AN ADVANCED DISC-BASED SYSTEM

by Michael Busch and Dan Gaines

There comes in the life of all hobby computerists the traumatic moment when the novelty of vanquishing squadrons of Klingons and delving into the mysteries of biorhythm theory begins to wear a little thin. And so it was, some months ago, that the authors began to search for a high-performance low-cost microcomputer system for serious business applications.

BACK TO BASICS

From experience in developing on-line business applications on minicomputer-based business systems, we knew that the key to successful business applications is fast file manipulation, *not* fast CPU performance. Proof of this can be found by looking at the BASIC/FOUR, one of the finest small business systems available in the $40,000+ category. Believe it or not, this minicomputer executes BASIC programs at a speed which is slower than most microprocessor BASIC interpreters, yet its relatively fast disc hardware and its excellent file management software permit the BASIC/FOUR to perform exceptionally well in a business applications environment.

Thus it was clear to us that an 8-bit microprocessor would be entirely adequate for our requirements. We were especially attracted to the Zilog/Mostek Z-80, not because of its high speed (which we did not really need) but because its extensive instruction set permits programs to occupy significantly less memory space than do other MPUs.

We were also convinced that it would be very advantageous to build our system around the S-100 bus, in order to provide maximum flexibility for future growth. To minimize the cost of memory and to eliminate the need for shielding and terminating the bus, it was decided to run the Z-80 at a conservative 2 MHz clock rate. Although a number of good Z-80 CPU boards are available for the S-100 bus, we determined that the one distributed by S. D. Sales Co. (Dallas, TX) was an exceptionally fine design and an outstanding bargain.

Now we faced a most critical decision: the selection of mass storage hardware and associated file management software. Even during our Startrek/Biorythm activities, it had become painfully clear that magnetic tape was a marginal medium even for program storage, and wholly unacceptable for manipulation of data files. On the other extreme, it was apparent that hard disc prices had not yet reached an affordable level. Floppy disc technology seemed to offer the best compromise between cost and performance, so we began to investigate the alternatives available to us in this area.

SORTING OUT THE FLOPPIES

It did not take us long to figure out that a two-drive diskette system was the minimum that we could consider, since a single drive would preclude disc-to-disc copying (for file back-up, etc.). We found that the total cost of a two-drive system which included all of the essentials (drives, controller, power supplies, cabinets, cables, and software) fell consistently in the $2,500 to $3,000 range for standard 8-inch floppies, and around $1,500 for 5.25" minifloppies. The minifloppy, at nearly half the price of others, seemed like quite a bargain until we realized that the mini had only one-third of the capacity of a standard floppy (89K bytes versus 250K to 300K bytes), and that its performance in a random-access application was three to six times slower (see Table 1). While the minifloppy is an excellent vehicle for storing programs and small sequential data files (i.e., a high-speed replacement for cassette tape), we decided that it was not suited to the needs of our applications.

Diskette Drive (Positioner Type)	Min. Seek (ms)	Max. Seek (ms)	Average Latency (ms)	Average Access (ms)	Average Accesses per sec.
PerSci 277 Dual (linear motor)	10	95	83	136	7.4
Shugart SA850 (3ms stepper)	18	243	83	214	4.7
iCOM, Wangco (6ms stepper)	18	468	83	326	3.1
Shugart SA800 (10ms stepper)	18	768	83	476	2.1
Shugart Minifloppy (40ms stepper)	50	1370	100	810	1.2

Notes:
(1) Seek times include settling time.
(2) Average access time = average latency time plus average of min and max seek times.
(3) Average accesses per second = reciprocal of average access
(4) Seek, settling, and latency times obtained from manufacturers' specifications.

Table 1. Performance of floppy disc drives.

