THE PDP-16/M SUBMINICOMPUTER

THE NEWEST MEMBER OF DEC'S FAMILY OF 16-BIT FUNCTIONAL COMPUTERS

digital

THE PDP-16/M IS A...

- GENERAL PURPOSE COMPUTER
- COMMUNICATIONS PROCESSOR
- INDUSTRIAL CONTROLLER
- COMPUTER TERMINAL CONTROLLER
- EDUCATIONAL TOOL
- DATA LOGGER
- PREPROCESSOR
- INSTRUMENTATION CONTROLLER
- PROCESS MONITOR

WHAT'S A SUBMINICOMPUTER?

Digital Equipment Corporation pioneered the concept of the minicomputer more than a decade ago. Some years later DEC developed the PDP-16 16-bit functional computer. Now DEC has combined the programmable capability of the minicomputer and the proven reliability of PDP-16 hardware to produce a true subminicomputer—the PDP-16/M.

This major innovation in 16-bit functional computers provides the user with all the power of the minicomputer at a fraction of the cost. The PDP-16/M incorporates the programmable read only memory to offer the most cost effective family of computers available today. And a variety of options makes the PDP-16/M a highly versatile device for the OEM, educator and systems designer. The PDP-16/M is a submini in size and price, but a giant in performance.

The basic chassis of the PDP-16/M is designed so that arithmetic unit, control memory, data memory and I/O controls can be conveniently plugged into the computer. Space is reserved within the chassis for 1K words of control ROM, 1.5K words of solid state RAM, 280 words of data ROM, 33 high speed registers, 6 control flip-flops, 2 serial I/O channels and 3 16-bit parallel I/O channels. An expander chassis can be attached to the back of the basic housing for additional PDP-16/M options, special logic functions or special interfacing hardware.

A reprogrammable read only memory is used within the basic PDP-16/M. Fusible link ROMs or Masked ROMs are used for large quantities of a given PDP-16/M configuration to obtain maximum cost effectiveness for each unit. All units are pin compatible and interchangeable, at the module level, regardless of the ROM chosen.

A PDP-8 computer based assembler is used for development of PDP-16/M programs. The output tape for the desired program is returned to DEC for loading into the PDP-16/M control memory. A ROM simulator program for the PDP-8 is also available. This allows the PDP-8 to simulate the PDP-16/M control memory.

DEC will offer an optional ROM loader, compatible with both the PDP-8 and PDP-16/M, at a later date.

PDP-16/M ORGANIZATION

HARDWARE

The figure below diagrams the basic hardware components of the PDP-16/M as well as available options.

The PDP-16/M contains four functional boxes: Control/Program Memory—holds from 256 to 1024 8-bit word programs. It fetches instructions from its program memory and instructs the three unit types listed below. Control also senses input (bit) conditions from other parts of the system. Control has a Program Counter (PC) which points to the instruction being executed. A subroutine stack holds the previous value of the PC when subroutines are called.

Arithmetic/Logic—contains 3 + registers and arithmetic/logic operations capability to carry out instructions on 16-bit data. The registers are: A and B (for holding arithmetic results); Link-bit (used as input to A and B for shift operations); Overflow-bit (for holding arithmetic overflow results from A and B); and Bus Sense (set on every register data transmission and able to be tested i.e., BS=0, BS<0, BS>0).

Memory—organized as bits (Flags), registers, read only and read-write arrays. Total memory size is approximately 1,600 16-bit words. Only data is stored in memory (not the program). The actual memory configuration can be up to 6 bits (Flags), 1 word register, 32 word scratchpad, 280 words read only constants and 1512 words read-write.

Input/Output—communicates with all the other boxes. Capacity of up to three sets of 16-bit parallel input and output lines and two sets of bit-serial input and output communications equipment lines. Also, 22 Boolean (bit) external inputs can be sensed by the control unit.

