ALTAIR CONVENTION

by David Bunnell

By the end of January, 1976, it is hoped that MITS will be in its new facility near the Albuquerque Airport. Internal construction has been in progress for several weeks, and plans call for moving the production department before Christmas. Administrative departments will move during January.

In connection with this move, there will be a combined open-house, World Altair Computer Convention, tentatively scheduled for March, 1976. All Altair owners will be invited to attend this convention.

While formal plans for this convention have not been finalized, an outline of this event calls for a weekend of seminars and demonstrations. The seminars will be conducted by MITS engineers and software writers and will cover a wide range of topics of interest to Altair users.

Hobby clubs and individual Altair owners are asked to bring their Altairs to Albuquerque for demonstration purposes. Prizes worth several thousand dollars will be awarded to the best demonstrations in each of an undetermined number of catagories. (We will be looking for unique applications as well as well-developed traditional applications.)

Since the new MITS plant is located close to the Albuquerque Airport, it will be very convenient for out-of-town Altair users to attend the convention. Hotel reservations can be made at the Albuquerque Airport Marina Hotel across the street from the terminal building, and from there to the factory it is just a short walk.

Once a definite time table has been established for the first WACC (World Altair Computer Convention), all Altair owners will receive a schedule of events and a personal invitation. Attendance to the convention will be free, and in addition to prizes for the best demonstrations, MITS will provide door prizes and at least one free luncheon.

People who have suggestions for this conference are asked to address their letters to: WACC/MITS
2450 Alamo SE
Albuquerque, NM
87106
Attention:
David Bunnell

With the help of the thousands of Altair users, it is hoped that this convention will become one of the most exciting computer conventions in the industry and that it will become an annual event.

BYTE Sponsors ACR Standards Meeting

Users and manufacturers of audio cassette data recording hardware attended a symposium sponsored by BYTE magazine in Kansas City, Missouri on November 7th and 8th, 1975. The purpose of the meeting was to investigate the various methods of recording data on audio tape and to make a recommendation on the method that would meet the requirements of a low cost, reliable system for interchange of tapes.

Those in attendance included representatives of BYTE magazine, Popular Electronics, MITS, Processor Technology, SWTP, Godbout, The Computer Hobbyist, Pronetics, LGC and Sphere.

The first portion of the meeting was concerned with the minimum requirements of the cassette recorder. It was found that speed tolerance was the biggest variable, with speed variance as large as ±10% possible. Minimum frequency response was set at 500Hz to 3KHz ±3db bandwidth, which is easily attained by most low cost cassette recorders. Other parameters such as start/stop time, remote control and tape characteristics were discussed. It was agreed that the standard should include provision for remote start/stop of the tape recorder, and that low noise audio tape (not bargain pack tapes) should be used.

The next part of the meeting concerned the bit format for recording data. The format decided upon was a UART style, 11-bit word to be recorded at 300 bits per second (300 baud). The 11-bit word consists of a start bit (logic Ø), eight data bits, and at least two stop bits (logic 1).

The rest of the meeting was spent primarily on discussion of modulation techniques. Of the methods presented, there were three main groups. One was pulse modulation (The Computer Hobbyist, Tarbell), which has the advantage of high data density and data rate, but the disadvantage of more complex hardware. A second type of modulation considered was the HIT system (Processor Technology, Popular Electronics). It was decided to be too software dependent and possibly amplitude sensitive. The last group of techniques discussed was frequency modulation, including the 103 type (MITS, LGC) and the Lancaster method (SWTP, Pronetics). Frequency modulation has the advantage of somewhat simpler hardware, without being software dependent. The meeting concluded with an agreement to investigate the Lancaster method as the possible modulation standard for hobbyist and user interchange. (The Lancaster method uses 2400Hz = logic 1, and 1200Hz = logic Ø with the advantage of being synchronous with UART clock inputs.)

COMPUTER NOTES

November/December, 1975

Volume One Issue Six

A PUBLICATION OF THE ALTAIR USERS GROUP

Across the Editor's Desk

by David Bunnell

As I recall, the last issue of Computer Notes was the October issue and you got it around November 11. Right?

This issue of Computer Notes should logically be the November issue, however, if you get it before 1976, the December issue won't arrive until February.

Therefore, this issue of Computer Notes is the Holiday Issue.

Now, that should make a lot of sense.

However, doesn't that mean that people who paid $30 to be members of the Altair Users Group will get one less issue of Computer Notes before their subscription runs out? No.

Effective upon the arrival of this issue, you automatically have a one-year subscription. Even if you subscribed back in July, your subscription now begins with the coming January issue (we've got our fingers crossed).

Hopefully, we will learn to be more punctual with this publication.

I think you'll find our Holiday Issue well worth the wait. Included are two four-page inserts from Byte Magazine and Creative Computing. These inserts were prepared entirely by the respective staffs, as was the PCC insert in October's C/N, and I hope they give you an idea about what these two publications are about.

Creative Computing is published by David H. Ahl, who in my opinion, is the best writer in the "popular" computing field. It is aimed primarily at the educational market.

Byte Magazine, which I reviewed in September's C/N, is aimed at the hobby market. There have been recent changes at Byte and I see these changes as resulting in better editorial content.

Both publications are excellent and since they have completely different editorial slants, I would recommend that the serious "computer freak" subscribe to both.

Another magazine trying to make it on a national scale is Interface, which is published by the Southern California Computer Society. They have a long way to go if they want to compete effectively with Byte and Creative Computing, but I understand their first issue is going like hotcakes. This issue features the two Altair computers on the cover in a stunning painting by Kim Behm (the artist who drew the "Created by Man" illustration). A list of articles includes: "Toward the design of a Micro-operating System," "Teleprinter Maintenance/Repair/Preventative Maintenance," "Altair Switch Memory," "1+1=10?" and "Soup Up Your T.V. Typewriter."

Enjoy this issue of Computer Notes, and have a happy holiday.

LOCAL USERS GROUPS

Atlanta Area Microcomputer Hobbyist Club
Jim Dunion, President
421 Ridgecrest Rd.
Atlanta, GA 30307
(404) 373-8990
Meetings are held the last Wed. of each month, 7:30 p.m., at
The Calhoun Co.
6000 Peach Tree Rd.
Atlanta, GA

Computer Hobbyist Group of North Texas
Bill Fuller
(214) 641-2909
Neil Ferguson
(817) 461-2867
Lannie Walker
(817) 244-1013

The Amateur Computer Group of New Jersey
Sol Libes
(201) 889-2000 (day)
277-2063 (eve)
George Fischer
(212) 351-1751

29 Palms California Area Group has changed the contact phone number and address to:
Sgt. Wesley Isgrigg
74055 Casita Dr.
29 Palms, CA 92277
(714) 367-6996

Homebrew Computer Club
Robert Reilling, Editor
193 Thompson Square
Mountain View, CA 94043

HP-65 Users Club
% Richard J. Nelson
2541 W. Camden Place
Santa Ana, CA 92704

UCLA Computer Club
3514 Boelter Hall
UCLA
Los Angeles, CA 90024

San Diego Club
% Gary Mitchell
Box 35
Chula Vista, CA 92012

-CONTINUED Page 24-

ALTAIR SERVICE DEPT.

Barbara Sims

New Coding Form

The Users Group Library has a new Coding Form for program submission that is much more efficient than our old form. A sample copy of the new form is enclosed -- it consists of a cover sheet (to give pertinent information about the program) plus an additional sheet of program listing space. Coding Forms are purchased from MITS in batches of 50. When you order the new Coding Forms, please specify how many cover sheets you want included in that batch. (One cover sheet for each program to be submitted.) If you still have copies of the old form, feel free to use them up before ordering the new one.

Programs submitted to the Users Group Library become available to other members much sooner when our coding form is used. The entry rules for the Software Library are as follows; your cooperation in following these rules will be greatly appreciated.

In order that programs can be reproduced for distribution to other Altair owners, they should be typewritten using a ribbon that produces fairly dark type. It is desirable that assembly language and machine language programs be submitted on official Coding Forms or copies of such forms. Coding Forms are available from MITS at $2.00 per 50 to cover printing, postage and handling.

All entries should be on 8 1/2" x 11" white paper. Teletype printouts and Xerox copies are not acceptable.

C/N Delivery Dates

I've had quite a few letters and calls regarding the delivery of Computer Notes. Perhaps it will help if I outline exactly how we mail Computer Notes to different areas:

I. Bulk Mail
  A. Any address with a zip code
    1. United States
    2. APO
    3. FPO

II. Third Class Mail
  A. Canada
  B. Mexico

III. Air Overseas Mail (AO)
  A. Countries overseas

—continued page 5—

Ramblings from Ed Roberts

by H. Edward Roberts,
President, MITS

This is a continuation of my random ramblings concerning the small computer industry, the Altair and MITS. If you have any comments on any or all of these subjects, please write me.

PROBLEMS:

The 4K memory cards are still in a heavy back order situation. The majority of the 4060-2 (4K RAM chips) we have received in the last few months have been diverted to replace the 2604's which were shipped in August. There are now two shifts operating in the Repair Department in order to eliminate this problem. If you have a new order for 4K memories or if you have a 4K board at MITS for repair, please bear with us for the next few weeks.

PHILOSOPHY:

There are a number of questions raised each month concerning the Altair design philosophy. Therefore, starting with this issue I will specifically discuss the most common ones.

1) FRONT PANEL READY: The front panel ready provides an active pullup to the system ready line. Therefore, in the run mode if a bus card pulls the ready line low it must fight the front panel pullup, this is an intentional design and does not create any problems. (Incidentally, a check of the Signetic data manual on the 8797 will show that this is an acceptable condition.) Nevertheless, if you are concerned about this, move the PRDY line to the XRDY line on the system bus wiring from the front panel. That is, move the wire from pin 72 to pin 3 coming from the front panel harness. The XRDY was designed into the system to allow for a simplified DMA which is no longer used by MITS. One mod which has been suggested is move the PRDY line on each card to pin 3 on the card, don't do it. This will cause major conflicts in the system if additional peripherals are added. Some of them are looking for a PRDY from other system cards. If you are using a non-Altair card that uses the ready line, make sure it is connected to PRDY and not XRDY. For that matter, don't use any modification in your system if it doesn't appear in Computer Notes or another MITS publication.

2) POWER SUPPLY: There have been a number of questions raised concerning the power supply used in the Altair. The basic Altair is shipped with a transformer which is rated at 8V and 8 amp. It turns out that in practice if the current load exceeds 3-4 amps (6-8 Altair cards) the unregulated bus voltage drops to the point where local regulation by the 7805 is marginal. But if you have a system which contains 6 or more MITS cards, we will provide you with a higher voltage transformer at no cost. Due to the heavy 5 volt current requirement of some of the 4K static memory cards (discussed in previous issue) you may run out of voltage with as few as 4 cards in the system. The rectifier in the power supply is rated at 10 amp. Which is 20% over the maximum current required by a fully expanded Altair using MITS cards, i.e., .5 amp average per card and 16 cards. The higher voltage transformer should not be used in small systems, i.e., under 5 cards, because it places an unacceptably heavy thermal load on the 7805's.

Another point raised about the power supply is why didn't we use the lower cost single regulator for the whole system, i.e., not use card regulation. There are two primary reasons: (a) the on card regulation significantly improves system noise immunity (b) and most important a castrophic failure in a single regulator system could destroy literally thousands of dollars worth of components.

3) CARDS: There are two comments raised concerning the DC cards used in the Altair. Some cards in the earlier Altairs did not have gold fingers, of course all Altair cards produced recently have gold fingers. Contrary to popular opinion the gold has more esthetic value than practical value. If you have an early Altair board which doesn't have gold fingers, effective immediately you have a 5 year warranty on the fingers.

A valid criticism that was made concerning early Altairs was related to the 1/16" thick motherboard. While this motherboard didn't create an electrical problem, it wasn't rigid enough to support the card guides properly, in a word it was sloppy. Of course, all the later Altairs have 1/8" motherboards. If you have an early Altair and if you are willing to remove the edge connectors from your mother card, let me know and we will send you a new 1/8" card free!

