

November 1978

16-bit Single-board Computer Maintains 8-bit Family Ties

By Robert Garrow, Jim Johnson
and Les Soltesz
Electronics

16-bit single-board computer maintains 8-bit family ties

Three-bus 8086-based board addresses a megabyte, communicates over expanded system bus

by Robert Garrow, Jim Johnson, and Les Soltesz, *Intel Corp., Santa Clara, Calif.*

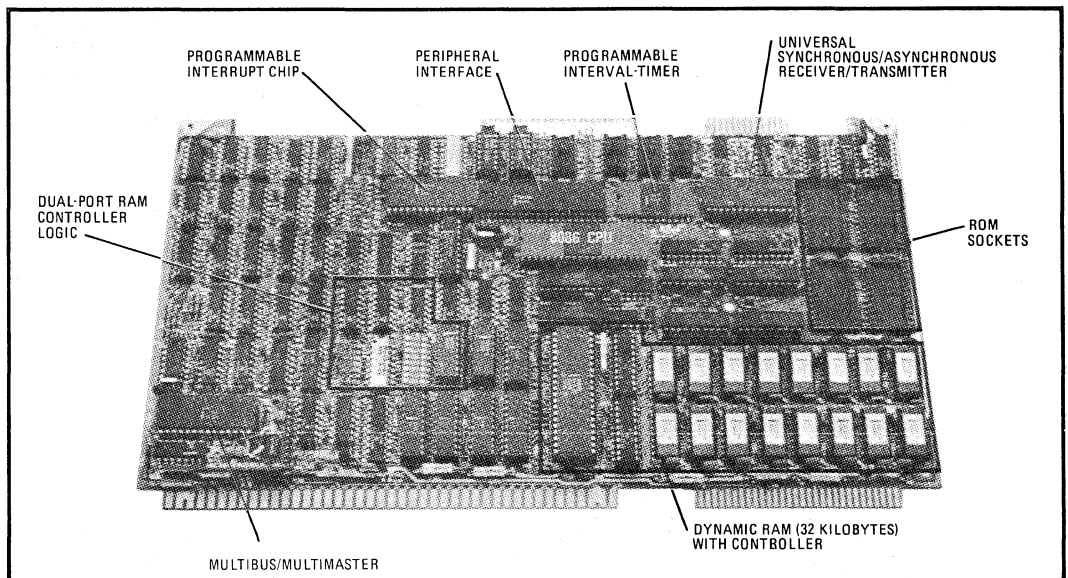
□ For the first time ever, 8- and 16-bit single-board computers can brainstorm over the same system bus. The iSBC 86/12 16-bit SBC has been designed to work intimately with its predecessors, the iSBC 80 family of 8-bit boards. What's in it for the user? Design flexibility—8-bit designs can be enhanced to 16 bits, developed software can be transported and, beyond that, 8- and 16-bit devices can be mixed in multiprocessing configurations. Several features make these options possible: a 16-bit CPU and instruction set designed for 8-bit compatibility; greatly expanded memory resources; and an extension of the Multibus specifications.

At the heart of the iSBC 86/12 is a 16-bit, high-performance metal-oxide-semiconductor 8086 central processing unit that operates at 5 megahertz. Because the 8086 instruction set is a superset to that of both the 8080A and 8085A 8-bit processors, the CPU can execute the full set of 8080A/8085A-type 8-bit instructions plus

