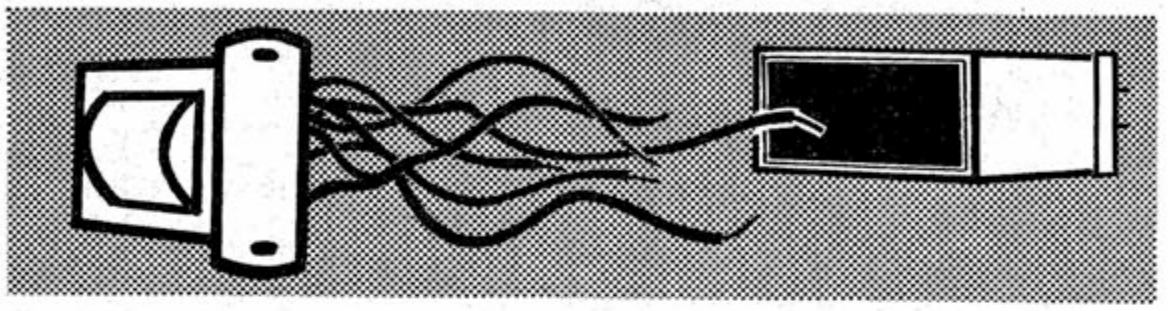
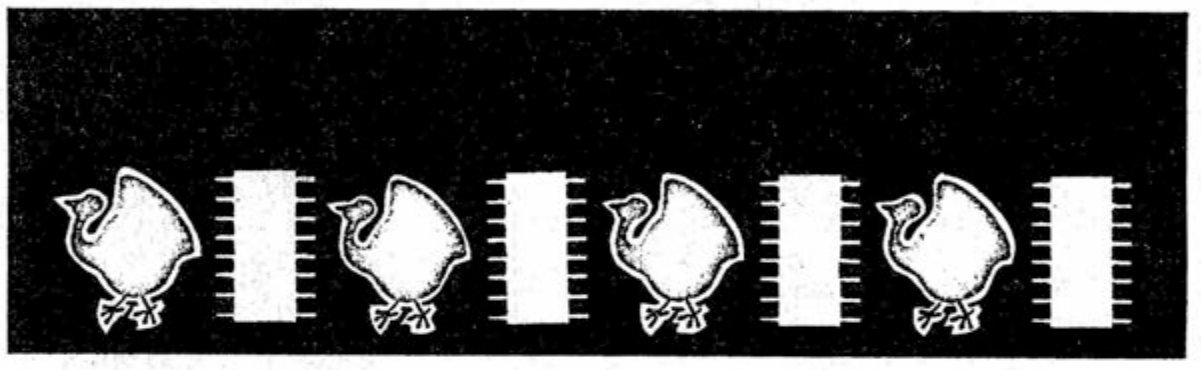
OCTOBER Vol2 Issue5

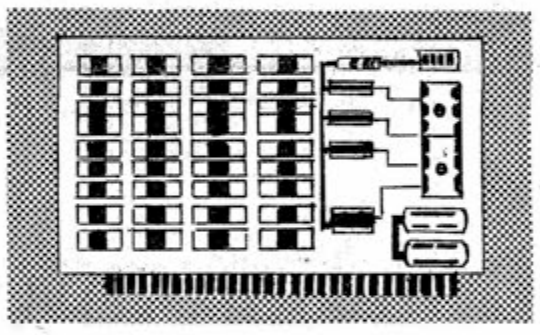
COMPUTER NOTES

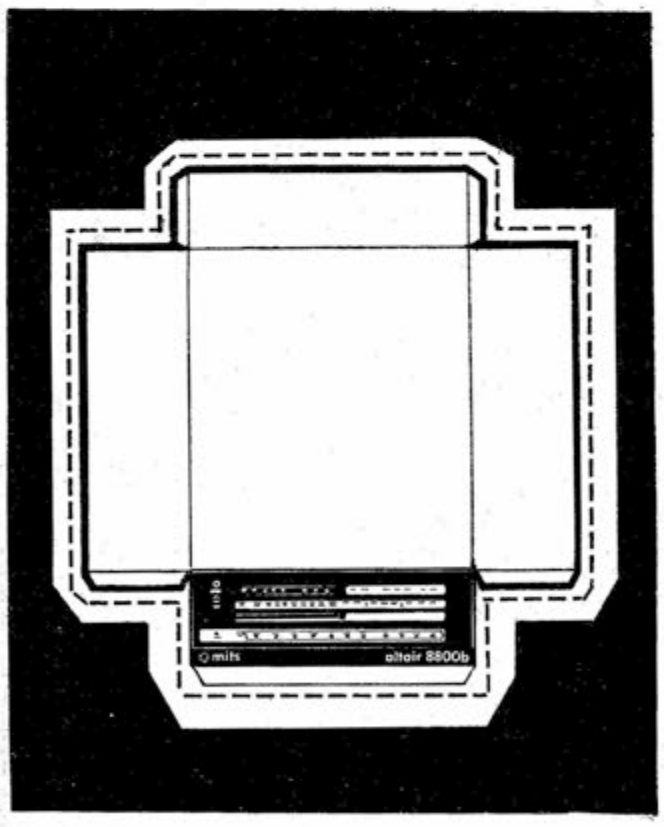
©mits

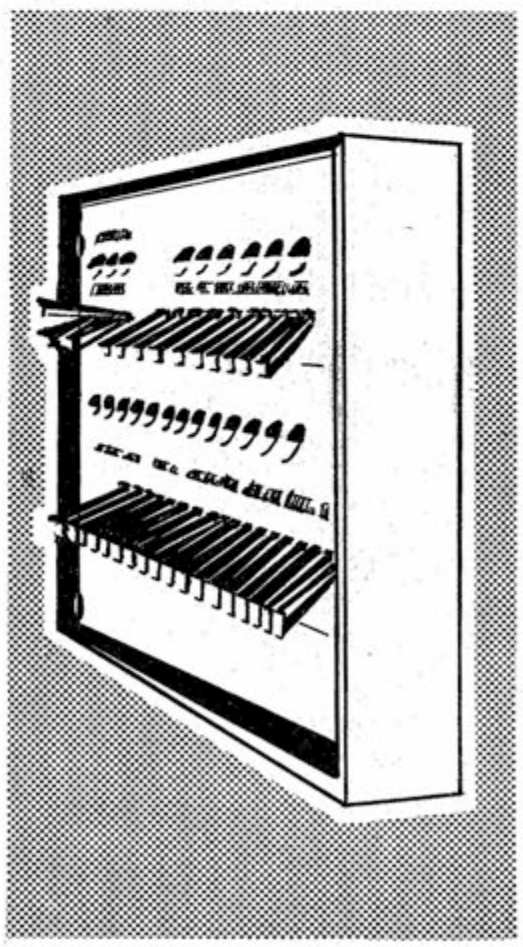


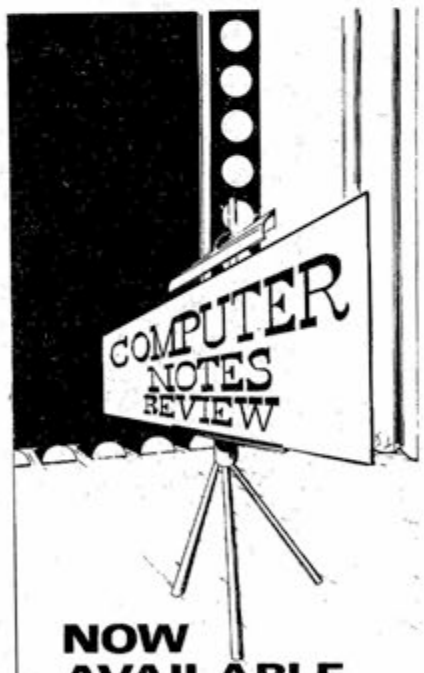


IN THIS ISSUE;









NOW AVAILABLE

Computer Notes Review, Volume I, **is a collection of reprinted articles from previous issues of Computer Notes (April, 1975 through July, 1976). We have eliminated all editorial, fictional and advertising materials and have printed only the most informative and technical articles pertaining to Altair hardware (specs, modifications, troubleshooting) and software. This 94-page book is arranged in an 8½ x 11 format and is ready to insert in a 3-ring binder. The price of Volume I is $12.00. (Altair customers who have already ordered the Update Service will automatically receive *Computer Notes Review, Volume I.*)**

LETTERS TO THE EDITOR

Dear Ed,

When I read C.N., I notice a downright lack of agricultural applications in your software. As foreman of the Hazza farming operation, I know firsthand the possibilities of controlling irrigation, thermostatic control, automatic feeding and keeping production records. Why, with the proper programming, the Altair could one day replace the Farmer's Almanac!