In comparing the available 8-inch diskette drives, we were surprised to find such a wide range of performance parameters within such a narrow range of prices. The drives manufactured by PerSci, Inc. (Los Angeles, CA) not only performed much better than any of the other available drives, but also outperformed the newly-announced Shugart SA850 "Fasflex" drive (not yet avail-

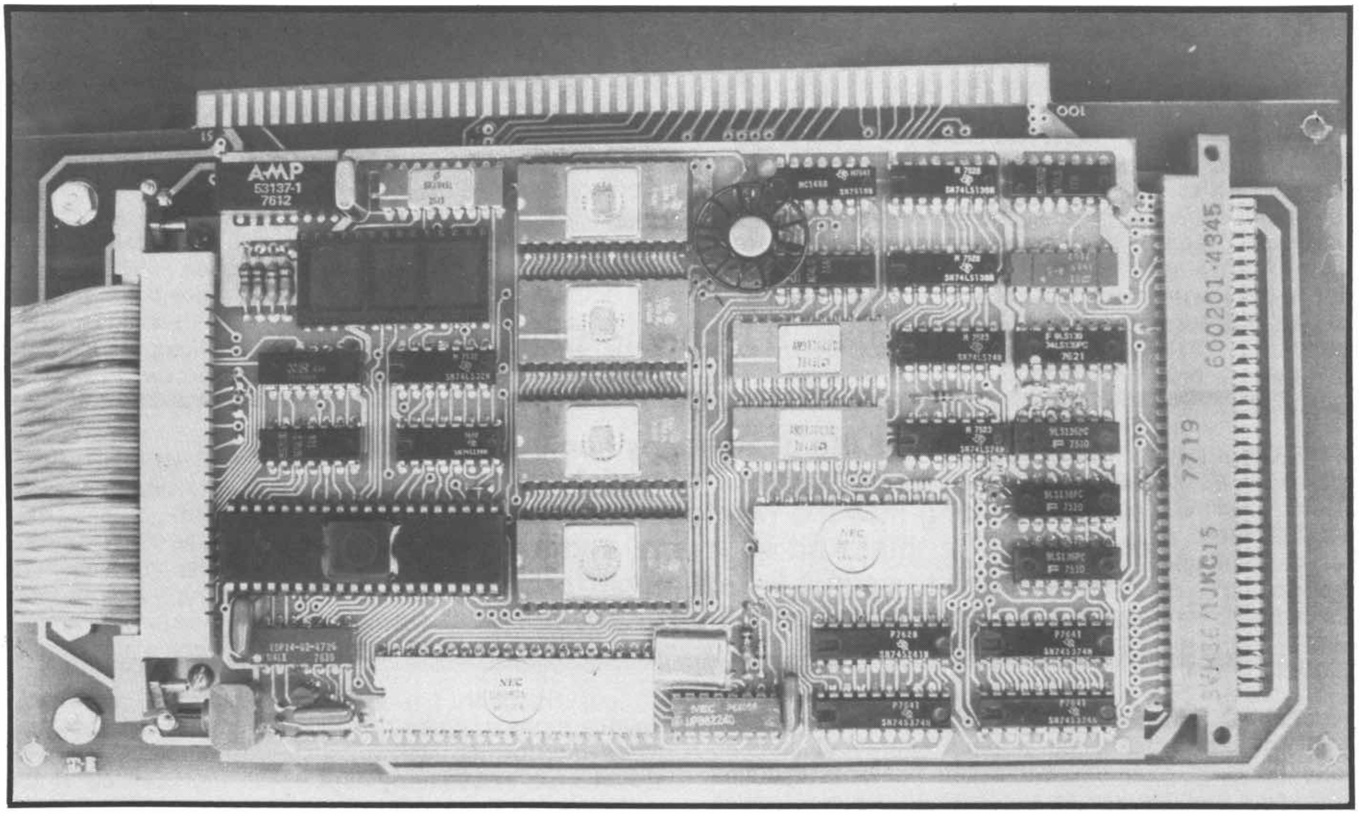
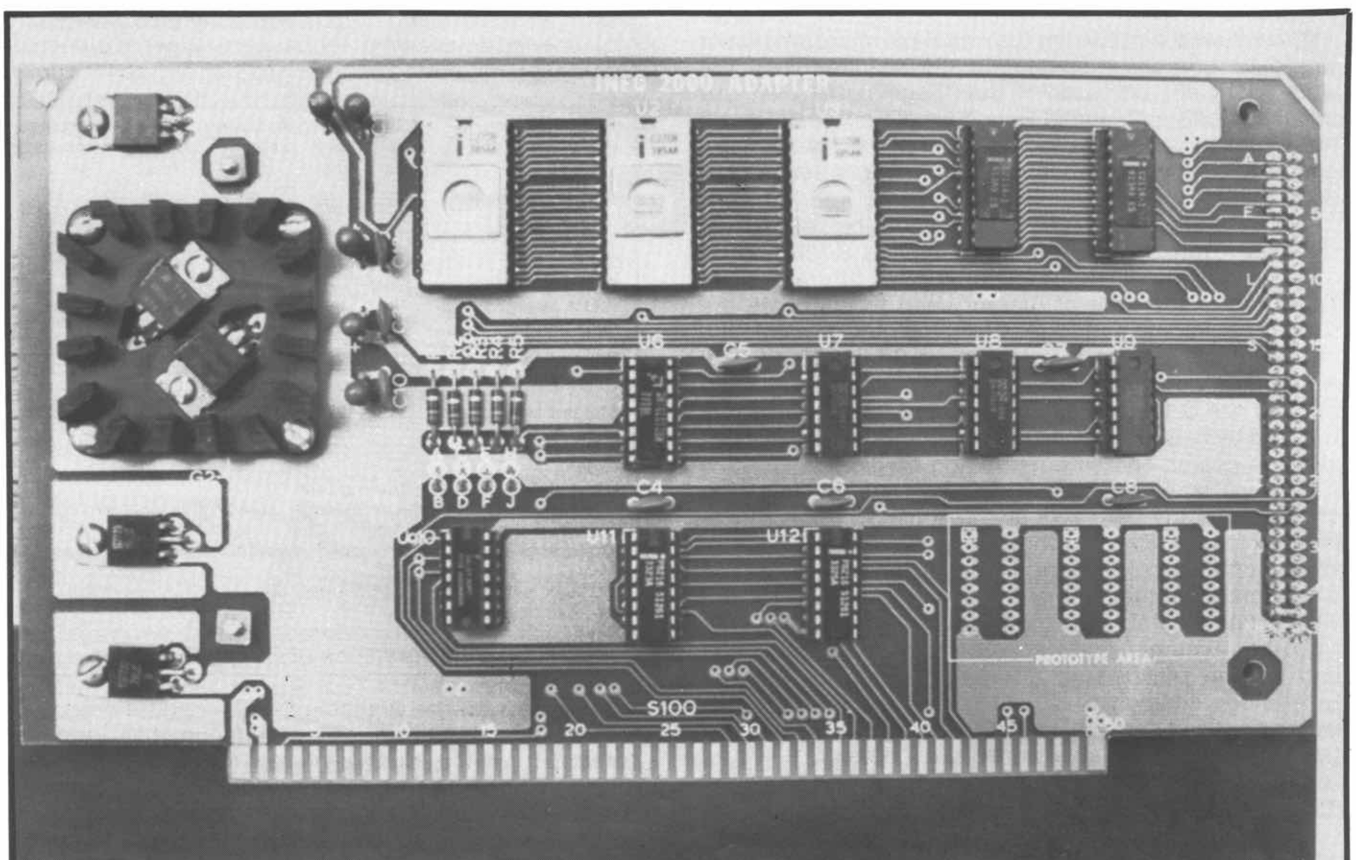


Figure 1. PerSci Model 1070 Intelligent Diskette Controller mounted piggyback on an INFO 2000 Adapter board for the S-100 bus.