Next month I will discuss other commonly asked questions or criticism.

PROCESSOR COMPARISON:

I have been asked by a number of people the results of our MPU (microprocessor unit) evaluations. We have probably had more experience in designing general purpose MPU hardware and software than any other group in the world. Therefore, our evaluations may be useful and hopefully interesting to anyone involved with small computers. I'm sure it goes without saying that we have a very active internal program concerning evaluation of all MPU's. Since we have no ties with IC manufacturers, our evaluations are essentially unbiased. Unfortunately, non disclosure agreements prevent us from discussing some of the newer unannounced MPU's. The following is a list of the processors which we have evaluated in detail.

Intel 8008, 8080, 4004, 4040
Motorola 6800
MOS Technology 6500 series
National PACE, IMP 8
Fairchild F-8

In order to keep this evaluation short I have reduced our internal criteria to the following major items:

1) Design Complexity of CPU - this is important only to the manufacturer and to the home brewer.

2) Hardware Flexibility - this is a measure of the flexibility of the system to interface to other hardware.

3) Memory Efficiency - a measure of how efficiently the system uses memory particularly important from a cost standpoint.

4) Power of Instruction Set - all measure of software power must be done on the basis of relatively large software packages, small or singular routines tell nothing about system power.

5) Overall System Speed - this is a measure of both hardware and software speed.

6) Availability of Software - a measure of which systems have the largest amount of manufacturer supplied software.

7) Reliability of Primary Vendor and Number of Second Sources - this will give an indication of the availability of new software and compatible hardware.

8) Overall System Speed - this is really the best single measure of system performance, but is only applicable when related to large real-world software packages, we use BASIC as a comparison. The following is a thumbnail chart.

—CONTINUED PAGE 4—

Ed Roberts, continued

	PACE	F-8	6500	6800	8080	8008
Complexity	G	E	E	E	F	D
Flexibility	F	G	G	G	E	G
Memory Efficiency	F	G	E	E	E	P
Power of Instruction	G	E	F	G	E	P
Avail. of Software	P	P	P	G	E	P
Efficiency of Software	F	E	G	G	E	E
Speed	F	E	G	G	E	E
Reliability of Vendor	F	G	G	G	E	E
Suitability as a G.P. Processor	P	P	F	E	E	F

E - Excellent
G - Good
F - Fair
P - Poor

There are several things that are apparent from the evaluation. The 24 bit byte oriented processors are in general superior to the 16 bit processor (PACE, the IMP 16 is not even in contention). The "old" 8008 is almost as good a processor from a general standpoint than some of the newer processors, of course, it is significantly slower. The 8080 is still the most powerful single chip LSI processor available but the 6800 is certainly good competition. As a matter of fact in simple minimal systems the 6800 is preferred due to the simplicity of interfacing, it is the logical choice for someone who is planning to homebrew a small system. If you are interested in a more detailed report on our evaluation, let me know and we will write a full and complete article on the subject.

CLUBS:

The majority of the computer clubs that exist today are a result of the catalyst provided by the MITS road show. We are very interested in continuing to provide whatever assistance we can to clubs. At the present time our typical customer is relatively sophisticated in terms of general electronics technology and/or software, but the vast majority of the potential customers in this hobby are not sophisticated. These new people are certainly needed by MITS and other manufacturers and their volume will benefit either directly or indirectly everyone in the hobby. A large percentage of the education and assistance getting these new groups up and running will have to come from the clubs, there is no other place. Our present efforts with the van only scratch the surface, incidentally the present van show is a basic technical presentation and not a sales pitch. We are actively working on a number of different courses which may prove useful to many of the clubs.

Any club which would like to have a technical group from MITS come to their meeting, please let me know and we will be there. If you would like an equipment demonstration and/or sales pitch we have representatives and dealers scattered all over the country who would love to have the opportunity to attend your meetings.

PARANOIA:

I have received a number of letters from readers of this publication concerning our counter attack criticism is very gratifying in the sense that it indicates that our typical customer is more than sophisticated enough to establish the attacks on MITS and the Altair for what they are. Therefore, in the future we will address these sort of comments only in response to letters you send us and in the general design philosophy section of this column.

NEW PRODUCTS:

As I indicated last month as a general rule we will not release information on new products until they are production ready. But there have been some inquiries concerning the -2 and -II designation on some of the newer cards produced by MITS. Cards designated in this way are designed to be compatible with the until now unannounced Altair II. Let me just say the Altair II will be at least as revolutionary as the original Altair was when it was introduced. All -II cards have downward compatibility with the Altair 8800 systems, non-II cards probably won't interface with the Altair II.

Another product which has gotten a good bit of unintentional publicity is the MITS CRT graphics terminal. This is a product that has been under active development for more than two years. Because of the great amount of interest in this product let me break my own rule and give you some preliminary data.

In the alpha display mode it will present 80 characters per line, 24 lines per page. Up to 32 pages can theoretically be stored within the machine. There are two graphic modes, in the low resolution mode there are 256 x 256 elements. Any number of graphics pages and alpha pages may be interlaced and displayed under real time software control. The brightness of each frame in an interlace mode may be controlled. Control logic allows the system to be used with a standard color monitor if desired, i.e., full color graphics and alpha numerics are possible.

The data transfer rate between display and CPU is at main frame baud rate. Of course, the display has upper and lower case capability, black on white and white on black and any of the other things you would expect in a terminal of this quality. The cost of this system will be significantly greater than the T-V typewriter type of display, but the capability will be much greater.

Our internal scheduling calls for shipments to begin in February, but you know how that goes. Nevertheless, we will not accept orders on this product yet, but if you would like to reserve a unit pending firm pricing and delivery schedules, write a letter to Barbara Sims and she will assign you a slot in production.

PUBLICATIONS:

We have mentioned the Computer Hobbyist in previous issues of Computer Notes, but I feel that it is such a valuable publication for the serious hobbyist it should be considered required reading. You can obtain a subscription by writing to the following address:

The Computer Hobbyist
Post Office Box 295
Cary, North Carolina 27511

Please send any comments or suggestions to me directly at MITS.

Highest regards,

Ed

David B. Porter
528 Carr Ave.
Rockville, MD 20850

Jim Gerow
2525 McCue #243
Houston, TX 77027

Peter Jarvis
10545 Ashworth N.
Seattle, WA 98133

Jerry K. Kozelsky
5711 Sarvis Ave. Suite 502
Riverdale, MD 20840

Jack J. Keith
3435 Mansfield Rd.
Falls Church, VA 22041

Olin A. Williams, Jr.
2871 Tony Dr.

continued

E	E	G	G	E	G	Flexibility
E	G	E	E	E	F	Memory Efficiency
G	E	E	G	E	P	Power of Instruction
P	P	P	G	E	G	Avail. of Software
F	E	G	G	E	G	Efficiency of Software
F	F	G	G	E	P	Speed
P	G	G	E	E	G	Reliability of Vendor
F	P	F	G	E	G	Suitability as a G.P. Processor

E - Excellent
G - Good
F - Fair
P - Poor

There are several things that are apparent from the evaluation. The 24 bit byte oriented processors are in general superior to the 16 bit processor (PACE, the IMP 16 is not even in contention). The "old" 8008 is almost as good a processor from a general standpoint than some of the newer processors, of course, it is significantly slower. The 8080 is still the most powerful single chip LSI processor available but the 6800 is certainly good competition. As a matter of fact in simple minimal systems the 6800 is preferred due to the simplicity of interfacing, it is the logical choice for someone who is planning to homebrew a small system. If you are interested in a more detailed report on our evaluation, let me know and we will write a full and complete article on the subject.

CLUBS:

The majority of the computer clubs that exist today are a result of the catalyst provided by the MITS road show. We are very interested in continuing to provide whatever assistance we can to clubs. At the present time our typical customer is relatively sophisticated in terms of general electronics technology and/or software, but the vast majority of the potential customers in this hobby are not sophisticated. These new people are certainly needed by MITS and other manufacturers and their volume will benefit either directly or indirectly everyone in the hobby. A large percentage of the education and assistance getting these new groups up and running will have to come from the clubs, there is no other place. Our present efforts with the van only scratch the surface, incidentally the present van show is a basic technical presentation and not a sales pitch. We are actively working on a number of different courses which may prove useful to many of the clubs.

Any club which would like to have a technical group from MITS come to their meeting, please let me know and we will be there. If you would like an equipment demonstration and/or sales pitch we have representatives and dealers scattered all over the country who would love to have the opportunity to attend your meetings.

PARANOIA:

I have received a number of letters from readers of this publication concerning our counter attack of the armchair experts and others. The majority of those letters have been critical in the sense that they seem to say we are wasting time attacking these characters. This criticism is very gratifying in the sense that it indicates that our typical customer is more than sophisticated enough to establish the attacks on MITS and the Altair for what they are. Therefore, in the future we will address these sort of comments only in response to letters you send us and in the general design philosophy section of this column.

NEW PRODUCTS:

As I indicated last month as a general rule we will not release information on new products until they are production ready. But there have been some inquiries concerning the -2 and -II designation on some of the newer cards produced by MITS. Cards designated in this way are designed to be compatible with the until now unannounced Altair II. Let me just say the Altair II will be at least as revolutionary as the original Altair was when it was introduced. All -II cards have downward compatibility with the Altair 8800 systems, non-II cards probably won't interface with the Altair II.

Another product which has gotten a good bit of unintentional publicity is the MITS CRT graphics terminal. This is a product that has been under active development for more than two years. Because of the great amount of interest in this product let me break my own rule and give you some preliminary data.

In the alpha display mode it will present 80 characters per line, 24 lines per page. Up to 32 pages can theoretically be stored within the machine. There are two graphic modes, in the low resolution mode there are 256 x 256 elements. Any number of graphics pages and alpha pages may be interlaced and displayed under real time software control. The brightness of each frame in an interlace mode may be controlled. Control logic allows the system to be used with a standard color monitor if desired, i.e., full color graphics and alpha numerics are possible.

The data transfer rate between display and CPU is at main frame speeds, that is it should be possible to update the display at approximately 100 K byte rate, this is equivalent to a 1,100,000 baud rate. Of course, the display has upper and lower case capability, black on white and white on black and any of the other things you would expect in a terminal of this quality. The cost of this system will be significantly greater than the T-V typewriter type of display, but the capability will be much greater.

Our internal scheduling calls for shipments to begin in February, but you know how that goes. Nevertheless, we will not accept orders on this product yet, but if you would like to reserve a unit pending firm pricing and delivery schedules, write a letter to Barbara Sims and she will assign you a slot in production.

PUBLICATIONS:

We have mentioned the Computer Hobbyist in previous issues of Computer Notes, but I feel that it is such a valuable publication for the serious hobbyist it should be considered required reading. You can obtain a subscription by writing to the following address:

The Computer Hobbyist
Post Office Box 295
Cary, North Carolina 27511

Please send any comments or suggestions to me directly at MITS.

Highest regards,

David B. Porter
528 Carr Ave.
Rockville, MD 20850

Jim Gerow
2525 McCue #243
Houston, TX 77027

Peter Jarvis
10545 Ashworth N.
Seattle, WA 98133

Jerry K. Kozelsky
5711 Sarvis Ave. Suite 502
Riverdale, MD 20840

Jack J. Keith
3435 Mansfield Rd.
Falls Church, VA 22041

Olin A. Williams, Jr.
2871 Tony Dr.
Lawrenceville, GA 30245

James T. Mattley
6417 Fernhurst Ave.
Parma Heights, OH 44130

CONTINUED FROM PAGE 2

Altair Service Dept.

