HUGH M. SOUTH

HIGH-SPEED PROCESSORS FOR SONAR

The Applied Physics Laboratory and its subcontractors have been leaders in the development of programmable sonar signal processors for 19 years. Several novel architectures for both hardware and software have resulted, giving APL the most powerful sonar signal-processing facilities in the United States. This article traces the historical evolution of sonar-processing hardware, with emphasis on the development of high-speed sonar processors in the Strategic Systems Department.

INTRODUCTION

This article traces the development of sonar signal processors from the primitive systems that relied only on the human ear to the programmable, high-speed multiprocessor systems now in use. To set a framework for the discussion of sonar processors, I begin with a description of sonar systems and the environment in which they operate. Next I describe passive sonars and the particular problems of detecting signals by passive systems, and outline the development of modern, programmable sonar signal processors. Since the 1970s, staff members of APL's Strategic Systems Department, along with many subcontractors, have made significant contributions to the use of distributed processing systems for sonar, and I detail the development of such processors for APL programs.

A sonar is any system that uses acoustic means to detect, localize, track, or classify objects. In this article the sonar is assumed to operate underwater, although that need not be the case. Sonars may be of two types: active and passive. Active sonars transmit pulses of acoustic energy and process the resulting echoes. They are used for navigation, underwater survey, and localization of objects such as fish, sunken ships, or mines. Passive sonars detect and classify objects by analyzing acoustic energy radiated by natural and man-made sources. Natural sources are wind and marine life; man-made sources are typically ships. I focus below on processors for passive sonars.

The earliest mention of an operational sonar was a passive system described by Leonardo da Vinci in the fifteenth century: "If you cause your ship to stop, and place the head of a long tube in the water and place the outer extremity to your ear, you will hear ships at a great distance from you."[1]

Sonar remained a curiosity that changed little from Leonardo's observation over the next 500 years. At the outbreak of the First World War, for example, sonars used for hunting submarines were much like Leonardo's tube, except that one tube for each ear was used. The need to protect shipping from submarines, coupled with the invention of the vacuum tube, finally led to significant advances in sonar processors and the introduction of processors other than the human ear.

PASSIVE SONARS

Understanding sonar processors requires some understanding of the passive sonar problem. Table 1 gives some of the physical properties of sonar signals and illustrates the difference in physical parameters between radar and sonar.

Even though radar and sonar have quite different propagation speeds and frequency ranges, the wavelength regimes overlap, and therefore radar and sonar sensor arrays can have comparable size.

Sonar signals are carried by sound waves propagating in the ocean. A plane wave propagating in a three-dimensional coordinate system $\mathbf{x} = (x,y,z)$ may be described by the standard relation

$$p(\mathbf{x},t) = \exp[j(\omega_0 t - \mathbf{k}_0 \cdot \mathbf{x})] \ , \qquad (1)$$

where t denotes time, ω_0 denotes the radian frequency, and $\mathbf{k}_0$ denotes the vector wave number, that is, the wave number in the direction of propagation. In sonar, we customarily write Equation 1 in the following form:

$$p(\mathbf{x},t) = \exp[j\omega_0 (t - \boldsymbol{\alpha}_0 \cdot \mathbf{x})] \ . \qquad (2)$$

Since the speed of propagation equals $1/|\boldsymbol{\alpha}_0|$, $\boldsymbol{\alpha}_0$ is called the slowness vector.

Taking the Fourier transform of the plane wave gives its four-dimensional wave number–frequency spectrum

Table 1. Radar and sonar signals.

Property	Sonar	Radar
Speed of propagation	1.5×10^3 m/s	3×10^8 m/s
Frequency range	0.1 to 10 kHz	300 to 30,000 MHz
Wavelength	0.15 to 15 m	0.01 to 1 m
Pulse duration	0.01 to 1 s	0.1 to 10 μs

$P(\mathbf{k},\omega)$. Each point in $(\mathbf{k},\omega)$ space corresponds to a plane wave propagating in a particular direction, given by $\mathbf{k}$, at a particular frequency, given by ω. Except in shallow water, propagation of sound in the ocean is nondispersive. The loci of points in $(\mathbf{k},\omega)$ space corresponding to waves propagating with the same speed c form a cone, as shown in Figure 1, based on the approach of Dudgeon and Mersereau.[2] To simplify the figure, only two wave number axes are shown. (Although sound speed in the ocean depends on temperature, depth, and salinity, this dependence is not important for the present discussion.)

In general, the sonar signal plus noise, $s(\mathbf{x},t)$, can be expressed in terms of plane waves by using the four-dimensional Fourier transform

$$s(\mathbf{x},t) = \frac{1}{(2\pi)^4} \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} S(\mathbf{k},\omega) \times \exp[j(\omega t - \mathbf{k}\cdot\mathbf{x})] \, d\mathbf{k} \, d\omega ,$$

where $S(\mathbf{k},\omega)$ is the wave number–frequency spectrum of the signal. In passive sonar, the goal is to separate the signal components by frequency and by direction of propagation. The signal processor must, therefore, realize a multidimensional filter designed to pass desired signal components and to reject noise or other components not of interest. In general, the multidimensional filter is designed to be both linear and shift invariant and to have some desired wave number–frequency spectrum $H(\mathbf{k},\omega)$. The output of the sonar processor, $F(\mathbf{k},\omega)$, then has the form

$$F(\mathbf{k},\omega) = H(\mathbf{k},\omega)S(\mathbf{k},\omega) ;$$

that is, the processor must implement a filter with passband near unity in the desired regions of $(\mathbf{k},\omega)$ space and near zero everywhere else.

The ideal multidimensional filter needed for passive sonar operates in both space and time. In practical systems, spatial and temporal processing are usually performed separately to reduce the number of computations required. The separation of temporal and spatial processing is not optimum in general, but it is nearly so for many operational cases.[3,4] Practical sonar processors are more restricted than the general multidimensional filter because the signal $s(\mathbf{x},t)$ is known only at the sensor positions rather than for all space, and the filtered sonar output $F(\mathbf{k},\omega)$ is needed only at the observing platform. Still, the multidimensional filter with separate spatial and temporal processors describes most sonar signal processors. Figure 2 shows the relationship between the spatial and temporal processors and the other components necessary to make a complete passive sonar. A sonar array typically contains between 25 and 100 sensors and has an output data rate between 2 and 6 MB/s.