Figure 2. INFO 2000 Adapter board mates the PerSci controller to the S-100 bus, includes: 3K of EPROM and 1K of RAM for the Disc Monitor, plus power regulation and interfacing logic for the controller.



able at the time of this writing). We decided to pay a visit to PerSci to find out how they do it. We were exceedingly impressed by what we saw.

STAIRS AND ELEVATORS

Most floppy disc drives, it turned out, use a positioning mechanism consisting of an incremental stepping motor which drives a lead screw along which the record/ playback head travels. The operation of a stepper-type positioner can be likened to running up a flight of stairs. Each seek operation consists of a sequence of track-to-track steps, followed by a settling period during which any head vibration has the chance to die down. Each step requires between six and ten milliseconds depending on the make and model of the drive (the minifloppy requires 40 ms per step), and the settling time is between eight and fifteen milliseconds (the faster the stepper, the longer the settling time). A maximum seek (76 tracks) takes 76 times as long as a minimum seek (1 track) if settling time is not considered.

The maximum speed of a stepper-type positioner is limited by its mechanical inertia. If the stepping motor is driven too fast, it may overshoot the desired track when the stepping pulses are removed. The newly-announced generation of floppy disc drives typified by the Shugart SA850 reduces inertia by doing away with the lead screw. The stepper motor drives the head carriage by means of a capstan and a taut steel band. This mechanism is capable of operating at three milliseconds per step without overshoot.

The PerSci drives are built with an entirely different type of positioner, a miniaturized version of the linear motor (also called "voice coil") positioning mechanisms which are used universally in large-scale hard-disc drives. The operation of this kind of positioner can be compared to that of an elevator in a 77-floor skyscraper. The head carriage in a PerSci drive travels along a track on almost frictionless ball bearings. During a long seek operation, the linear motor accelerates the head carriage to very high speed, and then gradually decelerates it as the desired track is approached. As a result of the smooth deceleration, no additional settling time is required. This mechanism is about twice as fast as a stepper on short seeks, and increases its advantage to as much as eight times on long seeks as indicated in Table 1.

Another feature of PerSci is their Model 277 dual diskette drive, which combines two floppy disc drives into a package the same size as an ordinary single drive. The 277 dual drive shares a single positioner and a single spindle motor between the two floppy discs. Of course, sharing one positioner results in some performance penalty: the PerSci dual drive is not quite as fast as two PerSci single drives (Model 70s), but it will handily outperform two of anybody else's drives in most applications. The dual drive is smaller and less expensive than two singles, and has drastically reduced power consumption (28 watts for one PerSci dual compared with 140-180 watts for two Shugart or Wangco singles). In fact, the device operates quite happily without any cooling fan.

DOUBLE TROUBLE

We next looked into the pros and cons of hard sectoring and double-density recording. Both are techniques for increasing the capacity of a floppy disc.

Hard sectoring makes use of a special type of diskette which has thirty-two sector holes punched in a circle surrounding its large center hole. These are used to define thirty-two 128-byte sectors on each track of the diskette. In contrast, IBM-compatible soft sectoring identifies twenty-six 128-byte sectors per track by means of magnetic headers recorded on each track. While hard sectoring provides six extra sectors on each track, soft sectoring is more flexible, has greater immunity to errors, and offers compatibility with a wide variety of computer systems and data entry devices. Either technique seemed acceptable for our needs.

Ordinary IBM-compatible single-density recording uses a simple technique called "double frequency" or "FM" recording. FM is similar in some respects to the biphase recording technique used by some hobbyist cassette interfaces (e.g., Tarbell) in that it is self-clocking and relatively immune to variations in speed and to media defects. Double-density recording has been successfully accomplished by means of several sophisticated encoding techniques (known as "MFM," "M²M," and "GCR"). Each of these techniques make it possible to record twice as much data on a diskette, but all of them are much less tolerant of speed and alignment errors, dirty heads, marginal diskettes, etc. After soliciting expert advice from several sources, we decided to forego double-density recording on the grounds that we'd rather drive a healthy Ford than a sick Ferrari!

