INTERDATA®

PROCESSORS

Model 8/32 Processor

PRODUCT DESCRIPTION

The Model 8/32, the second in the Interdata series of medium-scale 32-bit processors, is specifically designed for high performance applications. The 8/32 attains high performance through its basic architectural characteristics:

- Full 32-bit parallel architecture
- High speed Schottky logic
- Four-way interleaved memory
- Powerful instruction lookahead stacks
- Memory access controller
- Dual bus structure with fast direct memory access
- Four-level interrupt system
- Multiple register stacks

The Model 8/32, implemented in Schottky T^2 L logic, removes the bottlenecks of earlier minicomputer designs. It offers full 32-bit parallel architecture: 32-bit interleaved memory paths, 32-bit arithmetic unit, and 32-bit data paths.

The 8/32 uses the same 750-nanosecond core memory as the Interdata 7/32, yet it operates at an effective cycle time of 300 nanoseconds. Speed is achieved through dual 64-bit instruction lookahead stacks and four-way interleaved memory. Each stack consists of four 16-bit registers.

Interleaved memory takes advantage of the fact that each 16-bit wide bank can run independently of the other memory banks. Two banks operate in parallel to produce a 32-bit wide access. The memory controller begins a read cycle in both pairs of memory banks simultaneously and without waiting for restoration of the read-out data, transfers the two words of data in sequence at an interval of about 100 nanoseconds. The effective instruction cycle for the 32-bit word is 300 nanoseconds. Each memory bank consists of 32K bytes of core. Thus a minimum system of four 32KB banks provides 128K bytes of memory.

The CPU executes variable length instructions in varying execution times. Demand for memory constantly changes. In most computers, this causes the memory to wait for the CPU or the CPU to wait for the memory.

The Interdata 8/32 eliminates this delay with the two lookahead stacks that act as high-speed dual buffers, allowing the CPU and memory to run largely in parallel. While the CPU executes instructions from one 64-bit lookahead stack, memory loads the other stack.

In addition to the lookahead stacks, the CPU has two standard stacks of sixteen 32-bit general registers for rapid context switching among user and operating system programs.

The memory access controller (MAC) interposes sixteen registers between the CPU and memory and provides hardware segmentation, program relocation, and protection. MAC (in conjunction with the operating system) assumes the responsibility for placing a user program in core and protecting it.

The interrupt system includes 1023 vectored interrupts and four user implemented interrupt levels.

The strength of the input/output system lies in its dual bus architecture. Each bus deals with one of the major kinds of I/O a computer performs: either man/machine or machine/machine communication.

The man/machine channel is the Multiplexor Bus. It supports 1023 slow-to-medium speed devices, such as Teletype, alphnumeric CRT, card reader, and printer.

The Extended DMA (EDMA) bus is used for high speed machine/machine devices such as disc, mag tape, and CPU-CPU links. Standard devices interface to the EDMA via either an ESELCH (Extended Selector Channels) or a BSELCH (Buffered Selector Channel). Custom-designed devices interface to EDMA via the Universal Interface.

OPTIONAL FEATURES

Powerful optional features increase the Model 8/32 performance:

- Additional 128KB core modules to one megabyte maximum.
- Writable control store (WCS)
- High-performance floating point hardware
- High-performance data handling instructions
- Six additional sets of sixteen 32-bit general register
- Extended memory selector channel (ESELCH)
- Buffered selector channel (BSELCH)

Memory can be expanded in increments of 128KB to a maximum of 1,048,576 bytes. The CPU architecture allows it to address the whole megabyte directly.

Writable Control Store (WCS) is a processor enhancement for custom-tailoring the system to a specific application. Up to 512 words, 32-bits each, can implement macros, pseudo operation codes, or instructions in microcode to improve execution times by a factor of two to three.

The High Performance Floating Point Option implements both single and double precision floating point arithmetic. Single precision numbers consists of six hexadecimal digits, equivalent to 7.1 decimal digits. Double precision numbers are made up of 14 hexadecimal digits, equivalent to 16.7 decimal digits. Floating point arithmetic is extremely fast. For example, the execution time for a double precision floating point multiply is 2.5 microseconds.

The High Performance Data Handling Option provides facilities to compute the polynomial for the error check redundancy character used by most data communications protocols. The option includes a memory-to-memory byte string move with character translation and cumulative checksum calculation on either the data bytes or the translated bytes.

The six additional sets of general purpose registers allow the user to dedicate a complete set of sixteen 32-bit registers to frequently-run application programs. In a multiprogramming environment, the dedicated general purpose registers need not be stored and restored as the CPU switches control from one program to another.

The ESELCH provides selector channel facilities for up to 16 device controllers to transfer data directly between main memory and high-speed devices. Maximum transfer rate is two million bytes per second. Data can be transferred in either the byte or halfword mode.

BSELCH is a buffered selector channel that provides direct memory access to one million bytes of main memory. Effective buffer size varies between 4 and 32 bytes depending on the desired DMA transfer rate which can vary from 2.58 to 5.85 million bytes per second. BSELCH transfers data between its buffer and main memory in bursts of up to 28 bytes.

The Model 8/32 supports a total of 7 selector channels.

CENTRAL PROCESSOR UNIT

Figure 1 shows a simplified block diagram of the Model 8/32. All interconnecting data paths are 32 bits wide. Only the Multiplexor and EDMA busses are 16 bits wide to maintain compatibility with all other Interdata computers.