The only difference in AO and Air Mail is that the flap is taped rather than sealed for inspection purposes. It is impossible for MITS to take special requests for air mail due to the large amount of mailing we handle at one time. In the past, Computer Notes has usually been mailed around the 20th of the month. After the first of the year, we will try to start mailing a little earlier in the month. If for some reason you are not receiving your newsletter within approximately three weeks from the date it is mailed, then perhaps you should check with me.

Change of Address

When a school or company purchases the Altair, the sold-to address is used for our monthly mailings. If this is the case with you, and you would like to change the monthly mailing address to come directly to the user, drop me a post card with the purchasing name, 8800 order number and new address included. I will see that the mailing label is changed. If you mail or call in a change of address, please note that you are an Altair owner or user.

C/N Subscriptions

For some reason many of our customers think that the return of their warranty card initiates the subscription to Computer Notes. This is not true; your subscription begins in the month we ship your 8800 or 680 to you. Those customers with a computer printout invoice are immediately on the mailing list. Those customers who receive a white (7" x 8 1/2") invoice are put on our mailing list when their Altair is shipped. The same is true with time payment customers. If you purchased from one of our representatives or retail stores, then we add your name to our mailing list when we receive word from them.

Ordinarily, your membership in the Users Group will end in a year's time (in the same month you ordered your equipment). However, since there have been delays in getting Computer Notes out on a regular basis, no subscription will end until January 1977. This means if your ordered an Altair this year, your membership is good through December of next year.

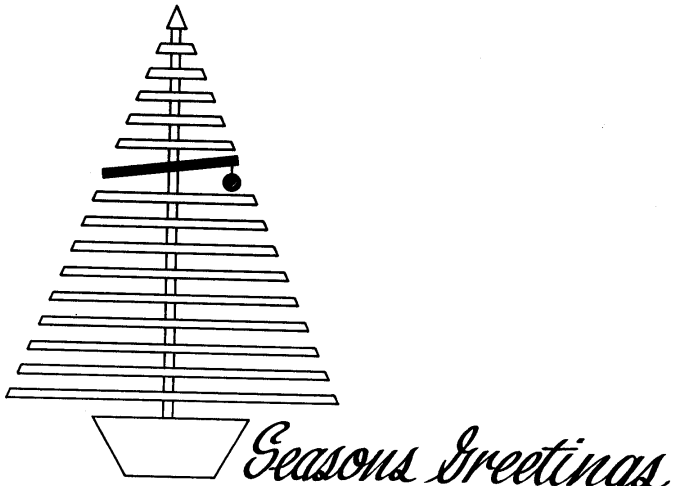
Invoices

Our invoicing has been changed to a new system which has caused a small amount of confusion to some customers. When you receive equipment from MITS, you will have a packing slip enclosed called our MITS order. A few days later you will receive a MITS invoice for this equipment. Please note on the invoice how your order was paid for. If you have prepaid the order, this invoice is for your records only. If your company is on a net 30 basis, the invoice is for payment. Hopefully this will clear up the confusion.

Classified Ads

A classified ads section is being introduced to our newsletter this month. Anyone who would like to advertise equipment for sale or trade, or has a want ad to place, should send a postcard to my attention with all the necessary information.

Barbara



The following 4 pages of *Computer Notes* are reprints of 8 pages from *Creative Computing*, "the magazine of recreational and educational computing." Each issue of *Creative Computing* carries 4 or 5 games in BASIC, most of which will run without modification in Altair BASIC. (The Jan-Feb 1976 issue is entirely devoted to games and puzzles!) This 72-page bi-monthly magazine is also crammed full of other good stuff of interest to Altair users. Subscriptions are only $8 per year and all subscribers will receive a $5.95 computer art book free as part of their subscription.

COMPLEAT COMPUTER CATALOGUE

We welcome entries from readers for the "Compleat Computer Catalogue" on any item related, even distantly, to computers. Please include the name of the item, a brief evaluative description, price, and complete source data. If it is an item you obtained over one year ago, please check with the source to make sure it is still available at the quoted price.

Send contributions to "The Compleat Computer Catalogue," *Creative Computing*, P.O. Box 789-M, Morristown, NJ 07960.

BOOKS AND BOOKLETS

TOWARD UNDERSTANDING THE SOCIAL IMPACT OF COMPUTERS

This very comprehensive 136-page book by Roy Amara is the distillation of the thoughts of 60 participants in four workshops held by the Institute for the Future. The principal message is that a real urgency exists to systematically understand how computers affect the decisions we make, the goods and services we produce, and the world we perceive. Specific topics discussed in-depth include: 1. Computer modeling and simulation as an aid to decision making, 2. Computers and financial processes, 3. Computer perceptions, attitudes, literacy, 4. Computers and individual access. A fifth could be added, "where do we go from here?" $10 to institutions, $7.50 to individuals and schools.

Institute for the Future, ATTN: Judy Flathman, 2740 Sand Hill Road, Menlo Park, CA 94025.

COMPUTERS IN EDUCATION

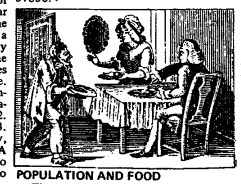
This ambitious little booklet (48 pp text, 14 pp appendices) boldly attempts to cover in three parts: How Computers Work, Administrative Uses of Computers, and Educational Uses of Computers. Indeed in the reading time of one hour, a person will get from this booklet a clear, concise and amazingly complete view of the computer in education. Last known price $1.00.

Computer Services, Province of Manitoba Dept. of Education, 103 Water Ave., Winnipeg, MB, R3C OJ2, Canada.

COMPUTERS BOOKLET

A 94-page booklet for 40 cents? Why, it's even difficult to get 94 sheets of blank paper for 40 cents. And yet here is one of the best basic descriptive books on computer hardware. It covers the birth of computers from the abacus and Babbage up to the present day, the anatomy of a computer including simplified explanations of some very sophisticated hardware, and finally some speculation about what's coming in the way of applications and large networks. The booklet simply titled *Computers* is written by William Corliss and published by the AEC, sorry ERDA. This one is a must. 40¢ each, even less in quantity. (Many other booklets are also available from ERDA; we recommend three: *Cryogenics, Teleoperators* and *Lasers*. Send $1.60 for all four.)

U.S. Energy Research and Development Administration, Technical Information Center, P.O. Box 62, Oak Ridge, TN 37830.



POPULATION AND FOOD

There are two interrelated and indisputable facts which many people have failed to comprehend: 1) world food production cannot keep pace with world population growth (not even with computer planning, miracle fertilizers, or plankton harvests from the ocean) and 2) "family planning" or new contraceptives cannot and will not, in the foreseeable future, check population growth. People are clearly looking for a miracle solution since the alternative, of course, is triage. Every person in the world, has an obligation to learn more about this situation which can only be described as a crisis. Send for a copy of the "Declaration on Population and Food" and a subscription to "The Other Side." Both free.

The Environmental Fund, 1302 Eighteenth St., N.W., Washington, DC 20036.

3 IFIP BOOKLETS

Three booklets have be
between 1971 and '74 by va
groups of the Internationa
for Information Processing.

*Computer Education fo
Secondary Schools – An Ou*
for those who are planning
the training of teachers.
gestions for the content
although the information
general nature. Includes
excellent pages on methodol

*Computer Education fo
Secondary Schools – Aim
tives in Teacher Training.* B
how society and educatio
and discusses the role of th
modern education. Lists bric
seven courses. 75¢.

*The Use of the Compute
and Learning.* Describes way
computer in education, the
benefits. Discusses the nece
for a successful program –
instructional material develc
ities, and hardware acquisiti
coverage is sparse (booklet
pages of text). $1.50.

AFIPS Press, 210 Su
Montvale, NJ 07645.

SMALL PRESS MATERIALS

*Edcentric, Haiku Magaz
kim, Feminist Art Journal, F
Nitty Gritty, Blue Pig, Algol
Quest* are just 11 of the ap
1800(!) little magazines and
listed and described in t
International Directory of
zines and Small Presses. If
different view of the world
strange, sometimes ecstatic,
refreshing, get this directory
sample copies of 10 or
magazines. Directory $5.9
postage.

Dustbooks, P.O. Box 10
CA 95969.

creative computing LIBRARY

Four sets of books covering major topics in computer literacy.

COMPUTER GAMES

101 BASIC Computer Games- Dave Ahl
An anthology of games and simulations- from Acey-Deucey to Yahtzee, all in the BASIC language. Contains a complete listing and sample run of each game, plus a descriptive write-up. Large Format. 256 pp. **$7.50**

What To Do After You Hit Return- Bob Albrecht
Another collection of games and simulations- all in BASIC- including number guessing games, word games, hide-and-seek games, pattern games, board games, business and social science simulations and science fiction games. Large format. 158 pp. **$6.95**

Fun And Games With The Computer- Ted Sage
Teaches problem-solving, flow charting and computer programming (in BASIC) in the context of well-known games of chance and strategy. 351 pp. **$5.95**

HAND CALCULATORS

Games, Tricks and Puzzles For A Hand Calculator- Wally Judd
This book is a necessity for anyone who owns or intends to buy a hand calculator, from the most sophisticated (the HP65 for example) to the basic "four banger". 110 pp. **$2.95**

The Calculating Book- Jim Rogers
Discover where you can buy gas during the oil crisis. An anthology of games, puzzles, puns, magic tricks and math problems that can be performed or solved with the pocket calculator. 125 pp. **$2.95**

150 Problems In Crypt-Arithmetic- Maxey Brooke
More grist for the calculator. 156 problems in which letters are substituted for numbers. Examine the number relations between the groups of symbols and solve the problem. 72 pp. **$1.25**

Advanced Applications For Pocket Calculators- Jack Gilbert
This book shows what you can do with your calculator after you balance your checkbook. As much as 80 or 90 percent of the capacity of the calculator is not used, and this book shows you how to use it- step-by-step- regardless of type. Explores all functions of all existing calculators, and all scientific and business applications. 304 pp. **$5.95**

COMPUTERS IN MATH

Problem Solving With The Computer- Ted Sage
Teaches problem solving, chart flowing, and computer programming (in BASIC) in the context of the traditional junior/senior high school curriculum precalculus math. 244 pp. **$4.95**

Problems For Computer Solution- Gruenberger & Jaffray
A collection of 92 problems in engineering, business, social science and mathematics. Oriented toward the FORTRAN language. 401 pp. **$7.25**

My Computer Likes Me- Bob Albrecht
This workbook introduces the BASIC computer language to young or old. The teaching examples are drawn from population problems and demographic data. Large format. 64 pp. **$2.19**

COMPUTERS IN SOCIETY

Electric Media- Les Brown and Sema Marks
How TV and computers affect individuals and society. Cable TV, air waves as public domain, popular and distorted views of the computer speed and storage capacities of computers, communicating with computers, artificial intelligence how computers learn, creative computers. Guest spots by Buzz Aldrin, Dick Cavett and Marshall McLuhan. 168 pp. **$4.95**

Freedom's Edge- Milton Wessel
The computer threat to society. The author, an attorney, shows some of the ways in which the computer is changing our lives- or soon will be. Discusses the data bank, point-of-sale marketing and free competition, computer related crime, controlling the computer, etc. 137 pp. **$4.95**

Computer Lib/Dream Machine- Ted Nelson
This book is devoted to the premise that everybody should understand computers. In a blithe manner the author covers interactive systems, terminals, computer languages, data structures, binary patterns, computer architecture, minicomputers, big computers, microprocessors, simulation, military uses of computers, computer companies, and much, much more. Large format. 127 pp. **$7.00**

Computers in Society- Donald Spencer
How can the computer help the Businessman, artist, or sports announcer? This book examines a wide range of up to date applications of the computer to medicine, engineering, transportation, business, the arts, education, law, process control and many other areas. 208 pp. **$4.95**

Please send me the following books:

Postage and handling	**$0.75**
Total	

☐ Payment enclosed
☐ Bill me ($1.00 charge)

THE CREATIVE COMPUTING LIBRARY
42 PLEASANT STREET NEWBURYPORT, MASS. 01950

Name _____
Street _____
City _____
State _____ Zip _____

Another new game from Creative Computing . . .