Further to complicate the issue, dual drives and double-density recording has recently been joined by two new notions called double-sided recording and double-track recording (see "Double Talk" box). Double-sided recording was recently demonstrated by Shugart, PerSci, and others at the National Computer Conference. It provides doubled capacity without any of the compromises of double-density recording, and will be available to hobbyists next year. Double-track recording doubles the number of tracks on each disc surface, but it will not be available until mid-1978 and its reliability is not yet well understood.

DOUBLE TALK

Floppy disc manufacturers talk of four distinct methods of doubling the storage capacity of a single drive, and the terminology can get quite confusing:

DUAL DRIVE: Two drives packages in a single unit with certain common components shared between them.

DOUBLE DENSITY: Data recorded on each track at twice the standard density, by means of either MFM, M²FM, or GCR encoding.

DOUBLE SIDED: Data recorded on both sides of a diskette, using specially certified media and two independent heads. Announced in June 1977, and soon to be available.

DOUBLE TRACK: Twice as many tracks recorded on each disc surface (154 tracks instead of the standard 77). Not yet announced.

To make matters even more confusing, a floppy disc system can employ combinations of these techniques to multiply storage capacity by four, eight, or sixteen. One manufacturer has used the phrase "quad density" to refer to a combination of double density and double track recording. All we can say is: watch your terminology.

HOW SMART SHOULD A CONTROLLER BE?

Having settled on the PerSci 277 dual drive and single-density recording, we next had to select a suitable controller. Several alternatives presented themselves. Alpha Microsystems, Processor Technology, and Tarbell Electronics all sell controllers which mate the PerSci drive to the S-100 bus. These are fairly simple devices which require extensive software support in the host microcomputer amounting to several thousand machine instructions. PerSci also supplies a controller of its own, a sophisticated "intelligent" controller, and the more we studied it the more we became intrigued with its potential.

The PerSci Model 1070 controller is a remarkable piece of engineering. Built on a tiny 4.5-inch by 7-inch circuit board (see Figure 1), the device incorporates an 8080 microprocessor and associated support chips, a Western Digital FD1771B single-chip LSI diskette controller (see INTERFACE AGE, October and November 1976 issues), 4K bytes of ROM or EPROM containing file management firmware, 1K bytes of RAM used for input/ output buffering, an 8-bit parallel microcomputer interface, and an optional RS232 serial interface. In short, the PerSci controller is a one-board computer devoted to managing the disc subsystem. The controller was not designed for the S-100 bus specifically, but we determined that interfacing it to S-100 was a simple matter requiring only three cheap ICs.

FIRMING UP THE FIRMWARE

The matter of controller firmware was not so simple. The exciting thing about an intelligent controller like the PerSci 1070 is that the controller's own firmware can take care of all of the complexities of controlling the drives and managing disc files. To the host microcomputer, the disc subsystem looks no more complicated than, say, a paper tape reader. This makes it exceptionally easy to use the disc with existing non-disc-oriented monitors, language processors, and other software.

We were disappointed to find that the existing firmware supplied with the PerSci 1070 controller was not very well suited to our requirements. But the concept and the engineering of the controller was so appealing to us that we hated to pass it up. Therefore, we approached PerSci and proposed to perform a complete rewrite of the controller firmware for them. The new firmware would support the following functions: diskette format initialization with optional sector interleave; maintaining and searching a directory of files on each diskette; allocation, de-allocation and reclamation of diskette space; sequential, random, stream and direct file access methods; blocking and unblocking of both fixed-length and variable-length records; creating, deleting, renaming and copying of files; error detection and error re-try; and diagnostic testing of the diskette drives.