Data is transferred among the arithmetic-logic, memory and I/O boxes via a 21-wire PDP-16 bus. Since this bus is a subset of the PDP-11 bus, PDP-16 and PDP-11 modules can be plugged into it. The simple PDP-16/M bus protocol facilitates the design of special logic interfaces.
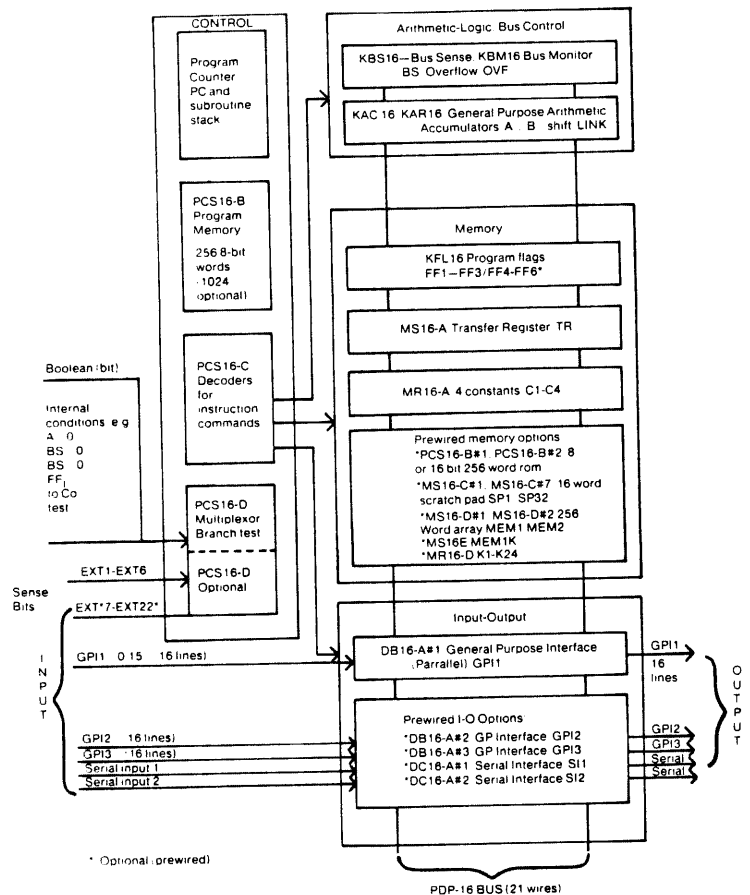
SOFTWARE

The PDP-16/M program is stored in a solid state, reprogrammable, read only memory. The basic memory module is 8 bits by 256 words to 1024 words maximum. Programs can be written using special software packages on the PDP-8. The PDP-8 can then be used to simulate the program during system debugging. Finally, a PDP-8 with a special front end or the memory loading service provided by DEC can be used to load the PDP-16/M control memory.

The PDP-16/M assembler uses a high level language consisting of only five basic instructions. Thus, the programming of a PDP-16/M is a simple matter even for the novice.

PDP-16/M INSTRUCTION FORMATS

There are five types of instructions which operate on or transfer data among the four sections. Instructions in the machine are specified as either an 8-bit or 16-bit word depending on whether addresses are present.
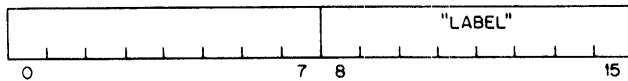
1. Register-transfer type

```
          ┌──────────────────────┐
          │       OPCODE         │
          └──────────────────────┘
          0                      7
```
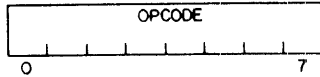
This instruction type either specifies arithmetic operations on the register or transfers data among the registers of the other sections. This includes accessing memory, input-output, and the arithmetic section registers. Some example assembler formats and their corresponding meanings are:

A ← A + B	add register A and B, transfer the sum to A
GPI1 ← A	transfer the register A to the first general purpose interface register, GPI1
A ← SP17	load A with the 17th scratchpad register
FF2 ← 1	set program flag 2 with a one

2. Unconditional Branch (i.e., GOTO) type

```
┌───────────────────┬──────────────────────┐
│                   │       "LABEL"        │
└───────────────────┴──────────────────────┘
0                   7 8                    15
```
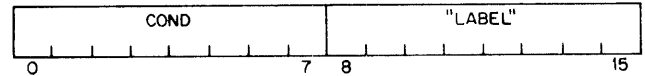
GOTO "Label"   place the address (number) corresponding to "Label" in the Program Counter (PC) register, i.e., transfer or jump

3. Conditional Branch type

```
┌───────────────────┬──────────────────────┐
│       COND        │       "LABEL"        │
└───────────────────┴──────────────────────┘
0                   7 8                    15
```

