions. This large computing area is

consistent with the numerically intensive nature of digital signal processing algorithms. The arithmetic and logic unit operates with 16/32-bit logic and the parallel hardware multiplication performs 16 × 16-bit two's complement multiplication in 200 ns. This high multiplication rate supports high-speed FFT computations. Programming is done in assembly language. The maximum clock rate is 20 MHz (Quarmby, 1985; Yuen *et al.*, 1989).

DSP56001

The DSP56001 was introduced in 1987 and has quickly gained widespread use in audio equipment, scientific instrumentation, and other applications. The chip is capable of 10 million multiplications, 10 million additions, 20 million data movements, and 10 million loop operations per second. To achieve these high speeds, the chip has a highly parallel architecture with a hardware array multiplier, two ALUs (arithmetic logic units), two independent on-chip memory spaces, and an on-chip program memory. Most important, the DSP56001 utilizes a parallel and pipelined architecture in which several independent units operate simultaneously (Mo, 1991). The Motorola DSP is also unique in its ability to perform a MAC in just one cycle, with the result available by the next cycle. Its features include 512 words of on-chip program RAM, 24-bit data paths providing a dynamic range of 144 dB, a data ALU, address arithmetic units and program controller operating in parallel, and a MAC time of 74 ns (Lee, 1988; Motorola, 1988).

14.2 HIGH-PERFORMANCE VLSI SIGNAL PROCESSING

Only recently has it become feasible to perform digital signal processing in real time. This is because the implementation of digital processing techniques requires high levels of computational throughput, particularly for real-time applications. This demand combined with the continually increasing levels of performance of VLSI has led to the development of VLSI digital signal processors such as the TMS320 and the DSP56001 discussed above. The trend in DSP design is toward more algorithm-based architectures. In other words, the ease with which VLSI design can be done today leads the designer to more specialized architectures.

As discussed previously, the most useful digital signal processing techniques include FFT computing, FIR and IIR digital filters. The implementation of these techniques requires only three types of operations. The required operations are storage, multiplication, and addition. The small number of operations required suggest the use of a repetitive modular architecture. This is indeed the case. The limited number of different operations required and the way in which VLSI technology is fabricated have led to several VLSI-oriented special-purpose architectures for digital signal processing applications. These architectures use multiprocessing

and parallel processing, array processors, reduced instruction set computer (RISC), and pipelining to achieve very high-speed processing rates.

The terminology used to describe and classify VLSI architectures is by no means standard. Various attempts have been made to establish a useful taxonomy, including Flynn's terminology based on instruction and data streams (Flynn, 1966). In the following discussion, we have tried to use the most commonly used terms. However, be aware that the literature is peppered with various terminology used to describe the similar architectures.

Parallel processing or multiprocessing uses multiple processors that cooperate to solve problems through concurrent execution. Pipelining is just an extension of multiprocessing that optimizes resource utilization and takes advantage of dependencies among computations (Fortes and Wah, 1987). Array processor typically refers to a two-dimensional array of processors that the data flows through. Pipelining is used to route the data through the array in the most efficient manner.

14.2.1 Parallel processing

The application of parallel processing to signal processing generally consist of two types (Yuen *et al.*, 1989). Figure 14.3 illustrates these architectures.