Temporal processing algorithms used in passive sonars are common to many fields: digital filters, integration, fast Fourier transform, and correlation, for example. The most commonly used spatial processor is a beam former. In $(\mathbf{k},\omega)$ space, a beam former is a filter with a passband centered along the intersection of a cone corresponding to waves of constant speed and a plane corresponding to waves traveling in a particular direction, as shown in Figure 3. The most common type is the delay-and-sum beam former, in which the output of each

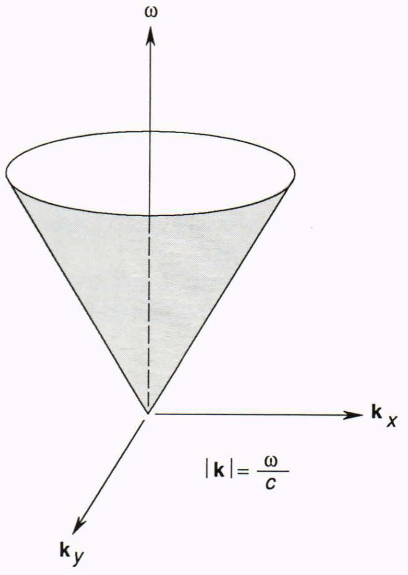


Figure 1. Loci of points in $(\mathbf{k},\omega)$ space corresponding to signals with a common propagation speed. (Only two wave number axes are shown.)

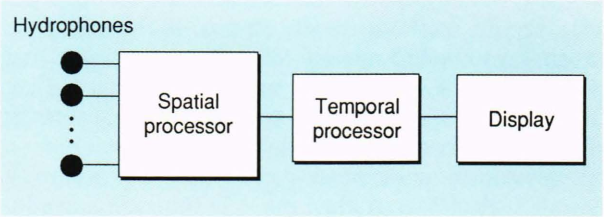


Figure 2. Functional components of a typical passive sonar.

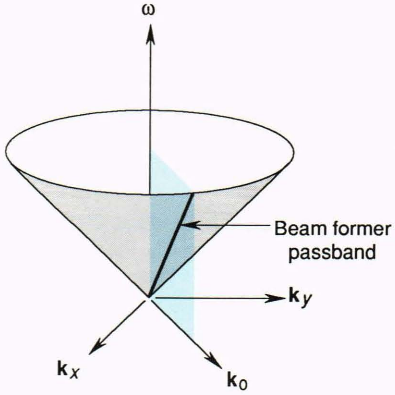


Figure 3. The $(\mathbf{k},\omega)$ space diagram of the passband of an ideal beam former.

sensor is multiplied by a weighting coefficient, delayed, and then summed with the other sensors, as shown in Figure 4. The weighting coefficients determine the shape of the passband in $(\mathbf{k},\omega)$ space, and the delays are used to center the passband along a particular direction. If it is necessary to detect signals from multiple directions, multiple sets of delays are used.

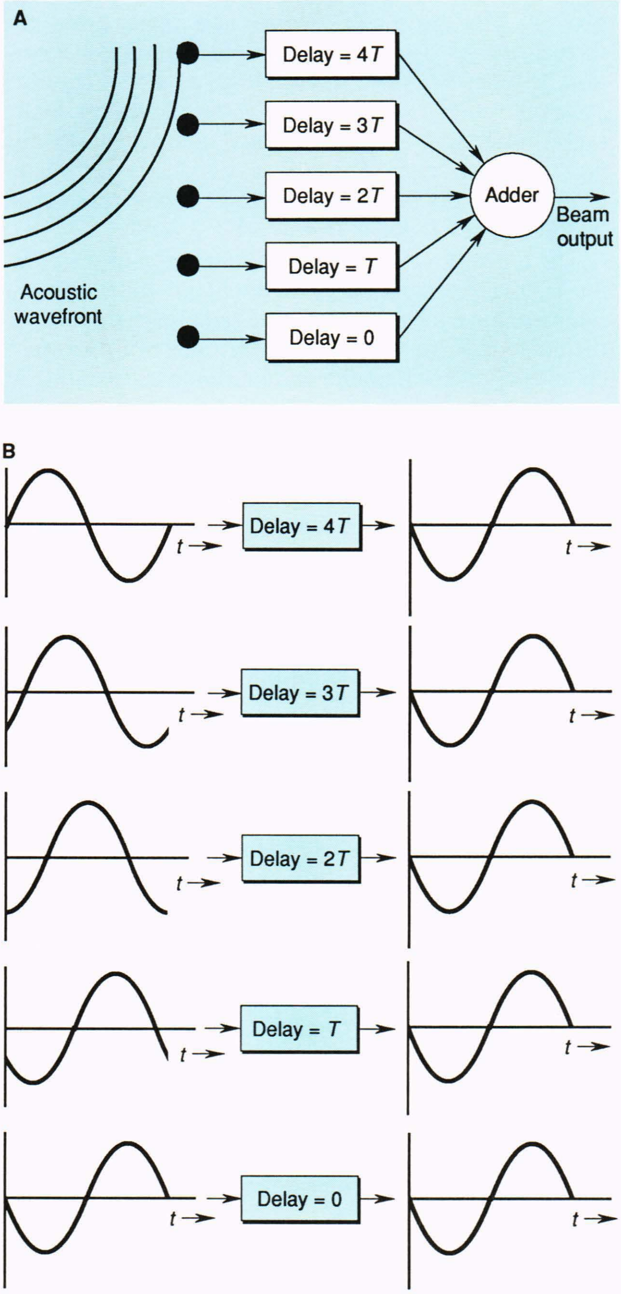


Figure 4. Delay-and-sum beam forming for a line array of hydrophones. **A.** The arrival of an acoustic wave and the manner in which the output of each hydrophone is delayed and then summed. **B**. With proper delays, the outputs of each hydrophone may be added in phase to provide maximum signal strength. (*T* denotes the time required for sound to travel between a pair of hydrophones and depends on the arrival of the acoustic wavefront; *t* denotes time axis.)