IF "CONDITION" "Label"   if the Boolean (bit) "Condition" specified is true (on, 1, etc.), GOTO "Label" address

4. CALL subroutine type

```
┌───────────────────┬──────────────────────┐
│                   │       "LABEL"        │
└───────────────────┴──────────────────────┘
0                   7 8                    15
```

CALL "Label"   call the subroutine at location "Label", save the Program Counter (PC) in the subroutine return stack

5. EXIT subroutine type

```
          ┌──────────────────────┐
          │                      │
          └──────────────────────┘
          0                      7
```

EXIT   return from the subroutine previously called

PROGRAM EXAMPLE

This program waits until an external input is true, it takes in a number in BCD, sums the integers up to the input value $S = 0 + 1 + 2 + ... + i$, then outputs the sum in BCD.
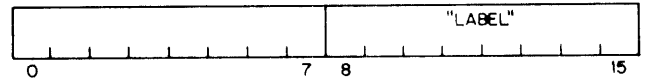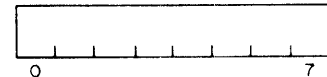
SENSE,	IF EXT1 START GOTO SENSE	WAIT FOR EXT1 INPUT BEFORE STARTING
START,	A ← GPI1 CALL BCD-BIN	TAKE IN DATA, IN BCD, ON INPUT 1 CONVERT TO BINARY USING SUBROUTINE
	B ← 0	BEGIN PROGRAM TO SUM INTEGERS TO INPUT
LOOP,	B ← A + B A ← A − 1 IF DP LOOP	LOOP IF A IS POSITIVE
	A ← B	END OF PROGRAM TO SUM INTEGERS
	CALL BIN-BCD	CONVERT TO BCD USING SUBROUTINE
	GPI1 ← A . . .	OUTPUT
BCD-BIN	. . .	BCD TO BINARY SUB
	EXIT	
BIN-BCD	.	BINARY TO BCD SUB
	EXIT	

PDP-16/M INSTRUCTION SETS

ARITHMETIC AND LOGICAL INSTRUCTIONS

The following instructions are specified by a 1 byte operation code, and operate on the KBM16, KBS16 Bus Control and KAC16, KAR16 General Purpose Arithmetic. Each instruction takes 2.4 microseconds. The instructions use the A and B registers as arithmetic and logical operands, and operation results are placed in either A or B. Other instructions use and affect the Bus Sense (BS), Overflow-bit (OVF) and shift LINK bit.

Instructions for A(arithmetic-logical):

A←0	clear A (transfer a 0 to A)
A←B	store B in A
A←A + 1	increment A
A←A + 1(S)	increment A, set OVF
A←A − 1	decrement A
A←A − 1(S)	decrement A, set OVF
A←A + B	add B to A
A←A + B(S)	add B to A, set OVF
A←A − B	subtract B from A
A←A − B(S)	subtract B from A, set OVF
A←AXORB	exclusive—or B to A
A←AORB	inclusive—or B to A
A←AB	and B to A
A←NOTA	negate A
A←A/2	rotate A and Link right
A←A/2(S)	rotate A and Link right, set OVF
A←AX2	rotate A and Link left
A←AX2(S)	rotate A and Link left, set OVF
*A←B/2	Set A equal to rotate B and Link right
*A←B/2(S)	set A equal to rotate B and Link right, set OVF

Instructions for the shift Link-bit:

L←0	set Link to 0
*L←OVF	set Link equal to OVF

Instructions for the Bus Sense (BS):

BS←A	load BS with A

* Optional

Instructions for B(arithmetic-logical):

B←0	clear B
B←A	store B in A
*B←A + 1	set B equal to A + 1
*B←A + 1(S)	set B equal to A + 1, set OVF
*B←A − 1	set B equal to A − 1
*B←A − 1(S)	set B equal to A − 1, set OVF
B←A + B	add A to B
B←A + B(S)	add A to B, set OVF
B←A − B	subtract A from B
B←A − B(S)	subtract A from B, set OVF
B←AXORB	exclusive—or A to B
B←AORB	inclusive—or A to B
B←AB	and A to B
B←NOTB	negate B
B←B/2	rotate B and Link right
B←B/2(S)	rotate B and Link right, set OVF
*B←AX2	set B equal to rotate A and Link left
*B←AX2(S)	set B equal to rotate A and Link left, set OVF
*B←A/2	set B equal to rotate A and Link right
*B←A/2(S)	set B equal to rotate A and Link right, set OVF

L←1	set Link to 1
*L←LNOT	compliment Link

BS←B	load BS with B

* These instructions are optional. Decoder #5 must be added to the basic machine.

