October 1977 $1.50

ohio scientific's

SMALL SYSTEMS JOURNAL VOLUME I NO.4

features page

The magazine for 6502 computer enthusiasts!

Introduction

This month we are continuing our format of producing the Small Systems Journal on a line printer. There is no question that it has facilitated our output, since corrections and insertions are far easier to make than on an electric typewriter. It is also a great deal neater, since it eliminates the need for erasures and correcting fluid.

We are bringing to a conclusion our three-part series on memory devices for microcomputers with a story on Random Access Memory. Also as a feature in this issue, we are presenting another program contributed by one of our faithful readers. The questionnaire on page 15 is designed to help us better respond to your needs as Ohio Scientific customers.

Needless to say, our progress in getting back on our printing schedule has yet to begin. The publishing of this magazine must be done simulaneously with a whole library of instructional and promotional literature. Moreover our staff is often directly occupied with the efforts involved in maintaining our booths at microcomputer conventions. Therefore we are in the process of changing our publication schedule to bimonthly. The coming winter months will enable us to concentrate our labor at our office.

Let us continue hearing from you. We will give due consideration to any articles or programs submitted for publication. We will also try to answer as many of your questions as possible in the space available. There are still copies of our Fall '77 catalog available for a dollar. Send in the coupon on p. 10 and all correspondence to:

Ohio Scientific Small Systems Journal
Box 36
Hiram OH 44234

1K Corner

Hex address and offset calculator program resides at 0DDE to 0EE4 and is entered at 0DDE. In the following operational description, LLLL, MMMM & XXXX are four-digit numbers and NN is a two-digit hexadecimal number. Note: This program is for use on video systems only.

<+> LLLL+MMMM=XXXX The <+> operator performs hex addition.

<-> LLLL-MMMM=XXXX The <-> operator performs hex subtraction.

<<> LLLL<NN=XXXX The <<> operator calculates the new address (XXXX) from current address (LLLL) and offset (NN).

<>> LLLL>MMMM=NN The <>> operator calculates the offset (NN) between addresses (LLLL) and (MMMM). If the offset is too large, a "?" is returned.

<cr> A carriage return resets the program at any time.

Ohio Scientific's Small Systems Journal is issued monthly by Ohio Scientific Inc., P. O. Box 36, Hiram OH 44234. The subscription rate is six dollars per six issues. Individual copies are $1.50. Published in Twinsburg OH by the Twinsburg Bulletin.

Vol. 1, No. 4 October 1977
Editor-in-Chief Gary Deckant
Production Manager Don Muchow
Contributing Editors Mike Cheiky
Bob Coppedge
Eric Davis
Rick Whitesel

	E	F	0	1	2	3	4	5	6	7	8	9	A	B	C	D
0DDE	D8	A9	D0	85	FF	A9	C5	85	FE	A0	00	A9	20	91	FE	C8
0DEE	D0	FB	A9	3A	91	FE	A2	00	20	ED	FE	C9	0D	F0	E1	20
0DFE	93	FE	30	F4	20	A9	0E	E6	FE	91	FE	E8	E0	04	30	E8
0E0E	F0	06	E0	08	30	E2	F0	33	A9	01	85	FD	20	ED	FE	C9
0E1E	0D	F0	8D	C9	2B	F0	1C	E6	FD	C9	2D	F0	16	E6	FD	C9
0E2E	3E	F0	12	E6	FD	C9	3C	D0	DF	48	A9	00	20	A9	0E	E8
0E3E	E0	05	F0	F6	68	E6	FE	91	FE	30	AD	C8	A9	3D	91	FE
0E4E	C6	FD	F0	10	C6	FD	F0	31	C6	FD	F0	33	A5	F3	10	04
0E5E	A9	FF	85	F2	20	C9	0E	B5	E8	4A	4A	4A	4A	20	BD	0E
0E6E	C8	91	FE	B5	E8	20	BD	0E	C8	91	FE	E8	E0	09	F0	E7
0E7E	20	ED	FE	C9	0D	D0	F9	F0	98	20	D7	0E	4C	65	0E	20
0E8E	D7	0E	A9	FF	45	F1	85	F1	E6	F1	90	04	10	07	30	02
0E9E	30	03	E8	10	C2	A9	3F	91	FE	10	D5	48	A0	04	0A	0A
0EAE	0A	0A	2A	26	F3	26	F2	26	F1	26	F0	88	D0	F4	68	29
0EBE	0F	09	30	C9	3A	30	03	18	69	07	60	18	A5	F3	65	F1
0ECE	85	F1	A5	F2	65	F0	85	F0	60	38	A5	F1	E5	F3	85	F1
0EDE	A5	F0	E5	F2	85	F0	60	EA	EA	EA	EA	EA	EA	EA	EA	EA

MEMORY TECHNOLOGIES FOR SMALL COMPUTERS

PART THREE

RAMS

The term RAM stands for Random Access Memory, but really indicates Read/Write Memory, that is, memory which can be both randomly accessed and directly modified by the computer. The user generally places in RAM all the programs and data of current or immediate interest. For this reason it is to the user's advantage to have as much RAM in his computer as possible. The amount of RAM in a computer is normally referred to in blocks of 1024 bytes, or K (from the prefix kilo-, "thousand"). Thus if a computer has 4K of RAM, it has 4x1024=4096 bytes of RAM. A byte typically consists of eight bits and this is always true in the case of microcomputers.