PerSci agreed, and the usual development steps ensued with their good and bad days. Three months later the new firmware was fully operational.

FLASHBACK TO THE Z-80

In the meantime, we searched for the best base of existing software to use with our system. We surveyed much of the available disc-oriented software such as Altair, Processor Technology, and Northstar Disc BASICs, Digital Research CP/M, Administrative Systems Inc. OPUS/ONE, but we were less than satisfied with all of them for reasons of inflexibility, performance, and/or large memory requirements.

At length the best library of program development software that we found turned out to be the non-disc-oriented software distributed by Technical Design Labs (Princeton, NJ). The TDL library includes 8K BASIC and 12K Super-BASIC interpreters, an ANSI FORTRAN compiler, a powerful text editor similar to DEC's well-known TECO editor, an excellent word-processing system, and probably the best macro-assembler available on any microcomputer system. All of TDL's software has been written expressly for the Z-80 to take advantage of its powerful instruction repertoire. Furthermore, TDL's software development staff consists of a half-dozen computer scientists at MIT who really know what they are doing.

Because all TDL software is written without embedded input-output routines, and depends upon calls to the monitor to perform all input-output operations, we felt that it would not be too difficult to marry the TDL software to a disc-oriented environment.

PUTTING IT ALL TOGETHER

Having decided to combine the S-100 bus, the Z-80 microprocessor, and the PerSci 277 drive with the 1070 intelligent controller, and TDL software, we faced a big task. Somehow we had to make all of these pieces work together. It turned out to be easier than we feared.

First came the task of interfacing the PerSci 1070 controller to the S-100 bus. To do this, we mounted the controller board piggyback on an S-100 prototyping card, added a few ICs and regulators, and wire-wrapped all into a reasonably presentable package. To our astonishment it worked the first time!

Next we had to get the disc working with BASIC and the other TDL software. Starting with TDL's paper-tape-oriented ZAPPLE monitor, we added patches to make available the full set of disc operations implemented in controller firmware, and to permit assigning disc files as the logical "reader," "punch," and "list" devices. This worked extremely well, but as the patches became more and more extensive, we finally decided to start with a clean slate and write our own TDL-compatible disc monitor.

Initially, the monitor was executed from EPROM on a Bytesaver board, and used the top several hundred bytes of RAM for stack and working storage. Eventually, we added 3K of EPROM and 1K of RAM to our wire-wrap piggyback board, thereby creating a completely self-contained hardware/firmware/software system on one board.

The final step was to modify TDL BASIC to permit full manipulation of disc files by BASIC programs. Because of the simple interface afforded by the intelligent disc controller, it required less than 100 bytes of software to add file handling to BASIC! The addition simply permits the PRINT and INPUT statements to be used with the disc as well as the console.

Everything was working fine. With the warm feeling that accomplishment brings, we set to work developing the disc-oriented applications which had motivated this whole train of events — with occasional Startrek breaks, you understand . . .

INFO 2000 IS BORN

The three Princes of Serendip are still amongst us. Some PerSci employees who were themselves computer hobbyists asked whether we could set them up with adapter boards and software of our design. After receiving a few other requests of a similar nature, we demonstrated our system at the First West Coast Computer Faire in San Francisco, and found much interest. So we took the logical next step.

We contracted with a first-rate circuit board house to make up the S-100 adapter cards, (see Figure 2), then added some enhancements to our disc monitor, and produced extensive user documentation for it (using the TDL editor and word-processor, naturally). To support and distribute these products, we started a company called INFO 2000.

Today our system and software are being used in more than 100 installations throughout the United States, and are on display in a growing number of computer stores. We have added an INFO 2000 adapter board for the Digital Group Z80 computer alongside our S-100 product, so that the PerSci disc and the TDL software library can be used on Digital Group systems. The editor warned us not to boast, but we do feel proud. We believe we have designed a good product.