The two standard sets of 16 registers provide 32-bit wide general purpose registers. All can be used as accumulators. Fifteen registers in each set can be used for indexing addresses.

The dual lookahead stacks consist of four 16-bit wide registers. Each stack can hold up to four halfword instructions, two fullword instructions, or one and one-third 48-bit instructions. A fullword or 48-bit instruction can cross a stack boundary.

As soon as the first halfword in a lookahead stack is transferred into the instruction register, the other lookahead stack is loaded from memory. Thus, the CPU waits for an instruction to be read from memory only when a Branch occurs.

The processor can be in either the STOP or RUN Mode. In STOP mode, the operator has total control of the processor and can examine or modify any register or memory location including the Program Status Word. In addition, the operator can execute one instruction at a time.

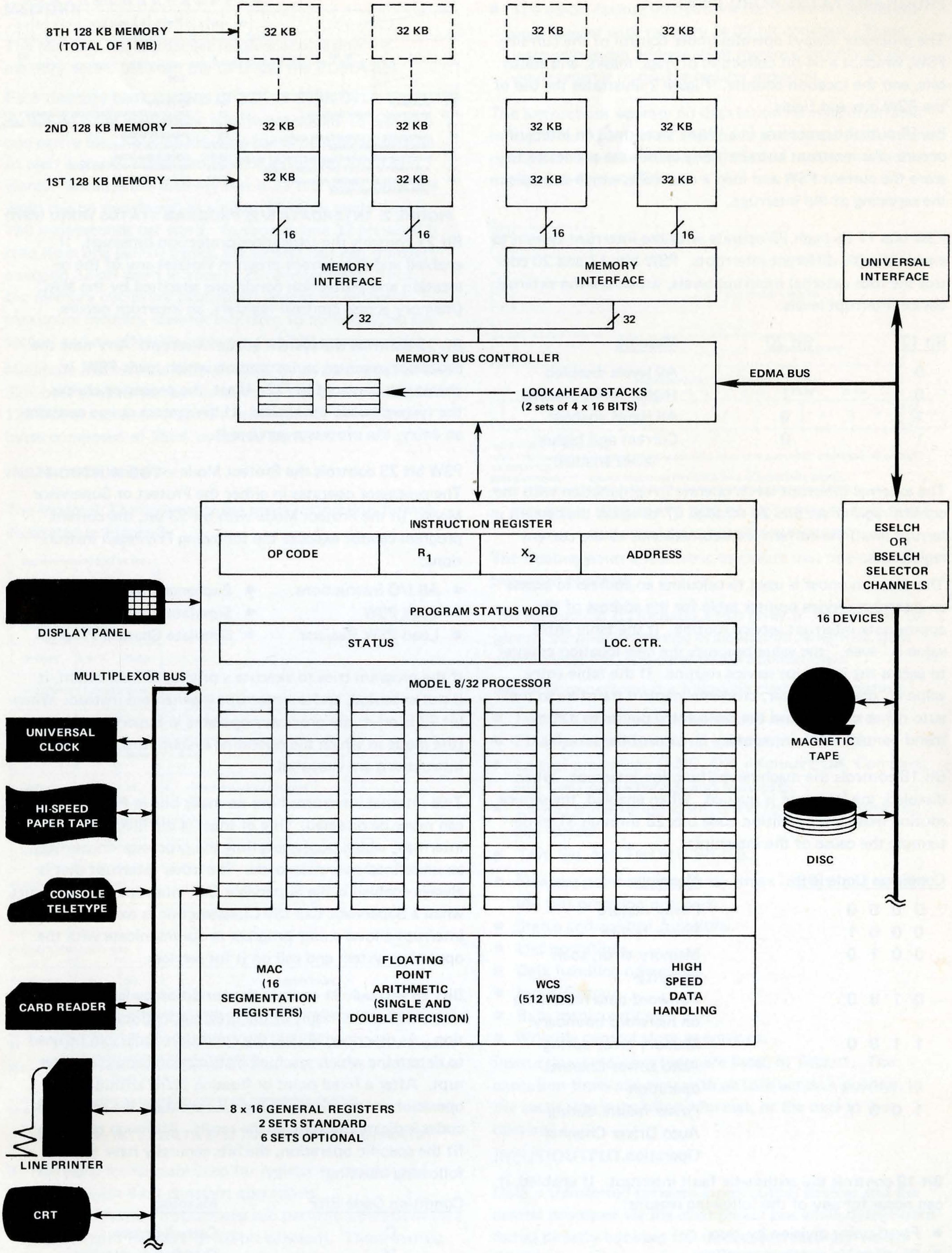


FIGURE 1. INTERDATA MODEL 8/32 BLOCK DIAGRAM

PROGRAM STATUS WORD (PSW)

The processor always operates under control of the current PSW, which is a 64-bit collection of flags, masks, and status bits, and the location counter. Figure 2 illustrates the use of the PSW bits and fields.

Bit 16 puts the processor in a WAIT state until an interrupt occurs. An interrupt automatically causes the processor to store the current PSW and load a new PSW, which controls the servicing of the interrupt.