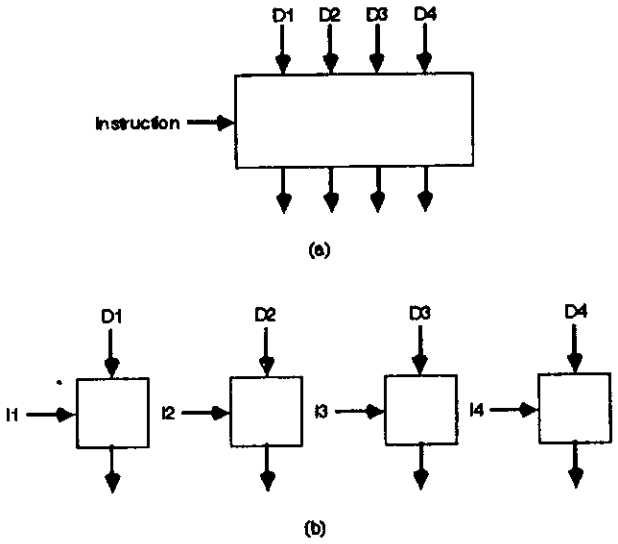

Figure 14.3 Classification of parallel processor architectures. (a) Single Instruction Multiple Data (SIMD), (b) Multiple Instruction Multiple Data (MIMD).

1. The single instruction multiple data (SIMD) operation in which the computational field consists of a number of data elements acted upon by a single operational instruction.

2. The multiple instruction multiple data (MIMD) operation in which a number of instruction streams act on multiple data elements. This method is used in image processing work.

Parallelism is achieved at the mathematical level by applying the residue number system (RNS) in architectural form (Szabo and Tanaka, 1967). In this system, the computational field is decomposed into a set of independent subfields. Computations in these subfields are performed in parallel in a SIMD-like structure. RNS provides for high-speed mathematical operations since addition and subtraction have no interdigit carries or borrows and multiplication does not require the generation of partial products.

14.2.2 Bit-serial processing

In bit-serial processors, operations are performed on only 1 bit in each word at a time. Bit-serial organization is usually applied to many words at once. Hence, the term bit-slice or bit-column arose. Figure 14.4 shows a simple bit-slice scheme.

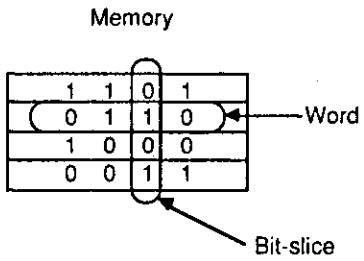

Figure 14.4 A simple bit-slice organization. The bit-slice of data is operated on, so one bit from each word is operated on simultaneously.

In bit-serial architecture, the digital signals are transmitted bit sequentially along single wires rather than simultaneously on a parallel bus (Denyer and Renshaw, 1985). This approach has several advantages over the parallel approach. First, communications within and between VLSI chips is more efficient. This is an important point given the communication-dominated operations involved in signal processing. Second, bit-serial architecture leads to an efficient pipelining structure at the bit level which leads to faster computations.

There are also several advantages to the bit-serial approach in terms of doing the actual VLSI chip layout. Bit-serial networks are easily routed on the chip since there is no need to make parallel connections to a bus. Also, since all signals enter and leave the chip via single pins, the number of input/output pins is reduced. Finally, bit-level pipelining distributes both memory and processing elements on the chip in a modular and regular fashion. This greatly facilitates ease of design, layout, and the application of silicon compilers (Yuen et al., 1989).

14.2.3 Systolic arrays

A systolic array takes a bit-serial architecture and applies pipelining principles in an array configuration. The array can be at the bit-level, at the word-level, or at both levels. The name systolic array arose from analogy with the pumped circulation of the bloodstream. In the systolic operation, the data coefficients and other information are systematically fed into the system with the results "pumped out" for additional processing. A high degree of parallelism is obtained by pipelining data through multiple processors, typically in a two-dimensional fashion. Once data enters the array, it is passed to any processor that needs it without any stores to memory. Figure 14.5 illustrates this data flow. The data flow is synchronized via a global clock and explicit timing delays (Duncan, 1990).

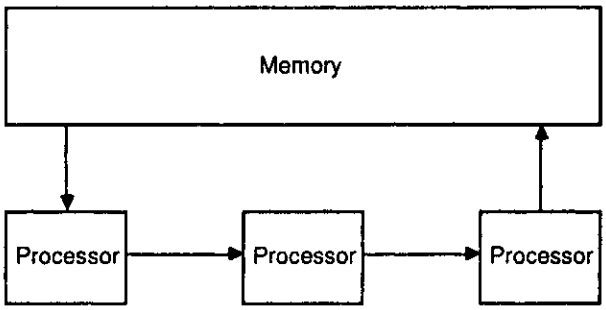

Figure 14.5 Systolic flow of data to and from memory.