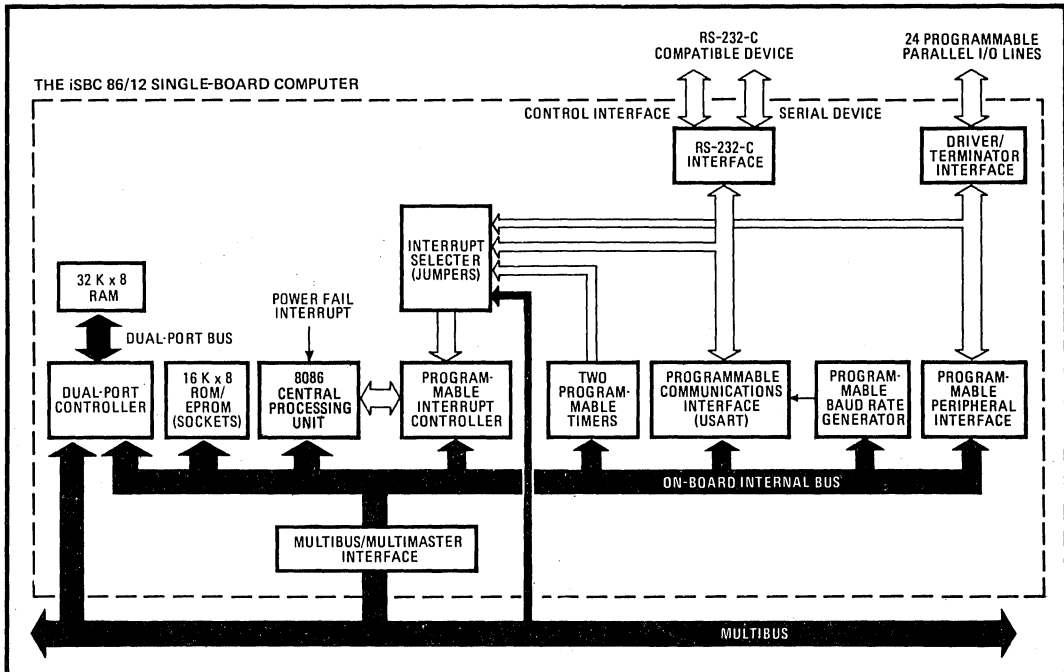
a new set of 16-bit instructions. Thus, programs generated for 8-bit-CPU systems can easily be upgraded to run on the iSBC 86/12 using the software tools available with the Intel microcomputer development system. Programs written in Intel's high-level programming language, PL/M, can be executed on both iSBC 80 and iSBC 86 products, preserving the software investment in 8-bit systems as a user moves into 16-bit applications.

Other features of the 8086 CPU are signed 8- and 16-bit arithmetic (including multiply and divide), efficient interruptible byte-string operations, and improved bit manipulation. Furthermore, the 8086 provides mechanisms for reentrant code, position-independent code, and dynamically relocatable programs.

This enhanced processing power is supported by the largest memory ever offered on a CPU board (Fig. 1). Memory address space has been extended over the iSBC 80 series to one million bytes. Up to 16 kilobytes of



1. What a board. The iSBC 86/12 has 32 kilobytes of RAM and room for 16 kilobytes of ROM. The 5-MHz 8086 CPU executes 8080A/8085A-type as well as 16-bit instructions, including multiply and divide. Address space has been increased to a megabyte.



2. LSI + SBC = 86/12. A number of programmable LSI devices take credit for the power and flexibility of the iSBC 86/12. Note their interconnection to the three-bus hierarchy. When the 8086 requests a resource, the system bus is used only as a last resort.

read-only memory can be installed on the iSBC 86/12 itself. Furthermore, an additional 32 kilobytes of dynamic random-access memory with on-board refresh may be accessed independently by the CPU or by the system bus (Multibus).

Like the iSBC 80/30, the 86/12's RAM has dual ports to extend its use off board for access by other Multibus masters, including single-board computers, direct-memory-access devices, and peripheral controllers [Electronics, Aug. 17, p. 109]. All memory operations on the board occur independently of the Multibus, freeing it for external parallel operations. For applications that require data integrity at all times, a separate bus supplies power to the RAM and support logic via the edge connector. An auxiliary power source energizes the RAM in the event of power failure.

Multibus—the new look

To exploit the greater performance of the 8086 CPU and simultaneously make the iSBC 86/12 fully compatible with the iSBC 80 family of SBCs and expansion products, the Multibus specification has been extended to support 20 bits of address and 16 bits of data. The control lines, too, have been expanded to direct 8- and 16-bit data transfer over the system bus. These improvements enable the iSBC 86/12 to address directly a full one megabyte of system memory, access data in 8- or 16-bit word lengths, and recognize and acknowledge a variety of interrupts.