PSW bits 17 through 23 operate with the interrupt system to enable/disable different interrupts. PSW bits 17 and 20 control the four external interrupt levels, which are the external device interrupt levels:

Bit 17	Bit 20	Meaning
0	0	All levels disabled
0	1	Higher levels enabled
1	0	All levels enabled
1	0	Current and higher levels enabled

The external interrupt levels operate in conjunction with the current register set bits 24 through 27 to select the current interrupt level; the current level corresponds to the current register set.

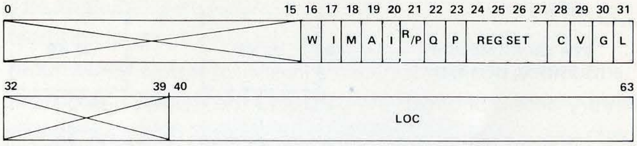
The device number is used to calculate an address to access an interrupt service pointer table for the address of the appropriate interrupt service routine. If the table entry value is "even," the value becomes the new location counter to access the interrupt service routine. If the table entry value is "odd," however, processor control transfers to the auto driver channel and the table entry points to a command control block in memory to control the transfer.

Bit 18 controls the machine malfunction interrupt. When disabled, the interrupt is queued. When enabled, the service routine must test condition code bits 28 through 31 to determine the cause of the interrupt:

Condition Code Bits	Meaning
0 0 0 0	Power restore
0 0 0 1	Power failure
0 0 1 0	Memory error, such as parity
0 1 0 0	Fullword data read/write on halfword boundary
1 1 0 0	Memory error during Auto Driver Channel operation
1 0 0 1	Power failure during Auto Driver Channel Operation.

Bit 19 controls the arithmetic fault interrupt. If enabled, it can occur for any of the following reasons:

- Fixed-point division by zero
- Fixed-point quotient overflow
- Floating-point division by zero
- Floating-point overflow or underflow



BIT	MEANING	BIT	MEANING
0-15	RESERVED – MUST BE ZERO	21	RELOCATION/PROTECTION INTERRUPT MASK
16	WAIT STATE	22	SYSTEM QUEUE SERVICE INTERRUPT MASK
17	IMMEDIATE INTERRUPT AND AUTO DRIVER CHANNEL MASK	23	PROTECT MODE
18	MACHINE MALFUNCTION INTERRUPT MASK	24-27	REGISTER SET SELECT
19	ARITHMETIC FAULT INTERRUPT MASK	28-31	CONDITION CODE
20	IMMEDIATE INTERRUPT AND AUTO DRIVER CHANNEL MASK	32-39	RESERVED – MUST BE ZERO
		40-63	LOCATION COUNTER

FIGURE 2. INTERDATA 8/32 PROGRAM STATUS WORD (PSW)

Bit 21 controls the relocation/protection interrupt. If enabled and the current program violates any of the relocation and protection conditions specified by the MAC (memory access control) registers, an interrupt occurs.

Bit 22 controls the system queue interrupt. Any time the processor executes an instruction which loads PSW, it checks the status of bit 22. If set, the processor checks the system queue for entries. If the system queue contains an entry, the processor services it.

PSW bit 23 controls the Protect Mode violation interrupt. The processor operates in either the Protect or Supervisor Mode. In the Protect Mode with bit 23 set, the current program cannot execute the following Privileged instructions:

- All I/O instructions
- Load PSW
- Load PSW Register
- Exchange PSW Register
- Simulate Interrupt
- Simulate Channel Program

If the program tries to execute a privileged instruction, it is not executed, and an interrupt is generated instead. When bit 23 is reset, the processor operates in Supervisor Mode (the mode in which the operating system runs), and all instructions are executed.

Two internal interrupts have no mask bits in PSW and thus, can never be disabled. One of these is the illegal instruction interrupt, which occurs anytime the processor encounters an undefined operation code. The other interrupt that is always enabled is the Supervisor Call interrupt, which occurs when a Supervisor Call (SVC) instruction is executed. This interrupt allows a user program to communicate with the operating system and call on it for services.

Bits 28 through 31 contain the condition codes, which are set to reflect the result of the previous instruction or operation. As described above, the condition codes can be used to determine which machine malfunction caused an interrupt. After a fixed point or floating point arithmetic operation, comparison, or logical operation the condition codes indicate the state of the result. Although coded to fit the specific operation, the bits generally have the following meaning:

Condition Code Bits	Meaning
C	Carry/borrow
V	Overflow/underflow
G	Greater than zero
L	Less than zero

MEMORY

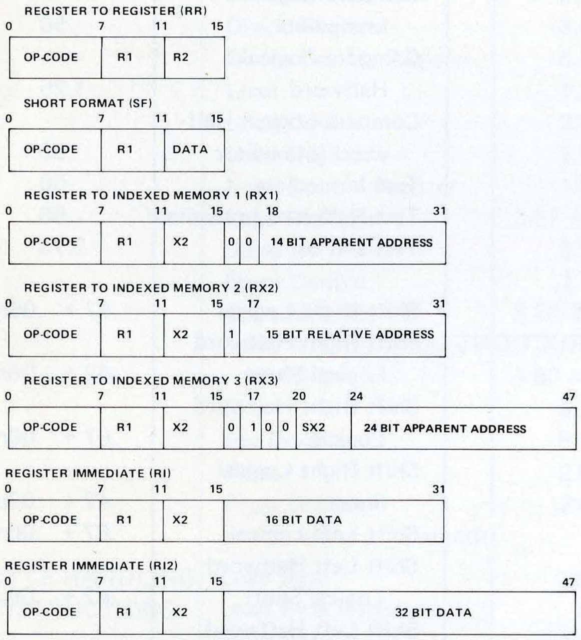
The Memory Bus Controller resolves contention for memory access between the CPU and the EDMA bus.