Systolic arrays are particularly well suited to VLSI implementation. In order to take full advantage of the ever increasing density of VLSI, chip layouts must be simple, regular, and modular. Systolic arrays use simple processing elements and interconnection patterns that are replicated along one or two dimensions on the chip. In fact, most connections involve only nearest neighbor communication. The general architecture for a systolic array is shown in Figure 14.6. The chip resembles a grid in which each point is a processor and each line is the link between them.

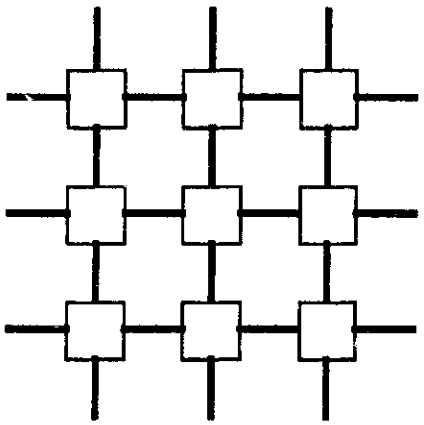

Figure 14.6 A simple systolic array configuration. Each box represents a processor or transputer and each line represents the link between the processors.

14.2.4 Wavefront arrays

One of the newest architectures is the wavefront array. Developed by Kung, the wavefront architecture is similar to the systolic array in that both are characterized by modular processors and regular, local interconnection networks. The difference is that in the wavefront array, the global clock and time delays are replaced by asynchronous handshaking. This eliminates problems of clock skew, fault tolerance, and peak power (Duncan, 1990).

The details of these architectures is beyond the scope of this text; the intent here is to give the reader an overview of the current capabilities and applications for VLSI digital signal processing (see Kung et al., 1985 and Kung, 1988 for details of VLSI architectures).

14.3 VLSI APPLICATIONS IN MEDICINE

VLSI devices are currently used in a wide variety of medical products ranging from magnetic resonance imaging systems to conventional electrocardiographs to Holter monitors to instruments for analyzing blood.

14.3.1 Portable ECG

The portable ECG machine was made possible largely due to progress in VLSI design. Today's portable ECG machines, such as the Elite (Siemens Burdick, Inc.), are fully functional 12-lead ECG machines equal in every respect to larger machines with the exception of paper size and mass storage capacity. A quick comparison points out the dramatic results that can be obtained with VLSI. The portable ECG is 15 times smaller, 10 times lighter, and half as costly as the full-size machine. It consumes 90 percent less power and uses 60 percent fewer parts (Einspruch and Gold, 1989).

14.3.2 Digital hearing aid

A traditional analog hearing aid consists of the parallel connection of bandpass filters. To provide accurate compensation, a large number of filters are needed. Size, complexity, and cost all limit the number of filters. A hearing aid with a large number of adjustable components is also difficult to fit and adjust to the needs of each individual patient.

With the advent of general-purpose DSPs and application-specific integrated circuits (ASIC), it has become feasible to implement a hearing aid using digital technology. The digital implementation utilizes many of the ideas discussed previously including A/D converters, D/A converters, and digital filtering. The biggest advantage of the digital design is that the transfer function of the filter used to compensate for hearing loss is independent of the hardware. The compensation is performed in software and thus is extremely flexible and can be tailored to the individual. The digital hearing aid is more reliable and the fitting process can be totally automated (Mo, 1988).

14.4 VLSI SENSORS FOR BIOMEDICAL SIGNALS

A recent spinoff from standard microelectronics technology has been solid-state sensors and smart sensors. Integrated circuit processing techniques have been used for some time to fabricate a variety of sensing devices. Currently under development is a new generation of smart or VLSI sensors. These sensors combine the sensing element with signal processing circuitry on a single substrate. Figure 14.7 shows the elements of a generic VLSI sensor.