Most small computers running BASIC will have anywhere from 12K to 32K of RAM if the BASIC interpreter is being placed in RAM and anywhere from 4K to 12K of RAM if the BASIC interpreter is resident in ROM (Read Only Memory, see September 1977, p. 8). It is difficult for a user to decide how much RAM he should have in his computer. For one thing, RAM is the most expensive type of mass storage. Moreover, it has the annoying feature of "forgetting" its contents once the power is turned off. A user may prefer to invest in devices such as disk drives and other peripherals instead of RAM memory. Generally, computer systems are purchased with the minimum amount of RAM recommended by the manufacturer. The owner then installs additional RAM memory based on his experience and contingent needs. For this reason, add-on RAM is the most popular accessory to any computer system. Therefore, the computer user should become intimately familiar with the technologies associated with RAM memory and the various types of RAM memories available for small computers so that he can make intelligent decisions in purchasing additional memory for his computer system.

RAM memory chips are organized in different fashions. For instance, there are chips organized as 1024 x 1, or 128 x 8, etc. If a chip is described as 1024 x 1, it means that it has 1024 addressable locations, each containing 1 bit. The smallest practical memory that can be made from these chips is a 1024-byte memory which will have a minimum of eight chips. A 128 x 8 memory, on the other hand, has 128 addressable locations, each containing a byte, which means that the smallest memory that can be made from it must be one byte wide, but need be only 128 bytes long. It might seem advantageous to use a memory chip organized as 128 x 8, instead of 1024 x 1, because it offers more versatility in configurations. This is true in very small memories such as 1K, but it is not at all true in large memory arrays such as 4K or 16K arrays because of the large number of interconnections required for a one-byte-wide, instead of a one-bit-wide, memory. This large number of interconnections requires a PC board of high complexity and also demands that the package containing the chip be larger because of all the pins on it. In this way, memory density is reduced. Therefore, memory chips designed as small buffer memories of only a few bytes are typically four bits or eight bits wide, whereas memory chips designed for large memory arrays are usually only one bit wide with a large number of addressable locations. This optimizes the utilization of PC board space.

There are two fundamental types of RAM memory: static and dynamic. Static RAM memory utilizes an internal structure made of flip-flops. Each flip-flop typically consists of three transistors within the silicon chip. Static RAM latches its data and holds it as long as the power is turned on. That is, when you write into a memory location that piece of data, or bit, will be retained until you write something else there, as long as the power is maintained. You can read that location as many times as you like without losing the bit that is there. Static RAMs require large areas of silicon wafers which means that they are not only expensive, but have high power dissipations. However they are quite simple to use and very reliable since you can be assured that whatever you write into a location will be preserved as long as you maintain power to the device.

The dynamic RAM utilizes a one field-effect transistor cell, where a bit is stored in the gate of the cell, which can be thought of as a small capacitor. When a bit is written into the cell, it charges the gate capacitance and holds that memory cell at that value for a while. That is, the charge drains off with time. Because of this charge drain-off or degrading, the memory cell must be refreshed periodically and this is where the memory becomes dynamic. If you write into memory locations, you must constantly reread those locations and rewrite into them. Typical dynamic memories must be refreshed every two milliseconds, i.e., 500 times a second! It is desirable for this reading and rewriting of memory locations to be transparent to the CPU so that the refresh operation does not interfere with normal program operation. Most modern dynamic memory boards utilize this transparent mode of refresh.

Obviously the dynamic memory system, or approach, is much more complex than the simple static mode, where you simply write into a location once and forget it. It would not be worth bothering with dynamic memories except for their much lower cost due to the smaller size of the chips and the greater simplicity at the actual IC level. Because of the small size of the chips, they generally have a very low power dissipation. However, it should be obvious that the chips are difficult to implement in a system and are inherently much less reliable than static memories since the chip naturally forgets very quickly. Any minor or temporary problem that can occur in a computer system will cause loss of memory in a dynamic memory system. Static memories are much more forgiving and can tolerate temporary problems, such as brownouts, much better than dynamic memories can because the latter are constantly being refreshed at least 500 times a second.

We will discuss some specifics of actual static and dynamic memory devices later, but to summarize this discussion, static memories typically have higher power dissipations at a higher cost, but are simple to implement in circuits, and are very reliable. Dynamic memories, on the other hand, have low power dissipations at a lower cost (at the chip level), but are difficult to implement, and currently have a much lower reliability than most static memories.