Each memory bank consists of 32K to 256K bytes organized as 16K to 128K halfwords; a halfword equals 16 bits plus one parity bit. Four 16-bit wide banks operate in parallel so each memory access can read or write two full 32-bit words. Because the memory bus is 32 bits wide, only one word can be transferred at a time. Memory cycle time is 750 nanoseconds per word. As soon as one 32-bit word is read from one pair of banks, the second pair can transfer a second 32-bit word in about 125 nanoseconds. Thus, the memory cycles are overlapped producing an overall maximum memory transfer rate over 10 million bytes per second and an effective cycle time of 300 nanoseconds.

Minimum memory consists of 128K bytes composed of one 32K-byte module per bank. Memory can be expanded in 128K-byte increments up to a maximum of one million bytes composed of 256K bytes per bank.

INSTRUCTION SET

The Model 8/32 implements the seven instruction formats illustrated in Figure 3.



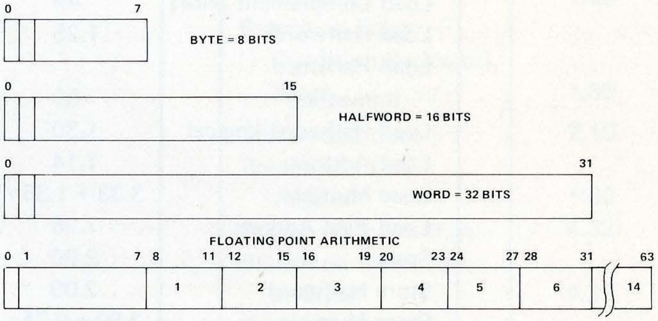
OP-CODE	HEXADECIMAL REPRESENTATION OF FUNCTION TO BE PERFORMED (ADD, MULTIPLY, ETC.)
R1	SPECIFIES ONE OF 16 G.P. REGISTERS AS A FIRST OPERAND.
R2	SPECIFIES ONE OF 16 G.P. REGISTERS AS A SECOND OPERAND.
X2	SPECIFIES ONE OF 15 G.P. REGISTERS AS AN INDEX VALUE (ADD TO APPARENT ADDRESS FIELD TO OBTAIN TRUE VALUE OF ADDRESS).
SX2	SPECIFIES ONE OF 15 G.P. REGISTERS AS A SECOND INDEX VALUE (TO BE ADDED TO SUM OF ADDRESS + FIRST INDEX VALUE).

A "RELATIVE" ADDRESS IS CALCULATED BY ADDING THE INCREMENTED LOCATION COUNTER TO THE SUM OF THE APPARENT ADDRESS AND THE INDEX VALUE.

FIGURE 3. INTERDATA 8/32 INSTRUCTION FORMATS

- Halfword formats are used for register-to-register and register with 4-bit constant operations.
- Fullword format instructions can perform operations on a register with memory or 16-bit constant. These instructions can address directly the first 16K bytes of memory and 16K bytes relative to the location counter. All of memory can be addressed via single-register indexing.
- The 48-bit format instructions can perform operations on a register with memory or 32-bit constant. These instructions can address all of memory directly or via single-register or double-register indexing.

The instructions operate on data in the formats illustrated in Figure 4. Fixed point arithmetic and logical instructions operate on words or halfwords. Bit and byte instructions manipulate bits and bytes.



BIT 0 = SIGN OF FRACTION
BITS 1 THROUGH 7 = EXPONENT OF HEXADECIMAL FRACTION; EXPONENT EXPRESSED IN EXCESS 64 NOTATION.
BITS 8 THROUGH 31 = SINGLE PRECISION FRACTION OF 6 HEXADECIMAL DIGITS.
BITS 8 THROUGH 63 = DOUBLE PRECISION FRACTION OF 14 HEXADECIMAL DIGITS.

FIGURE 4. INTERDATA 8/32 DATA FORMATS FOR FIXED AND FLOATING POINT ARITHMETIC

The floating-point arithmetic hardware uses one-word or two-word operands.

The instruction set includes a rich array of instructions for general purpose processing in simulation, scientific, data communications, and commercial applications.

Instructions perform the following classes of operations:

- Load/Store halfwords, words and multiple words.
- Fixed-point arithmetic on halfwords and words.
- Logical operations (AND, OR, eXclusive OR, Compare, and Test) on halfwords and fullwords.
- Logical and arithmetic shifts and rotates on halfwords and words.
- Test, Set, and Test and Set bits.
- Floating-point arithmetic on single (32 bit) and double (64 bit) precision operands.
- Status and control functions.
- List operations.
- Data handling operations.
- Input/Output.
- Byte manipulations.
- Writable control store operations.

Instruction execution times are listed in Table 1. The execution times can vary with an instruction's position in the cache, the instruction's format, or the data in the operand.

INPUT/OUTPUT

Data is transferred between input/output devices and the central processor via the multiplexor bus while data is transferred directly between I/O devices and memory via the EDMA bus. High speed devices require either an ESELCH (Extended Selector Channel) or BSELCH (Buffered Selector Channel) to transfer data over the EDMA bus.

