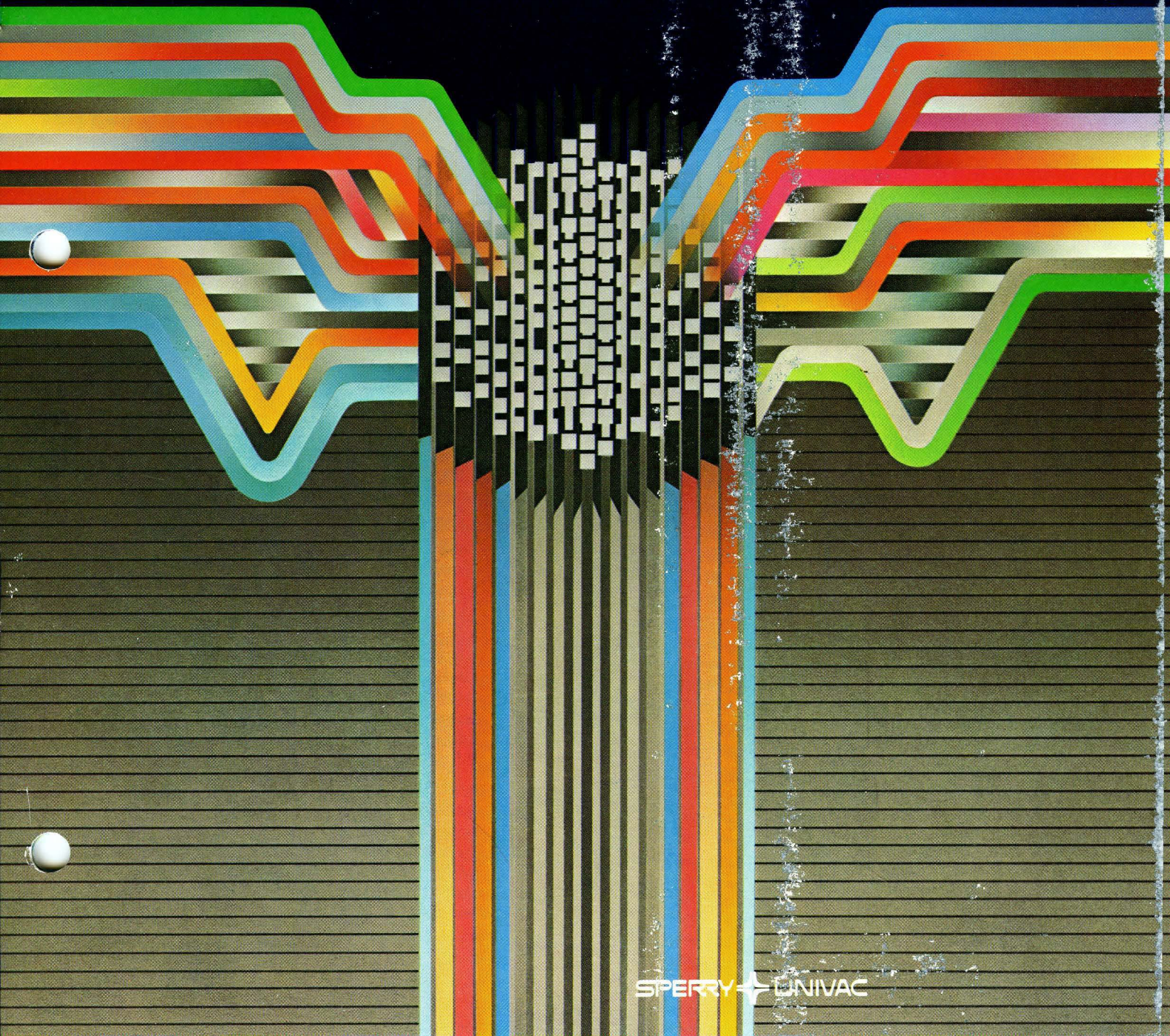


SPERRY UNIVAC  
**Series 1100**  
Array Processor Subsystem





The SPERRY UNIVAC Series 1100 has traditionally provided high-performance, large-scale computing systems that have led the way in applying computer technology for oil and gas exploration. In 1970, for example, Sperry Univac introduced the UNIVAC Array Processor, which enjoyed worldwide use for geophysical applications.

Now, Sperry Univac announces a new concept in scientific computing—the SPERRY UNIVAC 1100/80 Array Processor Subsystem (APS). Among the most powerful of supercomputers—and possibly the most advanced—the 1100/80 APS combines the speed of vector processing with the flexibility of general purpose computing to produce a unique and eminently usable system.

**General Description**

The SPERRY UNIVAC Array Processor Subsystem is a high-performance, *closely integrated* scientific processor available for the SPERRY UNIVAC 1100/80 computer family. The APS achieves a combination of functionality, system throughput and cost-effectiveness not approached by any other system.