LUNAR

by David Ahl

LUNAR, also known as ROCKET, APOLLO, LEM, etc. is, next to STAR TREK and SPACE WAR, the most popular computer game. It is certainly the most popular on smaller machines. (I remember a milestone of sorts when I managed to compress LUNAR to run on 4K PDP-8 BASIC while retaining full instructions and landing messages. I used every single character available.)

The version of LUNAR presented here was originally written in FOCAL by Jim Storer, a student at Lexington (Mass.) High School in the mid 60's. While everyone claims to be the original program author of LUNAR, I'm reasonably sure that Jim predates the others and therefore qualifies as the original, original author. I converted the program to BASIC in early 1970. It's a straight-forward version without side stabilization rockets or other goodies but, nevertheless, is quite a challenge to land successfully.

PLAYING THE GAME

Your mission is to achieve a soft landing of your LEM on the moon. You separate from the command ship 200 miles above the surface of the moon and, every 10 seconds, set the burn rate of your retro rockets to slow your craft. You may free fall (0 lbs./sec.) or burn at any rate between 8 lbs./sec. and 200 lbs./sec. Since ignition occurs at 8 lbs./sec., burn rates between 1 and 7 lbs./sec. may not be used. A negative burn rate automatically aborts your mission.

There are three popular ways to land:

1. Constant burn rate all the way down.
2. Free fall for a while, then maximum burn rate tapering off as you get close.
3. Gradually increase burn rate to a maximum, then taper off as you get close.

Recall from physics that Newton found the force of attraction (gravity) between two bodies varies directly with the mass of the bodies and inversely with the square of the distance between their centers. This may help you land successfully. Then again, it may not.

COMPUTER NOTES

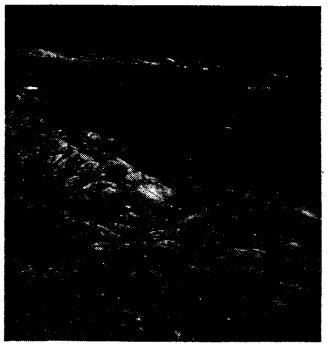
Convert the program to your version of BASIC. Multiple statements on one line are separated by a colon (:). Everything else is standard.

Some computers produce an error calculating the expansions (Statements 910 and 920) when you get close to the moon and the numbers get very small. If yours does, substitute the expanded form. Here it is for Statement 910:

$$-Q*(1+Q*(1/2+Q*(1/3+Q*(1/4+Q/5))))$$

You should be able to figure out the other one yourself.

Would you like us to print the other versions of LUNAR in *Creative Computing*? If so, write and let me know—DHA.



MAPPING THE MOON

This photograph pictures Mare Crisium, the large "flat" area near the eastern edge of the moon as seen from Earth. In the foreground is the mountainous terrain that forms the southern rim of Mare Crisium. Visible near the horizon, 285 miles across the mare, is its northern rim. Prominent at above right in the mare is the 24-mile-wide crater Picard. Photo was made from Apollo 10, the last flight before the lunar landing. (Photo Kodak)

SAMPLE RUN

```
RUNNH
LUNAR LANDING SIMULATION


CONTROL CALLING LUNAR MODULE.

YOU MAY SET THE FUEL RATE (K) TO ZERO OR ANY VALUE
BETWEEN 8 AND 200 LBS PER SECOND.  A NEGATIVE FUEL
RATE WILL ABORT THE MISSION.

YOU HAVE 16000 LBS OF FUEL.
ESTIMATED FREE FALL IMPACT TIME IS 120 SECONDS.
CAPSULE WEIGHT IS 32,500 LBS.

FIRST RADAR CHECK COMING UP. . .
BEGIN LANDING PROCEDURE

TIME(SECS)   WEIGHT(MI)    VELOCITY(MPH) FUEL(LBS)    FUEL RATE
 0             120           3600          16000        K? 0
 10            109.95        3636          16000        K? 0
 20            99.8          3672          16000        K? 0
 30            89.55         3708          16000        K? 0
 40            79.2          3744          16000        K? 0
 50            68.75         3780          16000        K? 0
 60            58.2          3816          16000        K? 0
 70            47.55         3852          16000        K? 200
 80            37.3656       3476.43       14000        K? 200
 90            28.2623       3072.94       12000        K? 200
 100           20.3232       2637.46       10000        K? 200
 110           13.644        2164.97       8000         K? 200
 120           8.33572       1649.14       6000         K? 200
 130           4.32958       1081.92       4000         K? 100
 140           2.2887        522.398       2200         K? 98
 150           1.22706       238.772       1302         K? 46
 160           .718289       127.493       900          K? 20
 170           .458009       80.3788       780          K? 22
 180           .237646       41.1685       480          K? 14
 190           .147722       23.5114       348          K? 10
 200           .859422E-1    20.9169       240          K? 10
 210           .317304E-1    19.8912       148          K? 12
ON THE MOON AT  218.451 SECS.
IMPACT VELOCITY OF  8.98172 M.P.H.
FUEL LEFT  38.5879 LBS.
VERY GOOD LANDING. NOT PERFECT YET.

TRY AGAIN (1 FOR YES, 0 FOR NO)? 0

CONTROL OUT

READY
```

PROGRAM LISTING

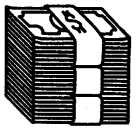
```
LISTNH
1 REM *** WRITTEN BY JIM STORER, LEXINGTON MS
2 REM *** CONVERTED FROM FOCAL TO BASIC BY  DAVID AHL, DIGITAL
10 PRINT "LUNAR LANDING SIMULATION":PRINT
20 PRINT:PRINT:PRINT "CONTROL CALLING LUNAR MODULE. ":PRINT
30PRINT"YOU MAY SET THE FUEL RATE (K) TO ZERO OR ANY VALUE"
40PRINT"BETWEEN 8 AND 200 LBS PER SECOND.  A NEGATIVE FUEL"
50 PRINT "RATE WILL ABORT THE MISSION.":PRINT
60PRINT"YOU HAVE 16000 LBS OF FUEL. "
70PRINT"ESTIMATED FREE FALL IMPACT TIME IS 120 SECONDS."
80PRINT"CAPSULE WEIGHT IS 32,500 LBS."
90 PRINT PRINT "FIRST RADAR CHECK COMING UP. . ."
100 PRINT "BEGIN LANDING PROCEDURE":PRINT:PRINT
110PRINT"TIME(SECS)","HEIGHT(MI)","VELOCITY(MPH)","FUEL(LBS)","FUEL RATE"
120 LET L=0:LET M=120:LET V=10800/3:LET N=32500:LET G=1500
130 LET G=.001:LET Z=1.9
200 PRINT INT(L+.5),M,V*3600/5280,N-M,"K",
220 INPUT K
225 LET T=10
230 IF K<0 GO TO 530
235 IF K=0 GOTO 310
240 IF K<8 THEN 260
250 IF K<=200 GO TO 310
260 PRINT "NOT POSSIBLE";,"K";
270 INPUT K  :GOTO 230
310 IF N-M> 001 C=0 GOTO410
320 IF T<.001 GOTO 210
330 LET S=T:IF M<S*C+M GO TO 350
340 LET S=(M-M)/K
350 LET I0=1  :GOTO 900
360 IF J <= 0 GOT O 710
370 IF V<=0 GO TO 390
375 IF J<0 GOTO 810
380 LET I0=1 GOTO600
410 PRINT"FUEL OUT AT ";L;"SECS. "
420 LET S=(-V+SQR(V*V+2*M*G))/G
430 LET V = V+G*S
440 LET L=L+S
510 PRINT"ON THE MOON AT ";L;"SECS. "
511 LET W = 3600*V
514 PRINT "IMPACT VELOCITY OF ";W;"M.P.H. "
520 PRINT "FUEL LEFT ";N-M;"LBS. "
530 IF W>1 GOTO 550
540 PRINT "PERFECT LANDING!  CONGRATULATIONS!!": GOTO 590
550 IF W >10 THEN 560
552 PRINT "VERY GOOD LANDING. NOT PERFECT YET.":GOTO590
554 IF W >= 25 THEN 570
562 PRINT "A FAIR LANDING. NO CRAFT DAMAGE.":GOTO590
570 IF W >= 60 THEN568
572 PRINT "CRAFT DAMAGE    HOPE YOUR OXYGEN HOLDS OUT UNTIL A"
574 PRINT "RESCUE MISSION ARRIVES!":GOTO 590
580 PRINT "SORRY, BUT THERE WERE NO SURVIVORS. "
585 PRINT "IN FACT YOU BLASTED A NEW LUNAR CRATER ";W* 277777;" FEET DEEP
590 PRINT:PRINT:PRINT "TRY AGAIN (1 FOR YES, 0 FOR NO)";
592 INPUT R :IF R=1 THEN 99
595 PRINT: PRINT "CONTROL OUT" : GOTO 1000
600 LET L=L+S
610 LET T = T-S
620 LET M=M-S*C
630 LET A=I
640 LET V=J
650 IF I0=1 GO TO 310
660 IF I0=3 GO TO 850
710 IF S< .005 GO TO 510
720 LET S= Z*M/(V+SQR(V*V+2*M*(G-Z*C/M)))
730 LET I0=2 :GOTO 900
810 LET A=(1-M*G/(Z*Z))/2
820 LET S=M*V/(Z*C*(A+SQR(A*A+V/Z)))+.05
825 LET I0=3  :GOTO 900
830 IF J<>0 THEN 710
840 GOTO 660
850 IF J>=0 THEN 310
860 IF V<=0 GO TO 310
870 GOTO 810
900 LET Q=S*V/M
905 IF Q<=0 THEN 1000
910 LET J=V+G*S*(+1+Q*(1/2+Q*(1/3+Q*(1/4+Q*(1/5))))))
920 LET I=A-G*S*S/2+V*S/Z*(+Q*(1/2+Q*(1/6+Q*(1/12+Q*(1/20+Q*(1/30))))))
930 IF I0=1 GOTO 360
940 IF I0=2 GOTO 680
950 IF I0=3 GOTO 830
1000 LET J=V+G*S
1010 LET I=M-G*S*S/2-V*S
1020 GOTO930
1099 END

READY
```

Apollo 14 Launch Control Center, Cape Kennedy, Florida

Puzzles and Problems For Fun

▶ The number 153 = $1^3 + 5^3 + 3^3$ Find all oth 3-digit numbers that have the same property. Ho about 4-digit numbers? To the 4th?

Bill Morris
Sudbury, Ma



▶ Mr. Karbunkle went to the bank to cash h weekly paycheck. In handing over the money, th cashier, by mistake, gave him dollars for cents ar cents for dollars.

He pocketed the money without examining and spent a nickel on candy for his little boy. H then discovered the error and found he possesse exactly twice the amount of the check.

If he had no money in his pocket befo cashing the check, what was the exact amount the check? One clue: Mr. Karbunkle earns less tha $50 a week.

▶ Can you find the missing number for eac diagram? You first have to figure the pattern whic may be horizontal or vertical with a relationshi between every number, every second or thir number. You may have to add, subtract, multiply divide, invert or do a combination of these thing Have fun!

A.

3	5		17
7		25	49
	37	73	

B.

	13	6
7	9	
12		10

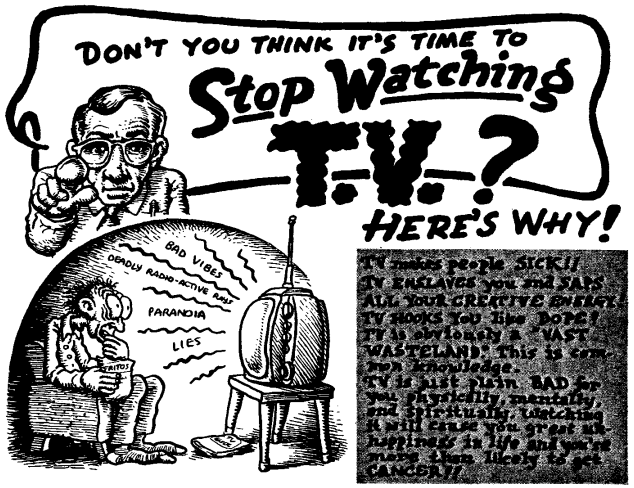
C.