TABLE 1. INTERDATA MODEL 8/32 INSTRUCTION EXECUTION TIMES

TYPE	INSTRUCTION	EXECUTION TIME IN USEC*
LOAD AND STORE INSTRUCTIONS	Load	1.25
	Load Register	.41
	Load Immediate	.50
	Load Immediate Short	.41
	Load Complement Short	.49
	Load Halfword	1.25
	Load Halfword Immediate	.50
	Load Halfword Logical	1.30
	Load Address	1.14
	Load Multiple	3.33 + 1.35n
	Load Real Address	7.96
	Store	2.00
	Store Halfword	2.00
	Store Multiple	2.50 + 0.75n
	Exchange Halfword Register	1.83
	(n = number of registers)	
FIXED POINT ARITHMETIC INSTRUCTIONS	Add	1.25
	Add Register	.41
	Add Immediate	.50
	Add Immediate Short	.41
	Add Halfword	1.25
	Add Halfword Immediate	.50
	Add to Memory	2.00
	Add Halfword to Memory	2.60
	Subtract	1.25
	Subtract Register	.41
	Subtract Immediate	.50
	Subtract Immediate Short	.41
	Subtract Halfword	1.25
	Subtract Halfword Immediate	.50
	Compare	1.74
	Compare Register	1.13
	Compare Immediate	1.38
	Compare Halfword	1.74
	Compare Halfword Immediate	.50
	Multiply	3.58
	Multiply Register	2.76
	Multiply Halfword	4.69
	Multiple Halfword Register	5.90
	Divide	6.00
	Divide Register	5.65
	Divide Halfword	8.61
	Divide Halfword Register	9.58
LOGICAL INSTRUCTIONS	AND	1.25
	AND Register	.41
	AND Immediate	.50
	AND Halfword	1.25
	AND Halfword Immediate	.50
	OR	1.25
	OR Register	.41
	OR Immediate	.50
	OR Halfword	1.25
	OR Halfword Immediate	.50
	Exclusive OR	1.25
	Exclusive OR Register	.41
	Exclusive OR Immediate	.50
	Exclusive OR Halfword	1.25
	Exclusive OR Halfword Immediate	.50
	Compare Logical	1.25
	Compare Logical Register	.41
	Compare Logical Immediate	.50
	Compare Logical Halfword	1.25
	Compare Logical Halfword Immediate	.50
	Test Immediate	.50
	Test Halfword Immediate	.50
	Test and Set	2.73
SHIFT INSTRUCTIONS	Shift Right Logical	.67 + .06n
	Shift Right Halfword Logical Short	.42 + .06n
	Shift Right Halfword Logical	.67 + .06n
	Shift Right Logical Short	.42 + .06n
	Shift Left Logical	.67 + .06n
	Shift Left Halfword Logical Short	.42 + .06n
	Shift Left Halfword Logical	.67 + .06n
	Shift Left Logical Short	.42 + .06n
	Shift Right Arithmetic	.67 + .06n
	Shift Right Halfword Arithmetic	.67 + .06n
	Shift Left Arithmetic	.67 + .06n
	Shift Left Halfword Arithmetic	.67 + .06n
	Rotate Right Logical	.67 + .06n
	Rotate Left Logical	.67 + .06n
	(n = number of bits)	

TABLE 1. INTERDATA MODEL 8/32 INSTRUCTION EXECUTION TIMES (Continued)

TYPE	INSTRUCTION	EXECUTION TIME IN USEC*
FLOATING POINT INSTRUCTIONS	Add	1.82
	Add Double	3.38
	Add Register	1.00
	Add Double Register	1.04
	Subtract	1.82
	Subtract Double	3.38
	Subtract Register	1.00
	Subtract Double Register	1.04
	Compare	1.45
	Compare Double	3.00
	Compare Register	.60
	Compare Double Register	.60
	Multiply	2.50
	Multiply Double	4.90
	Multiply Register	1.75
	Multiply Double Register	2.50
	Divide	4.45
	Divide Double	9.20
	Divide Register	3.60
	Divide Double Register	6.70
	Load	1.39
	Load Double	2.91
	Load Register	1.04
	Load Double Register	1.04
	Load Multiple	3.57 + 1.34n
	Store	2.23
	Store Double	2.75
	Store Multiple	3.59 + 0.90n
	Store Multiple Double Precision	4.50 + 1.80n
	Fix	5.35
	Fix Double	8.10
	Float	2.00
	Float Double	2.00
	(n = number of operands)	
BYTE HANDLING INSTRUCTIONS	Load Byte	2.50
	Load Byte Register	.41
	Store Byte	2.62
	Store Byte Register	1.17
	Exchange Byte Register	1.17
	Compare Logical Byte	2.65
BRANCH INSTRUCTIONS	Branch on True Condition	1.95
	Branch on True Condition Register	1.42
	Branch on True Condition Forward Short	1.90
	Branch on True Condition Backward Short	1.85
	Branch on False Condition	1.95
BRANCH INSTRUCTIONS (Continued)	Branch on False Condition Register	1.42
	Branch on False Condition Forward Short	1.85
	Branch on False Condition Backward Short	1.85
	Branch and Link	2.10
	Branch and Link Register	1.60
	Branch on Index High	4.32
	Branch on Index Low or Equal	4.32
	Convert to Halfword Value Register	2.64
COMMUNICATIONS INSTRUCTIONS	Cyclic Redundancy Check 12	6.94
	Cyclic Redundancy Check 16	7.68
	Translate	2.68
	Simulate Channel Program	8.36
BIT MANIPULATION INSTRUCTIONS	Test Bit	3.68
	Set Bit	4.51
	Reset Bit	4.75
	Complement Bit	4.51
STATUS AND CONTROL INSTRUCTIONS	Load Program Status Word	4.65
	Load Program Status Word Register	2.98
	Exchange Program Status Register	2.74
	Supervisor Call	5.00
LIST HANDLING INSTRUCTIONS	Add to Top of List	9.80
	Add to Bottom of List	10.00
	Remove From Top of List	11.05
	Remove From Bottom of List	9.01
DATA HANDLING INSTRUCTIONS	Process Byte	8.08
	Process Byte Register	6.77
	Move and Process Byte String Register	12.43n
	(n = number of bytes)	