RAM PARAMETERS

There are a few engineering parameters which are valuable, but which can be somewhat confusing, particularly in many advertisements for memories. The two most important parameters for memories are access time and cycle time, sometimes referred to as Read/Modify/Write time. The access time of a memory is the amount of time it takes after the address has been presented to the memory before one can be assured of having a valid data at the output of the chip. Access times can vary anywhere from an extremely rapid 15ns or 20ns for exotic ECL memory chips up to a microsecond or more for some of the very early semiconductor memory devices. The most popular parts currently being manufactured have access times ranging from 200ns to 450ns. Cycle time can be referenced from several points. One simple way of looking at it is the minimum amount of time required between the data output at one address and the data output at a different address. For many static RAM memories the access time is equal to the cycle time. However for all dynamic memories and some special static memories, the cycle time is considerably longer than the access time. Most microcomputers, including the 6502, are optimized for use with memories whose access time equals their cycle time so that it is not always possible to make full use of the fast access time of a memory chip if it has a long cycle time.

There are other important parameters under certain circumstances such as power supplies utilized by a memory chip and data-hold time. The latter specifies how long data is to be valid following the initiation of a write operation which loads the data into memory. Microprocessors such as the 6502 require the utilization of zero data-hold time devices. Virtually all modern memory chips have data-hold times of 0ns, however, some early devices such as the non-A-type 2102 do not, and cannot be utilized with such microprocessors without additional timing components. In general, a primary market for all RAM memory chips is for use as memory on microprocessor-based computer systems. Therefore all modern RAM memories have been designed to be interfaced to microprocessors as easily as possible.

RAM CHIP COMPARISON TABLE

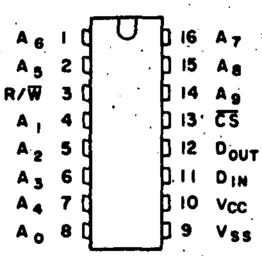
RAM	Size	Organization	Voltage	Power bit/w	Speed range(ns)	Options	Sources	Type
2102AL	1K	1024x1	+5	6K-12K	250-450	various speeds, low-power standby	Intel, TI, National, Signetics, Fairchild, Mostek, etc.	static
6810	1K	128x8	+5	3K	250-450	various speeds	Motorola, AMI, Fairchild	static
2114L	4K	1024x4	+5	8K-16K	250-450	various speeds	Intel, Synertek, EMM, etc.	static
4200	4K	4096x1	+12, +5, -5	112K	215	none	EMM	static
4027	4K	4096x1	+12, +5, -5	32K	200-375	various speeds	all producers	dynamic
4116	16K	16Kx1	+12, +5, -5	10K+	200-375	various speeds	all producers	dynamic

RAM CHIP COMPARISON

The following section is a discussion of each of the RAM memory devices utilized in Ohio Scientific computer systems.

2102AL. The 2102 is the oldest RAM chip in use in microcomputer systems and it is still possibly the most popular chip. Because of its long history, it is highly debugged and relatively inexpensive as all of its manufacturers have gone through their learning curve. They are often selling the part at a discount to reduce large inventories. For this reason, 2102-based memories yield the highest reliability at a very low cost. The 2102-based memory boards are available from Ohio Scientific on the 420 Memory Board, which can be populated as

2102



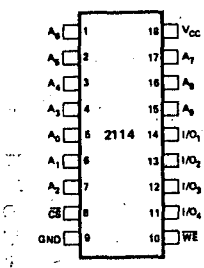
PIN NAMES

A0-A9	Address Inputs	DIN	Data In
R/W	Read/Write Input	VCC	Power (+5V)
CS	Chip Select	VSS	Ground
DOUT	Data Output		

4Kx8 or 4Kx12. The Model 500 CPU Board, which can optionally have up to 4K of memory, also utilizes the 2102. A low-power, high-speed 2102 is utilized on the CM-2 memory which is capable of operation at 2MHz, and a medium-speed, medium-power 2102 is utilized in the CM-1. These are also available in kit form as the 422 and 427 Kits, respectively. The disadvantages of the 2102 are that they have high power consumption and that many boards are required for a large memory array. For instance, it would take eight 420 PC Boards to construct a 32K byte memory, which would draw from 8 to 12 amps at +5V.

2114L. The 2114 and 2114L have just recently come onto the market and are now being produced by virtually every semiconductor manufacturer. The 2114 is designed to be a quad 2102 in virtually every respect. The timing relationships of the 2114 are virtually identical to those of the 2102 which makes implementation very simple. The part is arranged as a 1024x4 chip and is specifically designed for small to medium-sized memories on microcomputer systems. The advantages of this part are, of course, a higher packing density than on the 2102, and its low requirement of just two packages for a minimal memory of 1K byte. The disadvantages for the moment include its high cost because of its recent development.