The software shouldn't be too difficult to work out. Using Altair BASIC with an IF...THEN here and a GOTO there, here a PEEK, there a POKE and overall a FOR loop, it's a pretty simple operation.

But perhaps the problem is in hardware. I would suggest you all look into providing facilities for sending Interrupt and Interrupt Enable signals over the long distances encountered in modern farming operations. This Extended Interrupt-Enable I/O board(E.I.E.I.O.) would have a P.C. here and an IC there, here a chip, there a switch and everywhere a flip-flop.

Sincerely,

O. McDonald
Hazza Farms

customer service news

By Gale Schonfeld Oct. '76 Issue

This will be my last time as author of "Customer Service News". In the next issue of Computer Notes Sandy Koppenheffer will begin writing the articles for this column.

This month I would like to extend my appreciation to Fred Schonfeld, for his help in writing the software for the User Group and Computer Notes mailing program.

Beginning with the November issue, your mailing labels will appear slightly different than previously. The code number above your name will indicate geographical location, record number, a MITS or dealer purchase, number of subscription months left and type of mainframe owned. For example:

Geograp-ical loc-ation.	Record Number	Purchased From	Sub. months left	Owns an 8800
AA	1234	MITS	10	8

Appears above your name as: AA1234MITS108

When we begin using this new system we would appreciate you advising us if there are any errors on your mailing label. Send corrections or address changes to the attention of Sandy Koppenheffer - Customer Service.

On November 15th we will begin accepting Computer Notes renewals. If you have any questions regarding this, please contact Sandy.

Thank you.

COMPUTER NOTES

Editor	Andrea Lewis
Assistant Editor	Linda Blocki
Production	Tom Antreasian Al McCahon Grace Brown
Contributors	Ed Roberts Gale Shonfeld Ron Scales Bill Kuhn Rich Haber Bruce Fowler John Archer Bill Yates

SHORT MEMORY TEST

How it works:

When the user loads the program, the starting address must be patched into octal locations 077 and 100. The end address plus one must be patched into locations 004 and 005 (octal).

First, the program writes zeros into all addresses that are to be checked. Next, it starts again at the first address, reads a byte, and compares it to the pattern that was written in. If that byte matches the pattern that was written, the byte is incremented by one and stored into the address from which it was read. This process saves time by writing the next test pattern long before it will be needed in the next read loop.

The program continues to read, verify, and rewrite until it finds an error (byte read doesn't match reference pattern) or until all addresses have been checked.

If there are no errors, the program restarts at the beginning memory check address and checks the bytes against the new pattern that was written on the last loop. Because this process repeats again and again, the pattern will vary from 0 to 255 and back to 0, etc.

If there is an error, the address where the error occurred is saved at locations 075 and 076 (octal). (These locations will remain 377, 377 if no errors are found.) The reference pattern will be stored at 073 (octal). If no errors are detected, the reference pattern is stored anyway so that the user can stop the program and see how far it has gone. The bad data that is read is saved at location 074 (octal).

After saving the error information, the program loops at location 043 (octal) until the user stops it. For the jump to self, the user can substitute a jump to his own binary-to-ASCII conversion and I/O routines.

Assembly Listing

```
                DI              "DO NOT DISTURB"
                XRA A
                MOV C,A         C gets pattern for compare
                LXI D,XXXX      user loads end address +1 here
*WRITE ALL ZEROS FIRST TIME HERE
                LHLD STRT       get start address
                                start
                                address location
WRO:            MOV M,C
                INX H           next place to store
                MOV A,H         see if done:
                CMP D           high addresses equal?
                JC WRO          no - keep writing
                MOV A,L         highs were equal - check lows
                CMP E
                JC WRO          still not done
*MAIN LOOP - READ/COMPARE/WRITE NEXT PATTERN - DIE ON ERROR
RST:            LHLD STRT       back to start
                                start
                                address location
RCW:            MOV A,M         read a byte
                CMP C           same as expected?
                JZ GOOD         yes - skip error stuff
                SHLD ERRADR     no - save error address
                STA DATERR      and bad data
SELF:           JMP SELF        then die quietly (user can do I/O here)
GOOD:           INR A           write next pattern back where we just
                MOV M,A         read to save time later
                INX H           where we look next
                MOV A,H         unless already done
                CMP D           check high addresses
                JC RCW          not done - keep going
                MOV A,L         may be done - check low addresses
                CMP E
                JC RCW          not done
*END OF A GOOD PASS - NEW PATTERN WRITTEN
                INR C           new reference for compare
                MOV A,C
                STA PATT        save where user can see it
                JMP RST         start again
*VARIABLE STORAGE
PATT:           DC              0
DATERR:         DC              0
ERRADR:         DC              'FF'
STRT:           DS              2
```

Octal Listing

```
000     363
  1     257
  2     117
  3     021
  4     XXX end address   low
  5     XXX               high
  6     052
  7     077
 10     000
  1     161
  2     043
  3     174
  4     272
  5     332
  6     011
  7     000
 20     175
  1     273
  2     332
  3     011
  4     000
  5     052
  6     077
  7     000
 30     176
  1     271
  2     312
  3     046
  4     000
  5     042
  6     075
  7     000
 40     062
  1     074
042     000
  3     303
  4     043
  5     000
  6     074
  7     167
 50     043
  1     174
  2     272
  3     332
  4     030
  5     000
  6     175
  7     273
 60     332
  1     030
  2     000
  3     014
  4     171
  5     062
  6     073   pattern
  7     000   address
 70     303
  1     025
  2     000
  3     000   pattern stored here
  4     000   bad data stored here
  5     377   low   error address
  6     377   high  stored here
  7     XXX   low   start
100     XXX   high  address
```

See TROUBLESHOOTING 16K STATIC BOARDS Page 4

TROUBLESHOOTING 16K STATIC BOARDS

By Rich Haber, Bruce Fowler, and John Archer

Judging from the Repair Department, the 16K Static boards seem to be providing very reliable service to MITS customers. But since no board is perfect and repairing the 16K static boards is sometimes confusing, these troubleshooting hints should help cure some common ailments.

The symptoms aren't always indicative of the 16K Static board's real problem. So, it's important to first pay particular attention to any "physical" problems.

Any troubleshooting sequence should begin with a thorough visual inspection. Look at the components. Make sure that they are the correct ones and are properly polarized, etc. Watch out for bent pins on the RAMs (4200's) and latch (8212) and carefully inspect the back of the board in the vicinity of the RAMs for solder bridges. Some of these types of shorts are too small to be seen without the aid of a magnifying glass.

Ohm out the pins of the RAMs to test for shorts. Check from pin 1 to all other pin 1's then pin 2 to all other pin 2's, etc. There should be continuity to all pins with the same numbers except for pins 5, 6, 7 and 16 vertically and 5, 16 and 17 horizontally. On one RAM also check to see that there is no short between any two pins.

It's a good idea to polish the fingers (100 pins on the bottom of the board) with an eraser. This will insure the best possible contact to the bus. If a short is found, reinspect with a magnifying glass and scribe. Use the PCB layout to find out where the two lines are contiguous. If a short cannot be found visually, it may be necessary as a last resort to cut one of the shorted lines with an exacto knife. By dividing the trace into two equal sections with each cut, the short can be located with a maximum of five cuts. Repair the cuts with a short piece of bare wire.

Solder all the plated-through holes from the back of the board and then check to see that the solder flowed through the hole to the top of the board. Sometimes heat and/or flexing or other physical pressure applied to the board causes a problem with the solder flow. If that happens and the solder doesn't flow through, place a small piece of bare wire into the hole and solder from both sides. If the solder still didn't flow through it, you may have had an "open". Just find the holes that are accessible from only one side and insert a small portion of wire, then solder. That way the solder has a path to flow through even if the plate-through has failed.

Check the supply voltages on the RAMs themselves. Pin 1 = -5v, 11 = +5v, 18 = +12 volts and 22 = gnd. The voltage should be within 200 millivolts of spec. 300 millivolts of ripple is the maximum on the +5 and -5 lines. 500 mv is allowed on the +12v line. If excess ripple is found, see if the input to the regulator is at least 2 volts higher than the output and check the output of regulator under no load.

Operational Checks

1. Printed elsewhere in this issue is a convenient memory test you can use to find bad bits. Run this test as per instructions. If a bad bit (or bits) are found, juxtapose the bad RAM with another one and rerun the test to see if the RAM is really at fault. Sometimes merely switching the RAMs will correct a problem due to poor contacts.

2. Sometimes a problem can be found from the front panel. Deposit a "one" at each data bit and reexamine. Examine each address switch to see if that line goes HIGH. If a data or address line does not come on, see if it will come on if all the other switches are HIGH. By following through one can isolate a short between two lines.