CONTROL INSTRUCTIONS

A special set of instructions controls the program flow. All instructions may affect the Program Counter (PC). Instructions: unconditional branch (i.e., set PC)—the GOTO instruction; conditional branch (conditionally set PC)—IF instruction for testing Boolean (bit) conditions; CALL subroutines, or EXIT from subroutines. The instructions which change the PC with a new address are two bytes long. These instructions require 2.0 microseconds. The unconditional and conditional branch instructions are:

GOTO "Label"	a two byte instruction which places the address "Label" in the PC
IF "Condition", "Label"	if the Boolean (bit) "Condition" selected by IF instruction is true, the address "Label" is placed in the PC

There are two groups of Boolean input variables which can be tested. In order to specify which of the group is being tested, the MUX-bit (which is either 0 or 1) must be set properly to access either the 1st or 2nd group respectively. The MUX instructions are:

MUX0	set the MUX register for the selection of the first group of inputs (MUX$\leftarrow$0)
MUX1	set MUX to select the second group (MUX$\leftarrow$1)

The "Condition" input mnemonic Boolean names of the first group (MUX0) are:

DZ	(BS$=$0), result in Bus Sense zero
DP	(BS$\geq$0), result in Bus Sense positive
DN	(BS$<$0), result in Bus Sense negative
OVF	OVF is on
A$<$1$>$	A register bit 1 is a one
A$<$3$>$, A$<$5$>$, A$<$7$>$, A$<$9$>$, A$<$11$>$, A$<$13$>$, A$<$15$>$	Other bits
FF$_j$	Boolean flag (Flip-Flop) j is a one j$=$1,2,...,6

EXT$_j$	external condition j is a one, j$=$1,2,...,6
KF$_j$	optional Keyboard Flag, KF, is one, j$=$1,2
PF$_j$	optional Punch Flag, PF, is one, j$=$1,2

The "Condition" input mnemonics of the second optional group (MUX1) are:

PWOK	Power OK signal from the power supply
*A$<$0$>$,...,*A$<$14$>$	bits 0,2,4,6,8,10,12,14 of A
*EXT$_j$	external condition j is one, j$=$7,...,22
B$<$0$>$, B$<$15$>$	bits 0 and 15 of the B register
L	the Link bit

The subroutine CALL and EXIT instructions allow a subroutine to be called and terminated (exited) respectively. The CALL instruction is two bytes, and EXIT is one byte. They are specified:

CALL "Label"	go to the subroutine addressed by the address, "Label", place PC in the top register of the subroutine stack, and push the registers of the stack down one position
EXIT	return from the current subroutine by placing the top register of the stack into the PC and popping the registers of the stack up one position
*HALT	halt the PDP-16/M and wait for an external continue pulse or the start switch

*optional modules

MEMORY INSTRUCTIONS

BOOLEAN FLAG REGISTERS FF1, FF2, FF3, FF4*,...FF6*

The flag registers are each 1-bit (Boolean) registers which can be reset to 0 or set to 1 under program control. The conditional jump instructions also permit the flags to be used to control the sequencing (branching) if the flag is a 1. FF1,2, and 3 are standard and FF4,5, and 6 are optional.

$FF_j \leftarrow 0$ clear (reset) the jth flag
$FF_j \leftarrow 1$ set the jth flag

MS16-A TRANSFER REGISTER

The Transfer Register, $TR<0:15>$ is a 16-bit register used as a temporary register. It is normally wired to perform a particular bit transformation, but can be rewired for other uses. The normal instructions are:

$TR \leftarrow A$ load TR with A
$A \leftarrow TR$ load A with TR
$A \leftarrow TRU$ load $A<0:7>$ with $TR<0:7>$, load $A<8:15>$ with 0
$A \leftarrow TRL$ load $A<8:15>$ with $TR<8:15>$, load $A<0:7>$ with 0

PCS16-B #1,* PCS16-B#2* 8 OR 16 BIT 256 WORD DATA ROM DB16-A* ROM INTERFACE

The data ROM is implemented using the same solid state ROM's found in the control memory. They are interfaced to the PDP-16/M data bus using a standard 16 bit input-output interface. The lower 8 bits of the output interface are used for the ROM address. The output of PCS16-B #1 is read into the lower 8 bits and the output of the PCS16-B #2 is connected to the upper 8 bits. If only one PCS16-B option is used, the other 8 bits will be read as logic ones. These memories can be used to store text, conversion tables, interpolation tables for mathematic calculations, control data or program constants. Just like the control memories these memories can be erased and reloaded if desired.