2114



PIN NAMES

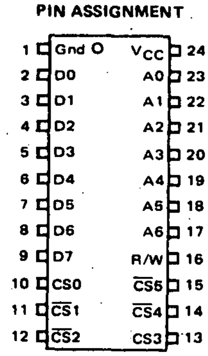
A0–A9	ADDRESS INPUTS	VCC POWER (+5V)
WE	WRITE ENABLE	GND GROUND
CS	CHIP SELECT	
I/O1–I/O4	DATA INPUT/OUTPUT	

It has only a slightly lower power consumption than the 2102, but it comes in a fairly non-standard 18-pin package. The 2114L is used in Ohio Scientific's Model 525 multi-port memory board in an 8K or 16K byte configuration. This memory board requires only +5V, so it can be used in the popular Challenger IIP. The 2114 was chosen as the memory for the multi-port memory because of its simple timing requirements which permit multi-port operation to be very simple. The 525 of couse can be populated as a single-port memory board. It is currently in production solely for use on our 74-megabyte disk systems as buffer memory. As soon as a large supply of 2114L memory chips are available, this board will be offered as an accessory board for all Ohio Scientific systems with primary use being add-on memory for Challenger IIPs. The board is not currently being offered because 2114L memory chips are in short supply at this time.

6810. The 6810 memory was developed by Motorola for use as a scratchpad RAM for industrial applications of the 6800 microcomputer family. It is ideally suited in situations where a small amount of RAM memory is required. However, because of its

6810

PIN ASSIGNMENT



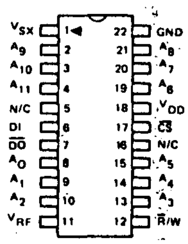
low manufacturing volume, it is high in cost. In addition it is high in power consumption, and is not suitable for use in large arrays because of its greater size--a 24-pin package. The 6810 RAM memory was used in Ohio Scientific's Model 300 Trainer and is currently used as a scratchpad RAM memory for restart and interrupt vectors on the Model 510 triple processor CPU Board.

4200. The 4200 4Kx1 memory chip is one of the very first 4K statics in production. It was used in large-scale memory systems as early as 1975. The part has a pinout equivalent and power supply requirements very similar to early 22-pin 4K dynamics and is intended to be a static replacement for such parts. Because the model 4200 has been available, it has enjoyed the most experience of any 4K static device on the market and currently has a very high level of reliability. The 4200 also has an automatic

4200

PIN	SYMBOL	FUNCTION
1	V_{SX}	Supply Voltage (-5V)
2	A_9	Address Input
3	A_{10}	Address Input
4	A_{11}	Address Input
5	N/C	
6	D_I	Data In
7	$\overline{D_O}$	Data Out
8	A_0	Address Input
9	A_1	Address Input
10	A_2	Address Input
11	V_{RF}	Supply Voltage (5V)
12	R/W	Read/Write Input
13	A_3	Address Input
14	A_4	Address Input
15	A_5	Address Input
16	N/C	
17	$\overline{CS}$	Chip Select
18	VDD	Supply Voltage (12V)
19	A_6	Address Input
20	A_7	Address Input
21	A_8	Address Input
22	GND	Ground

TOP VIEW



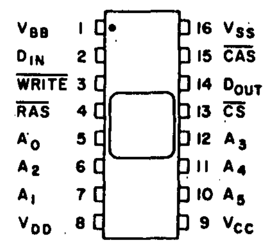
PIN ASSIGNMENT

power-down standby mode, so that when the device is not being accessed, it automatically reverts to standby. Because of this feature it has one of the lowest power requirements per bit of any semiconductor memory in existence. The only disadvantages of the 4200 are that it requires three power supplies (voltages) and is a medium-cost device. Because of its high performance, low power consumption, and high reliability, the EMM 4200 is currently the most popular memory chip in Ohio Scientific computers. It is utilized in our CM-3 16K memory board and as the main memory in virtually all assembled computer systems.

4027. The 4027 is a third-generation 4K dynamic, following the earlier 2107 22-pin dynamics and the first 16-pin 4K dynamic, the 4096. The 4027 is provided by virtually all semiconductor manufacturers. As a dynamic memory, it requires extensive support circuitry for refreshing. It also requires address multiplexing because it has only six address lines into which twelve addresses must be fed. This is accomplished by first presenting one set of addresses at the part, then providing the second set of addresses and latching them in. The 16-pin configuration is utilized with this multiplexing scheme to minimize circuit board layout complexity and also to allow high packing density of memory chips on a PC board. The dynamic memory chips, such as the 4027, are only economical in large arrays