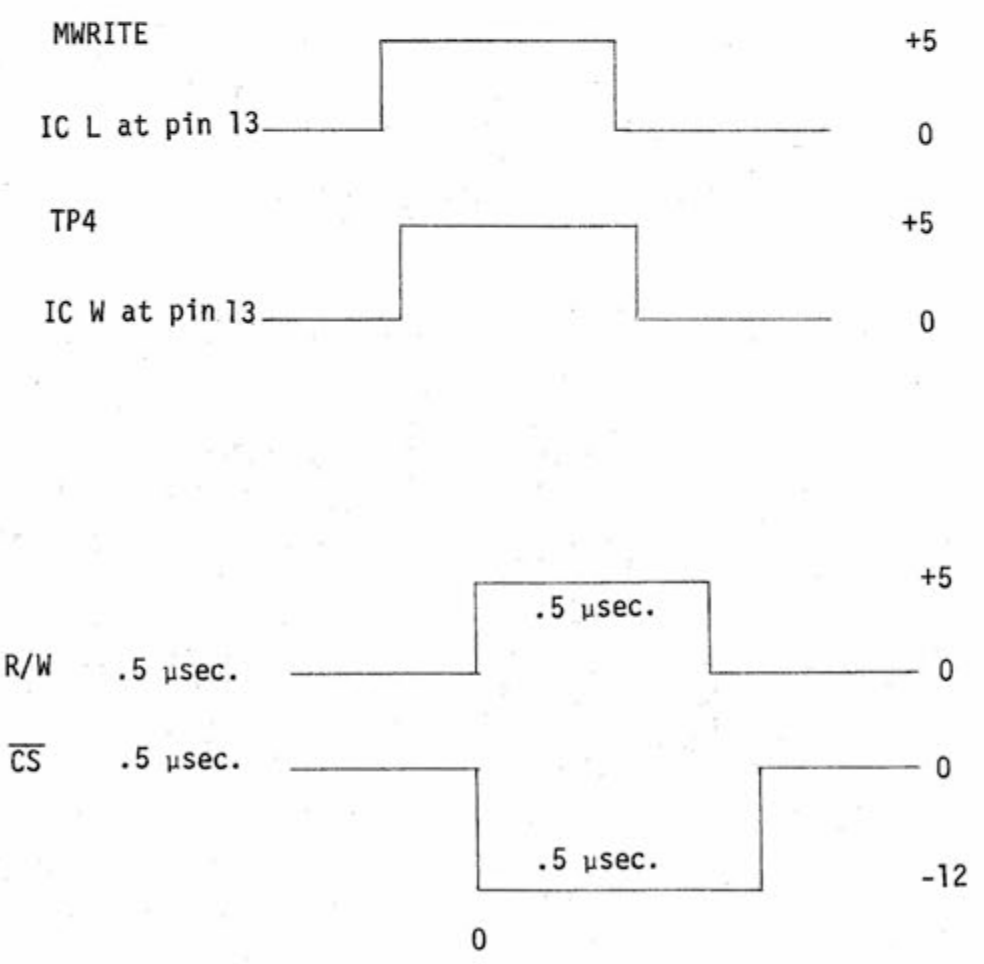
Try to deposit a 377 at each 4K row of memory. If a row can't be written, pin 17 (CHIP SELECT) is a likely culprit. Look for a fast negative pulse when depositing.

If you cannot deposit into the board at all, you can write a program in another memory board which will continually try to deposit into the 16K. This will allow you to observe waveforms otherwise very difficult to examine. A program that continuously outputs data from sense switches follows.

```
333
377
062
000
100 (switch 2 up in 16K board)
303
000
000 (other board is at location
    000)
```

Load program, examine 0, put up any combination of sense switches (A15-A8) and run.

Some signals to look for:



Program Progress

Due to the increased interest in developing programs for very specific needs, beginning next month we'll be offering "Program Progress" whenever possible. Designed to give readers an opportunity to suggest what programs they'd like to see in the Altair Software Library, the list of ideas will be published as long as you keep providing suggestions. We hope these suggestions for programs will be of particular interest to Altair owners.

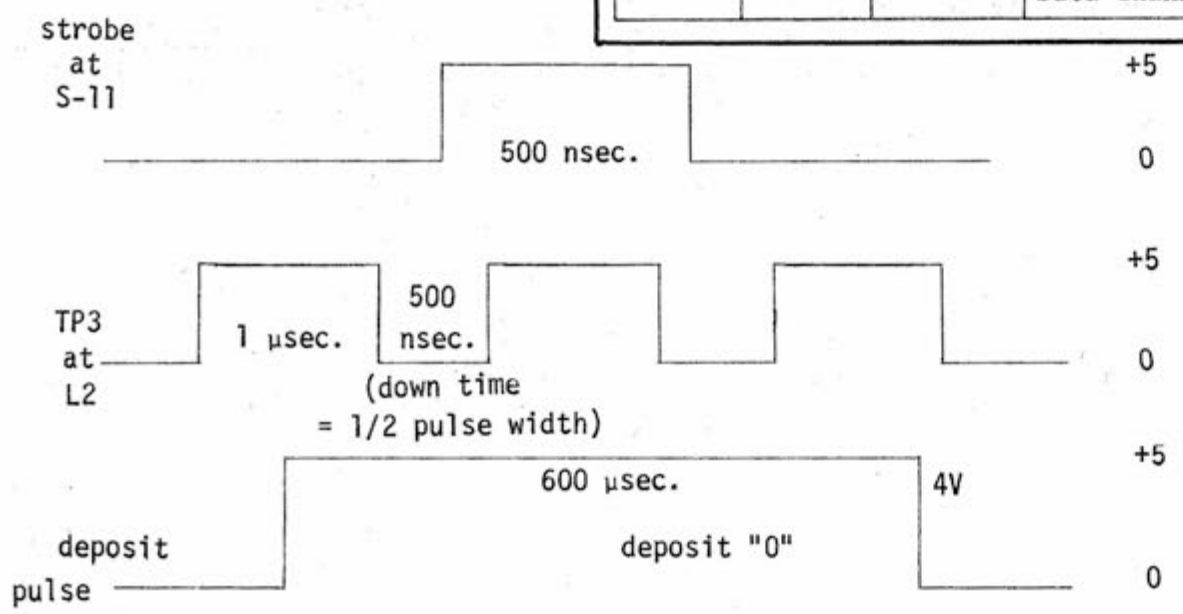
So send your ideas to COMPUTER NOTES, and we'll pass them on to our readers.

TROUBLESHOOTING 16K STATIC BOARDS

Continued from Page Four

The latch strobe and TP3 can be seen with just a jump program.

303 (board strapped to "0" address)
000
000



pin 6 of appropriate RAM repeat several times with a very slow sweep rate

Even if your scope is too slow to measure these pulses carefully, you can usually assume that the mere presence of a pulse indicates that this particular circuit is functioning.

Check the output of the latch to determine if data has been deposited. If the output of the latch is correct and the front panel indicates incorrect data, then the problem is probably in the buffers, IC V or W. Noisy buffers can cause problems that may be impossible to see.

If several bits at each address are registering incorrectly, check to see if these bits have an inverter or buffer in common.

Should you decide to send the board to us in Repair, you can expect a fairly rapid turn-around. Occasionally a board comes in which causes us to scream and pull our hair out for a week or two, but this is the exception. Please feel free to call or write to us should you desire some help on any of our products.

Happy Hunting!

4PIO Operation

By Bill Kuhn

The tremendous versatility and large number of software controlled options in the 4PIO often lead to confusion about the operation under a specific software initialization. The use of CA2 and CB2 as outputs seems to be particularly confusing. This article should clarify the operation of these two lines.

The most common initialization for both A and B sections is shown here. (See page 7, 4PIO Manual.)

A and B Section Control Register

Bit 5	Bit 4	Bit 3
1	0	1

The type of system in which this initialization might be used would be to use the 4PIO to interface a parallel data terminal to the Altair 8800 system. Section A data lines would be used as inputs, Section B as outputs. When the computer does an input from Section A Data Channel, a negative going pulse, 2 µsec. wide is produced at CA2. This would tell the terminal, "OK, I read the data. Send me another character."

When the computer does an output to the B Section Data Register, a negative going pulse of 2 µsec. width is produced at CB2. This would tell the terminal, "OK, there is a new character on the lines. Read it."

If the 2 µsec. pulses are too fast for your terminal, you may have to lengthen them at the terminal with Single Shot Multivibraters such as a 74123 TTL.

Clarification of another, less common initialization of CB2 as an output is shown in the following chart:

B Section Control Register			CB2	
Bit 5	Bit 4	Bit 3	Cleared	Set
1	0	1	LOW on positive going transitions of first E pulse following write on B Data Channel after Control/Status bit 7 is cleared by read of B Data Channel.	HIGH when C/S bit 7 is set HIGH by CB1 active transition.