HARDWARE IMPLEMENTATIONS

I now present specific hardware structures for sonar signal processing, progressing from general considerations to consideration of the first programmable sonar signal processor and the extensive development of sonar signal processors at APL.

The nature of the passive sonar problem suggests an architecture for a sonar processor. Spatial and temporal processing can be separated. In addition, processing for individual sensors and arrival directions can be separated. This separation suggests a distributed architecture: individual processors linked by a high-speed bus with some overall synchronization scheme. In such an architecture, the requirement to achieve real-time or faster-than-real-time processing rates can be accomplished by using multiple processors working on different data segments or different problems. Since the various sonar processes may execute at different rates, the individual processors must also have access to memory buffers to smooth the flow of results between units. A distributed architecture naturally provides modularity, which eases system integration and allows new processors to be added as requirements change. Finally, any architecture must support the programmability of algorithms and algorithm parameters. What follows is a discussion of the evolution of sonar signal processors that meet these requirements.

INITIAL PROGRAMMABLE PROCESSOR

The first programmable sonar signal processor with a distributed architecture appeared in 1974. Called the Sperry Programmable Acoustic Receiver (SPAR),[5] the system contained two processing units and an address processor, which also served as the system executive (Fig. 5). The SPAR accepted only three analog channels as input through the analog/digital unit. As a result, no significant spatial processing ability was needed. The two temporal processors, the digital filter and matched filter units, were microprogrammable custom units, and the PDP-8 minicomputer provided the user interface for setup and control.

The most unusual feature of this early distributed sonar system was the address processor, which maintained its own program counter and the program counters for each temporal processor on the bus. The address processor also controlled the memory accesses for the other units, allowing it to synchronize the entire system. The processor contained many features, such as a hardware arbitration network for memory requests, which supported high-speed processing. Although the SPAR was successfully implemented, the feasibility of using such centralized control in larger-scale machines remains questionable.

HARDWARE DEVELOPMENT AT APL

Five years after the SPAR appeared, APL and the Autonetics Marine Systems Division of Rockwell International built the first of two similar SPAN (Sonar Program Analyzer) processors. Both systems were built to support the SSBN Sonar and Command and Control System Evaluation Program in APL's Strategic Systems Department. This program has four principal objectives:

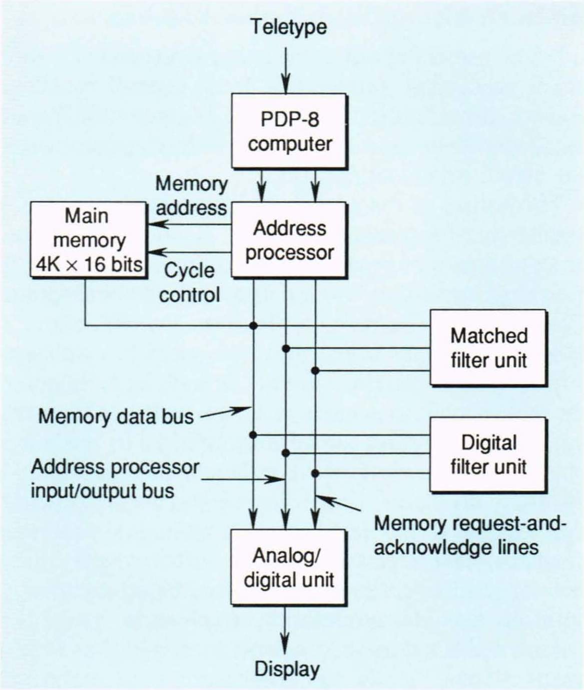


Figure 5. Block diagram of the Sperry Programmable Acoustic Receiver.

1. To assess the survivability of the submarine-launched ballistic missile deterrent.

2. To assess the effectiveness of submarine sonar systems.

3. To establish the potential for improvements, both to submarine sonar equipment and to operating guidelines for that equipment.

4. To determine sonar performance norms useful in detecting equipment degradation.

To achieve these objectives, the outputs of entire hydrophone arrays are recorded throughout a deterrent patrol, and the resulting recordings are returned to APL for processing and analysis. Recording of complete sensor arrays permits application of the most sophisticated signal-processing techniques, independent of any processing by submarine equipment. Timely evaluation of the resulting large amount of data requires a signal analyzer that can process the data in less time than needed to record them. Existing signal-processing systems could not meet that requirement, necessitating the development of new systems.

The first new processor, SPAN-A, was an order of magnitude more powerful than the SPAR. Since SPAN-A has already been the subject of an article in the *Technical Digest*,[6] it will only be described briefly for comparison to subsequent processors. As originally delivered, SPAN-A consisted of five processing units and four memory units interconnected by a unidirectional ring bus, an architecture similar to that described below for SPAN-I. Both the array processors and the memory units were commercial equipment. During the years since delivery, SPAN-A has been upgraded by the addition of a reconfigurable signal processor and a long-array beam former (LABF). SPAN-A is also the host of the memory-linked wavefront array processor.[7]

The ring bus used to interconnect the units remains the most unusual feature of SPAN-A. This bus provided a peak throughput of 50 MB/s, and used a communication protocol similar to that for SPAN-I. In SPAN-A, however, a single unit could send information to only one other unit at a given time. As will be seen, this restriction was relaxed in SPAN-I.

In 1983 the successor to SPAN-A, called SPAN-I, appeared. This processor, which had three times the throughput of SPAN-A, also resulted from a collaboration between Rockwell International and APL. Figure 6 shows the overall signal flow in SPAN-I and relates SPAN-I signal processing to the functional components of the typical passive sonar. The greater complexity of the temporal processing is typical in passive sonar. SPAN-I can handle a peak input data rate of 5 MB/s.