Address space has been enlarged to 1 megabyte by adding four address lines, A_{10} – A_{13} . Next, 8- and 16-bit

data operations have been defined to permit both types in the same system. This is done by reorganizing the memory modules, adding one new signal and redefining another. The memory is divided into two 8-bit data banks, which form a single 16-bit word. The banks are organized such that all even-byte-addressed data is in one bank (D_0 – D_7) and all odd-byte-addressed data is in the other bank (D_8 – D_{15}). A new bus-address signal has been defined to control the odd-byte bank called byte high enable (BHEN) during 16-bit operations. When active, BHEN enables the high byte of the data word from the addressed boards on the D_8 – D_{15} Multibus data lines. A_0 controls the even byte bank and, when inactive, enables the low byte of the data word on the D_0 – D_7 Multibus data lines. All word operations must occur on an even-byte-address boundary with BHEN active for maximum efficiency. (A_0 is inactive for all even addresses—see the table.) Word operations on odd-byte boundaries will be converted to 2-byte operations by the 8086, one for low-byte, one for high-byte. Byte operations can occur in one of two ways. The even bank is accessed when BHEN and A_0 are both low. This puts the data on D_0 – D_7 . To access the odd bank (normally placed on D_8 – D_{15} during a word operation), a new data path has been defined. The active state of A_0 and the inactive state of BHEN are used to enable a swap-byte buffer, which places the odd data bank on D_0 – D_7 . This permits an 8-bit master access to both bytes of the data word while controlling only A_0 . A_0 therefore specifies a unique byte and is not part of the word address, since all word operations are on even-byte boundaries.

Flexibility: LSI chips are the key

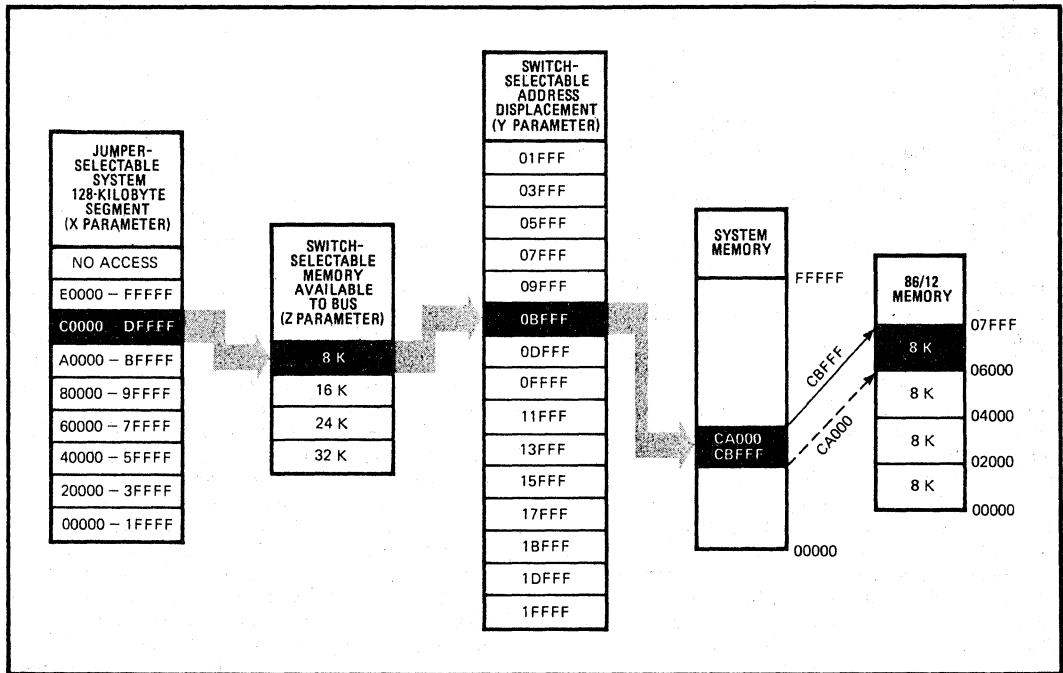
The iSBC 86/12 owes much of its flexibility to programmable large-scale integrated devices. An 8255A peripheral interface chip provides 24 programmable I/O lines that may be tailored to the customer's needs by simply programming the device for input, output, or bidirectional modes with or without handshaking abilities. In conjunction with the 8255A's configuration the user may then select appropriate line drivers and terminators for the I/O lines that can be inserted into sockets on the iSBC 86/12 board.

An 8251A universal synchronous/asynchronous receiver/transmitter is included to provide an RS-232-C interface for serial communication with other computers, RS-232-C-type peripherals (cassette tape, modems, etc.) or cathode-ray-tube terminals. The 8251A enables the user to customize the communication link. Synchronous/asyn-

chronous mode, data format, control character format, parity and baud rates from 75 to 38.4 kilobauds are all under program control.

For system timing functions an 8253 programmable interval timer provides two programmable timers, each of which may be used as a square-wave generator, retriggerable one-shot multivibrator or as an event counter.