4	20	5
	8	8
6	54	9
7	45	7

D.

27
— —
12 6 3
— 4 — 1

▶ Send us your favorite puzzles for this column!!



IF YOUR CREATIVE INSANITY IS SEEKING A REAL OUTLET, THEN READ

creative computing

It's the magazine that answers to NO ONE!! CREATIVE COMPUTING is the Last outspoken bastion of TRUTH left in America today. Read it and you shall be free! Contained within these pages are mind-blowing glimpses of REALITY as it REALLY IS! This magazine BREAKS THROUGH the TV-induced stupor, for it is ANTI-MEDIA! It has things for you to ACTUALLY Do! It's got the medicine for the BLUES, and has been known to turn MENTALLY ILL persons into HEALTHY, GOOD-HUMORED FREE-THINKERS. You too will benefit from reading CREATIVE COMPUTING!!

Only $6 per year ($4 for students)
CREATIVE COMPUTING, Dept. C,
P.O. Box 789-M, Morristown, NJ 07960.

creative computing

BACK ISSUES

NOV-DEC 1974 – CHARTER ISSUE (Out of print)

JAN-FEB 1975 – GENERAL TOPICS (Out of print)

MAR-APR 1975 – COMPUTER CAREERS (Out of Print)
A Computer Career for You?
Profile of the Computer Industry – Job opportunities.
Shorthand Instruction via Computer.
Computer Cartoons – a complete analysis.
Games: Rabbit Chase, Not One, Binary.
Review of BASIC books – Summary and first four reviews.

MAY-JUN 1975 – SPACE EXPLORATION
Games: Super Star Trek, Lunar Landing, Parachute Jump on Any Planet, ICBM Missile Intercept, Seawar, Geowar.
Escape – A simulation about escaping the earth's gravitation.
Mr. Spock's 7th Sense – real computer sensors.
Trends in Mathematics Curriculum Research.
Computer Simulation of the Atmosphere.
Structured Programming.
The Automobile and Air Pollution – Writing your own computer simulation.
Mr. Spock centerfold poster.

SEP-OCT 1975 – COMPUTER LITERACY
Activities that don't require a computer.
Games: Guess, Wumpus, Civil War, Reverse, Schmoo.
Languages for learners: LOGO, Smalltalk.
Toward a Human Computer Language.
Building A Computer of Your Own – Part 1.
"Compleat Computer Catalogue" – (First installment)

NOV-DEC 1975 – THE COMPUTER IMPACT ON SOCIETY
Isaac Asimov in a new story "The Life and Times of Multivac" looks at what happens after a nuclear holocaust when all the computers on earth link up to support the remaining human survivors.
Surveys, the Census, and Privacy – a new classroom activity.
Point of Sale Computer Systems.
Computer Crime and the White Collar Criminal.
Computer Security – A mini course.

Back issues are $1.50 each paid in advance. A billing charge of $1.00 is added to all orders without payment enclosed.

COMING SOON

JAN-FEB 1976 – GAMES AND PUZZLES
Re-inventing the Wheel – A collection of commonly-used routines and algorithms.
Chess programs.
Games, games, games.
The Complete Computer Catalog (3rd installment).

MAR-APR 1976 – ARTIFICIAL INTELLIGENCE AND FUTURE COMPUTERS
Smaller, faster, cheaper computers.
Access to computers in museums, libraries and storefronts.
At what point does a computer start to think?
Review of every BASIC book in print (6th and last installment).

MAY-JUN 1976 – COMPUTER ART
This fantastic double-size, full color, quality paper, large format issue will feature the works of 40 leading computer artists from around the world. Guest editor for this special issue is Ruth Leavitt, an outstanding computer artist.

Closing date for articles or other material for issues is 3½ months before the cover date. If you want an acknowledgement of your submittal, please include a self-addressed stamped envelope.

BULK SUBSCRIPTIONS

Bulk subscriptions are now available to *Creative Computing*. All copies must be sent to the same address; minimum term of the subscription is one year.

Quantity	*Price*
10-24	$7.00 each
25-49	6.00
50-99	5.00
100+	4.00

MEMBER COSMEP
COMMITTEE OF SMALL MAGAZINE EDITORS AND PUBLISHERS

CREATIVE COMPUTING SUBSCRIPTION FORM

Published bi-monthly – 6 issues per year

All subscribers will receive a $5.95 computer art book FREE!

1. *Quan.*	*Type*	*Term*	*Price Each*
____	Institutional	1-Year	$15.00
____	Institutional	3-Year	$40.00
____	Individual	1-Year	$ 8.00
____	Individual	3-Year	$21.00
____	Student	1-Year	$ 6.00
____	Student	3-Year	$16.00
____	Bulk (10-24 copies)	1-Year	$ 7.00
____	Bulk (25-49 copies)	1-Year	$ 6.00
____	Bulk (50-99 copies)	1-Year	$ 5.00
____	Bulk (100+ copies)	1-Year	$ 4.00
____	Sample Copy		$ 1.00
____	Foreign Postage		$1.50 per year

2. *New or Renewal*
☐ New
☐ Renewal
☐ Gift

3. *Billing Instructions*
☐ Payment enclosed
☐ Please bill me ($1.00 billing charge will be added except on institutional and bulk subscriptions.

4. *Name and Address*

Name _____ Title/Dept. _____

School or Company _____

Street Address _____

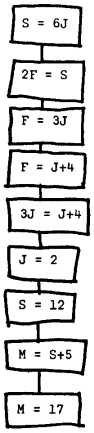
City _____ State _____ Zip _____

Return form to: CREATIVE COMPUTING, P.O. Box 789-M, Morristown, New Jersey 07960

contributed by Thomas D. Thomas
Salem, Oregon

Part 1

Problem Definition: Sue has 6 times as many apples as Joe. Fred has half as many apples as Sue. Joe has 4 fewer apples than Fred. How many apples has Mary, if she has 5 more than Sue?



1. Define the number of apples that Sue has, in terms of the number that Joe has, from the problem definition.
2. Define the number of apples that Fred has in relation to Sue, from the problem definition.
3. Define a relationship between Fred and Joe, from steps 1 and 2 above.
4. Define the relationship between Fred and Joe, from the problem definition.
5. Substitution for simultaneous solution of steps 3 and 4 above.
6. Joe has 2 apples, from step 5.
7. Sue has 12 apples, from steps 1 and 6 above.
8. Relationship of Mary to Sue, from problem definition.
9. From steps 7 and 8.

Part 2

Discussion:

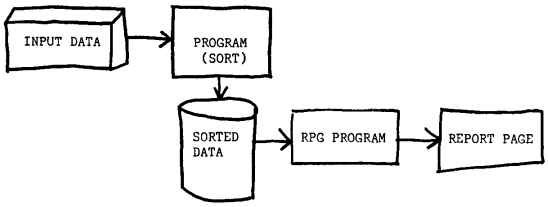
We are ready to ask ourselves, why flowchart at all? From the previous example, we see that Flowcharting forces us to clarify our thinking, by translating our solutions into a step-by-step procedure. Secondly, we have stated the problem, and its solution, in a language that is more universal than most programming languages.

We see then, that Flowcharting:
1. Helps us to formulate our solutions
2. Helps us to communicate our solutions to others.

As an example, let us communicate the solution to a data processing job to our employee, the programmer.

First, we define the problem:
A. Define the input information (data deck).
B. Define the output information (report).
C. Define the tools (language, computer, etc.).

Second, we communicate the general form of the solution in the form of the solution's flowchart:



Notice that the problem's solution is immediately clear, from the above flowchart, and that the programmer should have no difficulty in understanding what is wanted by his supervisor. That is, the input data is to be sorted and printed in report form, by two different programs. This simplistic flowchart may seem to add little to the understanding of the problem statement. However, look again at page 1, and compare the problem statement to the flowchart, to see that the flowchart may often tremendously clarify a problem statement.

Letters to the Editor

Last September - 15, I paid $30 for membership on your Altair Users Association - see MITS order 6768.

A few days after this date, I received the August issue of Computer Notes. Having been for over 15 years in the D/P industry, where the rather staid styles of the IBM and ACM publications set the standard modes of communication, I found your lively and informal news letter full of inspiration and human touches.

Perhaps someday you will be remembered not so much for your in-expensive hardware, but for having helped in the "humanization" of the computer industry.

Congratulations! Keep up the good work.

Mario F. Maffioli
Ponce, Puerto Rico 00731

Dear Mr. Bunnell:

I see in Computer Notes for October that there are many quotes from PCC for September. On pages 12 and 13 of the same issue of PCC there were some other comments of considerable interest to Altair owners.

Specifically:
1. Conditioning of the reset switch as per diagram shown.
2. Modification of the dynamic boards.
3. C-7 no greater than 4700 pF. Under "Arrowhead Tips" the final value is given at 0.01 uF.
4. Rewiring the PRDY to pin 3 of the bus.

Would you please give me your comments on these suggested modifications.

Dr. George L. Haller
Naples, FL

Dear Dr. Haller:

None of the modifications mentioned in the PCC article are required, and some will even damage your machine. But specifically to answer your questions:

1. Bounce in the reset switch can cause an undefined condition in the status latch, and this can result in a loss of data in the 4K memory boards. At this time there is only one solution to this problem: don't use reset once the machine is loaded (normally the case anyway). The modification proposed in PCC was tested internally and the only effect it had on our machines was to suppress the operation of the reset switch totally. If that's the goal, a simpler mod would be to remove the reset switch. In addition to the fact that it didn't work in the majority of the machines in which it was tested, it will permanently damage the reset switch due to high surge currents during the capacitor discharge. Never discharge a capacitor with a mechanical contact without using some sort of current limiting protection.

CONTINUED ON PAGE 21

Build this "Record and Play Switching Unit" for your Altair ACR

contributed by Craig Pearce

How many times have you run your output bootstrap program to record material onto tape, only to find that the cable in the back was connected to the 'Tape Play In' jack? If you're like me, probably about as many times as I've left the plug in 'Record Out' when I wanted to read a program into the core.

I immediately saw I was in a losing battle with what I call my "memory" (I think my brain's refresh circuitry has gone bad), so, I quickly roughed out and threw together a handy, simple switch box, then tacked on a few conveniences to make the unit a little easier to use.

With the "RPSU" both the record-out and play-in jacks on the back of the ALTAIR are connected to this unit at the same time. Coming from the unit are two cables that connect to the recorder's microphone input jack and the line output jack. A two-position rotary switch is used to select between the record (called 'store' for reasons explained later) and play modes.

Also, the box contains a built-in, 1-watt amplifier module for monitoring of signal on playback (and during record, if the tape recorder allows this), plus a seven-segment LED display to indicate the mode. (An "S" for Store--or record--and a "P" for Playback. The reason for the term 'store' now becomes obvious, since an R (for record) looks more like an "A" on a seven-segment display.)

The builder can use any enclosure and any 6-volt DC source that he wishes. The parts used by this author are listed in a table to follow.

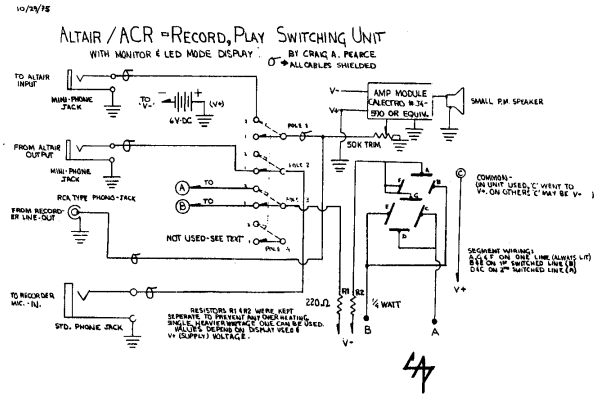
Construction is quite straightforward. The various parts to be mounted on the front panel should be laid out first and then the holes drilled. A series of holes, or one large one with a grill cloth, should be made for the speaker. One IMPORTANT NOTE: Always check to see component placement on the front panel does not interfere with internal component placement. The switches and speaker do displace some area inside and many times have I found a homebrew project does not fit together because of this problem.