Integrated solid-state sensors or smart sensors contain four parts: (1) custom thin films for transduction, structural support or isolation, or encapsulation; (2) microstructures formed using micromachining techniques; (3) interface circuitry; and (4) signal processors (Einspruch and Gold, 1989). A sensor built by Najafi and Wise to measure the extracellular potentials generated within neurons illustrates the possibilities of combining VLSI with current sensor technologies.

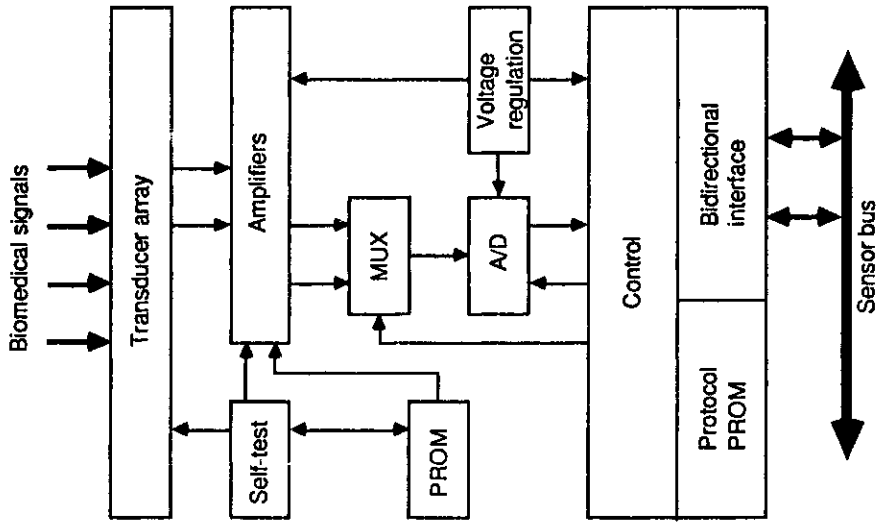

Figure 14.7 Block diagram of a generic VLSI sensor. The sensor is addressable, self-testing and provides a standard digital output (adapted from Einspruch and Gold, 1989).

Figure 14.8 shows the overall structure of the sensor. The electronics on the sensor includes functional circuits such as amplifiers, an analog multiplexer, a shift register, and a clock. Indeed most of the sampling functions discussed in Chapter 3 are integrated right onto the sensor. The sensor is 3–4 mm long, 200 μm wide and 30 μm thick.

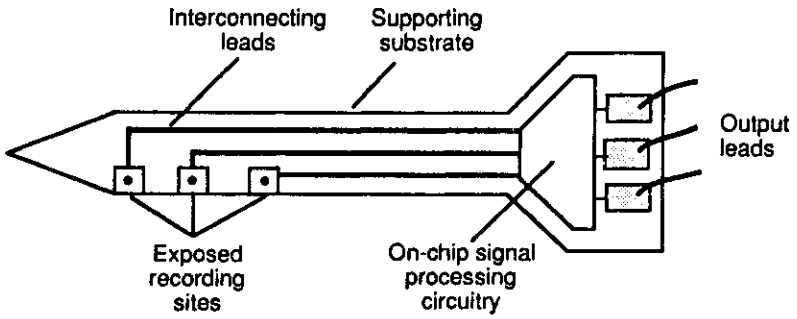

Figure 14.8 The overall structure of the smart sensor built by Najafi and Wise (adapted from Einspruch and Gold, 1989).