The hardware architecture of SPAN-I is shown in Figure 7. Of all the components in the figure, the three most significant are the reconfigurable signal processor (RSP), the LABF, and the SPAN-I bus itself. On SPAN-I the interconnecting bus is a unidirectional ring bus operating at a sustained transfer rate of 64 MB/s; the direction of the data flow is indicated by the direction of the arrows in the figure. This bus, called a Pierce ring,[8] was chosen because it supports simultaneous data transfers between units without interference. In the Pierce ring, all communication, whether software load, control, or data, occurs by the transfer of messages between adjacent units. Table 2 gives the structure of the bus messages. The data typically consist of four 8-bit integers, but 16- and 32-bit floating-point formats are also used.

At initialization the bus is filled with blank messages (messages with message type equal to zero). Data to be transmitted between units are first formatted into messages. Then each time a blank message is received it is replaced with a message containing the data to be transmitted and the identification of the destination unit. Every unit on the bus has a unique position code. The interface hardware common to each unit receives bus messages and compares the destination unit identification of the message with its position code. If the two codes do not match, the message is simply passed on to the next unit. If the codes do match, the message is accepted into input memory, and a blank message is substituted. The output data rate of each unit is less than

Table 2. Structure of SPAN-I bus messages.

Item	Size (bits)
Sending unit identification	6
Destination unit identification	6
Address at destination	16
Message type	4
Data	32
Total	64

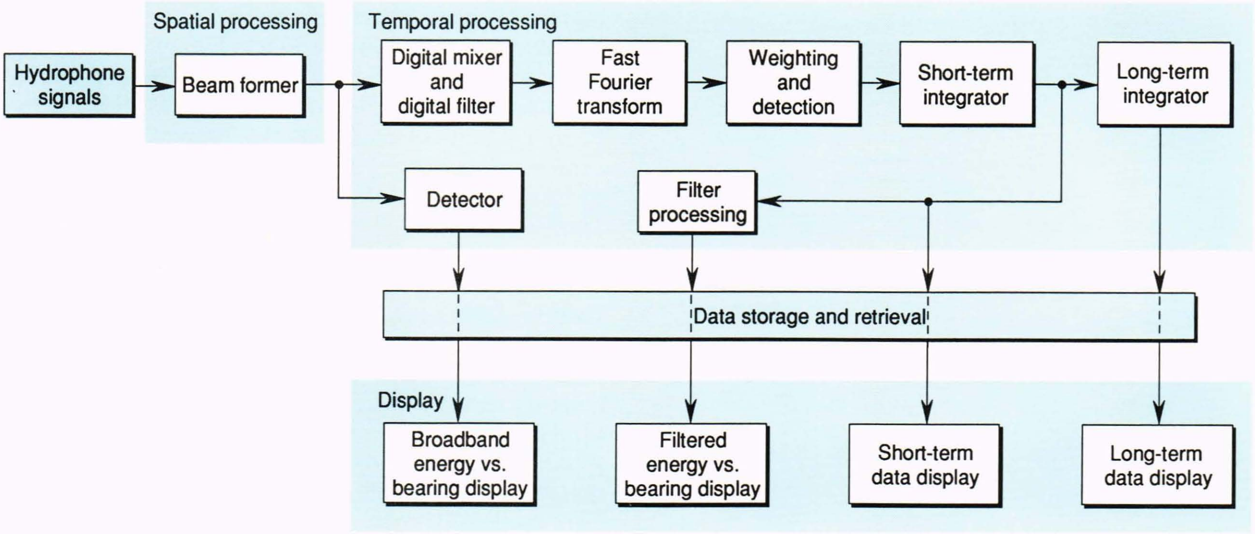


Figure 6. Simplified SPAN-I signal flow and functional components.

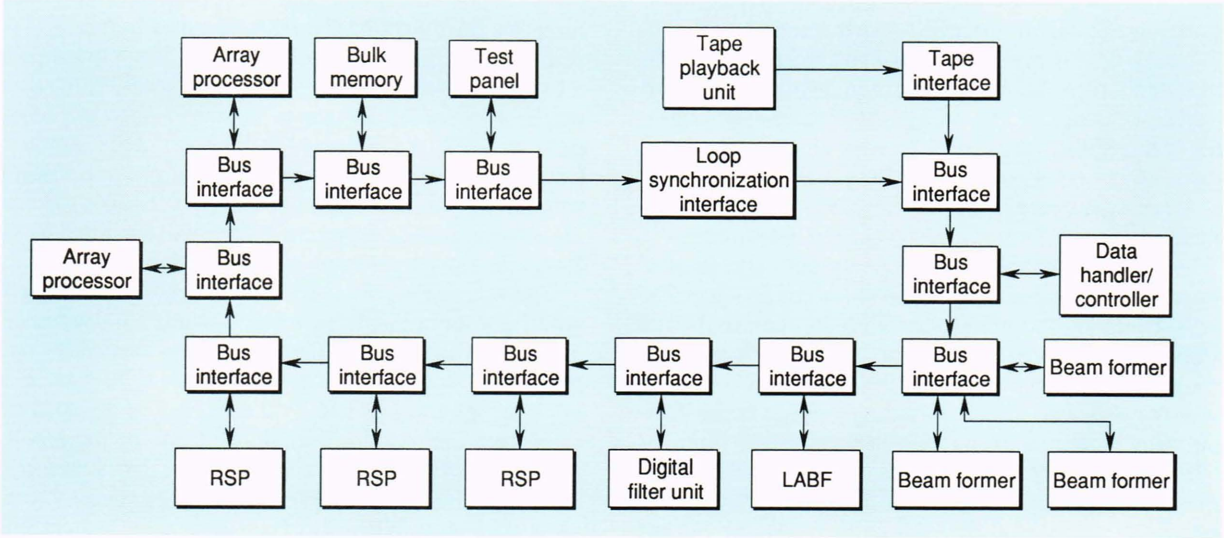


Figure 7. Block diagram of SPAN-I. (Direction of data flow is indicated by arrows.)

the bus rate; therefore, each unit always passes along some blank messages, ensuring that units may always place data onto the bus. Data transfer operations can occur smoothly as long as the peak load at any single point does not exceed the bandwidth of the bus. In the Pierce ring, the total bus capacity is usually much greater than the peak rate because most messages travel only a short distance before removal from the bus. This method of communication by messages is the essence of the Pierce ring, and SPAN-I is the largest system to use a Pierce ring as its main bus.