While the APS has been specifically designed to provide peak performance for seismic processing, it can also be effectively used in reservoir

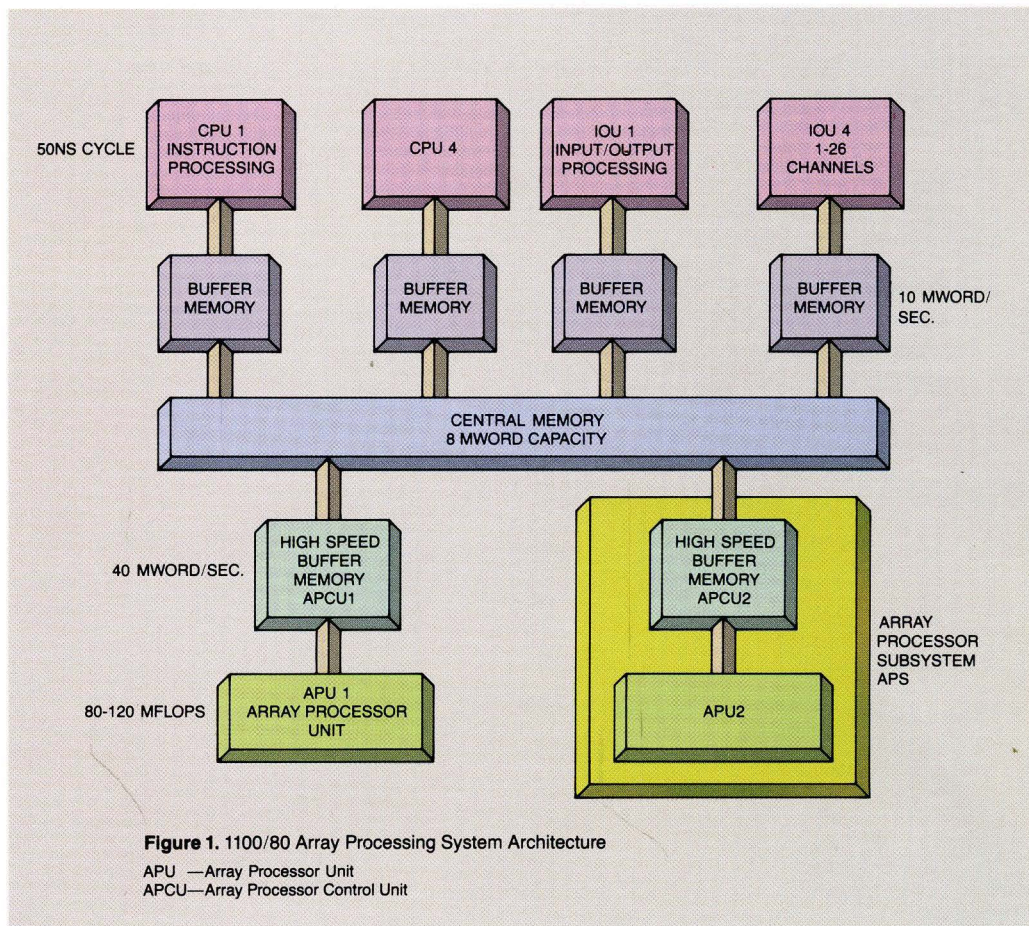
simulations, nuclear codes, electric power flow analysis, image processing, linear programming and large physical system modeling.

The APS functions as an extension of the 1100/80 system, which was first delivered to Sperry Univac customers in 1977.

The 1100/80 APS features separate logical and physical units for scalar processing (CPUs), Input/Output and communications processing (IOUs) and vector/array processing (APU). See Figure 1. Each functional unit is attached to, and operates directly on, a very large central memory via high speed Buffer Memories which operate as caches for data. The caches serve not only to transfer often-used data to and from the function units at very high speeds, but also to buffer the central memory bandwidth against extreme data request rates by the functional units.

Each APU can operate at speeds of up to 120 million floating point operations per second (MFLOPS). Buffer memories for the CPU and IOU are known as Storage Interface Units (SIU) and can transfer data to the CPU and IOU logic units at 10 million words per second.

Buffer memories for the APU are known as Array Processor Control Units (APCU) and can transfer data to and from APU logic elements at 40 million words per second. Each IOU in the system can accommodate up to 26 high speed I/O channels. The system is modular and can be expanded easily, in stages, from the minimum system of one CPU, one APU and 1 million words of central memory. In multiple, redundant unit configurations, the system exhibits resiliency and user availability, allowing redundant units to be isolated from the configuration and restored during production without system reboot.



**Figure 1. 1100/80 Array Processing System Architecture**

APU — Array Processor Unit  
APCU—Array Processor Control Unit

This document contains the latest information available at the time of publication. Sperry Univac reserves the right to change without notice specifications, performance data and availability dates contained herein.