14.5 VLSI TOOLS

The most commonly used VLSI design tools in the academic environment are the Berkeley VLSI tools. The system allows the designer to lay out the design at the device level and simulate the design for correct operation. A typical design would begin with a functional block diagram. One next moves progressively lower and lower toward the actual device layout. The device level layout is done in MAGIC, a VLSI layout editor. MAGIC is analogous to a mechanical CAD program. The designer actually draws each and every transistor in the design just as it will be fabricated. Obviously this can be a painstaking task since a typical chip may contain several hundred thousand transistors. MAGIC contains many features that simplify the design and often portions of the layout are redundant. As portions of the design are completed, programs such as CRYSTAL (a VLSI timing analyzer) and ESIM (an event-driven switch level simulator) are used to test the logic and timing operation of that portion of the design. This is an important point! It is important to thoroughly test each portion of the layout as it is completed rather than waiting until the whole design is done. Continual testing will help ensure that the design will be functional when all the pieces are connected together in the final layout.

Once the entire layout is complete and it has been successfully simulated, it is fabricated. The MOSIS (MOS implementation system) fabrication facility at the University of Southern California is often used for low-volume experimental work. Finally, the device is tested and revised as necessary.

14.6 CHOICE OF CUSTOM, ASIC, OR OFF-THE-SHELF COMPONENTS

When approaching an instrument design, one must decide between simply using off-the-shelf chips, ASIC (application specific integrated circuits), or fully custom chips. In reality, the finished design will usually contain some combination of these approaches. Ten years ago, using off-the-shelf chips was the only option. However, in recent years it has become feasible for even small companies to design their own ASIC or fully custom chips via tools similar to those discussed above. The design choices are based on the needs of the particular project and the availability of suitable off-the-shelf chips. The design time and cost generally increase as one moves from a design containing only off-the-shelf chips to a fully custom design.

14.7 REFERENCES

Denyer, P. and Renshaw, D. 1985. *VLSI Signal Processing: A Bit-serial Approach.* Reading, MA: Addison-Wesley.

Duncan, R. 1990. A survey of parallel computer architectures. *Computer*, **23**(2): 5–16.
Einspruch, N. G. and Gold, R. D. 1989. *VLSI in Medicine*. San Diego: Academic Press.
Flynn, M. J. 1966. Very high speed computing systems. *Proc. IEEE*, **54**: 1901–09.
Fortes, J. A. B. and Wah, B. W. 1987. Systolic arrays—from concept to implementation. *Computer*, **20**(7) :12–17.
Kung, S. Y. 1988. *VLSI Array Processors*. Englewood Cliffs, NJ: Prentice Hall.
Kung, S. Y., Whitehouse, H. J., and Kailath, T. 1985. *VLSI and Modern Signal Processing*. Englewood Cliffs, NJ: Prentice Hall.
Lee, E. A. 1988. Programmable DSP architectures: Part 1. *IEEE ASSP Magazine*. 4–19.
Mo, F. 1991. Basic digital signal processing concepts for medical applications. *Biomedical Science and Technology*. **1**(1): 12–18.
Motorola. 1988. Technical Data #DSP56001/D, Motorola Literature Distribution, P.O. Box 20912, Phoenix, Arizona, 85036.
Quarmby, D. (ed.) 1985. *Signal Processor Chips*. Englewood Cliffs, NJ: Prentice Hall.
Szabo, N. S. and Tanaka, R. I. 1967. *Residue Arithmetic and Its Applications to Computer Technology*. New York: McGraw-Hill.
Yuen, C. K., Beauchamp, K.G., and Fraser, D. 1989. *Microprocessor Systems in Signal Processing*. San Diego: Academic Press.

14.8 STUDY QUESTIONS

14.1 Describe the difference between general-purpose microprocessors and DSPs.
14.2 Why are MAC times used as benchmarks for DSPs instead of the usual MIPS or MFlops?
14.3 Discuss why VLSI is well suited to the design of DSPs.
14.4 Define the following terms: (1) parallel processing, (2) pipelining, (3) array processors, (4) SIMD, (5) MIMD.
14.5 Describe the difference between systolic arrays and wavefront arrays. Which would operate more efficiently?
14.6 List at least four advantages of using VLSI-based designs for medical applications.