In addition to a position code, each unit on the bus may have an alternate identification code called a "nickname." The ability to use multiple identifications for a single unit of hardware allows identical data to be sent to multiple units without separate transmission. This "party line" input is especially useful for sending a single set of outputs from a spatial filter to multiple, different temporal filters. One disadvantage of the SPAN-I bus is its overhead. As Table 2 indicates, only one-half of each message is data.

Each unit is connected to the ring bus through a common interface, as shown in Figure 8. The bus-interface input module scans the destination code of each message on the bus and buffers selected messages in a first-in, first-out memory. Data in these selected messages are sent to the connected processing unit. The bus-interface output module performs three functions: passing a message to the next unit, stripping a message and substituting a blank message, or stripping a message and transmitting an output message from the attached processing unit. The bus-interface output module also contains a first-in, first-out memory for buffering data to be sent to the ring bus.

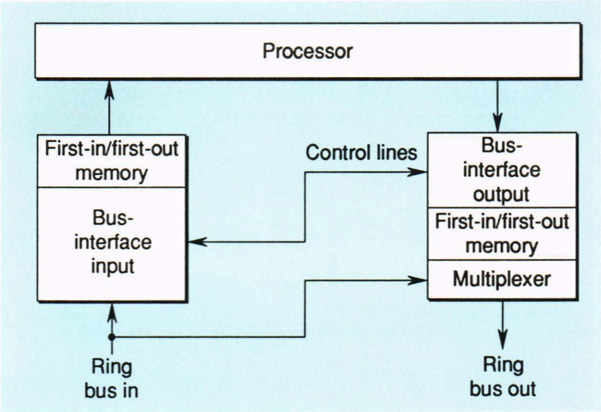


Figure 8. Block diagram of the SPAN-I bus interface.

The LABF, which provides the spatial processor in SPAN-I, is a hardware implementation of the time-domain, delay-and-sum beam forming algorithm described above. Specifically, the LABF is a single drawer containing 16 delay-and-sum processors, each accepting 256 sensor inputs; it can provide up to 8192 weighting coefficients and up to 32,768 delays. Aggregate throughput of all 16 elements is 256 million delay-and-sum operations per second. An overall LABF block diagram is shown in Figure 9. Because each delay-and-sum processor operates independently, 16 different sensor arrays with different sampling rates may be processed simultaneously.

The memories internal to the delay-and-sum processors may be partitioned through software to allow the user to trade off the number of sensors, the number of beams, and the maximum delay. Thus, the LABF may be configured nearly optimally for most sensor arrays. The processors can also interpolate between input samples using either a finite impulse response filter or Lagrangian interpolation. The LABF uses fixed-point arithmetic and may accept 8- or 16-bit, two's-complement input samples and produce 8- or 16-bit output beams. Weighting coefficients are also 16-bit two's-complement values. Also shown in Figure 9 are two digital/analog converters to allow monitoring of the input or output data streams. Overall LABF control is provided by a Motorola 68000 microprocessor.

Most of the temporal processing on SPAN-I is provided by the three reconfigurable signal processors (RSP's). Their design grew out of the observation that digital signal processors could be built out of modular digital filter and fast Fourier transform (FFT) sections.[9] An adaptive filter section, as shown in Figure 10, was added, however. All processing functions are implemented using fixed-point arithmetic to maximize speed and minimize hardware. All signal processors represent a tradeoff between programmability and speed. In the RSP, speed is obtained by using dedicated hardware modules that are parametrically programmable; that is, a digital filter module can only be used to implement digital filters, but many different filters can be generated by passing parameters to the unit.

The RSP processing functions are divided into three parts: digital filter, FFT, and postprocessor. The digital filter is a hardware implementation of a two-pole, two-zero section, as shown in Figure 11. The RSP performs a two-pole, two-zero filter in 200 ns, and longer filters can be synthesized by using the scratch memory to cascade sections. As indicated in Figure 11, the digital filter section may also be programmed to provide absolute-value detection and complex mixing functions. Placing the absolute-value function before the filter section allows the RSP to perform envelope detection.

The FFT section of the RSP uses a complex butterfly with fixed-point arithmetic to implement a decimation-in-time FFT algorithm. The RSP always computes complex FFT's. A real FFT of size N points is computed using a complex FFT of size $N/2$, followed by an extra pass to correct the real and imaginary parts of the result.[10] The FFT coefficients, so-called twiddle factors, are quantized to 16 bits, and the products are rounded to 16 bits after each multiplication. Inputs to each butterfly are

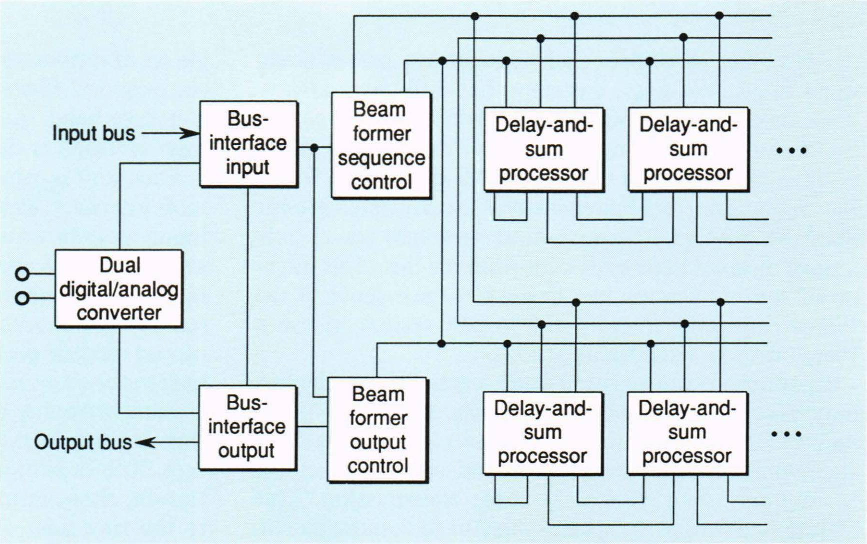


Figure 9. Hardware architecture of the LABF. (For simplicity only 4 of the 16 delay-and-sum processors are shown.)