TABLE 1. INTERDATA MODEL 8/32 INSTRUCTION EXECUTION TIMES (Continued)

TYPE	INSTRUCTION	EXECUTION TIME IN USEC*
INPUT/OUTPUT INSTRUCTIONS	Autoload	6.45 + 1.86L + 2.66n
	Simulate Interrupt	6.44
	Read Data	3.90
	Read Data Register	1.57
	Read Halfword	3.32
	Read Halfword Register	1.58
	Read Block	9.01
	Read Block Register	3.31 + 2.66n
	Write Data	3.33
	Write Data Register	1.73
	Write Halfword	3.31
	Write Halfword Register	1.73
	Write Block	4.92 + 2.76n
	Write Block Register	3.35 + 2.76n
	Sense Status	4.11
	Sense Status Register	1.76
	Output Command	4.25
	Output Command Register	2.70
	(n = number of data bytes) (L = number of Leader bytes)	
WRITABLE CONTROL STORE INSTRUCTIONS	Branch to Control Store	1.28
	Enter Control Store	1.00
	Write to Control Store	2.46 + 2.17n
	Read from Control Store	2.94 + 1.32n
	(n = number of words)	

*Minimal time (Partial listing)

MULTIPLEXOR BUS

Input/output transfers over the multiplexor bus can occur in several different ways:

- One byte or one halfword between an external device and any general purpose register or memory location under control of an I/O instruction.
- A block of data between a device and memory in burst mode under direct processor control.
- Multiplexed blocks of data between devices and memory under control of the auto driver channel.

External devices interface to the multiplexor bus via controllers. Each controller connects to one of the four Model 8/32 interrupt lines; line 0 has highest priority and line 3 the lowest. When multiple devices connect to a line, relative distances from the processor determine priorities. Up to 1023 devices can interface to the multiplexor bus.

Device requests for service access an interrupt service pointer table. If the pointer address is even, the processor transfers control to a software routine that transfers a byte or halfword of data between a device and a general purpose register or a memory location. If the pointer address is odd, the processor transfers control to the auto driver channel, the pointer addresses a channel command block (CCB) in memory. CCB contains the control parameters: function (read or write), describes two buffers, specifies translation of characters, designates calculation of longitudional and cyclic redundancy check values, and contains transfer address for error subroutine. The auto driver channel is implemented in firmware.

EXTENDED DMA (EDMA) BUS

Blocks of data are transferred between memory and high-speed devices over the EDMA bus under control of the extended selector channel (ESELCH), buffered selector channel (BSELCH) or EDMA Universal Interface.

Data are transferred over ESELCH one halfword at a time for a maximum transfer rate of two megabytes per second. The BSELCH can buffer up to 16 halfwords of data, and it can transfer data between a device and memory in bursts of up to 14 halfwords. Maximum aggregate transfer rate is 5.9 megabytes per second. The size of the BSELCH buffer is optional for a system. For most applications, optimal throughput will be obtained using data transfers in bursts of 8 halfwords for an aggregate transfer rate of 5.05 (read) or 5.67 (write) megabytes per second.

The EDMA Universal Interface contains the minimum logic to insure the integrity of the 8/32 EDMA bus. Integrated circuits (up to 113) can be added to implement custom designed interfaces. Maximum transfer rate in burst mode via this interface is 6 megabytes per second.

Two additional major interfaces are available for connecting optional components to the EDMA bus: Memory Access Multiplexor (MAM) to service data communications lines or character-oriented devices, and Multiport Memory Interface (MMI) to allow multiple processors to share memory.

MAM makes Extended Direct Memory Access (EDMA) block transfer facilities available to as many as 63 half-duplex devices or 31 full-duplex devices. Maximum throughput rate is 195K bytes per second for byte-oriented devices and 375K bytes per second for halfword-oriented devices. Up to seven MAM's can interface to one Model 8/32.

MMI provides access to a multiport memory system of 32KB to 1MB by up to 14 processors or device controllers. Maximum throughput is up to 2.4MB per second per memory bank in the multiport memory system, which can support up to eight memory banks for aggregate throughput of 19.2MB per second.