4027

PIN CONNECTIONS



PIN NAMES

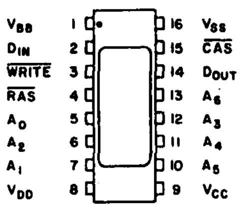
A_0-A_5	ADDRESS INPUTS
$\overline{CAS}$	COLUMN ADDRESS STROBE
$\overline{CS}$	CHIP SELECT
D_{IN}	DATA IN
D_{OUT}	DATA OUT
$\overline{RAS}$	ROW ADDRESS STROBE
$\overline{WRITE}$	READ/WRITE INPUT
V_{BB}	POWER (-5V)
V_{CC}	POWER (+5V)
V_{DD}	POWER (+ 12V)
V_{SS}	GROUND

because of the support circuitry required. Early dynamics, particularly in the hobbyist market, suffer from extreme reliability problems. However the 4027 chip is a very highly refined dynamic memory and the introduction of custom large-scale support circuitry for dynamic memories has improved the reliability of dynamic memories considerably. The 4027 should have acceptable reliability for all but the most demanding small computer applications. The 4027 requires +12, +5, & -5 volt supplies. It is an attractive component for microcomputer manufacturers, because it has relatively low power dissipation, and is available at very low cost in large quantities. This part is being used by several S-100 computer manufacturers in their $350-450 class of 16K dynamic memory boards. It should be pointed out, however, that the 4027 dynamic memory board typically will not demonstrate the reliability of more expensive static memory boards. Ohio Scientific has developed a universal 32-position dynamic memory board which can accept 4027, thus providing 16K bytes, or 8K devices, or the new 16K dynamics discussed below. This board will be offered initially in a 16K byte configuration utilizing 4027s in the spring of 1978.

4116. The 4116 was first developed by Mostek as a 16K dynamic. It has rapidly gained acceptance as the standard for 16K memories, winning out over the Intel 2116 design. Virtually every semiconductor manufacturer is getting on the 16K bandwagon, however to date, only Mostek and Intel have delivered 16K memories in production quantities. Companies such as Fairchild, Signetics, Motorola, Zilog, Fujitsu, and National have announced, and sample parts are now being delivered. Intel has developed a Mostek-compatible 16K part, the 2117, officially recognizing the Mostek part as the standard. The 4116 16K device is patterned after the 4027, utilizing virtually the same pinout. It requires fourteen address lines which are multiplexed into seven pins, where the seventh pin is gained by the loss of the chip enable of the old 4027. Power requirements are similar to the 4027, and the operating procedures are the same in that the 4116 requires refreshing ever two milliseconds and address multiplexing. The 16K dynamic will ultimately be the least expensive memory available. Since the parts are very new and unproven, they are fairly costly and somewhat unreliable at the moment. Ohio Scientific has been actively evaluating small quantities of 16K dynamics from most

4116

PIN CONNECTIONS



PIN NAMES

A_0-A_6	ADDRESS INPUTS
CAS	COLUMN ADDRESS STROBE
D_{IN}	DATA IN
D_{OUT}	DATA OUT
RAS	ROW ADDRESS STROBE
WRITE	READ/WRITE INPUT
V_{BB}	POWER (−5V)
V_{CC}	POWER (+5V)
V_{DD}	POWER (+12V)
V_{SS}	GROUND

manufacturers and has developed a Model 530 PC Board which can accept either 16K or 4K dynamic chips. When the 530 is fully populated with 16K dynamics, it will typically be configured with 24 chips yielding a 48K dynamic memory board. Because of the newness and high price of the 16K dynamics, we are currently projecting a second quarter 1978 introduction of 16K dynamic-based products from Ohio Scientific. We are waiting simply to give the industry a chance to refine the 16K chip before introduction.

Conventional Typewriter

This program provides a means of using the OSI-65V when interfaced to a printer to be used as a conventional typewriter and also modify the text for a data file.

Consider the program as having two parts or functions. The first part allows you to type text in much the same way you would on a typewriter, that is, without formatting PRINT instructions in BASIC. You need type only what you want to be printed.

After clearing the video screen, the program starts by asking the user "CHARACTER WIDTH?" This tells the program how many characters per line to limit. The maximum width would be determined by your printer or video display format. Next the computer displays the line number of the text which you are typing. An up-arrow appears on the screen (via a POKE instruction in the program) indicating where your line must terminate. This becomes a very useful feature. In the event that the line-width is exceeded, an OVERWIDTH error message comes up and you are free to rewrite the line. When the text input has been completed, input a right-arrow, which signals the program to jump to its next section. The correct format is then printed out on your display, following which the screen is cleared. Next the program lists three options. The first asks "PRINT DATA"--inputing a 1 causes the printer to turn on via a POKE instruction. The correct text is then printed in correct format and the printer is turned back off via another POKE instruction. The program now displays the option list again.

The second option asks "FILE DATA"--inputing a 2 displays the message "SET RECORDER". At this point a delay loop in the program allows you enough time to turn on the recorder in the record mode. When the loop times out, each line of the text is preceded by "100X DATA". Output to tape and display on screen are as in the following condensed example:

```
1000 DATA Now is the
1001 DATA time for all
1002 DATA good men
```

The line numbers are incremented, the word "DATA" is inserted, and the text follows.

When inputing back from the tape, any program could be written to use the data. The usefulness of the program should be readily apparent.

The third option asks "ADD DATA"--inputing a 3 allows you to continue with the text from the point where you left off.