The interrupt structure of the iSBC 86/12 encompasses nine levels with vectored priority. Eight of these levels are handled by an 8259A programmable interrupt controller chip, which may be configured for different priority processing modes in accordance with the application. One nonmaskable interrupt is available to immediately alert the CPU to catastrophes like a power failure, in which case the CPU can branch to an appropriate routine in memory to effect an orderly system shut-down.



3. RAM, please. The 8086's view of on-board memory is fixed from zero to 07FFFFH. When an outside master accesses this space, the DP controller performs the translation. Here, locations 06000H to 07FFFFH are available to another master by addressing CA000H to CBFFFFH.

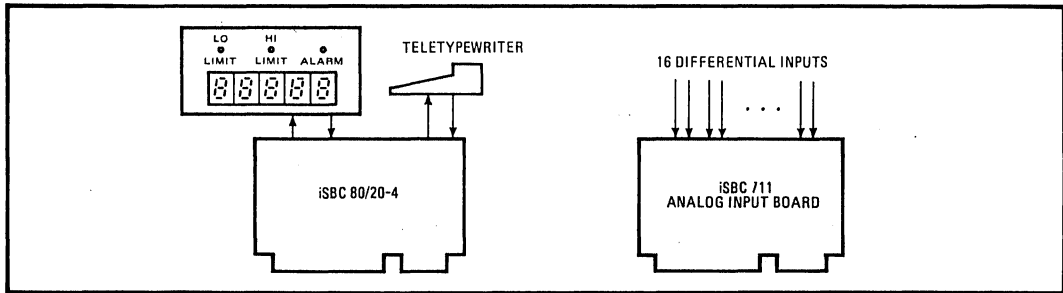
Since all 8-bit accesses via Multibus are done on the lower byte of the data word, the iSBC 86/12 can access 8-bit memory or I/O devices from the system bus. This makes the iSBC 86/12 compatible with all iSBC 80 Multibus modules.

More interrupts, too

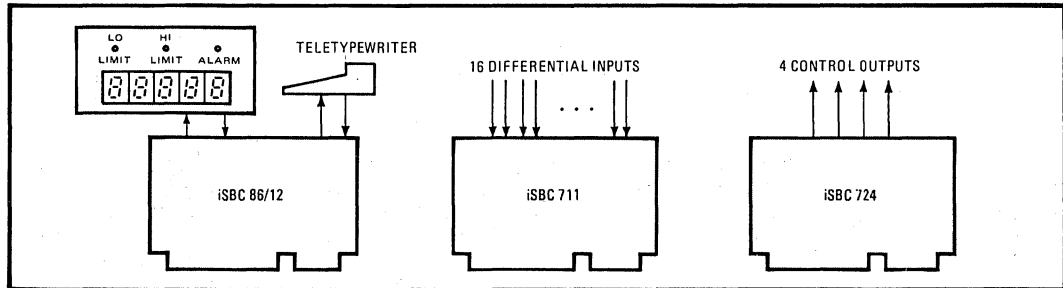
The iSBC 86/12 expands the previous Multibus definition of interrupts by creating two distinct types: non-bus-vectored (NBVI) and bus-vectored (BVI) interrupts. Each Multibus interrupt line can be individually defined

through software to be a BVI or NBVI. Using BVIs, the interrupt capability of a Multibus system can be increased to 64 bus-vectored-priority interrupts.

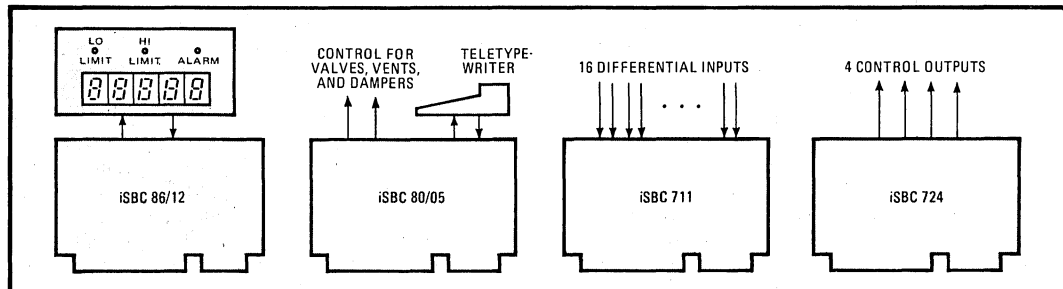
Using NBVIs, a slave module activates an interrupt line and the interrupted bus master generates its own restart address to service that interrupt. The Multibus address or data lines are not used. A BVI uses the Multibus address and data lines to communicate with the interrupting slave. When the slave module generates an interrupt, the bus master requires that module to generate the restart address. One additional command signal is