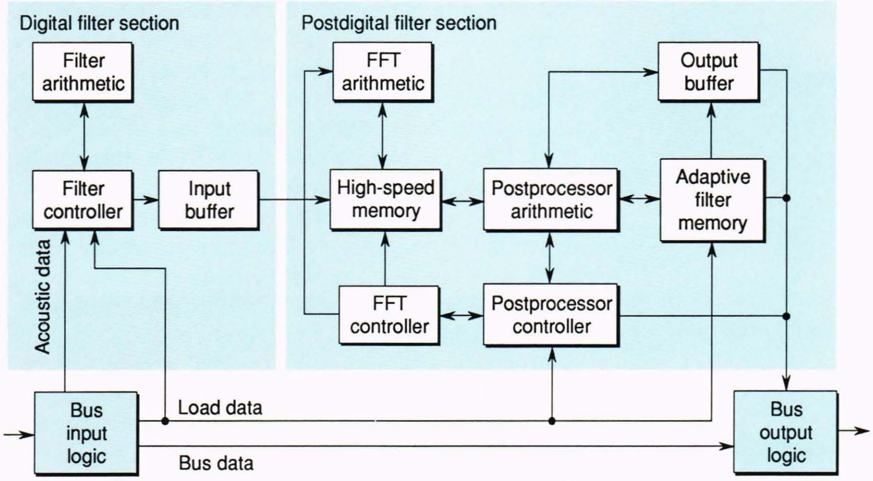


Figure 10. Architecture of the RSP.

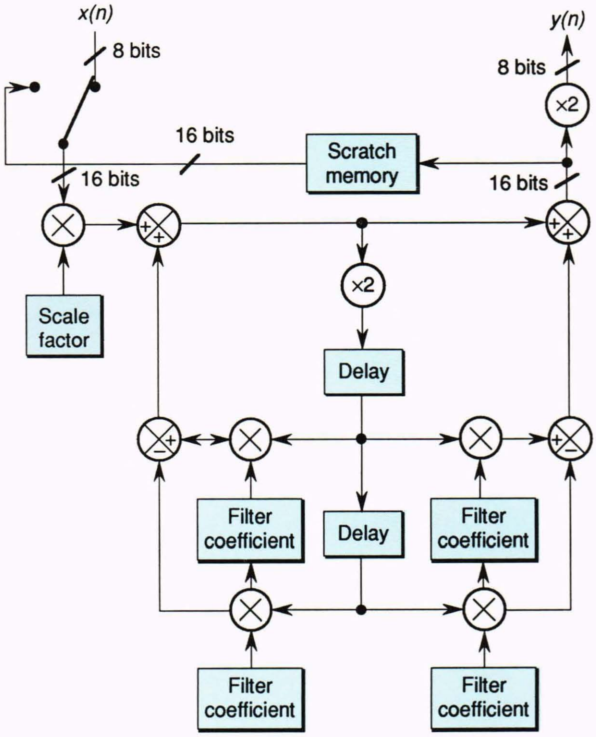


Figure 11. Digital filter section of the RSP. (For clarity, some internal scaling operations are not shown.)

multiplied by one-half and truncated to 16 bits to prevent overflow.

The FFT size is programmable from 256 to 2048 complex points. Each complex butterfly takes 400 ns; that is, a 4096-point real FFT, which requires 13,312 complex butterflies, executes in 5.2 ms. The theoretical peak computation rate of the RSP is approximately 92 million fixed-point arithmetic operations per second. In practice, computation rates of 95% of the theoretical maximum have been achieved.

The RSP's represent a successful design that is still in use. The units achieve both high throughput and efficiency, but these features are gained by sacrificing flexibility. As powerful as the RSP's are, the units implement only specific algorithms, and the order of the algorithms is determined by the hardware.

In its control structures, SPAN-I is a data-flow machine. Each processor on the ring bus maintains a queue of signal-processing jobs to be executed. In the simplest type of control, software running in one RSP counts the number of samples input to the digital filter section. When a prespecified number has been received, the RSP sends a message on the bus instructing the other processors to execute the next job in the queue.

From a manufacturing viewpoint, SPAN-I electronics are relatively simple. All electronics were built from off-the-shelf transistor–transistor logic (TTL) or TTL-compatible components. Components used were expected to operate only over the commercial temperature range. Modules were either wire-wrapped or double-sided printed circuit boards approximately 8 × 10 in. in size. Interconnections between modules required no special techniques other than twisted pairs. In all, SPAN-I contains 36 different custom circuit boards plus the boards found in the commercial equipment. Both SPAN processors are shown in Figure 12.

The success of SPAN-I and its precursor SPAN-A led to requests that APL develop an improved sonar signal processor called TSPAN (Trident Sonar Program Analyzer). The new processor had to be able to process sonar arrays up to 1000 elements and handle input data rates of up to 30 MB/s, six times the input rate of SPAN-I. Also, the increasing complexity of algorithms for spatial and temporal processing required a processor using 32-bit floating-point arithmetic rather than the fixed-point arithmetic used in SPAN-I. Minimum required throughput was 3.5 billion floating-point operations per second, with even higher throughput as a design goal. Early work to meet this goal led to the design of the memory-linked wavefront array processor.[7] Competing with the need for high throughput and data handling were three other requirements: flexibility in implementing algorithms in any order, minimization of development cost and development time by using commercial

Figure 12. Partial view of current SPAN laboratory processing equipment; SPAN-I (left), SPAN-A (right).

equipment, and improvement in maintainability by using few special-purpose modules.