MEMORY ACCESS CONTROLLER (MAC)

MAC monitors all programs running in user mode and translates all memory addresses. It relocates 20-bit logical addresses into 20-bit physical addresses and detects invalid addresses. It prevents execution of instructions from specified blocks of memory, and it denies write access to certain areas of memory.

MAC can divide programs into up to 16 segments and relocate segments on 256-byte boundaries. A segment can vary in size from 256 bytes to 64K bytes.

The operating system runs in supervisor mode with MAC turned off, thus it addresses physical memory directly. EDMA I/O operations also address physical memory directly.

WRITABLE CONTROL STORE (WCS) OPTION

The WCS option provides up to 512 words (32 bits each) of control store to allow users to tailor a Model 8/32 to a specific application. WCS is a processor enhancement which can improve overall processor performance by 20% to 80% depending on code utilization. The more the macros or algorithms implemented in microcode and stored in WCS depend on registers and CPU functions, the more performance is improved.

A microassembler with debug utilities allows users to develop microprograms in a higher level language under the OS/32MT operating system. A runtime system maintains WCS during power fail/auto restart.

The WCS option includes four instructions: Enter Control Store, Read Control Store, Write Control Store, and Branch in Control Store. Only Enter Control Store is available to the user program. The other instructions are privileged, and the operating system uses them to maintain and control WCS.

HIGH PERFORMANCE FLOATING POINT OPTION

The high performance floating point option includes 34 instructions to perform both single and double precision floating point arithmetic. Operations also include Compare, Fix and Float. Fix converts a floating point operand into an integer format. Float converts an integer into a floating point operand format. Data formats are shown in Figure 4.

Instruction execution times are listed in Table 1.

HIGH PERFORMANCE DATA HANDLING OPTION

The high performance data handling option is primarily used for data communications applications. It provides three instructions: Process Byte; Process Byte Register; and Move and Process Byte String Register.

Process Byte and Process Byte Register calculate a cumulative checksum based on an old checksum and a data byte. The cumulative check can be for BISYNC, SDLC (Synchronous Data Link Control), or any other protocol. Longitudinal or Cyclic Redundancy Checking (LRC, CRC) options can be elected.

The Move and Process Byte String Register instruction uses a block of six registers to define the operation. A string of data can be moved from one space in memory to another. In transit, each data character can be translated in accordance with a translation table and a cumulative checksum can be calculated.

In addition to its data communications applications, this instruction can also be used for a byte string move.

SOFTWARE

The Model 8/32 runs under control of OS/32MT, an event-driven operating system, which provides concurrent support for foreground multitasking application programs and background batch oriented program development.

Foreground tasks are scheduled according to four types of events: operator request, hardware interrupt, timer event, or inter-task event.

The background is provided primarily for on-line program development. Only one background program can be in the system at a time.

OS/32MT requires a Model 8/32 Processor with 128KB of memory, Display Panel, Power Fail/Auto Restart, Memory Access Controller, Operator Command Console, Universal Clock, and a cassette or 9-track magnetic tape unit, or 10MB disc.

LANGUAGES

Program development languages include the most popular languages for commercial, scientific, and simulation program development.

MACRO CAL

This optimizing assembly language has a powerful macro-preprocessor. CAL (Common Assembly Language) is cross compatible with Interdata 16-bit and 32-bit computers. Users can write programs in CAL to run on any Interdata computer.

FORTRAN V

The FORTRAN V compiler supports all the features of ANSI FORTRAN Standard X3.9-1966. In addition, the Purdue Instrument Society of America (ISA) real-time extensions are supported, including bit and byte manipulation features. The compiler permits device-independent I/O on a wide range of media.

FORTRAN V is useful when memory space is at a premium. It runs under OS/32MT in 96KB of memory.

FORTRAN VI

FORTRAN VI is a block optimizing compiler system, providing efficient generation of machine code from FORTRAN source. The only language difference between FORTRAN V and VI is that FORTRAN V has address variables and VI does not.

FORTRAN VI operates under OS/32MT Revision 2 (or higher) on 8/32 with 60KB of memory over and above that required by the operating system. At least one disc volume is needed for scratch files and overlays. The operating system must include support for single and double precision floating point.

COBOL WITH ISAM

Interdata's implementation of COBOL conforms to American National Standard X3.23-1974. It was developed in an IBM-like environment to make full use of the power of the Interdata's 32-bit architecture and to maximize transportability and compatibility with existing compilers.

COBOL is packaged with ISAM (Indexed Sequential Access Method), an enhanced file management system designed to meet the data processing requirements of a wide range of commercial data base applications. ISAM utilities provide for allocation and deallocation of up to 32 contiguous files on each of 32 different disc volumes. COBOL is used as the data description and data manipulation language.
COBOL with ISAM runs under OS/32MT on a Model 8/32 Processor equipped with a console device, 10 megabytes of disc storage, and a minimum of 160KB of main storage.

BASIC 32

BASIC 32 utilizes Interdata's Basic II, a powerful, easily learned, interactive programming language. It uses the conventions of Dartmouth's BASIC but offers significant enhancements for both the experienced and novice programmer.

Multi-user support is provided by establishing one copy of BASIC in a reentrant library partition and allowing any number of user programs to share this one core-resident copy of BASIC II.

The time-slice feature of OS/32MT provides true time-sharing operations.

The BASIC interpreter requires a Model 8/32 configuration to support OS/32MT plus 19KB of memory and a minimum of 1KB per user.