4. Open loop. Shown above is a simple alarm and monitoring system. The iSBC 711 analog-input board samples 16 differential inputs and the 8-bit iSBC 80/20 compares the inputs to the high and low limits. An alarm condition illuminates an LED and gets logged on a teletypewriter.



5. Closed loop. Suppose the system in Fig. 4 needs to be upgraded to handle a closed-loop system. For this application an iSBC 86/12 replaces the 80/20-04 to cope with the higher processing. The output control variables are handled by an iSBC 724 analog-output board.



6. Multi/master. To enhance the control system in Fig. 5, add a dedicated CPU to control valves, vents, and dampers that, in turn, affect pressure and flow parameters in the system. This has been done by adding an iSBC 80/05 in a Multibus/multimaster arrangement.

defined—interrupt acknowledge (INTA)—to request the restart address from the slave module.

The iSBC 86/12 board architecture, like that of the 8-bit iSBC 80/30, is organized around a three-bus hierarchy: an on-board bus, a dual-port bus and a system bus (Multibus). All three buses have been expanded over their 80/30 counterparts to incorporate 20 address lines and 16 data lines.

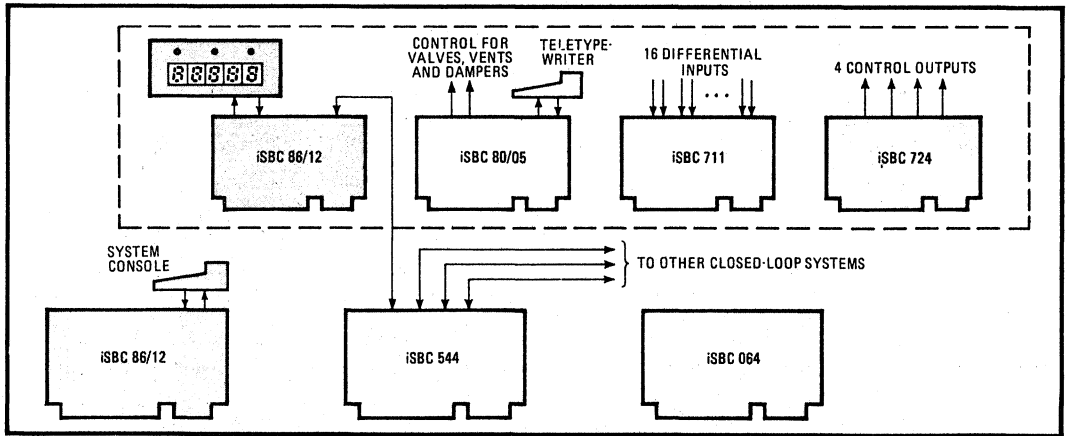
The iSBC 86/12 architecture

The on-board bus links the 8086, all the I/O peripherals, and the read-only memory. Next in the hierarchy is the dual-port bus, which connects to the DP controller, 32 kilobytes of dynamic RAM, and the dynamic RAM controller. Finally, the system bus permits expansion of system resources through Multibus modules (Fig. 2).

The bus protocol of the iSBC 86/12 dictates that each

of the three buses communicate with an adjacent bus or operate independently. When the CPU makes a request for a resource, the on-board and dual-port buses simultaneously determine if their hardware can fulfill the request. If the on-board bus is able to acknowledge the request, it does so and the DP bus is not disturbed. (The DP bus is not interrupted to determine whether it can acknowledge the request.) The 8086 always controls the on-board bus, and requested operations can be completed without delay. If the DP bus is needed, it is requested and the dual-port controller grants the use of the bus to the processor. Thereafter, the dynamic-RAM controller completes the operation and generates an acknowledge.

If neither the on-board nor the DP bus can satisfy the request, the CPU asks for the system bus. The 8086 must use the on-board and dual-port buses to communicate



7. Four loops. An iSBC 86/12 can be used in conjunction with an iSBC 544 intelligent communications controller to perform a supervisory function for four closed-loop systems. The iSBC 544 controls the line protocol and the iSBC 86/12 processes the 544's data.

with the system bus. The 8086 takes control of the DP bus when the system bus is granted. This prevents lock-out problems with the DP bus—that is, when the processor requests the system bus while another bus master has control of it and is accessing the dual-port RAM.