On the basis of previous experience, a distributed architecture with a centralized, high-speed interconnecting bus seemed the best approach. The final TSPAN design, being built by the Submarine Signal Division of Raytheon Company, is shown in Figure 13. As can be seen from the figure, commercial processors make up the majority of TSPAN units. The TSPAN bus consists of two Aptec IOC-200 input/output computers, which provide an aggregate bus bandwidth of 400 MB/s. Each IOC-200 contains 8 MB of memory to buffer data on transfers between processors. The computational ability of TSPAN is about evenly divided between four floating-point units, which perform both spatial and temporal processing, and 32 XL70 array processors (manufactured by CSPI), which perform additional temporal processing. In all, TSPAN contains about 100 processors, each of which can be programmed independently. The peak computation rate of TSPAN is 4.8 billion floating-point operations per second, or about 60 times the peak computation rate of the Cray 1-S.

Processors are connected to the Aptec bus in one of two ways. The tape interface unit, special processing unit, and floating-point units are connected via high-speed interface modules made by Aptec Computer. These modules have a peak transfer rate of 40 MB/s on 32-bit transfers and a Motorola 68020 to manage the transfers, as well as an independent direct-memory-access controller. The manager software keeps track of logical buffers in the Aptec memory, initiating processing when a buffer is full. Since TSPAN has many processors, the manager software tracks buffers that may be filled from multiple sources.

All processing units connected using high-speed interface modules are designed for high-input/output-rate processing such as beam forming and digital filtering. All these processing units have a similar architecture, shown in Figure 14. The individual floating-point processors can perform 40 million floating-point operations per second, and there are 16 processors per floating-point unit. In addition to the data buses shown in the figure, each floating-point unit also contains a standard commercial bus for system start-up and control.

Processing that requires more complexity but is less time-critical is performed in the 32 XL70 array processors, each of which can perform 70 million floating-point operations per second. To reduce cost, the array processors are grouped in units of eight, attached to the bus via a 12.5-MB/s interface called an openbus input/output processor. The method of attachment is shown in Figure 15. Each array processor can have a second bus interface added if future needs warrant.

Real-time control of TSPAN is by data flow, as was that for SPAN-I. The data flow is managed by the high-speed interface manager software discussed previously. System start-up and monitoring are provided through the Digital Equipment Corporation MicroVAX II com-

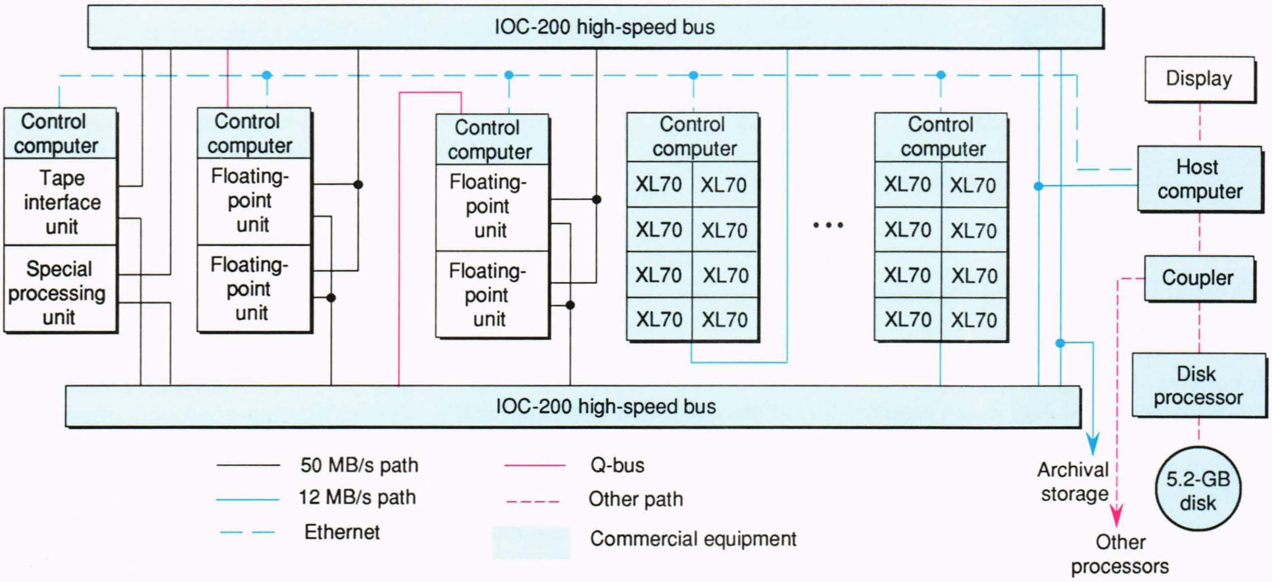


Figure 13. Architecture of the TSPAN signal processor. (Ellipsis points represent two additional banks of XL70 array processors not shown.)

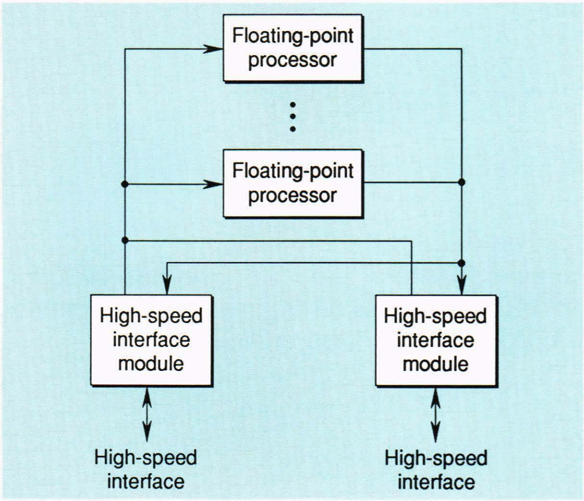


Figure 14. Hardware diagram of the floating-point units used in TSPAN.