When CB1 goes active, Control/Status Register bit 7 is set HIGH (see top chart, page 6, 4PIO Manual). Anytime bit 7 is HIGH, CB2 is also set HIGH. To clear bit 7, the computer must do a read of the B Section Data Channel. Once this is done and bit 7 is cleared to a LOW, CB2 may be cleared by doing a write to the B Section Data Channel. The actual clearing will take place on the positive going edge of the first E pulse following the write.

ADDRESS CHANGES

Send notification of change of address to the attention of Customer Service/MITS, Inc./ 2450 Alamo SE/Albuquerque, NM 87106. Include your most recent mailing label from Computer Notes. Allow 4-6 weeks.

Notification of Change of Address

NEW ADDRESS:

NAME _____

ADDRESS _____ **APT.#** ___

CITY _____ **STATE** _____ **ZIP** ___

AFFIX COMPUTER NOTES LABEL HERE

88-PC PROCESS CONTROL BOARD

By Bill Kuhn

With the new Altair 88-PC Process Control Board, your Altair can now talk to the real world of motors, switches, relays, alarms, fans, lights, heaters, contacters, solenoids, and a host of other electromechanical devices. Installation of the 88-PC enables your Altair to water the lawn, turn lights on or off, control solar heating systems, or even act as a combination alarm clock and cook that will start fixing breakfast while you're catching those few extra minutes of sleep. The new 88-PC Process Control Board allows the Altair to do all these things and more!

The board has eight relay outputs, each capable of switching 1 amp at 120 VAC, and eight optically isolated inputs which can be configured by the user to accept a wide range of input signals. Two pairs of optically isolated "handshake" lines for control of and communication with external devices are also included. All lines are isolated and balanced for operation in electrically noisy environments.

With the 88-PC, the Altair can increase the effectivness of your security system by enabling it to act discriminately in dangerous situations. When a break-in occurs, the Altair can now "decide" whether a phone call should be made or an alarm should be sounded in the building or at the police station.

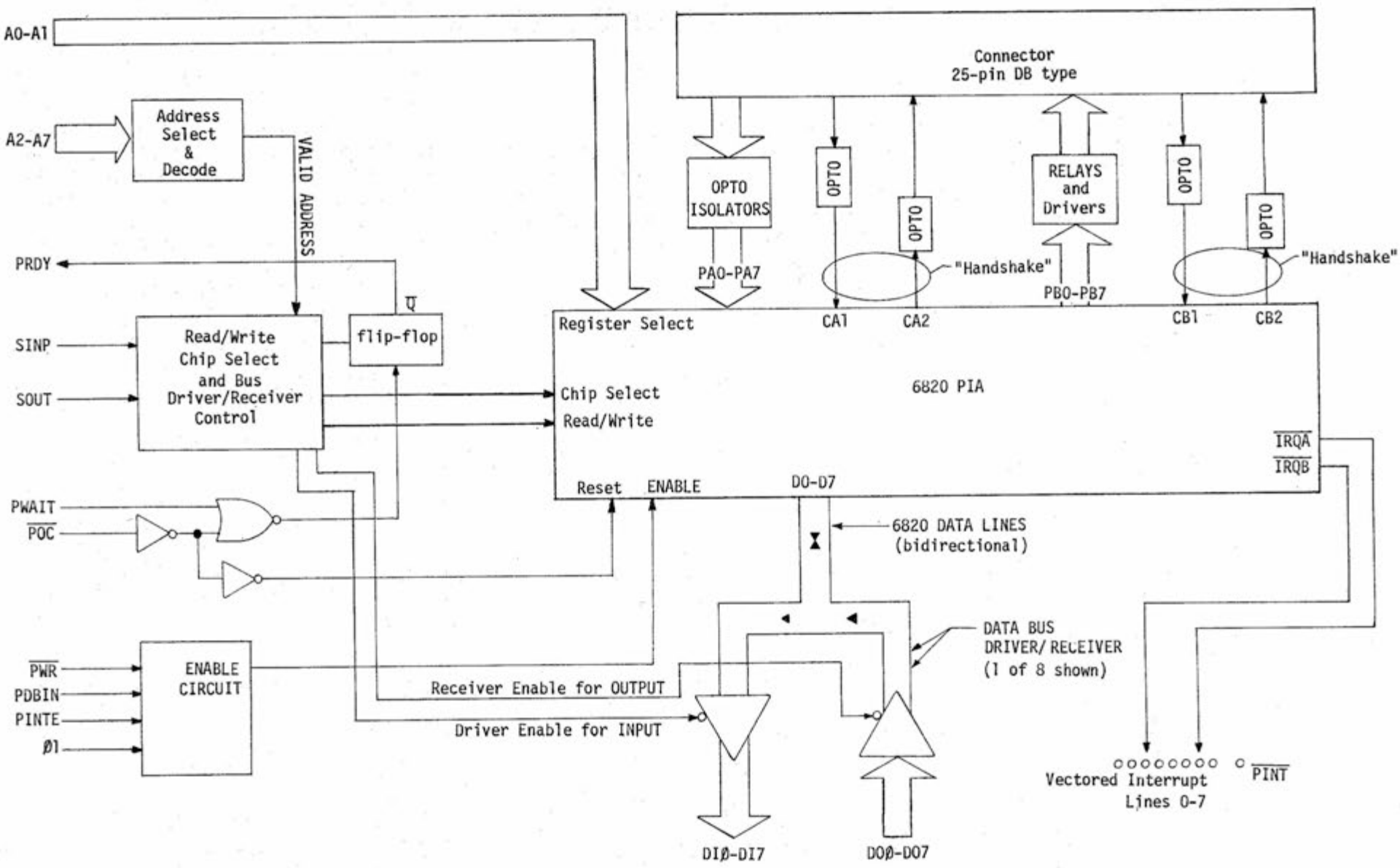
The 88-PC Process Control is quite simple to use. The manual contains extensive instructions for board set-up and initialization to adapt the board to the user's specific application. Once the board is set up and initialized, all that's necessary to control the relays is to output an eight bit word to the relay control channel. The lines set to a logic "1" by this method will energize their respective relays and those set to "0" will deenergize. An input from the OPTO isolator input channel reads the data from the OPTO isolated input lines.

The interrupt structure and use of the handshake lines are under software control. The interrupt request lines can be either enabled or disabled while the handshake lines can be set up to be high or low active. There are also a number of options for setting and resetting status flags.

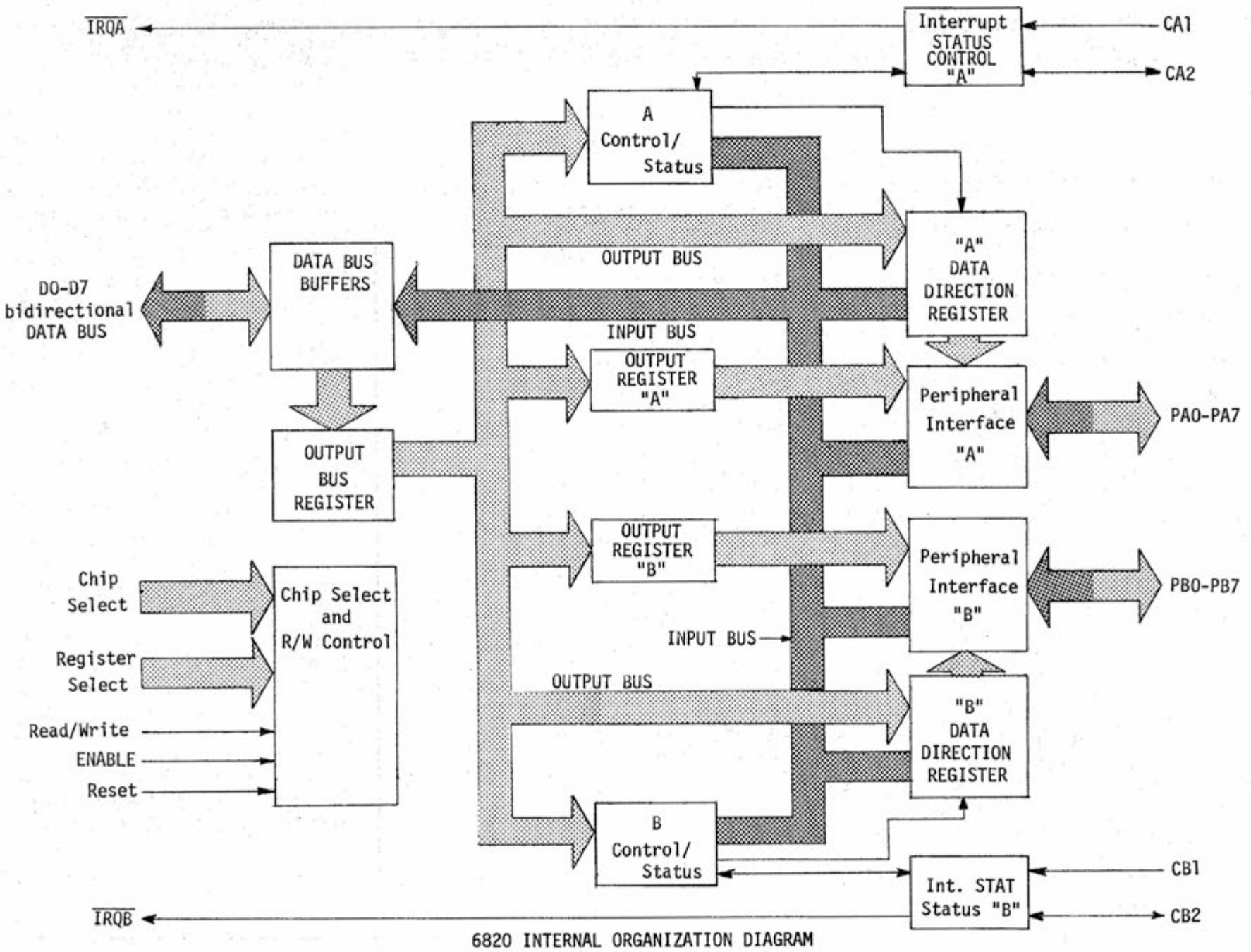
The following is just one example of how the 88-PC Process Control Board can be used for practical, real-world applications:

LAWN SPRINKLER CONTROL

Naturally the most important consideration is the amount of moisture in the ground. A humidistat could be used to signal the computer when the lawn needs watering. If this was the only consideration, a computer wouldn't be necessary--you could just hook the humidistat to a solenoid valve in your system.



88-PC PROCESS CONTROL BOARD BLOCK DIAGRAM



6820 INTERNAL ORGANIZATION DIAGRAM

But there are other important considerations, which include when and how much water is absorbed compared to how much might evaporate. This will help determine the most effective time to do the watering. A simple parameter to use here might be wind speed. A simple go/no-go device could be set up which would inhibit watering if wind speed exceeded a certain amount.

Temperature and relative humidity also have an effect on evaporation and could be sensed directly. A more simple system could be based on the fact that for any 24-hour period, optimum temperature and humidity occur at night. So a photo cell might be used to tell the computer if it's day or night.

Another important consideration is when the lawn is so dry that it's not possible to wait for optimum watering conditions. A more complex humidistat or simply an additional one with a different setting could be used to sense this situation.

Other possible inputs would be necessary if you wanted to sense and water separately in different parts of your lawn. This could be done with additional humidistats and solenoid valves for each different area of your lawn. If your city has premium prices for water usage in heavy use hours (this is becoming more common), an input from a clock telling the computer when the lower rate was in effect could be used. If there's an irrigation system as well as a normal water system in your area, you might want to make a point of knowing when there's water in the irrigation ditch. When water is present, the computer could pump water from the ditch to water the lawn.

The possibilities for use of the 88-PC in industrial process control applications are endless. Some of the following applications will be topics for future articles on detailed uses of the 88-PC.

--Monitor and correct temperature, pressure, pH, viscosity, etc., in chemical production processes.

--Automatic rejection or sorting of assemblies on production line based on presence or absence of parts as sensed by photo cell or finger switches.

--Control of test processes such as AC power, heater/cooler, electro-mechanical loading and collecting data either with just 88-PC boards or in conjunction with the 88-A/D Converter and the 88-Multiplexer.

--In a ham radio shack, switching on or off and controlling transmitters, receivers and recording instruments.

Continued on Page Eight

88-PC PROCESS CONTROL BOARD

Continued from Page Seven

88-PC Process Control Board Addressing

A0 (RS0)	A1 (RS1)	C/S RA-2	C/S RB-2	
0 (1)	0 (0)	X	X	A Section Control/Status
0 (1)	1 (1)	0	X	A Section Data Direction Registers
0 (1)	1 (1)	1	X	A Section Data (OPTO Inputs)
1 (0)	0 (0)	X	X	B Section Control/Status
1 (0)	1 (1)	X	0	B Section Data Direction Registers
1 (0)	1 (1)	X	1	B Section Data (Relay Control Output)

0=Logic Low (false)
1-Logic High (true)
X=Don't Care

88-PC Process Control Board

Preliminary Specifications

*OPTO Isolators

Isolation: Resistance, input to output: 100 G ohms
Voltage: 1500 volts min.

Input active current: 10-100 mA.

Propagation delay: Varies with diode current from input to data (D) line, diode and transistor:
turn on: 6 µsec., typ.
turn off: 10 µsec., typ.

*Relay Outputs

Isolation: Contact to contact: 750 VAC, 250 megohms
Contact to coil to frame: 2000 VAC

Contact data: Unloaded mechanical life: 20×10^6 operations
1 amp at 120 VAC: 5×10^5 operations
Contact resistance: 100 milliohms max (silver, gold overlay)

Propagation delay: Pull in: 3.5 nsec.
Release: 4 nsec.
Bounce: 1.2 nsec.

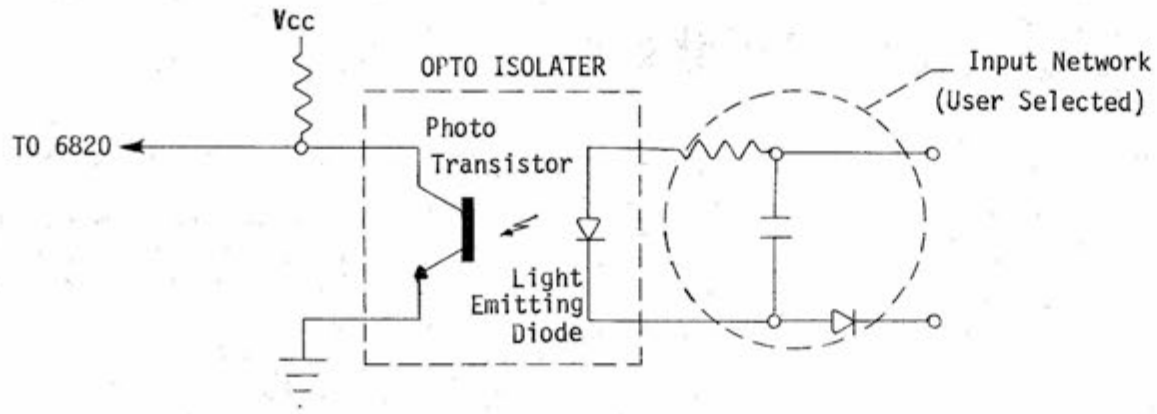
*Handshake Lines

Input: Same as OPTO Isolators
Output: Off leakage current @ 25°C: 10^{-9} amps
On current: approximately 30 mA.

*Supply Requirements

Unregulated 8 volt bus: Idle: approximately 150 mA.
All relays energized: add 200 mA.
CA2, CB2 on: add 100 mA.

OPTO ISOLATER INPUTS



Continued on Page Nine

OCTOBER SOFTWARE CONTEST

The software contest only had five entries this month, but the usefulness of the programs made up for the lack of quantity.

First place goes to John Robison for his Keyboard Entry/Display program. This is an interrupt driver program that can display and alter memory locations. It uses a keyboard for input and the upper eight address lights for a display. Since it's an interrupt driver, it can be used simulaneously with another program.

Robert Wilcox wins second place for his Basic Terminal Change program, which allows users who don't have the Console command to switch back and forth between two terminals. Data is supplied so the program can be used with version 3.2 4K, 8K, and ACR BASIC.

A rather unusual number guessing game wins third place for Alan Miller. The game involves trying to set the middle of a 3 x 3 matrix to 0 and add other entries to 1.

Philip Romanik submitted the winning subroutine. It splits an output string at the end of a word if the string would otherwise extend past column 72.

FIRST PLACE MAJOR PROGRAM
#9-27-761
Author: John Robison
Length: 265 bytes assembler
Title: Keyboard Entry/Display

Interrupt driven storage display/ modification program.

SECOND PLACE MAJOR PROGRAM
#9-17-761
Author: Robert Wilcox
Length: 16 lines Altair BASIC, 70 bytes assembler
Title: Basic Terminal Change

Program to allow version 3.2 of BASIC without Console command to switch to alternate console.

THIRD PLACE MAJOR PROGRAM
#9-8-761
Author: Alan R. Miller
Length: 50 lines Altair BASIC
Title: Teaser

Matrix number guessing game.

First Place Subroutine
#10-4-761
Author: Philip Romanik
Length: 20 lines HP BASIC
Title: Printer Subroutine

Program to break strings at word boundaries.