Naturally, the fewer the buses it has to access, the faster the iSBC 86/12 completes a transaction. The on-board bus always operates at maximum board speed. But the DP bus operates at maximum board speed only if it was not busy or taken up with a memory refresh cycle. When the system bus is brought into play, the processor speed depends on the overhead in acquiring it and the type of Multibus module being accessed.

With this three-bus architecture the iSBC 86/12 can be operating over its on-board bus at the same time as another Multibus master is using the system bus. It does so by accessing data from the DP RAM at no reduction in board speed. The on-board bus permits access only from the 8086. Thus all I/O and ROM are private to the 8086.

The dual-port controller has two independent address decoders—one for the 8086 and one for the Multibus. The 8086 decoder fixes the 8086's RAM addresses from hexadecimal 00000 to 07FFF using a fusible-link programmable ROM. The Multibus decoder allows the user to select any address range for the on-board RAM by specifying two parameters—a top-of-memory pointer and the size of the accessible memory. The TOM pointer (as seen by another Multibus master) can be set to any 8-kilobyte boundary in the 1-megabyte memory space. The amount of memory on the iSBC 86/12 accessible by another master can be set to 8, 16, 24, or 32 kilobytes (or no access) with an on-board jumper. For example, fixing the accessible memory size to 24 kilobytes provides the 8086 with 8 kilobytes of RAM that only it can access. This private area can be used for the processor's stack, interrupt jump table and other special system parameters that are generally protected from other Multibus masters. The only addressing restriction is that the memory block accessible to the Multibus cannot cross a 128-kilobyte boundary.

Suppose a Multibus master wants to load a program

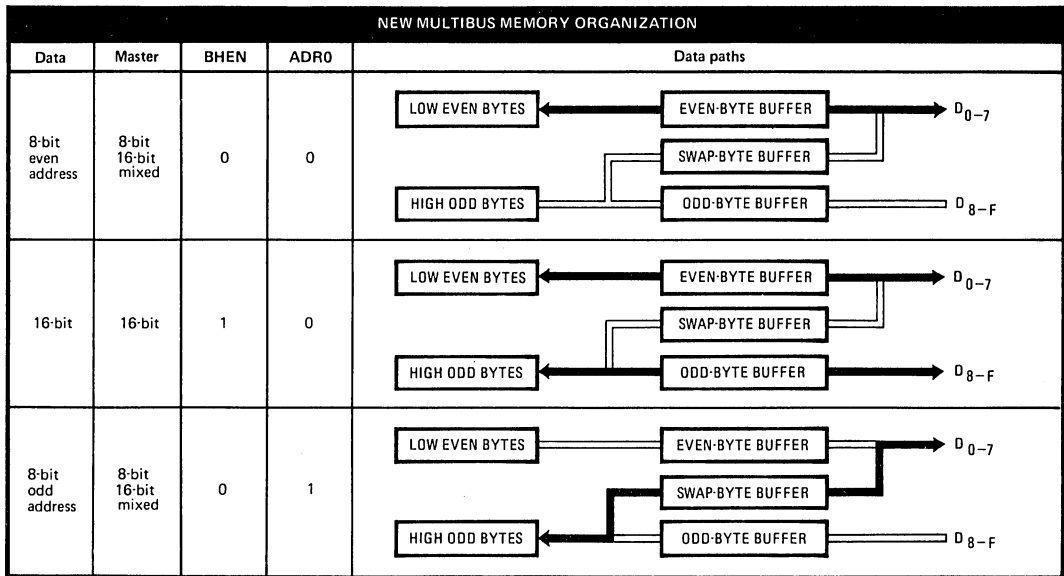
into the iSBC 86/12's dual-port RAM for execution. Since the 8086's view of the DP-RAM address space is fixed, the Multibus address must be translated into the on-board 8086 memory space. The DP controller performs this translation by mapping the TOM pointer (as seen by other Multibus masters) to 8086-address 07FFFH, the top of the 8086's on-board RAM. Pointer-1 is mapped to the top of 8086 on-board RAM-1, and so on.