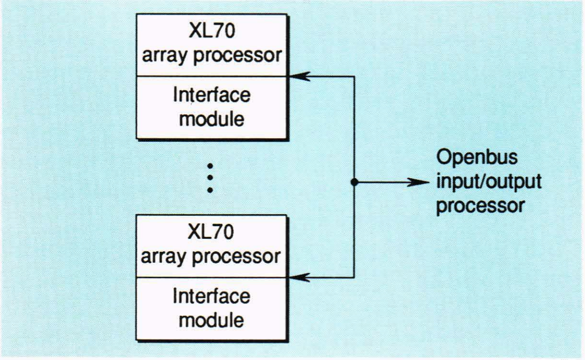


Figure 15. Architecture of the array processor units used in TSPAN.

puters connected to each major processing unit. An Ethernet link ties the MicroVAX II's to the VAX-8530, which holds the system software and provides the programming environment.

TSPAN contains only six custom modules, compared to 36 for SPAN-I. The TSPAN modules are 14-layer printed circuit boards using two controlled impedance layers. Its modules use the standard 9U Eurocard form factor; therefore, the modules are slightly smaller than 19 × 15 in.

FUTURE DIRECTIONS

Changes in the nature of the ocean and the nature of submarine warfare will require the use of more complex sonar signal-processing algorithms than those routinely used today. Such algorithms will in turn require computational potentials at least an order of magnitude greater than provided by TSPAN. Achieving that goal will require both faster processing elements and the use of more elements.

In addition, processing systems such as TSPAN produce large quantities of output that must be examined by human sonar analysts. Just the storage and display of this output is a difficult task, which is addressed by projects under way at APL. Ideally, much of the output of sonar signal-processing systems should be scanned by other types of processors, such as expert systems, which would act as useful aids to the human analyst. Such postsignal-processing systems are under active investigation at APL and elsewhere, but no successful system of this kind has been produced.

REFERENCES

[1] MacCurdy, E., *The Notebooks of Leonardo da Vinci*, Garden City Publishing Company, Inc., Garden City, N.Y. (1942).

[2] Dudgeon, D. E., and Mersereau, R. E., "Processing Signals Carried by Propagating Waves," Chapter 6, in *Multidimensional Digital Signal Processing*, Prentice-Hall, N.Y., pp. 289-339 (1984).

[3] Schultheiss, P. M., "Some Lessons from Array Processing Theory," in *Aspects of Signal Processing, Part 1*, Tacconi, G., ed., D. Reidel Publishing Co., Dordrecht, pp. 301-331 (1977).

[4] Van Veen, B. D., and Buckley, K. M., "Beamforming: A Versatile Approach to Spatial Filtering," *IEEE ASSP Mag.* **5**, 4-24 (1988).

[5] Harshman, J. V., "Architecture of a Programmable Digital Signal Processor," in *Digital Signal Computers & Processors*, Salazar, A. C., ed., IEEE Press, Piscataway, N.J., pp. 93-97 (1977).

[6] South, H. M., "Digital Recording and Signal Processing Systems for Hydrophone Arrays," *Johns Hopkins APL Tech. Dig.* **4**, 212-218 (1983).

[7] Dolecek, Q. E., "QUEN: The APL Wavefront Array Processor," *Johns Hopkins APL Tech. Dig.* **10**, 198-207 (1989).

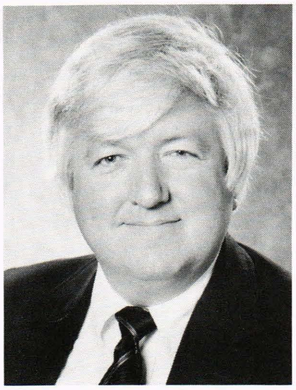
[8] Pierce, J. R., Coker, C. H., and Kropfl, W. J., "An Experiment in Addressed Block Data Transmission Around a Loop," in *IEEE International Convention Digest*, pp. 222-223 (1971).

[9] Allen, J., "Computer Architecture for Signal Processing," *Proc. IEEE* **63**, 624-633 (1975).

[10] Brigham, E. O., *The Fast Fourier Transform*, Prentice-Hall, Englewood Cliffs, N.J. (1975).

ACKNOWLEDGMENTS: The Director, Strategic Systems Programs, Department of the Navy, funded most of the APL work described in this article. Additional funding was provided by the G. W. C. Whiting School of Engineering of The Johns Hopkins University. The development of SPAN-A, SPAN-I, and TSPAN was conducted with the support of APL program managers Luciano P. Montanaro, Donald L. Eddins, and Douglas L. Geffert. In addition, valuable technical work was provided by APL staff members Michael H. Robinson, Quentin E. Dolecek, Frederick J. Everly, Daniel E. Chandler, John A. Cristion, and Gregory A. Niswonger. These sonar processors could not have been built without the efforts of many subcontractors to the Laboratory. The work of Paul M. Brewer, W. Don Ashcraft, Richard A. Smithson, and Steven F. Bergmann of Rockwell International and Raymond A. Janssen, Ken Houston, Steve Martin, Russ A. Wessner, and Douglas Byrum of Raytheon Company was especially important.

THE AUTHOR



HUGH M. SOUTH was born in Houston, Texas, in 1947. He received a B.S. degree in electrical engineering from Rice University in 1971 and a Ph.D. degree in electrical engineering from The Johns Hopkins University in 1981. Dr. South joined APL in 1975 as a member of the Sonar Evaluation Program Office, where he planned at-sea sonar performance tests and analyzed sonar data from SSBN patrols. He became supervisor of the SPAN Laboratory in 1979 and supervisor of the Signal Processing Group in 1985. He served as technical problem sponsor for the development of SPAN-I and is currently co-sponsor for the development of TSPAN. Dr. South is on the editorial board of the *APL Technical Review* and is also a member of the Underwater Acoustics Technical Committee of the IEEE Acoustics, Speech, and Signal Processing Society.