SPERRY UNIVAC and UNIVAC are trademarks of Sperry Corporation.

**Design Objectives**

The primary objectives of the APS focus on extremely high system performance and functionality by:

- Providing high floating point arithmetic performance in the Array Processor Unit (APU)
- Providing sufficient system data bandwidth direct from host to APU to sustain the very high internal APU performance
- Permitting full user microprogramming capability in the Array Processor Unit
- Providing functionality to enable the APS to be used for numerically intensive scientific problems—wherever some vector processing is present—particularly seismic data reduction and modeling and simulation of physical systems
- Allowing Array Processor algorithms a large, linear address space, up to 8 million 36-bit words
- Producing identical floating point arithmetic results to the Series 1100 Central Processing Units (36-bit)
- Minimizing additional supporting host operating system software and task execution overhead
- Providing for efficient sharing of the APS in a multiprogramming and time-sharing environment
- Providing extensive hardware-assisted statistics to enable users to observe, analyze and improve system performance.

Each APS consists of two major components: The Array Processor Unit (APU), which connects directly to an 1100/80 system (Figure 1), and the Array Processor Control Unit (APCU).

**Array Processor Unit (APU)**

The APU provides control and pipelined arithmetic units, program/instruction memory and local data scratchpad memory.

**APU Architecture**

The APU, as shown in Figure 2, consists of: four control sections which interpret instructions and compute addresses; program (instruction) memory; scratchpad memory; and four pipelined arithmetic sections. Each parallel pipeline has one floating point multiplier and two Arithmetic Logic Units (Figure 3).

**APU Specifications Summary**

- Basic speed is 25 nanoseconds (NS) effective 36-bit floating point multiply-add time (one multiply and one addition result each 25 nanoseconds).
- Four pipelined array arithmetic sections (Figure 3), each having one floating point multiplier and two Arithmetic Logic Units (ALU), which perform 60 arithmetic and Boolean operations on two 36-bit inputs in six categories:
  - Control
  - Logic
  - Floating Point Arithmetic
  - Fixed Point Arithmetic
  - Conversion
  - Comparison.

Each pipeline is capable of producing a result every 100ns for effective sustainable performance levels of up to 80 megaflops (80 million floating point operations per second) per APU in suitable algorithms, with maximum performance of 120 megaflops in bursts.

- Four control processors to decode instructions, generate addressing and collect statistics
- User microprogrammable by means of 288-bit microinstructions—8K instructions (expandable to 16K)
- 65K words of scratchpad data memory (36-bit words plus parity, expandable to 262K words)
- Data bandwidth of up to 40 million words/second (up to 80 million words/second if local data memory is used simultaneously)
- Addressing capability of up to 8 million words per APU application (algorithm)
- Unit-wide busing/multiplexing of data
- Full internal parity checking and memory protection
- Maintenance facility provided with dedicated microprocessor and breakpoint, fault isolation via scan/compare of circuit gates
- Hardware assisted algorithm statistics gathering (accumulated execution time of algorithm subroutines, number of times subroutine is called, memory conflicts.)

**Floating Point Numerical Representation**

The APU Arithmetic Logic Units and Multipliers produce identical, normalized 36-bit floating point results just as the 1100/80 CPU(s) does. This format provides a range of  $10^{38}$  to  $10^{-38}$  with eight decimal digit precision.