Note that some of the added features provided in this program via the POKE commands (e.g., screen erase and printer on-off at programmed points) cannot be appreciated in the hard copy printout.

General notes: a space inputed instead of characters yields a blank line. The text is limited to 256 lines of memory capability. The program will not recognize commas. If you type a line containing a comma, an error message "EXTRA IGNORED" will appear in the succeeding line.

This same program can also be used on OS-65A (serial-based systems) with the following restrictions:

No up-arrow appears on the screen as a prompter to indicate where a line should end.

The screen is not cleared at the start of each operation (on video-based system the clearing of the screen occurs as a result of lines 60 and 300; on serial-based systems, these lines cause merely a triple-carriage return-linefeed).

```
LIST

 10 PRINT
 20 PRINT "PROGRAM BY
 25 PRINT "GARY SMITH"
 30 PRINT "4322 Watterson St.
 35 PRINT "Cincinnati OH 45227"
 50 GOSUB 220
 60 PRINT:PRINT:PRINT
 70 INPUT "CHARACTERS PER LINE";D
 80 I=256
 90 DIMA$(I)
 100 PRINT
 110 FOR I=1TO256
 120 PRINTI
 130 IFD=>25ANDD=<49THENPOKE54149+(D-25),94
 140 IFD=>49ANDD=<70THENPOKE54181+(D-50),94
 150 IFD=<22THENPOKE54149+D,94
 160 IFD=24THENPOKE54148,94
 170 IFD=23THENPOKE54172,94
 180 INPUTA$(I)
 190 IFLEN((A$(I)))>DTHENPRINT"OVERWIDTH":I=I-1
 200 IFA$(I)=">"THEN260
 210 NEXTI
 220 FORC=53348TO54268
 230 POKEC,32:NEXTC
 240 RETURN
 250 POKE 64258,1
 260 GOSUB 220
 270 FORJ=1TOI-1
 280 PRINTA$(J)
 290 NEXTJ
 300 PRINT:PRINT:PRINT
 310 POKE 64258,0
 320 PRINTI-1;" DATA LINES"
 330 PRINT
 340 PRINT "INPUT OPTION..."
 350 PRINT
 360 PRINT"   1=PRINT DATA"
 370 PRINT"   2=FILE DATA"
 380 PRINT"   3=ADD MORE DATA"
 390 INPUTE
 400 PRINT
 410 IFE=3THENI=I-1:GOTO210
 420 IFE=1THENPOKE64258,1:GOTO260
 430 IFE<>1ANDE<>2THENPRINT"BAD INPUT ?":PRINT:GOTO340
 440 PRINT"SET RECORDER"
 450 NULL10
 460 FORL=1TO10000:NEXT
 470 GOSUB 220
 480 PRINT "1000 DATA ";I-1
 490 FORK=1TOI-1
 500 A=1000+K
 510 PRINTA;"DATA ";A$(K)
 520 NEXTK
 530 NULL0
 540 END
OK
RUN



PROGRAM BY
GARY SMITH
4322 Watterson St.
Cincinnati OH 45227


CHARACTERS PER LINE? 20
```

```
 1
? THE QUICK BROWN FOX JUMPS OVER THE
OVERWIDTH
 1
? THE QUICK BROWN FOX JUM
OVERWIDTH
 1
? THE QUICK BROWN
 2
? FOX JUMPS OVER
 3
? THE LAZY DOG
 4
? >
THE QUICK BROWN
FOX JUMPS OVER
THE LAZY DOG


 3  DATA LINES

INPUT OPTION...

   1=PRINT DATA
   2=FILE DATA
   3=ADD MORE DATA
? 1

THE QUICK BROWN
FOX JUMPS OVER
THE LAZY DOG


 3  DATA LINES

INPUT OPTION...

   1=PRINT DATA
   2=FILE DATA
   3=ADD MORE DATA
? 2_3

 4
? STEALTHILY
 5
? >
THE QUICK BROWN
FOX JUMPS OVER
THE LAZY DOG
STEALTHILY


 4  DATA LINES

INPUT OPTION...

   1=PRINT DATA
   2=FILE DATA
   3=ADD MORE DATA
? 2

SET RECORDER
1000 DATA  4
 1001 DATA THE QUICK BROWN
 1002 DATA FOX JUMPS OVER
 1003 DATA THE LAZY DOG
 1004 DATA STEALTHILY

OK
```

Now You Can Play Star Wars

Since the debut of "Star Wars," an overwhelming preoccupation with outerspace adventures has arisen. Therefore this program, which makes use of the POKE instruction to update the display, seems of natural appeal for those second-generation Star Trek freaks.