#9-10-761
Author: N. Craig Brown
Length: 20 lines Altair BASIC
Title: Christmas Tree

Program to print christmas tree.

PROM PROGRAMMER

By Bill Yates

The 88-PPC is a PROM Programmer designed to work with the 8800 (a or b) Altair systems. It programs the standard 1702A (256 byte) erasable PROMs in less than three minutes. The unit consists of a separate chassis (dimensions: 10.6" x 4.2" x 11") with a 24 pin zero insertion force socket. This is connected to the 8800 through its own interface card which plugs into the Altair bus. The programmer has a self-contained power supply, but the interface card requires about 500MA from the +8 volt bus in the 8800.

The programmer requires an 8800 (a or b) computer with an 88-PMC (PROM Memory Card), an SIO or 2SIO (Serial Input/Output Card) and some type of terminal device. The 88-PMC is required because the software driver for the programmer will be supplied on PROM. The serial I/O card and terminal are necessary for the verify program used to check programmed PROMs.

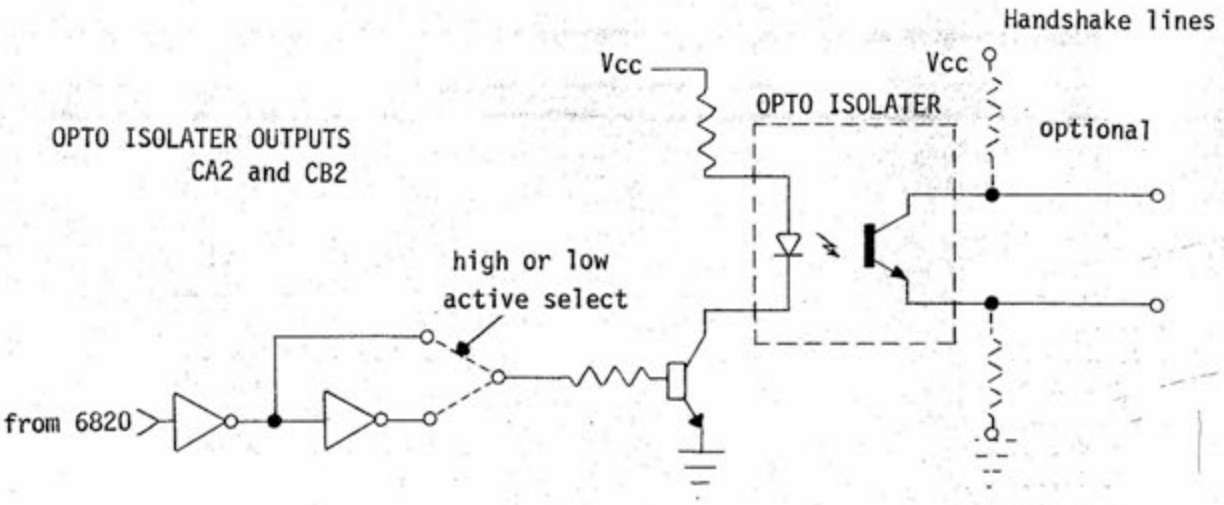
Functionally, the programmer looks like an addressable output port with two channels (even and odd). The even address channel outputs to a "control latch" in the programmer. The odd address channel outputs to either an "address latch" or a "data latch" depending on the state of the fourth bit in the "control latch". The "address" and "data latch" outputs drive the address and data inputs on the PROM through level shifters that generate the appropriate programming voltages. The outputs of the "control latch" drive the three control inputs to the PROM, the front panel indicators, and the "address/data latch" selection gating.

The 8800 controls all the programmer timing through its software driver. The PROM is programmed by doing a 256 byte block transfer from the 8800 memory (RAM or PROM) to the programmer. The driver does not currently have provisions for programming only part of the PROM. The programmer also does not have provisions for reading out the data from a programmed PROM. So programmed PROMs must be checked by using a standard PMC and doing a byte for byte compare.

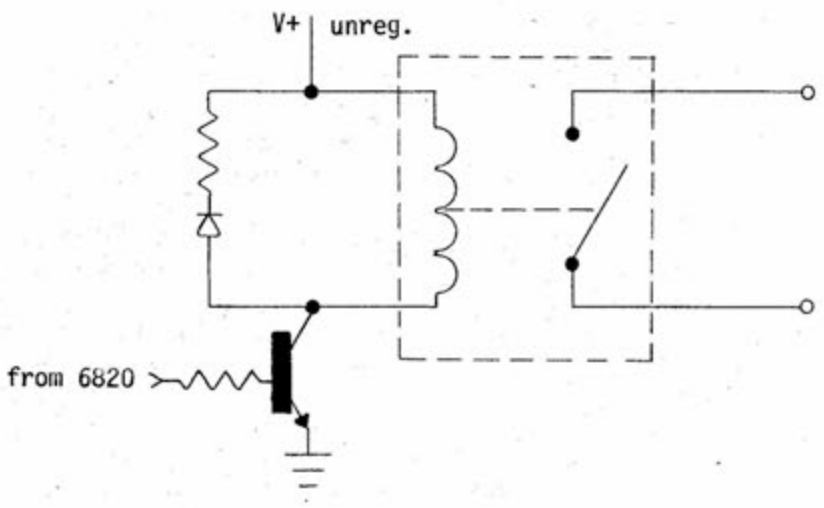
The Altair 88-PPC Programmer will be offered only as an assembled unit ($456). The driver will be supplied on PROM.

88-PC PROCESS CONTROL BOARD

Continued from Page Eight



Relay and Driver Outputs



classified advertising

For Sale: Intel SDK-80 Microcomputer with 512 bytes RAM, 2K PROM with System Monitor. Sacrifice $250.00. I would appreciate your help as I plan to use this money for a 16K RAM Board. Write to:

Steve Mastrianni
Sup., Applications Software
Gerber Garment Tech.
265 Prestige Pk. Rd.
East Hartford, CT 06108

or call:

(203) 528-9711, Programming Dept.

BASIC Super Star Trek.
Has features of Super Star Trek (as found in "The Best of Creative Computing--Vol I") but is optimized to fit in Altair 8800 with 16K total memory (including MITS 8K BASIC). Will run with other systems. High quality paper source includes 15-page description of game, program variables and routines; $3.00. MITS compatible tape cassette supplied on high-quality Maxwell UD tape; $6.00. Send to:

D. C. Mitchell
2S624 Mulberry Ct.
Warrenville, IL 60555

altair users

Anthony J. Bacchi
60 Frederick Ave.
Medford, MA 02155

Phil Schneider
3523 Bromley Dr. S. E.
Grand Rapids, MI 49508

616-245-9813

Edwin F. Hampton
6950 SCT4 GP
Box 36
APO New York 09193

Toivo Rusokallio
Karikkokatu 4
53100 Lappeenranta 10
Finland

Daniel A. Enger
(new address)
928 St. Joe
Rapid City, SD 57701

Jose Mauricio Campos Salaverria
25 Ave. Sur y 6 Calle Pte 1405
San Salvador, El Salvador

John Swain
3687 NE County Line Road
Buck Creek, Indiana

Skip Pearce
8013 North 11th Ave.
Phoenix, AZ 85021

Ken Wolfe
150 No. Railroad St.
Palmyra, PA 17078

Installation of Second Transformer for the 680b

By Ron Scales

To expand the 680b computer to three boards, it may be necessary to add a second transformer. When this second transformer is properly connected, the supply current will be doubled without increasing the supply voltages.

In order to meet the above requirements, these two transformers must be wired in parallel and it is important to wire them so that their phase relationship is identical.

Pay close attention to the following procedure because each step is critical. Check both of transformers and make sure they match each illustration exactly. Failure to wire these transformers correctly may result in permanent damage to your unit.

1. Disconnect the first transformer completely from the 680b back panel by unsoldering the primary leads from the terminal strip and removing the terminal pins from the secondary leads. Be sure to leave all the wires long enough to reach their destination.

2. Place both of the transformers upright in front of you and straighten and separate all of the wires coming from the coil winding.

3. Place the transformers so that the side with the two black wires is facing you. If the new transformer has two blue wires on the same side as the black wires, cut off the blue wires.

4. Tie a knot in the middle of the black wire that enters the coil winding the closest to the left side. (See Figure 1.)

5. Turn the transformers around so that the side with five wires coming from the coil winding is facing you.

6. Tie a knot in the middle of the red wire that enters the coil winding furthest to the right side. Do the same to the white wire that is furthest to the left on each transformer.

7. Strip off approximately 3/8" insulation from each of the secondary wires on both transformers. Now position the red wires with knots in them according to Figure 2. Wrap the bare end of one knotted red wire once around the other and solder them together. Heat shrink (if used) may be inserted now on the wires before crimping on the terminal pins.