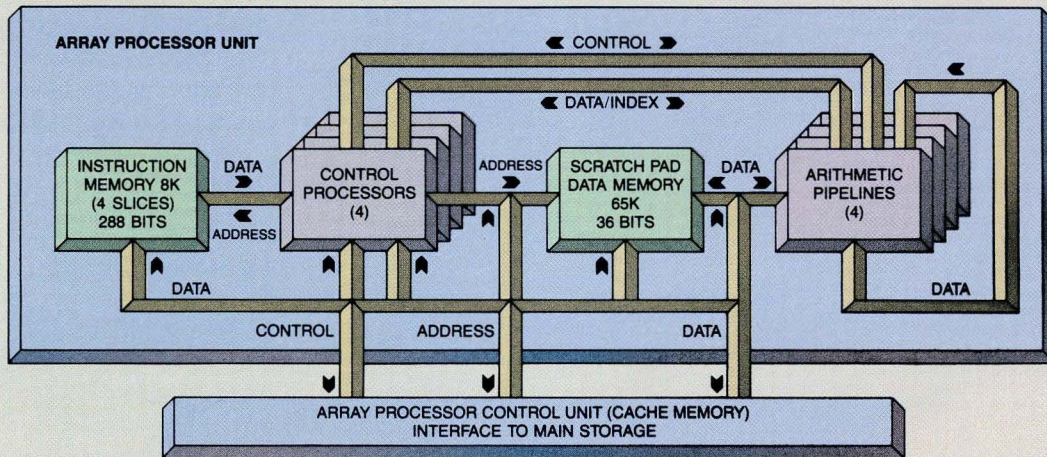


Figure 2. Array Processor Unit Architecture

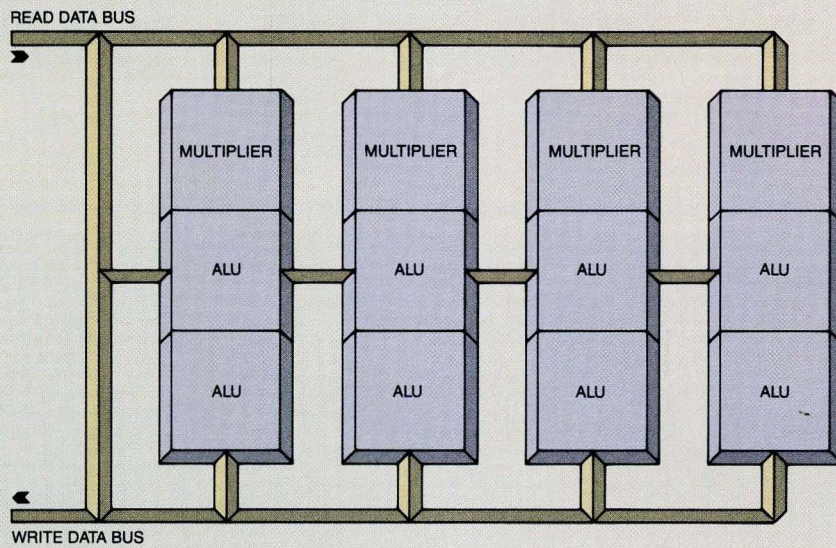
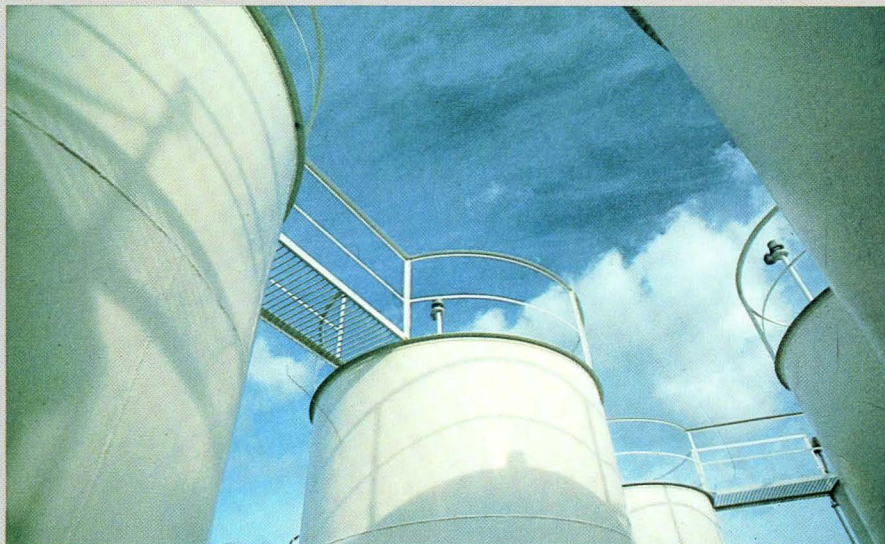
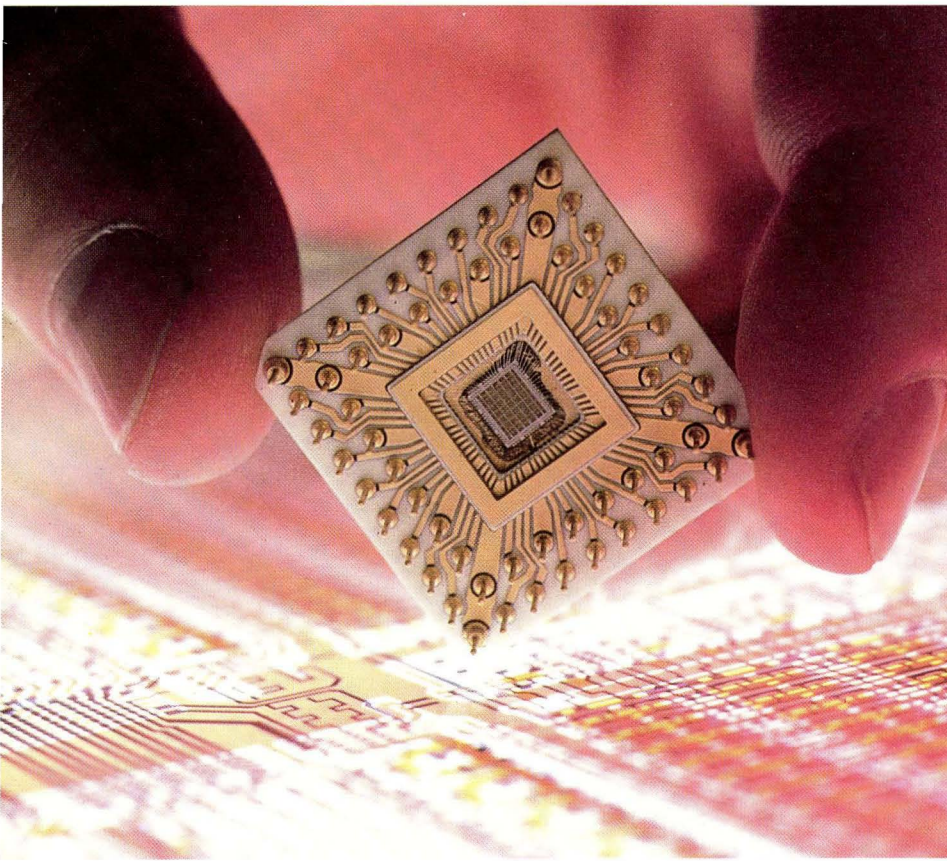