COMMUNICATIONS

Interdata Telecommunications Access Method (ITAM/32) is a complete data communications framework for implementing an application. It provides support at two user levels:

- Structured – for complete compatibility with defined systems. ITAM handles communications conventions automatically.
- Customized – for complete design flexibility in developing a specialized communications system.

ITAM/32 can be used to develop applications for a 2780/3780 RJE expansion environment, data concentration, front-end off loading, message switching, and packet switching.

The ITAM/32 system operates under control of OS/32MT and requires the following minimum system configuration:

- Model 8/32 Processor with 128KB memory
- Operator Console Panel
- Universal Clock
- OS/32MT Software System
- ITAM/32 Software Package
- Console Device
- Dual Magnetic Storage Media
- System Load Device
- Appropriate Communication Line Adapters

SPECIFICATIONS

Technology

Processor	Schottky T^{2}L, MSI & LSI
Read only Memory	50ns BiPolar LSI
Main Memory	750ns core
Cache Memory	50ns BiPolar RAM
Writable Control Store	50ns BiPolar RAM

PROCESSOR

Instruction Length	16, 32, 48 bits
Data Length	1, 8, 16, 32, 64 bits
Hardware Registers	8 sets of sixteen 32-bit general registers 1 set of eight 32-bit floating point registers 1 set of eight 64-bit double precision floating point registers 1 set of eight 32-bit micro registers 1 set of sixteen 32-bit program relocation registers
Addressing	Direct to 1,048,576 bytes Relative to ± 16,384 bytes Single and double indexing to 1,048,576 bytes Absolute to 16,384 bytes.
Arithmetic	Two's complement (fixed point ± 2,147,483,648); Sign/magnitude (floating point -5.4 x 10^{-79} to 7.2 x 10^{75})
User Instruction Repertoire	173 instructions total including optional 17 S.P. floating point, 17 D.P. floating point, 3 data handling, and 4 writable control store instructions.
Micro Instruction Repertoire	163 instructions
Data Paths	32-bits
Instruction lookahead	2 sets of 64-bit lookahead stacks (4 16-bit registers per stack)

Input/Output Characteristics	Maximum Number of I/O Devices — 1023 Number of Auto Driver Channels — 1023 Number of EDMA Ports — 7
I/O Transfer rates	Block transfer 387K Bytes/sec Auto Driver Channel 64K Transfers/sec (8 or 16-bit transfers) Memory Access Multiplexor (MAM) 195K bytes/sec (byte devices) 375K bytes/sec (halfword devices) EDMA Transfer 2.0M bytes/sec halfword mode via ESELCH; 2.4M bytes/sec halfword via Memory Multiplexor Interface (MMI); 5.9M bytes/sec via BSELCH; 6.0M bytes/sec (burst mode)
Priority Interrupts	Four levels with automatic device identification and vectoring
Interrupt Response Time	5.9usec
Avg. Interrupt Latency Time	2usec
Hardware I/O Timeout	25usec timeout on all micro processor operations to the I/O system to ensure the processor will not lockup due to I/O device failure.

BUILT-IN MAINTENANCE FEATURES

- Memory module failure isolation
- Control store address match stop, and set
- Processor clock control
- Arithmetic clock control
- Processor state displays

MAIN MEMORY

Type	750nsec core
Data Length/Access	16, 32 or 64 bits
Data Paths	32 bits
Parity	Optional, 1 bit per 16 data bits
Organization	64 bits interleaved 32 x 2
Memory Size	131,072 bytes to 1,048,576 bytes

PACKAGING

- 70" (177.8 cm) high x 24" (61 cm) wide x 30" (76.2 cm) deep system cabinet, forced air cooled, 300 CFM
- 19" (48.3 cm) RETMA chassis, 14" (35.6 cm) high, 28" (71.1 cm) deep, 16 slots
- 15" (38.1 cm) x 15" multiwire printed circuit boards. with ¼" (0.64 cm) stiffeners packaged horizontally.
- Printed circuit backpanel
- Pin to dual contact receptacle connections with locating pins.
- Basic system includes 8 processor boards and 4 memory boards, plus system cabinet, power supplies, binary display panel, power fail/auto restart, and current loop interface to a TTY.
- Power requirements 115 or 230 VAC ± 10%, 50 or 60 Hertz, 32 amps max (at 115 volts)

ENVIRONMENTAL

Temperature	0–50°C
Humidity	0–90% (non-condensing)
Vibration	5–55 CPS at 1.25 G's
Storage Temperature	–55°C to +85°C

PROCESSOR OPTIONS

- Additional General Register Sets (2 sets in the Basic System)
- Single Precision and Double Precision Floating Hardware
- Parity
- Writable Control Store
- High Performance Data Handling

INTERDATA PRODUCT NUMBERS

M83-025	Model 8/32C Processor with 128KB of memory
M83-103	High Speed Data Handling Option
M83-107	Processor/Memory Parity Generation and Checking
M83-108	Writable Control Store
M83-111	High Performance Floating Point

The information contained herein is intended to be a general description and is subject to change with product enhancement.

Printed in U.S.A. March, 1977

INTERDATA
A UNIT OF
PERKIN ELMER DATA SYSTEMS
2 CRESCENT PLACE ● OCEANPORT, NEW JERSEY 07757