$A \leftarrow ROM$ register A gets the output of both ROM s
$B \leftarrow ROM$ register B gets the output of both ROM s
$RMAR \leftarrow A$ ROM MAR gets register A only, bits 0-7 are used
$RMAR \leftarrow B$ ROM MAR gets register B only, bits 0-7 are used

* Optional

MR16-A AND MR16-D* CONSTANTS MEMORIES

These small, read only memories are addressed explicitly by word. The MR16-A is a 4-word Diode ROM. The contents are specified by presence (1) or absence (0) of diodes. The MR16-D is a 24 word ROM. The MR16-A and MR16-D have access times of 0.2 microseconds and 1 microsecond respectively.

For MR16-A:
$B \leftarrow C_j$ read constant j into A, where j=1, 2, 3 or 4

For MR16-D:*
$B \leftarrow K_j$ read constant j into B, where j=1, 2, ...,24

MS16-C,* 16 WORD SCRATCHPAD MEMORIES

The 16-word scratchpad is an optional read-write memory which is addressed in an explicit fashion as though each word were an independent register. The two scratchpads are denoted SP(1:16) and SP(17:32). There are 16 instructions to read and 16 instructions to write each word of the 16 word scratchpad.

For MS16-C#1 (words 1:16)*
$SP_j \leftarrow A$ load scratchpad register j with A

$A \leftarrow SP_j$ load A with register scratchpad j when j=1, ...,16

For MS16-C#2 (words 17:32)*
$SP_j \leftarrow A$ when j=17,...,32
$A \leftarrow SP_j$

MS16-D#1*, MS16-D#2*, AND MS16-E* ARRAY ORGANIZED MEMORIES

The three memory arrays are 256, 256 and 1024 words, respectively. The cycle time of each memory is roughly two microseconds. They all operate in exactly the same way. Each array has a memory address register (MAR) which holds the address of a word being accessed. After MAR is loaded the word (specified by MAR) can be either read from A or written into A.

For MS16-D#1 (256 words)*
MAR1←A   load the memory address of the first 256-word memory with least significant 8 bits of A
A←MEM1   read the word accessed by MAR1 into A
MEM1←A   write the word accessed by MAR1 from A

For MS16-D#2* (256 words)
MAR2←A   load the memory address register
A←MEM2   read the word accessed by MAR2 into A
MEM2←A   write the word accessed by MAR2 from A

For MS16-E* (1024 words)
MAR1K←A   load the memory address register using least significant 10 bits
A←MEM1K   read the word accessed by MAR1K into A
MEM1K←A   write the word accessed by MAR1K from A

* Optional

INPUT-OUTPUT INSTRUCTIONS

DB16-A≠1,2* and 3* GENERAL PURPOSE INTERFACE (GPI) MODULES

The GPI provides independent input and output 16-bit word channels to the computer. GPI contains a 16-bit register, GP<0:15> which holds the 16 bits on output. An instruction loads GP. On input the 16 lines are read by an instruction. One GPI is standard, and two additional GPI's are prewired options.

$GPI_j \leftarrow A$ load the jth output register, GP from A
$A \leftarrow GPI_j$ read in the jth input to A where $j = 1,2$ or 3

DC16-A≠1*, ≠2* SERIAL INTERFACES (SI) MODULES

The Serial Interface allows the computer to interface with teleprinters and serial, asynchronous communication lines. It contains a transmitter, a receiver and built-in crystal clock. The transmitter and receiver are completely independent of one another providing full-duplex communication. Standard data rates are 110, 150, 300, 600, 1200, and 2400 Baud. The bits per character and Baud rate are selected by jumpers on the serial interface definition module.

Standard 20mA current loops are available from a connector mounted on the side of the module. This connector is the same as the PDP-11 teleprinter.