Figure 3. APU Parallel Pipeline Architecture





The Array Processor Control Unit (APCU) functions as a fully associative cache or high-speed buffer memory to the 1100/80 memory. It can stream data to and from the APU at rates of 40 million words/seconds (36-bit words). See Figure 4.

This high speed data rate assures that the APU can *sustain* its high performance. High system bandwidths are achieved by integrating the Array Processor Unit with the SPERRY UNIVAC 1100/80 system by means of the cache buffer memory within the Array Processor Control Unit, thereby minimizing impact on the host. A main storage reference is made only when data required by the APU is not available in the APCU cache or when the "fetch ahead" mechanism foresees that data will soon be required by the APU.

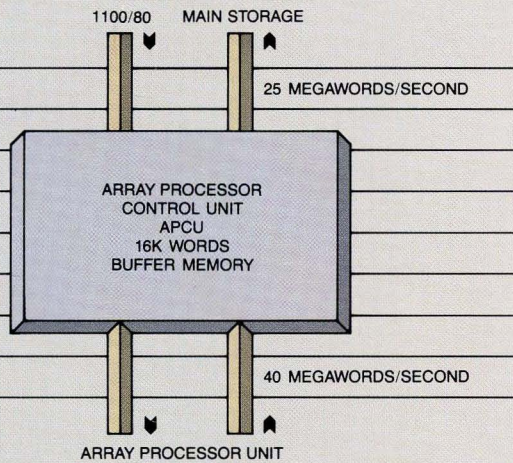


Figure 4. Data Paths and Capacities

### Magnetic Tape Subsystems Characteristics

	UNISERVO 30	UNISERVO 32
Recording density (PE)	1600 bpi	1600 bpi
(NRZI)	200, 556, 800 bpi	-
(GCR)	-	6250 bpi
Transfer rate (PE)	320,000 fps	120,000 fps
(NRZI)	40,000, 111,200, 160,000 fps	-
(GCR)	-	468,750 fps
Tape speed	200 ips	75 ips

### Disk Subsystems Characteristics

CHARACTERISTIC	8430	8433	8450
Drives per subsystem	2-16	2-16	2-32
Capacity per disk pack (BYTES)	100,000,000	200,000,000	302,000,000
Average access time (in milliseconds)	27	30	23*
Data transfer rate (BYTES per second)	806,000	806,000	1,260,000

Figure 5. 1100/80-APS TAPE/DISK Peripherals

The 1100/80 CPU and IOUs are versatile functional units that support scalar processing and advanced peripheral complexes directly (Figure 5). The entire system operates under control of the Series 1100 Operating System, first introduced in 1965. The 1100 OS provides a proven, stable platform for the extension of the Array Processing System to the 1100/80. Under control of the 1100 OS, the 1100/80 APS *directly* supports interactive timesharing/graphics, remote job entry, real-time and batch modes of user access to any component of the system.

Some of the most prominent features of the SPERRY UNIVAC 1100/80 system design are:

- Large, real-system memory—up to 8 million 36-bit words
- Large, high-speed buffer memories totaling up to 32K, 36-bit words, 100ns cycle time per 36-bit word
- High performance, multiple scalar processors, 50ns cycle times each CPU
- Scientific Accelerator Module (SAM), a high-speed CPU arithmetic instruction execution unit
- Basic instruction repertoire of 200+ instructions
- Independent Input/Output processors (IOU)
- High performance peripherals
- Byte-oriented and word-oriented I/O channels (104 maximum).