In the example shown in Fig. 3, the Multibus address selection is divided into three parts—two selecting the TOM pointer (X and Y) and one selecting the size of the accessible memory (Z). The TOM pointer is equal to a 128-kilobyte segment (X) plus address displacement (Y) from that segment. In this example, X is set to C0000H and Y is set to 0BFFFH, so the TOM pointer equals CBFFFH. Next, the size of the accessible memory (Z) is set, in this case to 8 kilobytes. This address translation makes the top 8 kilobytes of the 8086's RAM locations 06000H to 07FFFH available to another Multibus master when it addresses locations CA000H to CBFFFH. The 8086 still has 24 kilobytes (00000H to 05FFFH) of private memory.

Multiprocessing schemes

In multiprocessing systems, a master must be able to access the system without another master obtaining the bus. The iSBC 86/12 incorporates bus-arbitration logic to effect these transactions. Since the system bus is only requested when a system resource is needed, the iSBC 86/12 can perform true parallel processing with other iSBC 80 or 86 masters.

A typical example is the use of a common memory location that contains the status byte (busy/not busy) of a floppy-disk controller. When the floppy disk is needed, the master must first read the location and, if not busy, write the status word without another master obtaining the bus (to use the floppy disk). A bus-lock function on the iSBC 86, once enabled, allows the iSBC 86 to maintain control of the system bus until the lock is disabled by program control. This bus-lock function may



be activated in one of two ways—by an output bit from the resident 8255A peripheral-interface chip or by a software prefix on any 8086 instruction. The iSBC 86 can perform the test and set function by exchanging the accumulator with the memory location, preceding the instruction by a lock prefix. For example, the status word is read into the accumulator and, without another intervening bus cycle, a busy status is written. The accumulator is then tested: if busy—try again (writing a busy does not destroy status as it was already busy); if not busy, the floppy disk is now under the master's control and the status location is set to busy.

The iSBC 86/12: a design tool

For system debugging and full-speed execution, the iSBC 86/12 can be linked to the Intellec microcomputer development system. Programs generated on the Intellec system can be downloaded into the iSBC 86/12 RAM via cables. Through a virtual-terminal capability, the Intellec console can directly access an iSBC-resident monitor, which provides commands for software debug. Once the debugging cycle is completed, the user has the option of uploading the software back to the Intellec for storage on diskettes.

The Multibus and form-factor compatibility of the 8-bit iSBC 80 and 16-bit iSBC 86 single-board computers provide a degree of design flexibility previously unobtainable. Initial design problems can be solved with low-cost 8-bit hardware. As product requirements evolve, 16-bit performance can be added. Eventually, 8- and 16-bit multiprocessor solutions can be conveniently implemented.

Consider the application shown in Fig. 4, an alarm and monitoring system in a typical process plant. Sixteen differential inputs from pressure and flow transducers are sampled once every second, then compared to high and low limits previously entered through thumbwheel

switches. The iSBC 711 analog-input board takes care of sampling the inputs and the 8-bit iSBC 80/20-4 compares the data to the high and low limits. Whenever these limits are exceeded, an alarm LED lights up and the alarm condition is logged on the system teletypewriter along with input identification, high limit, low limit and sampled value.

Closed loops

Instead of an open-loop system, suppose the design must be enhanced to control four output variables—thereby making it a closed-loop system. The sampling rate must be increased to once every third of a second and more processing will be required to run through the control algorithm and output the control-loop data. For this application, an iSBC 86/12 replaces the 80/20-4 to handle the higher processing requirements. An iSBC 724 analog-output board is also added to provide the four output-control variables (see Fig. 5). Carrying this example one step further, one may want to dedicate another processor to controlling valves, vents, and dampers that in turn affect pressure and flow parameters in the system. This can be done by adding an iSBC 80/05 in a multimaster arrangement as shown in Fig. 6.

Finally, an iSBC 86/12 can be used with an iSBC 544 intelligent communication-controller to supervise four closed-loop systems of the type shown in Fig. 6. The 86/12 of each system interfaces with the supervisory system via its serial interfaces, which are connected to the iSBC 544's serial ports (see Fig. 7). The iSBC 544 performs the control functions associated with the line protocol. The supervisory iSBC 86/12 can access the iSBC 544's dual-port memory and can perform further processing of the data received from the four closed-loop systems. In this configuration large amounts of memory may be required; since the iSBC 86/12 can address up to 1 megabyte, this presents no problem. □