The transmitter section contains an 8-bit register, T<0:7>, which holds the character being transmitted, and a Punch Flag, PF, which indicates when a new character can be loaded into T. The instructions for the transmitter section are:

$SI_j \leftarrow A$ load the transmitter register T_j from A<8:15>, clear PF, (j = 1,2)

IF PF, "Label"   if the Punch Flag (PF) is true, then GOTO location called "Label" (i.e., PC is set to address "Label")

The receiver section contains an 8-bit register, R<0:7> which holds the character while it is being received from a keyboard or the attached paper tape reader. The keyboard flag, KF, indicates when a new character has arrived in R. The instructions for the receiver section are:

$A \leftarrow SI_j$ load the receiver register into A, clear KF, (j = 1,2)

$TAPE_j$ start the paper tape reader and have it read 1 character

IF KF, "Label"   if the Keyboard Flag (KF) is true, then go to location "Label"

The PDP-16/M is also wired with a simple interface which allows it to control many of the PDP-11 peripherals. The instructions used are:

$A \leftarrow DATI$ DATA from the PDP-11 peripheral addressed by GPI1 is read into A

$DATO \leftarrow A$ DATA in the A register is transmitted to the PDP-11 peripheral addressed by GPI1

* Optional

APPLICATIONS

The PDP-16/M is applicable to small digital systems designs which would possibly be hardwired and for larger systems designs which might require a mini-computer. PDP-16/M has the program capability to adapt, yet many desirable hardwired characteristics for easy interfacing. In general, applications tend to be where a minicomputer may be too costly and a strictly hardwired design is too time-consuming (costly). Also, systems tend to have a small amount of read-write memory. PDP-16 systems have been applied in the following areas—and the PDP-16/M promises to be even more cost-effective solution to similar problems. Since there is a ROM control there is no need for costly consoles and equipment to load programs once the system is designed.

COMMUNICATIONS

Many communications systems have related problems requiring little or no memory. The PDP-16/M is cost-effective in the following areas:

Time-division-multiplexors. Multiple communications lines are multiplexed on a single (faster) line.

Time-division-switching. Multiple communications lines can be switched arbitrarily by time sampling of the basic digital waveform and retransmitting. Systems for switching multiple Teletype or alpha-numeric terminals can be switched among multiple computers and line speed inputs.

Remote sampling and data transmission. Remote (unattended) sites can be sensed (e.g., analog) and data transmitted to a central site.

Telephone digit checking. A PDP-16/M can check to see if a particular phone is calling within its restricted area.

"Smart-terminal" control. Multiple terminals are interconnected, controlled, and their data trans-mitted on a time-multiplexed basis. The flexibility of PDP-16/M allows functional specialization of the terminals.

CONTROL

Continuous (sampled) and discrete control systems find applicable areas. In many applications the information about the behavior can also be analyzed and recorded or transmitted.

Conventional plant (sampled) closed loop. A single computer can control a large number of loops.

Crane control. Switch input conditions are exam-ined and the crane is moved accordingly via controls outputs to stepping motors and motor contactors.

Conveyor control. The status (items) of a conveyor is held in PDP-16/M's memory, and at appropriate times, gates are opened to transfer the items to appropriate storage bins.

Transfer machine control.

EDUCATION

Students build systems using microprogramming techniques. The very simple interface illustrates principles without tiresome detail required in large minicomputer interfacing.

Computer interfacing. Most interface logic is present, but other interfaces can be designed.

Computer microprogramming. The structure of the 16/M is similar to that of microprogrammed machines. Students are free to develop interpreters for instruction sets.

Computer programming. The microprogram-program interface can be explored.

Hardware-software interface. Additional, more complex, hardwired instructions can be added for greater speed.

APPLICATIONS

DATA LOGGING, WAVEFORM ANALYSIS, WAVEFORM SYNTHESIS

The PDP-16/M provides far more flexibility than hard wired data loggers.

Data logging and alarm scanning. Analog inputs are compared with preset values, and out-of-range alarm conditions can be printed out or transmitted via communications lines.

Waveform data logging. Input (sampled) data is analyzed to compute average, rate of change, high, low, and histogram in the same way that a conventional data logger would be used.

Data reduction. Time sampling and transmission of of information of waveform reconstruction information.

Digital filtering and signal processing. Fourier transform, auto and cross-correlation.

DATA FORMAT AND MEDIA CONVERSION

Data in one form (e.g., 8-bit BCD) or media (e.g., 6-hole paper tape) is converted to another format and/or media.