**System Configurations**

The 1100/80 system is designed around a multiple, independent processor concept. The minimum configuration (1100/81) consists of one central processor and one Input/Output processor. Memory and peripheral complements can vary. The largest mainframe configuration with two array processors is the 1100/84 with four central processors and four IOUs. This configuration, with fully shared peripherals, offers fully redundant and fail-safe capabilities (Figure 6).

UNISERVO 34	UNISERVO 36
1600 bpi	1600 bpi
—	—
6250 bpi	6250 bpi
200,000 fps	320,000 fps
—	—
781,250 fps	1,250,000 fps
125 ips	200 ips

Legend:  
 bpi = bits per inch  
 fps = frames per second  
 ips = inches per second

8470	7053 CACHE DISK	7053 SOLID STATE DISK
2-32	2-16	1-4
0,000	562,000,000	302,000,000/645,000,000
23	1	.2
000	2,097,000	1,260,000/2,097,000
		5,000,000

MAIN STORAGE	SCALAR PROCESSORS (CPU'S)	VECTOR PROCESSORS (APU'S)	INPUT/OUTPUT PROCESSORS (IOU'S)
(36-bit words) 1048 K	1-2	1	1-2
2096 K	1-4	1-2	1-4
4192 K	1-4	1-2	1-4
8384 K	1-4	1-2	1-4

**Figure 6** 1100/80-APS Configuration Flexibility

**VAST™ Compiler FORTRAN Interface**

User scientific applications for the Array Processing System are written in FORTRAN. A new or existing FORTRAN program, conforming to ANSI-1977 FORTRAN, can be submitted to the VAST precompiler, an APS vectorization utility that translates appropriate FORTRAN source statements to APS vector operations. At this first level of interface to the APS (Figure 7), user FORTRAN remains completely transportable, with no executable FORTRAN source statements changed or added.

The VAST precompiler provides a number of benefits for the APS user:

- **APS Application Transparency:** no changes to working FORTRAN source code are required to use the APS for faster computations
- **Portability:** programs written for other systems may be moved to 1100/80 APS system without change, if written in standard FORTRAN.
- **Reduced Training:** an application programmer need only learn how to invoke the VAST software in order to use the great speed of the APS; there is no requirement to become familiar with the APS hardware.
- **Efficiency:** VAST software includes features that chain together APS invocations, reducing transition overhead between the 1100/80 host and the APS.

In addition to a modified FORTRAN output program, VAST software also produces a listing of the input program with an analysis indicating why certain

loops were not vectorized. The application programmer may then modify the program and reprocess it with the VAST precompiler.

The VAST analyzer is actually only one component of the Vectorizer Utility, which includes the APS output option (to produce code for the APS after VAST analysis), the Chainer and the Interpreter. The Chainer aggregates separate APS operation invocations into a single "description block," which is then dispatched to the APS. In the APS, the Interpreter processes each descriptor block, performing the various separate operations on the APS. A descriptor block may even include scalar operations if a brief sequence of them intervenes between vector constructs. In this way, the Chainer-Interpreter combination reduces the number of transitions between the host 1100/80 and the APS, decreasing system overhead and increasing performance.

VAST is a trademark of The Pacific Sierra Research Corp.