The requirements are minimal:

1) Ohio Scientific's 8K BASIC by Microsoft (cassette, paper tape, disk or ROM version)

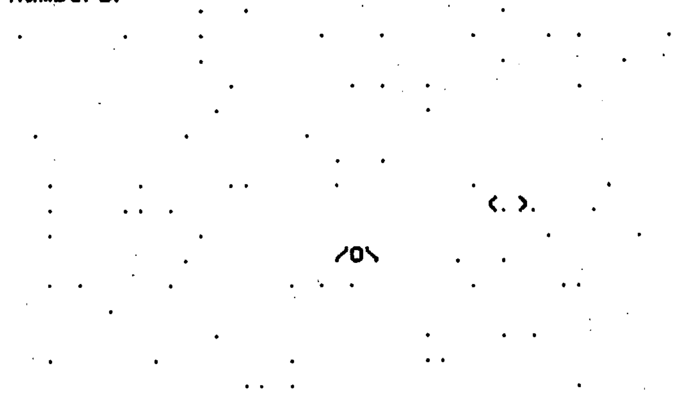
2) OSI's 440 Video Board at DXXX with a keyboard at DFFF

3) at least 4K of RAM at 0000 up, in addition to BASIC in ROM or RAM.

In principle the game is simple, yet it appeals to persons of just about all ages. At game start the screen is blanked and brief instructions follow. The player enters any random set of numbers to provide a basis for the arrangement of stars and the movement of the Enemy ship The screen first fills with a random display of "stars," following which the Enemy "/0\" appears on the screen along with the sight of your turret "< >". The Enemy moves at random. You control the movement and firing of your turret by means of the following keys:

1-up	3-right	5-stop
2-down	4-left	6-fire

If an Enemy is shot down the message "NEW TARGET APPROACHING" appears momentarily on the screen. The game ends when the enemy shoots you down (this occurs randomly), or when you shoot down the Enemy in sufficient numbers.



There are as always many possible modifications and/or improvements that can be made. One such possibility would be to use SPST momentary contact switches and the parallel port on a 430 Board in place of the keyboard. Another option would be to alter the game so that a projectile travels from base (ground) to the sights of the gun. By using the PIA you could also generate sound effects or trigger several 555s set at different frequencies and duty cycles.

```
5 REM****ROBERT L.
6 REM****COPPEDGE
340 FOR X9=53250 TO 54268     clear screen
345 POKE X9,32
350 NEXT X9                   V=USR(V)
351 Q=53776
352 PRINT"DESTROY ENEMY(/O\)"
353 PRINT"1-UP; 2-DN; 3-LFT; 4-RGT; 5-STOP; 6-FIRE"
360 PRINT"ENTER SEED #";
365 INPUT Z1
370 FOR Z2=53960 TO 54145     V=USR(V)
380 POKE Z2,32
390 NEXT Z2
```