Typesetting tape conversion. Data in one code and format is converted to another.

Paper tape to magnetic tape.

Sampled analog to paper or magnetic tape.

Card to magnetic tape.

Magnetic tape to plotter. By specifying only lines and character a large data reduction is possible.

COMPUTER INTERFACING WITH PRE AND POST PROCESSING OF INFORMATION

The 16/M provides two capabilities which can be beneficial for these applications: a simple interface; and processing. The former allows systems to be operational earlier than with hardwired logic. By having programming capabilities much pre and post processing can be carried out within the interface prior to placing final data at the main computer's access.

SCIENTIFIC EXPERIMENTATION

Many of the previous applications arise in this area: control, data logging and analysis, communication, etc. Because of the low cost, many scientific instruments can now be computer interfaced.

Experiment control.

Data logging, analysis and recording.

Data conversion. Providing access to other computers without tying up a minicomputer.

Computer interfacing. Interfacing capability to existing minicomputers.

SERVICE, TRAINING AND SUPPORT

DISCOUNT
An attractive discount policy offers substantial reductions in the already low price of the PDP-16/M.

TRAINING
DEC recognizes the importance of training both the OEM's employees and customers. To this end, DEC offers meaningful training courses in Maynard, Mass., at the OEM's plant or at his customer's facilities.

OEM SERVICES
The OEM marketing groups at DEC provide a variety of services to the OEM customer:
—Arranging seminars to the OEM sales force
—Promoting the OEM's products and services in the *Digital News* (circulated to over 20,000 computer users).
—Assisting the OEM in the preparation of articles and releases for publication in the trade media.
—Providing brochures and promotional literature on DEC equipment for the OEM's marketing effort.
—Referring DEC customers to OEM's who can provide a system to meet customer requirements.
—Providing special support to the OEM at trade shows.

FIELD SERVICES
With over 900 Field Service Engineers located in service centers throughout the world, DEC has the largest maintenance organization among minicomputer manufacturers and is able to offer the user complete service support virtually anywhere in the world. Service agreements are very flexible and enable the user to tailor DEC maintenance to his special needs and those of his customer:
—Contract coverage may be selected in one-month increments.
—Unused portions of the coverage may be passed on to an OEM's customers.
—Reinstallation of systems and/or addition of peripherals may be contracted for a fixed cost.

Those who prefer to perform their own maintenance may take advantage of DEC's ECO-LOG. This service keeps the user up-to-date on the latest revisions and engineering developments both with equipment and in service techniques.