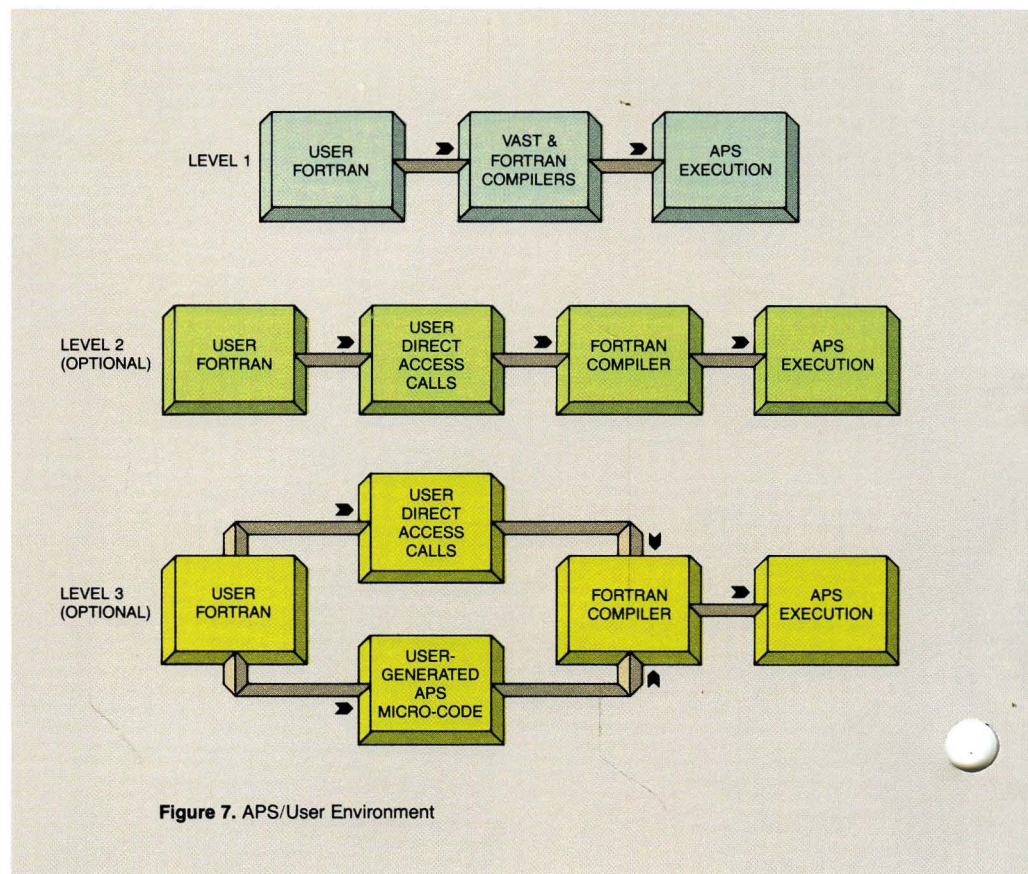


Figure 7. APS/User Environment



**FORTRAN Interface for Direct Access**

There is an optional second level of user program interface to the APS. It permits explicit FORTRAN subroutine CALLs of the APS vector operation to be inserted for replacing or augmenting existing FORTRAN statements. This second level offers more efficiency and higher performance. At this level, applications written in other languages may also use the APS.

Applications requesting APS execution normally use four FORTRAN CALL linkages:

- **CALL APIMP (parameter list):** normally used once per program to initialize the APS access method
- **CALL APDEF (parameter list):** used to define the real memory address space containing vectors/arrays that the APS operates on. This may be used more than once during a program.
- **CALL APXQT (parameter list):** used to execute a particular vector array operation (algorithm) from the APS Algorithm Library and to define the vector/array to be operated upon. It may be used more than once during a program.
- **CALL APDPRT (parameter list):** used to conclude the program's access to the APS.

**APS Microcode Interface**

At the optional third level of interface to the APS, vector operation microcode may be written to replace selected FORTRAN functions that do not exist as current APS vector operations.

To support the development of additional APS algorithms, a cross-assembler, library builder and simulator are provided. With these aids, users may construct and register new algorithms or applications for execution by the APS.

The cross-assembler allows the 1100 user to utilize a macrolanguage to design an algorithm or application, which is then generated automatically as an object microprogram (288-bit instructions). This macrolanguage provides for direction of APU control and arithmetic processors, data path definition and memory access protocols.

The library builder allows different user programs to use different algorithm libraries and allows linking of separate algorithms (subroutines) to form single applications.

Debug simulation of user-coded algorithms using the host APU simulator saves significant time and system resources in early checkout stages.

All vector operations, complex signal processing and matrix operations are directed by microcode contained within the APU. The microcode controls and configures each functional unit in each pipeline during execution of every cycle. The standard set of vector microcoded algorithms supplied with the APS are shown in Figure 8. The user Direct Access FORTRAN interface uses this set of APS operations.

This algorithm set may be modified or expanded. Full APS microprogramming facilities are provided by the system software.

Name	Operation
FFT Class	
FFT	Basic FFT Butterfly
IFFT	Inverse FFT Butterfly
UNSC	Unscramble to Natural Order
FFTN	FFT Normalization
CMCJ	Complex Conjugate
SRCB	Single Real Coefficient Builder
ISRC	Inverse Single Real Coefficient Builder
COEF	Two Real Coefficient Builder
ICOE	Inverse Two Real Coefficient Builder
Convolution/Correlation Class	
CONV	Convolution Multiply
CONA	Convolution Addition
CORR	Correlation Multiply
CORA	Correlation Addition
Vector Reduction Class	
VECP	Vector Inner Product
SSSQ	Sum of Signed Squared Vector
SUMR	Sum Reduction
SMSQ	Sum of Squares
Vector Element By Element Class	
VECA	Vector Add
VECM	Vector Multiply
CPXM	Complex Vector Multiply
VECS	Vector Squared
VESS	Signed Squared Vector
VMAG	Vector Magnitude
VNEG	Vector Negative
VNMG	Vector Negative Magnitude
PARM	Partial Matrix Multiply
LFnn	Logical Function nn
Utility Class	
SVH	Scan Vector High Value
SVHM	Scan Vector High Magnitude
SVL	Scan Vector Low Value
SVLM	Scan Vector Low Magnitude
EXCV	Exchange Vector
COPY	Copy Vector
CLRM	Clear Memory

**Figure 8.** Basic Algorithm Library



**Extended Algorithm Library**

The Extended Primitive Library is an additional set of APS algorithms supplementing those in the basic library. Although intended primarily for use with VAST translated programs, these algorithms may themselves be useful and are available separately.