Modules of the Program

LINES	FUNCTION
340--350	Clears screen
351--390	Sets Z1=value of seed and clears instructions off screen
391--394	Places random starfield on screen based on seed value
403--430, 460--480	Places enemy and turret sight at random locations on screen
481--485, 1100--1160	Determines which control key is pressed and selects proper program module
1700--1750	Up command (subtracts 32 from Q and blanks out old sight)
1200--1250	Down command (adds 32 to Q and blanks out old sight)
1400--1450	Left command (subtracts 1 from Q and blanks out old sight)
1500--1550	Right command (adds 1 to Q and blanks out old sight)
1600--1650	Stop (freezes sight)
2000--2050	Fire (displays shot and checks hit)
550--761	Random movement of Enemy
860--935	Controls Enemy fire
1005--1096	Controls explosion when enemy makes a hit
2500--2650	Ending messages

```
391 FOR Q5=1 TO 100
392 Q4=INT(920*RND(Z1)+53248)
393 POKE Q4,46
394 NEXT Q5
403 C3=47:C4=15
410 A2=INT(920*RND(Z1)+53248)
420 A3=INT(A2/32):A4=A2-(A3*32):IF A4>26 THEN 410
430 GOTO 460
440 Q1=Q-1:Q2=Q+1:POKE Q1,40:POKE Q2,41
445 F2=100*RND(Z1)
446 IF F2<2 THEN 860
447 D2=RND(Z1)
450 IF D2<.5 THEN 550
455 GOTO 481
460 B2=A2-1:C2=A2+1
461 Q1=Q-1:Q2=Q+1
462 POKE Q1,40:POKE Q2,41
470 POKE B2,47:POKE A2,79:POKE C2,92
480 B9=B2:A9=A2:C9=C2
481 Q3=PEEK(57343)
485 GOTO 1100
495 ON Q9 GOTO 1200,500,1400,1500,500,1700
500 GOTO 440
550 E2=INT(9*RND(1))
560 ON E2 GOTO 580,600,620,640,660,680,700,720
580 A2=A2-33
590 GOTO 740
600 A2=A2-32
610 GOTO 740
620 A2=A2-31
630 GOTO 740
640 A2=A2+1
650 GOTO 740
660 A2=A2+32
670 GOTO 740
680 A2=A2+31
690 GOTO 740
700 A2=A2-1
710 GOTO 740
720 A2=A2+33
730 GOTO 740
740 GOTO 744
744 A7=(INT((A2-53346)/32))*32
745 IF A2-53346-A7<2 GOTO 720
750 IF A2-53346-A7>26 GOTO 700
751 IF A2<53280 GOTO 720
752 IF A2>54250 GOTO 600
755 POKE A9,32:POKE B9,32:POKE C9,32
761 GOTO 460
860 F5=B2:F6=C2
870 W=INT(8*RND(Z1))
871 F1=A2:ON W GOTO 900,905,910,915,920,925,930,935
872 F1=F1+S:POKE F1,40
873 FOR M7=1 TO 25
874 G1=F1:F1=F1+S:F2=F1+S:F3=F2+S:F4=F3+S
875 F5=F4+S:F6=F5+S:POKE F1,39:POKE F2,39:POKE F3,39:POKE F4,39
876 POKE F5,39:POKE F6,39:POKE G1,32
877 IF W<>2 GOTO 879
878 IF F1>54244 GOTO 1005
879 NEXT M7
895 POKE F1,32:POKE F2,32:POKE F3,32
898 GOTO 975
900 S=31:GOTO 872
905 S=32:GOTO 872
910 S=33:GOTO 872
915 S=1:F1=F1+S:GOTO 872
920 S=-31:GOTO 872
925 S=-32:GOTO 872
930 S=-33:GOTO 872
935 S=-1:F1=F1+S:GOTO 872
975 J8=B2-32:J7=A2-32:J9=C2-32
976 POKE J8,32:POKE J7,32:POKE J9,32
980 GOTO 440
1005 FOR Z4=1 TO 170
```

```
1010 POKE F1,Z4
1020 NEXT Z4
1040 FOR N9=1 TO 8
1045 N8=F1+N9:N7=F1-N9:N6=F1-(N9*32):N5=F1-(N9*33):N4=F1-(N9*31)
1050 POKE N8,42:POKE N7,42:POKE N6,42:POKE N5,42:POKE N4,42
1055 O5=N5:O6=N4
1056 O7=N6:O8=O7
1060 FOR O9=N9 TO 1 STEP -1
1063 O7=O7-1:O8=O8+1
1065 O5=O5+32:O6=O6+32
1069 POKE O7,42:POKE O8,42
1070 POKE O5,42:POKE O6,42
1071 NEXT O9:NEXT N9
1080 FOR X9=53260 TO 54268
1090 POKE X9,32:NEXT X9
1096 GOTO 2500
1100 IF Q3=177 GOTO 1700
1110 IF Q3=178 GOTO 1200
1120 IF Q3=179 GOTO 1400
1130 IF Q3=180 GOTO 1500
1140 IF Q3=181 GOTO 1600
1150 IF Q3=182 GOTO 1300
1160 GOTO 495
1200 POKE Q1,32:POKE Q2,32
1210 Q=Q+32:Q9=1
1250 GOTO 500
1300 GOTO 2000
1340 Q9=2
1350 GOTO 500
1400 POKE Q1,32:POKE Q2,32
1410 Q=Q-1:Q9=3
1450 GOTO 500
1500 POKE Q1,32:POKE Q2,32
1510 Q=Q+1:Q9=4
1550 GOTO 500
1600 POKE Q1,40:POKE Q2,41
1610 Q3=128:Q9=5
1650 GOTO 500
1700 POKE Q1,32:POKE Q2,32
1710 Q=Q-32:Q9=6
1750 GOTO 500
2000 FOR Q8=1 TO 200
2010 POKE Q,Q8:NEXT Q8
2025 POKE Q,32
2030 IF Q=A2 GOTO 2150
2040 IF Q=A2-1 GOTO 2150
2050 IF Q=A2+1 GOTO 2150
2100 GOTO 1340
2101 GOTO 2600
2150 POKE A2,42:POKEB2,42:POKE C2,42
2160 POKE Q1,40:POKE Q2,41
2170 Y5=Y5+1: IF Y5=8 GOTO 2600
2171 PRINT"NEW TARGET APPROACHING."
2175 FOR Z4=1 TO 100
2176 NEXT Z4
2180 FOR Z3=54080 TO 54145
2190 POKE Z3,32
2200 NEXT Z3
2210 POKE Q1-32,32:POKE Q2-32,32
2220 GOTO 403
2500 PRINT"TURRET KNOCKED OUT."
2505 PRINT"DESTROYED ";:PRINTY5;:PRINT" ENEMY"
2508 PRINT"IN THE PROCESS."
2510 GOTO 2650
2600 PRINT"YOU HAVE SUCCESSFULLY"
2610 PRINT"PUT AN END TO THEIR"
2620 PRINT"MAD DREAMS OF"
2630 PRINT"TOTAL DOMINATION OF THE"
2635 PRINT"FREE UNIVERSE!"
2640 PRINT"AND YOU'RE STILL ALIVE!"
2650 END
OK
```