8. Now crimp the terminal pin on the wire and use a little solder to insure a good connection (see Figure 3). Repeat steps 7 and 8 with the remaining secondary wires making sure that the two knotted white wires are tied to each other and also that an extra 8" wire is tied to the two orange wires before crimping the terminal pin on. The third wire connected to the two orange wires is to insure a good chassis ground for the system. After all the secondary wires have been crimped on, it should resemble Figure 4.

9. Separate each of pins by cutting tabs as close as possible to each of the terminal pins.

10. Beginning with the two knotted white wires, orient the terminal pin and the 5-pin female connector according to Figure 5. Now, push the pin into the female connectors' slot until it locks into place.

11. Make sure you put the remaining wires into the connector in the correct order (white, white, orange with ground, red, red) according to Figure 6. When all the wires have been inserted into the connector, it may be necessary to insulate the bare wires with either heat shrink or electrical tape.

12. With the secondary wiring completed, install both transformers according to Figure 7. The terminal strip is mounted with the top screw over the second transformer.

13. Instructions for wiring up the terminal strip line cord and fuse holder are covered in the 680b Assembly Manual for the back panel. But it's essential to remember that the two primary wires with knots on them must be tied to the same terminal on the terminal strip. It's also advisable to replace the 1/2 amp fuse to a 1 amp slow blow fuse.

If the preceding steps have been followed exactly, the finished back panel should look similar to Figure 8. Your 680b power supply should then be able to handle 3 add-on boards to your system without any problems.

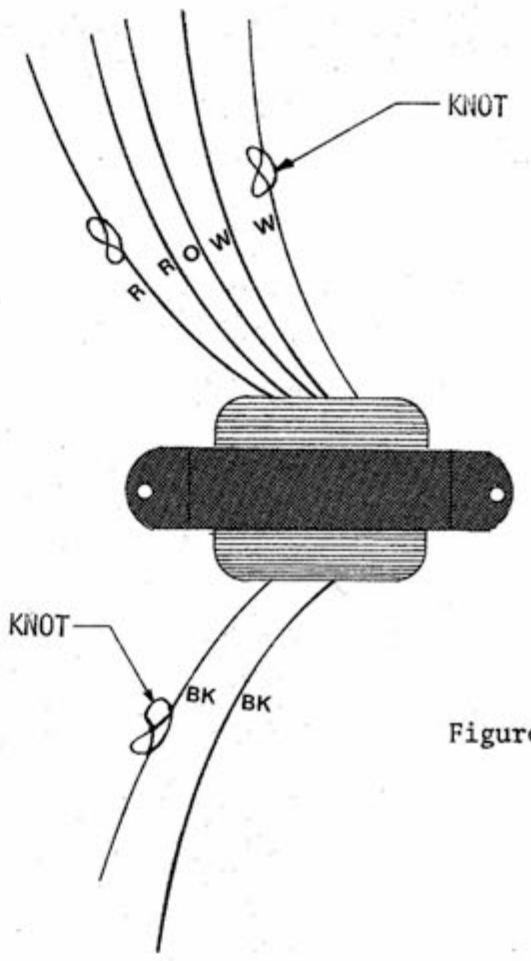


Figure 1

Figure 2
Knotted Red Wires

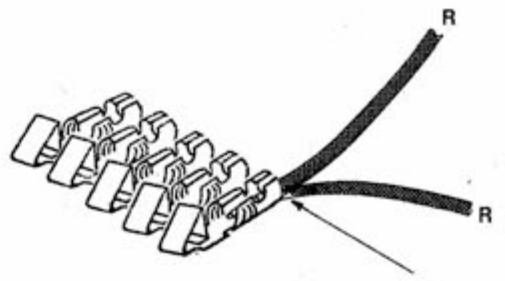


Figure 3

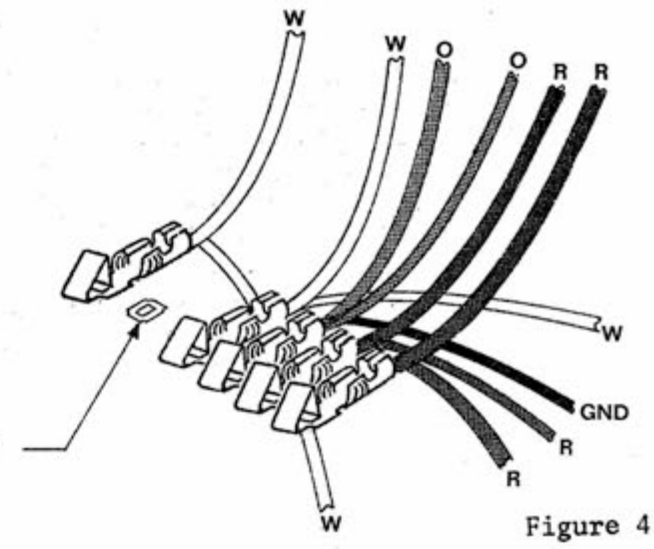


Figure 4

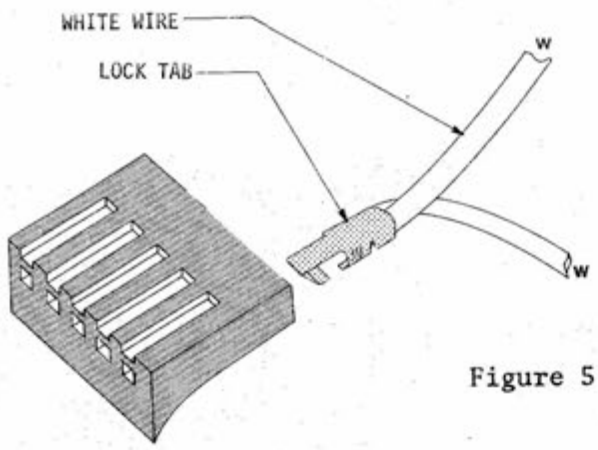


Figure 5

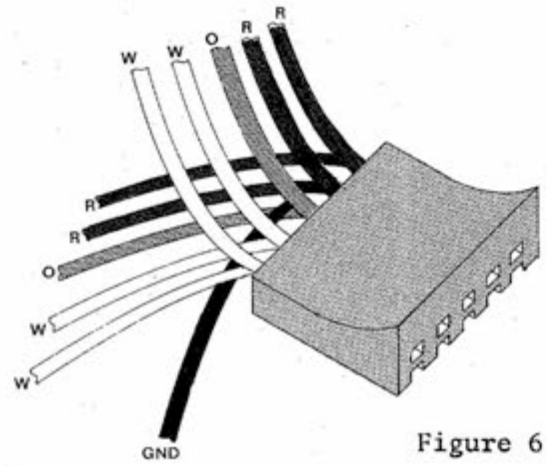
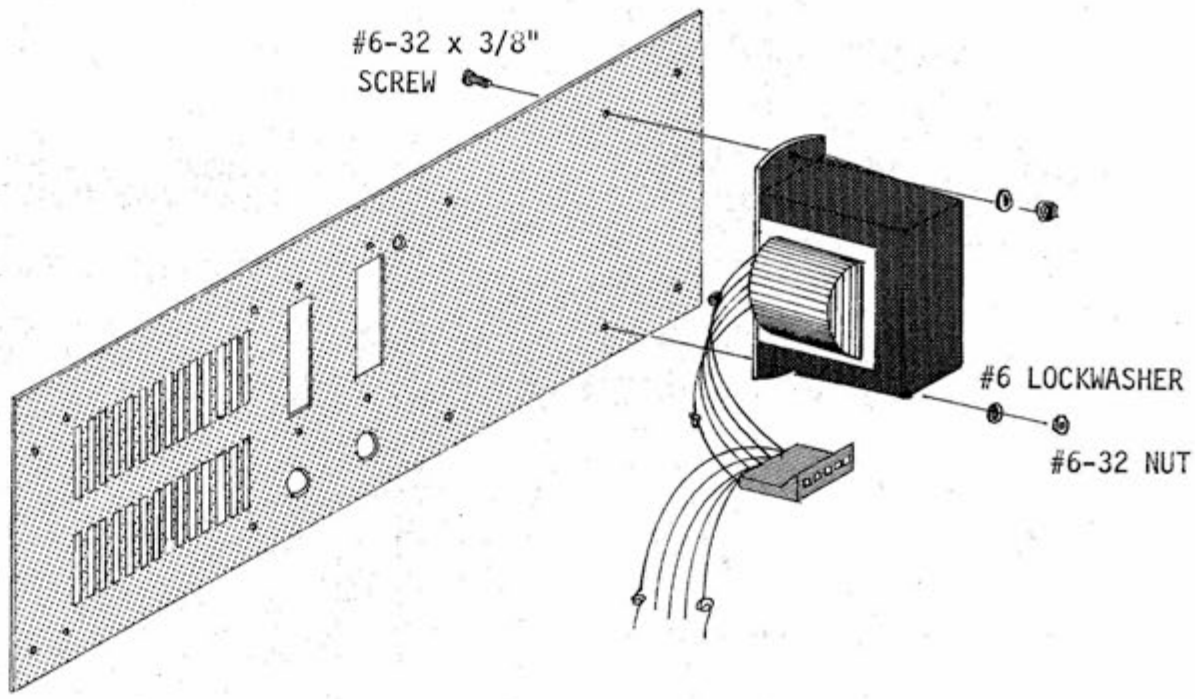