The extended algorithms include: integer operations, relational operations, logical operations, indirect addressing (use of vector components as indices), scalar and vector division, square roots, transcendental functions and combined scalar/vector expressions.

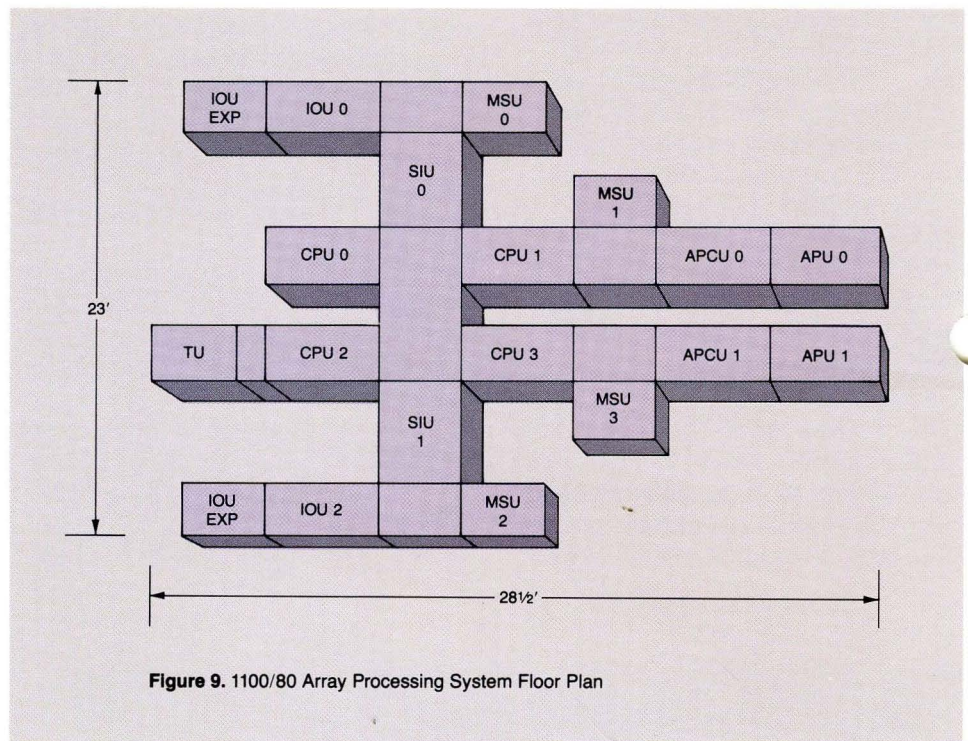
Sperry Univac also provides consultant services for the development of additional algorithms tailored to your requirements.

Because the 1100/80 APU is entirely air cooled, the expense of installation is minimal. In addition, the floor space needed for the maximum configuration of the mainframe is less than 700 square feet (Figure 9). The minimum configuration is about half that amount.

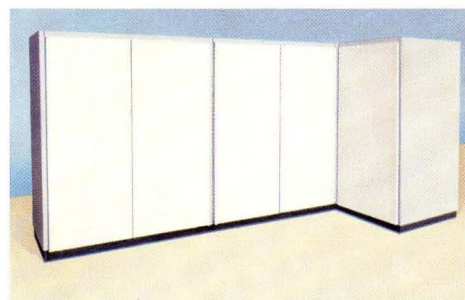
The Array Processor Subsystem is shown in Figures 10 and 11. The APS is contained in two cabinets, each six feet high, five feet long and 2.5 feet wide. As seen in Figure 11, the single APU logic deck occupies only eight cubic feet of the APU cabinet.

The 1100/80 APS has been designed with sustained high speed in mind. Unique reliability and maintenance features include:

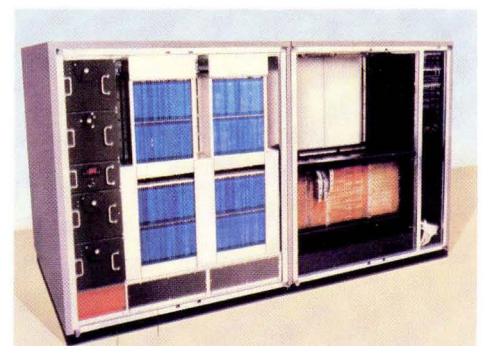
- Parity checks throughout data paths in the processors and semiconductor memories
- A special maintenance control computer scans and sets gates in the APS microprocessors, checking the expected against the observed conditions.



**Figure 9.** 1100/80 Array Processing System Floor Plan



**Figure 10.** Array Processor Subsystem External Appearance



**Figure 11.** Array Processor Subsystem Internal View



SPEERY  UNIVAC

*The computer people who listen.*