DOCUMENTATION
DEC publishes a wide variety of technical literature on the PDP-16/M as well as other DEC products. *The PDP-16 Computer Designer's Handbook—Second Edition* contains extensive information of interest to the functional computer user. Copies are available from all local Digital Equipment Corporation Sales Offices.
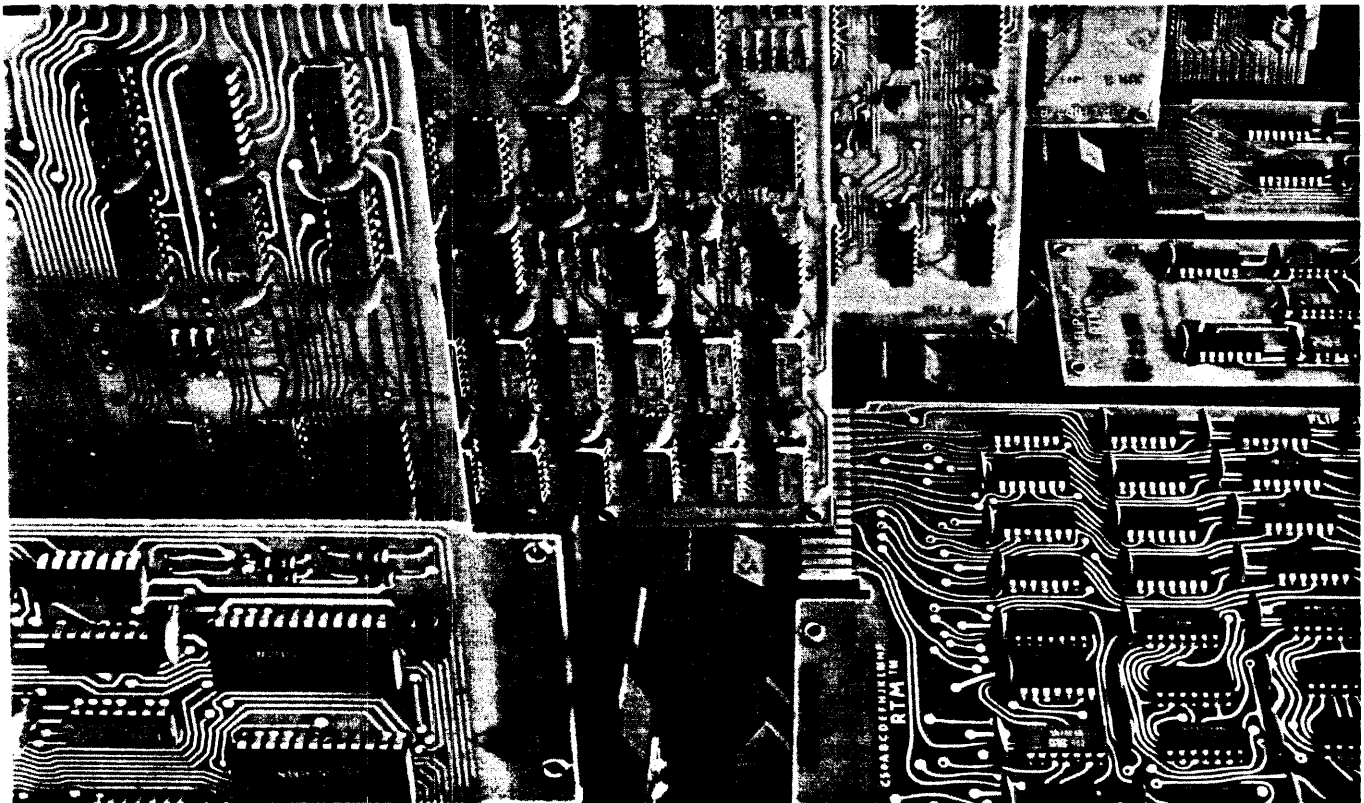
PDP-16/M SPECIFICATIONS

- Low cost: $1995 with discounts approaching 40%
- Word length: 16-bit (8-bit instructions)
- Applications-oriented: solves problems for which minicomputers may be uneconomical—a true subminicomputer
- Low systems design costs: a factor of 5-10 faster than hardwiring
- Modular and flexible: constructed from proven PDP-16 Modules, with a wide range of prewired options, extensions via the PDP-16 Bus using PDP-11 and PDP-16 modules.
- Reliable: constructed from PDP-16 components; programs are fixed in read only memory providing reliability of hardwired systems
- Software: programs assembled and simulated on a PDP-8 in high-level assembly language
- Instructions times: 1.4 to 3 microseconds
- Logic: TTL (ground and + 3 volts)
- Environment: 0 to 70° C; 20 to 95% relative humidity, non-condensing
- Mounting: 10-1/2" (high) X 19" (wide) X 14" (deep) with slides for RETMA 19" rack; weight approximately 35 lbs.
- Power: 95 to 130 volts, single phase, 47 to 63 Hz., 3.0 amps; 190 to 260 volts, single phase, 47 to 63 Hz., 1.5 amps

- Instructions: 113 basic with expansion to 253 in five types—register-transfer, unconditional and conditional branch, subroutine call and subroutine exit
- Arithmetic and Logic: 16-bit data words using two arithmetic accumulators (A and B) with LINK and Overflow (OVF) bits
- Interface (input-output)
  - via standard computer:
    3 16-bit word-parallel full-duplex channels to 0.5 mHz;
    2 serial (8-bit) standard asynchronous full-duplex communications channels to 2400 Baud
    22 input conditions can be sensed
  - via standard PDP-16 options: additional parallel and serial interfaces and new product options via standard PDP-11 options: paper tape, printers plotters, displays, etc.
  - via custom designed PDP-16 compatible modules: PDP-16 bus (21-wire, fully-interlocked) is straightforward and inexpensive to use
- Memory for program: 256 to 1024 8-bit Reprogrammable/ Read Only Memory
  for 16-bit data: 3 to 6 bits (flags), 1 to 33 registers, 280 ROM constants, 256, 512 and/or 1024 word read-write arrays