Another point to remember is always use a NON-SHORTING type switch. This prevents the in and out lines on the ACR Modem from getting shorted together. (A non-shorting switch is said to "break before make," or disconnect one lead before connecting the other. In contrast, a shorting, or "make before break" type switch will intentionally short adjacent contacts as it is moved from one position to the next.)

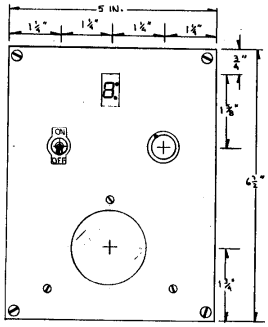
For those who wish to assemble the unit like the author's, a complete set of drawings accompanies this article. (see next page)

Hopefully, your leads will no longer be misconnected when using the ACR-Modem. Unless, of course, you forget to throw the switch.

--see page 13 for parts list--

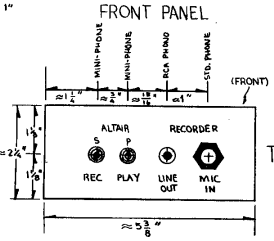


— see construction drawings on following page —

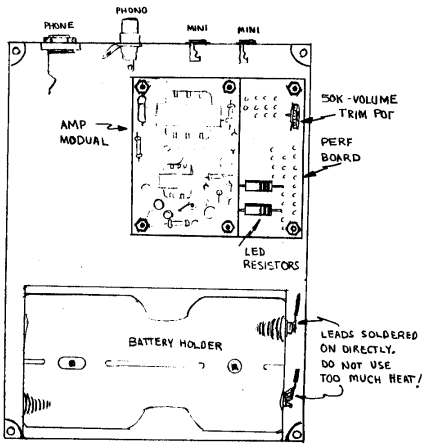


SCALE: ¾" = 1"

FRONT PANEL



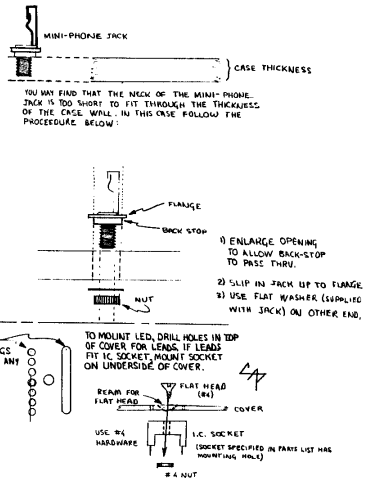
TOP VIEW



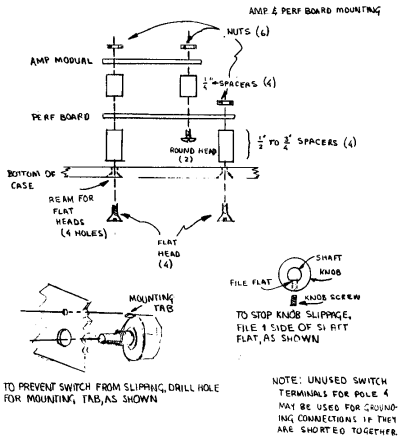
INTERIOR VIEW - COMPONENT LAYOUT

(USE SHIELDED CABLE ON ALL AUDIO SIGNAL CARRING CONNECTIONS)

MINI-JACK INSTALLATION ~ I.C. SOCKET USE



CONSTRUCTION HINTS & TIPS



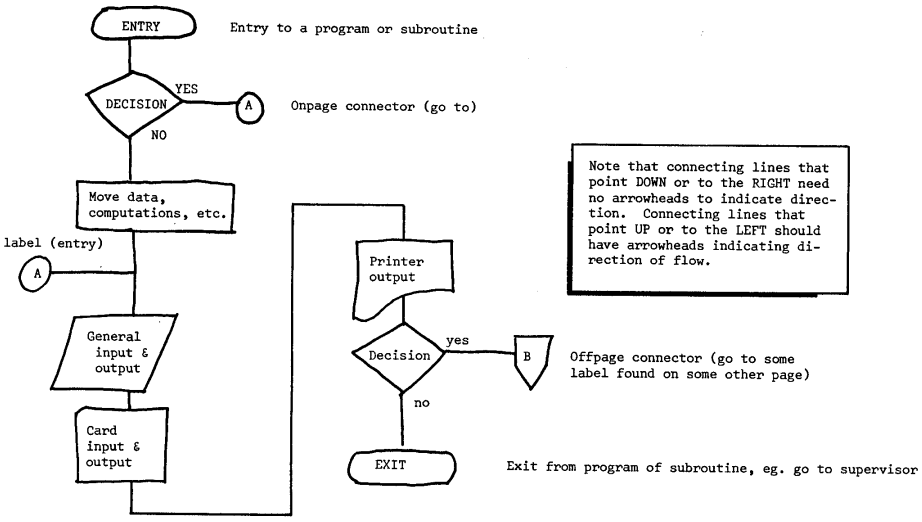
NOTE: UNUSED SWITCH TERMINALS FOR POLE 4 MAY BE USED FOR GROUNDING CONNECTIONS IF THEY ARE SHORTED TOGETHER.

CONTINUED FROM PAGE 10

Flow Charting

Part 3

Definition of symbols commonly used by programmers.



--continued from page 6

PARTS LIST for RPSU

1	Seven-segment LED display chip	(Calectro #J4-975 or equiv)
1	IC socket (for display)	(" #F2-998 ")
1	Audio amp module	(" #J4-590 ")
2	220 ohm, 1/2 or 1/4 w 10% resistors	(" #B1-376 ")
1	50K ohm trimmer pot (for volumn adj.)	(" #B1-645 ")
1	Two-inch speaker	(" #S2-202 ")
1	SPST light duty switch (ON-OFF)	(" #E2-130 ")
1	2-position rotary switch with 3 or > poles: NON-SHORTING	(" #E2-167 ")

PANEL MOUNTING JACKS:

2	Mini-phone type	(Calectro #F2-845 or equiv)
1	RCA Phono type	(" #F2-806 ")
1	Standard phone jack	(" #F2-848 ")
1	Case	(Calectro #H4-729)

CABLES: (Bought or made)
2 with male, mini-phone plugs on eigher end
1 with RCA type phono plugs on either end*
1 with std. male phone plug on either end**

1 6V (4 cell) C-battery holder (Calectro #D3-064 or equiv)

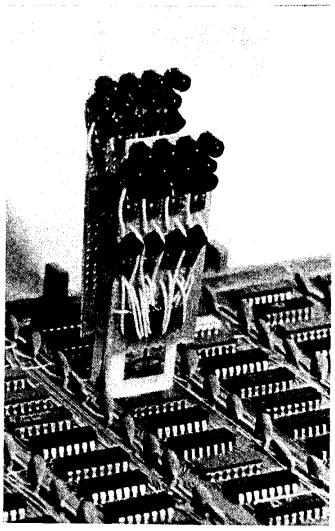
MISCELLANEOUS:
Shielded cable, C-cells, hook-up wire, mounting hardware, perf board and solder

*one end should have plug to fit the out jack on your tape recorder
**one end should fit the microphone in jack on your tape recorder

Powerless IC Test Clip

circuit by
John Errico

written by
Robert Baker



This test clip operates like the expensive, commercially available clips selling for $85 or more without requiring batteries or external power. All types of ICs may be tested (TTL, DTL, MOS, etc.) and LEDs are used to indicate the logic state of each pin being tested.

The heart of the test clip is a Texas Instruments TID125 diode array which costs about $3.75. Two diode arrays are used to determine the pin with the highest voltage (Vcc) and the pin with the lowest voltage (ground). These pins are then used to power the LEDs on the test clip itself, thus taking power from the IC on the board and eliminating the need for an external or separate supply. The circuit is straight forward and may be expanded to make a 24- or 40-pin test clip. The larger test clip, however, may be difficult to use due to the size of the LED display.

The basic IC clip is a standard item available from AP Products Inc., Box 110-Z, Painesville OH 44077. The 16-pin clip is part number 923700 (TC-16) and sells for $5.75 each.

The diode arrays are 14-pin dip packages and were chosen to make the test clip more compact. To cut down the cost, 16 general purpose silicon diodes may be used in place of each diode array IC. The transistors used to drive the LEDs may be any NPN transistor capable of handling the LED current. Any small size LED may be used; however, the 1k resistance value may have to be changed. Choose a value which gives about 2 mA current through the LED; this should give sufficient brightness without loading down the circuit supply.

Construction is very simple and parts layout is not critical. Use a small piece of 0.1" grid perforated board bolted to each side of the IC clip to mount components on. Try to keep the overall physical size of the boards as small as possible to make the finished test clip easier to handle. The LEDs should be mounted along the top edge of the perforated boards so they are visible from above the clip when it is attached to an IC. I would suggest wrapping a small piece of dark tape or using a short piece of dark tubing around each LED to improve visibility of the finished LED display. One of the TID125 diode arrays is mounted on each piece of perforated board along with the associated resistors and transistors, positioned wherever convenient. Remember to run two wires between the two perforated boards to connect the Vcc and ground outputs of the diode arrays together. These wires should be stranded to withstand the movement of opening and closing the test clip when in use.

Using the test clip is the simplest part of all. Just clip it over the desired IC. Don't worry about how to position the test clip on the IC; pin 1 may be at either end and the test clip will still work properly. With the test clip installed on an IC package the LEDs will indicate the logic level of each pin:

ON = Logic 1 (HIGH) or Vcc pin
OFF = Logic 0 (LOW) or ground pin

On 14-pin ICs disregard the two pins not attached.

Who said building an IC test probe is hard? ■

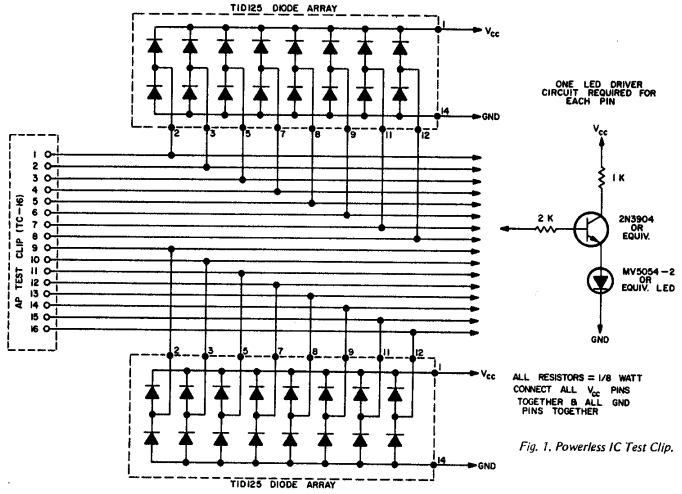


Fig. 1. Powerless IC Test Clip.

Reprinted from *BYTE Magazine*, December, 1975.

Recycling Used ICs

by
Carl Mikkelsen

The surplus market is saturated with used printed circuit boards from early computer systems which offer a very inexpensive per chip source of ICs. Used boards typically contain 50-200 chips of small scale or medium scale integration, usually with many simple two input gates and four bit data registers. Common part numbers include 7400, 7402, 7404, 7408, 74126, 74174, 74175, etc. Through careful shopping, I have found boards with large numbers of multiplexors such as 74151, 74153, and even scratch pad registers – 7489. After removing chips from the boards and eliminating any non-functional units, cost per chip is from 3 to 8 cents, resulting in an overall cost of about one fourth to one tenth of the individual chip cost through other surplus outlets.