Figure 6



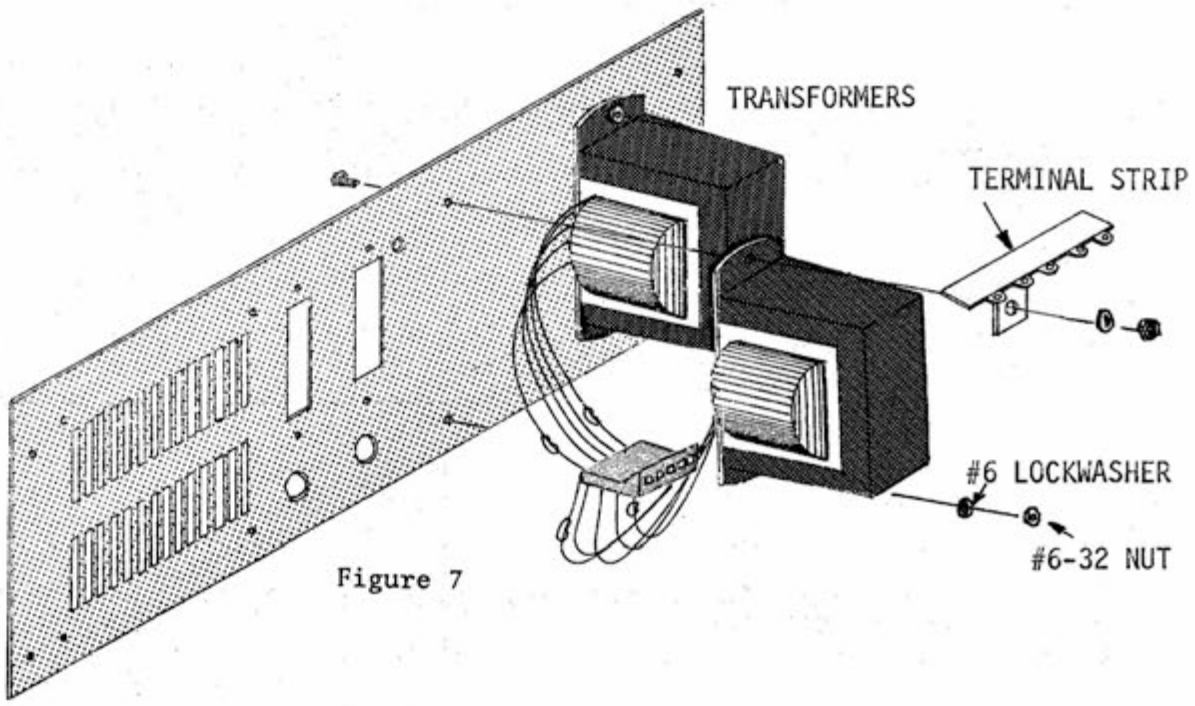


Figure 7

Figure 8

kilobaud

Another New Computer Hobby Magazine!

Ho-hum.

Ho-hum yourself! That's the very reason that a NEW magazine is needed and that's why *you* will be reading it. Have you ever heard a ho-hum for Wayne Green's *73 Magazine*? Well no one will be ho-humming *Kilobaud* either; it will be brought out by the same staff.

Do you know why a recent review of ham magazines in a computer hobby magazine totally avoided any mention of *73*? Could it be because *73* is a competitor and has been running an impressive computer hobby section (I/O) for almost a year — a collection of the best stuff in the field?

HERE ARE TWENTY REASONS WHY YOU CAN'T ESCAPE READING KILOBAUD:

#1 Innovations...

Kilobaud will publish detailed descriptions of the latest hardware, written by the chaps who designed the equipment. Who knows it better? These articles will explain the reasons behind the circuit designs, why particular chips and configurations were chosen and, of course, the trade-offs. The authors are fully aware that a truly candid approach is necessary for credibility. For starters you can look for an article on the brand new Z-80 CPU by MITS!

#2 Fundamentals...

Since the growth of the computer hobby field depends on newcomers, expect a lot of very basic articles in *Kilobaud* — hopefully at least 1/3 of the magazine — to help get novices up to speed. And most of us are novices at one thing or another — at hardware, software, systems or servicing There are already more than enough publications for PhDs.

#3 Programs...

Wouldn't you like a continuing library of smaller programs and algorithms . . . routines . . . things to save you time when you are working on software? The rate of payment for articles in *Kilobaud* should result in a lot of published software. Hopefully, this will eliminate the present need for organizing user groups to swap software.

#4 Money-making opportunities...

These will be a BIG thrust of *Kilobaud*. Computer hobbyists will have a golden opportunity to cash in on their hobby by writing articles and books, by developing hardware for the manufacturing firms, by starting their own companies to manufacture and market their own products, by opening computer stores, and by writing programs to be sold in bulk via *Kilobaud* to the computer stores. *Kilobaud* will keep you abreast of the many ways to capitalize on the enormous growth of small computers . . . how to get into business . . . how to obtain lines to sell . . . how to service computers . . . things like that. If you don't turn this hobby into a big profit maker for yourself, you are missing the boat.

#5 Problem-solving...

Have you ever wondered what problems others have had with the equipment or kits presently on the market? *Kilobaud* will be publishing a lot of dope on how other hobbyists have solved their problems. You'd do well to keep an accurate log of every problem you've run into with your own system — missing parts, bum instructions, and of your troubleshooting techniques and solutions. These notes will make invaluable letters to *Kilobaud* to help others over the rough spots.

#6Your claim to fame...

There is really only one way to develop a reputation in any field — be published. *Kilobaud* will pay you well for your writing time. And then there's the fame involved with being published (this sure helps at raise time or when you're looking to change jobs). It particularly doesn't hurt if you are working in the computer hobby field. At about $50 per page, *Kilobaud*'s rates nearly double those of any other magazine. It's easy to write for *Kilobaud* . . . send for the poop sheet from *Kilobaud* — that's all you need to get started.

#7 - #20

There are thirteen more important reasons why you will be reading *Kilobaud* and enjoying it. We haven't thought them up yet, but we will.

#21 This is a bonus...

The *Kilobaud* lab is already hard at work setting up all of the hobby computer systems . . . Altair 8800 with floppy disk system . . . Wave Mate, Kim, Mek, SWTPC, Holt, Imsai, Sphere, etc. Not only will this really familiarize the *Kilobaud* staff with these systems, but it will enable us to test all software sent in on real working systems. No other magazine has anything like it. No other is so deeply involved with the field.

The first issue of *Kilobaud* will be out in late November — the January 1977 issue. Don't take any chances on missing this issue. We suspect that it will soon be in short supply and probably worth more than the Charter Subscription price. And we promise not to stash away any first issues from Charter Subscribers, so we can sell them later at a premium price . . . or wasn't that what was going on at Atlantic City?

Readers of *73 Magazine* have been asking if *Kilobaud* will be more software and system oriented. Articles on ham computer applications will continue in *73* along with fundamental articles to bring amateur radio operators up to speed in this new field. Almost 1/3 of the computer hobbyists at Atlantic City got into computers via the *73*'s I/O computer section and the nearly 100 articles published on computers in 1976. Yes . . . the *73* I/O section will continue!

Kilobaud is going to be big — we're hoping to start out with a 160 page issue. It will be professional, stimulating and a lot of fun. The magazine will sell for $2 through computer stores and is available for $15 per year from *Kilobaud*. A special Charter Subscription price (until December 31) is $12 for one year and $25 for three years. Master Charge, BankAmericard or American Express charge card numbers will take care of everything by mail, or phone 1-800-258-5473 or 1-800-251-6771 (subscriptions only). For subscriptions or information write *Kilobaud*, 73 Magazine Street, Peterborough NH 03458.

kilobaud

charter subscription

☐ 1 Year @ $12 ☐ 3 Years @ $25

Name _____

Address _____

City _____

State_____ Zip _____

☐ Check enclosed ☐ Cash enclosed ☐ Bill me

☐ Bill BankAmericard # _____

☐ Bill American Express # _____

☐ Bill Master Charge # _____

Interbank # _____

Expiration date _____

Signature _____

or call TOLL FREE (800) 258-5473 OR (800) 251-6771

Kilobaud Magazine, Peterborough NH 03458 CN