Removing chips from boards offers advantages over purchasing chips surplus which makes them attractive for reasons other than price. Primarily, the companies which originally built the boards used top-quality, fully spec'ed components. All chips have already been tested, and most have already served in equipment.

Given that you've found a serendipity of well soldered chips, it's necessary to unsolder them without either burning them or cracking their cases. Desoldering individual leads can be done, but usually the chip is made unnecessarily hot by the prolonged application of heat. Also, pulling each lead out separately results in bent, often broken leads. Devices are available which will heat all 14 or 16 pins of a small IC, but again a long time is needed to melt the solder since the total amount of energy available is limited to a small soldering pencil heating element. Most available boards are two sided and four layer boards aren't uncommon. Multi-layered boards make the required amount of energy even higher.

When a board is built, the ICs are positioned in place with all other components, and the board is soldered by a three step process.

1. The underside is washed by hot, bubbling, liquid flux.
2. The clean board is passed over a small fountain of solder, so that the board just touches it.
3. After cooling, the board is immersed in FREON gas to remove any remaining flux.

As you can see, the board is subjected to high temperatures during the soldering phase, which takes around 5-10 seconds.

The blow torch method of IC removal duplicates conditions during board soldering by heating all pins simultaneously; removing the IC is a single step.

Equipment Needed

To use this technique, you will need:

A torch. Non-oxygenated propane and acetylene gas has been used.

Clamps or a vise to hold the board fairly rigid during chip removal.

A way to grip the chips, depending on how they are packed next to each other. Components, small vise grips, a small screw driver and a fine point awl should be all that are needed.

A place where splashed solder will not be serious.

Some form of eye protection.

WARNING 1

Using this method involves heating PC boards to high temperatures. Some boards release Hydrogen Chloride (HCl), which becomes hydrochloric acid in your lungs. Do this only in a well ventilated area, and stop to allow air to clear if irritation develops.

WARNING 2

When an IC is pulled from a board, the board often snaps back to its original position. This is especially true if it isn't fixed very rigidly in place. When the board flips, solder is often sprayed away from the back side of the board. I ruined a pair of pants by not considering this before I started. I, therefore, wear old clothes and if you don't want solder on the floor, cover it with newspapers.

Enough warnings ... following is how I pull ICs from boards:

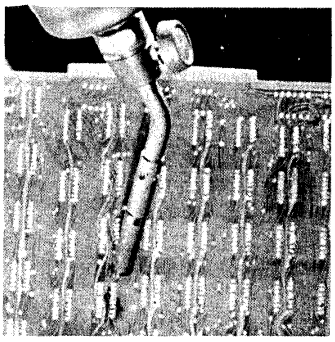
First I clamp the board to my bench so that I can get my vise grips on about half the ICs (this is with a 10" x 14" board). I adjust the vise grips so I can grip a 14 pin IC without the vise grips locking and then light the torch. The flame on my Benzo-matic torch with the narrow tip is about an inch long.

Beginning with the lowest IC I can reach, I heat it with the torch by sweeping the torch over its pins (you obviously heat the non-component side). Especially when using a torch with a narrow flame it is necessary to move the flame over the pins. One complete sweep should be done once or twice a second. After a second or so, the IC should be gripped, and rocking tension away from the board applied. It helps to rock the IC, especially if corner pins have been bent over to hold the IC in place during assembly. The IC should very rapidly become loose, and in another couple of seconds should come free of the board.

When the IC is removed, quickly drop it on the bench and move the torch and pliers to the IC above the one removed. Heating the lower IC pre-warms the board above, making the next removal easier. Also, the board position just heated will cool faster, thereby reducing the amount by which the board will be damaged.

As each column of ICs is removed, the next is done. When all ICs on one half have been removed, reposition the board so the other half is accessible. I've found that the half-way point often can be a good excuse to let the room ventilate and drink a beer.

No matter how carefully and rapidly I've worked, I always burn the board at least once because I have trouble removing an IC, or my pliers slip, or for some other reason. If you consistently burn each board position, your flame is probably too hot. If, however, it takes longer than 5 to 10 seconds to remove an IC, your flame is too cool.



Sweep the blow torch over the IC's pins—one complete sweep once or twice a second.

A certain amount of care is necessary when gripping the ICs. Too much pressure may crack them. Too little pressure will let the pliers slip, costing time to reposition them and marring the cases.

When attempting to remove the larger ICs such as 74181s and 74154s, which come in 24 pin DIPs, I have trouble gripping them, so I remove them as a two step process. First, I place an awl under the middle of one side, say between pins 6 and 7. I heat that pin row and, with the awl applying leverage, pull out that row. I then grip the IC on its thinnest dimension, heat the remaining pins, and remove the IC.

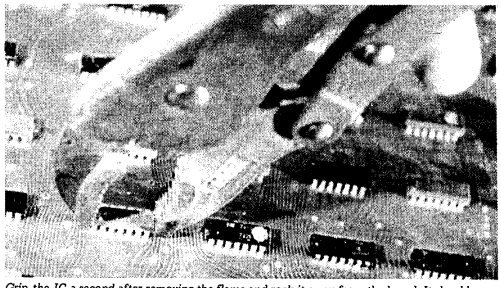
So far, by using this technique, my friends and I have removed about 1000 ICs from surplus boards which have about 80-100 ICs each. I tend to break 2% of the chips I pull by applying too much force with the pliers. But a friend has never broken one, so it clearly is an individual matter. Of those chips removed unbroken, we have tested around 250, and have never found a bad chip.

As an unrecommended demonstration of the ruggedness of ICs, I accidentally grossly overheated one, so that when I gripped it in vise grips, the chip was bent in a curve. The plastic case must have softened significantly. After allowing it to cool several minutes to the point where I could handle it by hand, I plugged it into a circuit, expecting it to have failed totally. It worked, although I didn't check out its ac characteristics. Out of general paranoiac distrust for a device so intensely mistreated, I discarded it.

After removing ICs from boards it is usually necessary to clean and straighten the pins. Boards with plated through holes often lose their plating around the IC lead.

I have found this method useful as a means of quickly building a stock of ICs ready to use in any project. It is limited mainly by the availability of exotic surplus chips, but most standard 7400 series TTL is easily available. The price of 4 cents/chip can't be beat, and the time required – about 10 to 20 minutes/80 chip board – is rather small.

This technique provides a fast, cheap, safe means of removing chips. I hope it proves as effective for you as it does for me.



Grip the IC a second after removing the flame and rock it away from the board. It should come free in a couple of seconds.

There's More to Blinking Lights Than Meets The Eye

A blinking light peripheral is an inexpensive, entertaining addition to your computer system. The use of multiple indicator lamps under computer control to produce moving patterns can lead to many hours of creative programming and pleasant amusement.

Playing with blinking lights is nothing new to people working with computers. Early systems, to say nothing of science fiction movie caricatures, tended to have monstrous front panels with row after row of indicator lamps. These lamps were used to reveal various machine states. As computers became more sophisticated, the need for many of these indicators diminished, but the fascination of making the indicator lamps dance and gyrate in interesting patterns has remained.

As a simple example of moving lights, consider a single 8 bit byte of memory in your computer, which might be called BLINK. Assume that you have also constructed an 8 bit output data latch which drives eight LEDs as shown in figure 1. When a certain program is started, BLINK might be initialized as follows:

0 0 0 0 1 0 0 0

Suppose this value is sent to the display, which is set up so that a 1 bit lights an LED, then the display will look like this:

● ● ● ● ○ ● ● ●

The key element in creating an illusion of motion is time. If the program starts out with data as shown above, waits a short time, then executes a left rotate instruction, a new pattern will be obtained. The new binary value 0001 0000 can be sent to the display:

● ● ● ○ ● ● ● ●

In the simplest of all motion programs, these three steps are repeated in an endless loop:

1. Rotate BLINK left one bit position.
2. Send BLINK to the display.
3. Wait n milliseconds.
4. Go to step 1.

With a program executing these four steps, the pattern of lit indicators will be seen moving to the left, disappearing on the left in the same step at which it reappears on the right. By changing the program delay (step 3), the speed of the pattern's apparent motion can be changed.

Figure 1 shows two quad latches which are used to drive 8 indicator lamps. The 220 Ω resistors are typical values for LEDs as indicators. This value allows reasonable brightness with most LEDs. The complement outputs of the 7475 ICs used as latches produce a lit LED for each 1 bit received. When you purchase LEDs for a blinking light display, make sure they are all the same, as the display will not look as attractive if lamps of different types are mixed.

Now, suppose you use four latches for 16 bits of data and 16 LEDs. How do you program an 8 bit computer to do the shifting

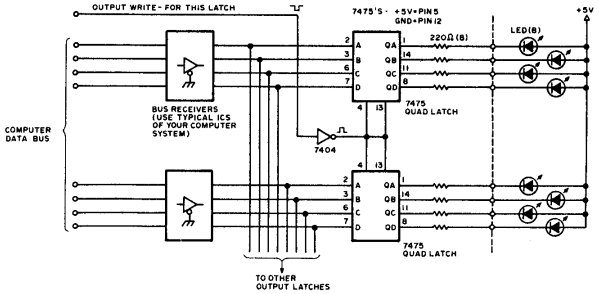


Figure 1: The basic "straightforward" approach to display lamps. One latch is assigned to each of several output ports of the computer. The output port decoding logic will determine when the latch is addressed for output. The result of decoding is a WRITE signal which latches the data presented at the bus receivers. The sample program of figure 2 assumes two such display registers. In principle the idea can be extended to many registers in groups of eight, limited only by the computer's input output addressing capability and the available power for lighting LEDs. In this circuit, TTL fanout limitations on the bus receivers would limit expansion to a total of 10 latches. The LEDs can be mounted on a separate display panel with connections by means of dual in line header plugs and sockets. A neutral tinted glass or plastic cover plate and an attractive wooden frame are good finishing touches for the visible portion of the project.

Figure 2: The CATERPILLAR program listing. Opcodes and addresses are specified in octal notation for the 8008 CPU. Comments at the right are designed to help convert the program to other machines.

```
address   octal code  label    op.   operand     commentary

010/000   250          START    XRA                clear carry and A;
010/001   036 377               LDI   11111111B    first caterpillar word;
010/003   046 000               LEI   00000000B    second caterpillar word;
010/005   304          MARCH    LAE                right byte to A;
010/006   022                   RAL                shift left into carry;
010/007   340                   LEA                then saved for next time;
010/010   303                   LAD                left byte to A;
010/011   022                   RAL                shift left into carry;
010/012   330                   LDA                then saved for next time;
                    *   note that RAL shifts old carry into A each time,
                    *     thus the above accomplishes shift from E's high
                    *     order bit into D's low order bit via carry...;
010/013   177                   OUT   37           write into left lamps;
010/014   304                   LAE                fetch right value;
010/015   175                   OUT   36           write into right lamps;
010/016   026 300               LCI   192D         set delay loop constant;
010/020   307          DELAY    LAM                use
010/021   307                   LAM                   several
010/022   307                   LAM                      longish
010/023   307                   LAM                        instructions
010/024   307                   LAM                          to
010/025   307                   LAM                          s t r e t c h
010/026   307                   LAM                        out
010/027   307                   LAM                     the
010/030   307                   LAM                loop
010/031   307                   LAM                (a BYTE in time saves nine);
010/032   021                   DCC                decrement delay count;
010/033   110 020 010           JFZ   DELAY        if non zero then repeat delay;
010/036   104 005 010           JMP   MARCH        back for another step;
                    *   note that during the operations at addresses 010/013
                    *     to 010/036 the carry bit value set by the last RAL
                    *     is retained unchanged, so that the RAL at 010/006
                    *     will shift the old high order bit of the left register
                    *     (D) into the low order bit of the right register (E);
```

This program uses only the internal CPU registers for its data, and assumes that 8008 output ports 36 and 37 are assigned to latched 8 bit displays so that a visible pattern can be seen.

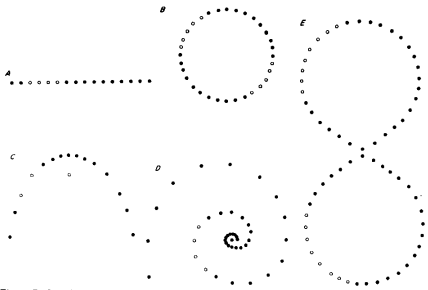


Figure 3: Creativity in the arrangement of the lamps of your blinking light peripheral can make the show more impressive and interesting. A: The traditional blinking light display, a la control panel, is a straight line of sixteen lamps in a row using two output latches. B: Tradition is nice, but how about a bit of circular thinking, using thirty two lamps in a circle with four output latches. C: Don't rule out the bouncing ball effect either. Here we make a parabola shape using sixteen lamps, one at the focus, with two output latches. D: With enough lamps, we can make a spiral show with one or more arms. Here are forty lamps using five output latches. E: There are an infinite number of patterns to be made. In this case, infinity consists of sixty four lamps using 8 output latches.

light pattern function for 16 bits? The basic way in which this is accomplished is to shift the bits through the carry flag of your machine.

Figure 2 illustrates a program which does a 16 bit shift in an 8008 microcomputer, and sends the data to the output ports reached by the OUT36 and OUT37 instructions (octal 175 and 177, respectively). The result is a moving display of 16 lights. A band of several bits is always marching right to left around the display at a steady rate.

But, why let the imagination end at a mere 16 bit display? The use of IO ports can be extended without too much cost (given limitations on power supplies, of course). The cost of two 7475s, eight resistors, and eight LEDs is about $3 (using BYTE's advertising pages as a source of prices). Thus a very reasonable display of 64 LEDs will set you back only $24 and the time it takes to put it together.

Using a bit of imagination, the extension of the program can lead to interesting patterns running around non-linear configurations such as those illustrated in figure 3. Variations in the patterns result in displays of light. Beyond the scope of this short article are more complicated programs using the same display peripheral: Programs in which the light patterns do more interesting things than simply chasing around the race-track at uniform speeds.

Try building a simple 16 bit blinking light display version first, then go on to bigger and better things. Then see if you don't agree that there's more to BLINKing lights than meets the eye! ■

Reprinted from *BYTE Magazine*, January 1976.

JIM GEROW WINS SOFTWARE CONTEST AGAIN

by Bill Gates

Another fourteen programs were added to the library this month, and I think a lot of Altair users will find them useful.

Jim Gerow, whose FORTRAN cross-assembler #521751 has been the most requested program in the library, has sent in a FORTRAN simulator for the 8800 (#1123751) which is the winning major program this month. When a file is assembled by the cross-assembler there is an option to output the assembled code as a file that the simulator can read and execute. The simulator allows the user to set breakpoints, dump the registers, dump memory contents, change the registers and change memory contents.

The second place major program is S. Armstrong's tape labeling program (#117751). This program is very useful for users who have lots of paper tapes.

The third place major program is Jerry Ford's JAMON (#117752). Commands are provided which make it unnecessary to use the front panel switches.

The winning subroutine is George Rompot's GET routine (#1117751). This program is especially notable for its good documentation. Unlike Mr. Rompot, most Altair users haven't programmed for very long. A hard-learned lesson about programming is that good documentation is not just an aesthetic, but is indispensable. It forces a programmer to be aware of what subroutines he has embedded in his program, what calling sequences should be used and explains the assumptions and techniques used in each section of code.

The second place subroutine is Jack Coats' BCD multiply routine (#113751). BCD multiplication is more complicated than binary multiplication. For people who like numbers with hundreds of digits and understand BCD (or want to), this routine can be set up for up to 512 digits per number without much trouble.

Software Library

#1020751
Author: Gerhald Hansel
Length: 46 lines of Basic
Title: Addition problems
Produces a group of math addition problems and answers. Written to use PRINT USING, a feature of Extended Basic, to format its output.

#1020752
Author: Lee Eastburn
Length: 290 locations
Title: Print program
Dumps a program with page headings, an address field and the octal contents of the address. Columns are provided to fill in labels, instruction mnemonics and comments.

#1121751
Author: Gary Tack
Length: 152 bytes
Title: Random Magic Squares
Generates 3 numbers which are used to make a magic square (a 3x3 grid in which the sums of the numbers in each row, column and diagonal are equal). All "magic squares" can be generated since 3 numbers characterize a magic square. Each group of 3 numbers generates a distinct magic square. Stores magic square information in memory. Doesn't do any input/output.

#1117751
Author: George Rompot
Length: 41 bytes
Title: GET - a "Lifeline" subroutine
A well-documented subroutine to perform the functions of the TGET and LGET subroutines specified in the lifeline article in Byte magazine. (October 1975, pp. 34-41) Includes test routine.

#1117752
Author: George Rompot
Length: 25 bytes
Title: CHECK
A well-documented subroutine to calculate a 16-bit sum of an area of core specified in the calling parameters. Optionally checks the computed checksum with a checksum provided in the calling sequence. A testing program is included.

#117751
Author: S. Armstrong
Length: 124 instruction bytes (not including embedded NOPS)
342 data bytes
Title: Punch tape label
Punches paper tape labels by using 5 tape frames to make a block letter. Can "print" any alphabetic, numeric, or common delimeter. Additional characters can easily be added. "Prints" an entire line at once with separating blanks.

#117752
Author: Jerry Ford
Length: 318 bytes
Title: JAMON - a teletype monitor
This monitor allows examining (dumping) of memory blocks, depositing into memory blocks and jumping to a specified address. All inputs and outputs are in octal. Subroutines for character input, character output, octal input and octal output are included.

#113751
Author: Jack Coats
Length: 83 bytes
Title: BCD multiplication subroutine
Currently set up to multiply two 8-digit numbers in memory and give a 16-digit number in memory for a result. Can be changed for different size numbers without much difficulty.

#1030752
Author: John Trautschold
Length: 16 bytes
Title: VLCT load program
Allows entry of a load address and program data in octal from a VLCT keyboard. The entered data is displayed for verification. Documented with high 8-bits of start address as "XXX" so the program can easily be located at the start of any memory page (a block of 256 consecutive memory locations, the first location of which has an address divisible by 256 [decimal]).

#1030751
Author: Lee Eastburn
Length: LOAD section-71 bytes
DUMP section-96 bytes
Title: ASCII Octal Loader and Dumper
The DUMP section outputs the contents of a block of memory in ASCII octal (3 characters ["0" - "7"] for each byte). Twenty bytes are printed on each line of output. The LOAD section loads a tape of the same format, ignoring all characters except "0" through "7." The start and end locations are specified by changing LXIs in the programs themselves.

#1021751
Author: Jack Coats Jr.
Length: 46 bytes
Title: A Dual-Tasker
An interrupt level routine to switch between two tasks. Context is saved on the stack and the stack pointer is saved in memory.

#1027751
Author: Jack Coats
Length: 43 bytes
Title: Time of Day routine
Using the real-time-clock interrupts this program keeps track of the hours, minutes, seconds and hundredths of seconds that elapse. The data is stored in binary coded decimal.

#1027752
Author: Roger J. Walker
Length: 53 bytes
Title: TVTDR-I/O handler for TVT-II
TVTDR is designed to overcome a deficiency in the TVT-II that prevents it from blanking to the end of the line when a carriage return is output. The appropriate number of spaces are output instead of a carriage return. Instruction for interfacing to Altair Basic are given.

#1123751
Author: Jim Gerow
Length: 33 pages
Title: A FORTRAN simulator for the 8080
A FORTRAN program to take as input the code generated by Mr. Gerow's Cross Assembler (#521751) and simulate the Altair's execution of the code. Extremely useful for debugging. A manual is included with the source listing. A list of changes to be made to cross assemblers received before December 1, 1975 to allow them to work with the simulator are given.

SOFTWARE

The Status of BASIC

by Bill Gates

Some of the more persistent and expert users of Basic have noted a few bugs in some of the versions of Altair Basic.

SPR (Software Performance Report) #1-
CSAVE sees only three zeros.
Symptoms: Some CSAVEd files will not CLOAD. Happens infrequently. Found by Bill Gates and later noted by Steve Grumette and J. Scott Williams.
Present in: 3.0 and 3.1 cassette of 8K.

Change: Made in 3.2. In 3.1 8K cassette if 10075 base 8 (4157 decimal) contains a 6, change the location after this (10076) from 4 to 3 (POKE 4158,3). If 10075 doesn't contain a 6 and 10076 (4158) contains a 6, change the next location (10077) from 4 to 3 (POKE 4159,3).
Files CSAVEd before the fix will CLOAD properly after the fix.

SPR #2-
Unreferenced non-array variables.
Symptoms: Assigning an array element the value of a formula containing a non-array variable which has never been referenced before is done improperly. A correct program will never do this. Found by Steve Grumette.
Present in: 3.1's and 3.0 4K
Change: Made in 3.2.

SPR #3-
String compare.
Symptoms: The string comparisons: A$>B$, A$>=B$ and A$<>B$ return zero if A$ is an extension of B$. That is, B$=LEFT$(A$, LEN(B$)) and Len (A$)>Len (B$). Found by Steve Grummette.
Present in: 3.0 and 3.1 8K's.
Change: Made in 3.2. Decimal numbers in []. Patch for 3.1 8K cassette:

```
[2778] 5332/ from 267 [183] to 326 [214]
[2779] 5333/ from 57 [47] to 1 [1]
[2780] 5334/ from 310 [200] to 330 [216]
```

Patch for 8K 3.1 paper tape:

```
[2765] 5315/ from 267 to 326
[2766] 5316/ from 57 to 1
[2767] 5317/ from 310 to 330
```

For a piece of software that's been running for 9 months, with over a thousand copies in use and 9 different versions, it's pretty respectable to have had only five bugs--none of them serious. (2.0 wasted stack space when "FOR" loops terminated and handled non-square matrices with more than one dimension improperly.)

Beware of using a loop variable, first in an outer loop that never completes and then in an inner loop. Example:

```
10  FOR I = 1 to 10
20  IF Y(I) = 7 THEN 40
30  NEXT I:  PRINT "ERROR":STOP
40  FOR J = 1 to 13
50  FOR I = 1 to 5
    :
    :
60  NEXT I
70  NEXT J
```

When a "FOR" statement is executed, a check is made to see if any uncompleted loops use the same loop variable as the "FOR" which is just starting. If so, the original loop and any started after it are automatically terminated. Therefore in this example, the execution of line 50 would terminate the loops started in lines 10 and 40.

Users wishing to modify the way Basic does I/O to the terminal should use a machine language program or the PEEK function to find the four different locations where terminal I/O is affected:

```
#1 OUTPUT:  IN   0
            ANI <mask for output ready>
            JZ or JNZ to the IN 0
            POP  PSW
            OUT  1
            RET

#2 INPUT:   IN   0
            ANI <mask>
            JZ or JNZ back to the IN 0
            IN   1

#3 CONTROL-C check:
            IN   0
            ANI <mask>
            RNZ or RZ
            CALL D

#4 (not in 4K) High Speed Control-C
                check:
            IN   0
            ANI <mask>
            CZ or CNZ
```

The addresses of these four code sequences varies from BASIC to BASIC.

Two special options are available in BASIC for $15.00 each. (NOTE: Cassette files will only CLOAD in a BASIC with the same special options as the BASIC that CSAVEd the file.) One option adds a CONSOLE command that allows a user to change the terminal he is using with a single command. The other option adds LLIST and LPRINT which are the same as LIST and PRINT, except their output goes to the line printer. Patches are not given for BASICs with special options.

Besides the changes already mentioned, 8K BASIC 3.2 will have two other enhancements:

1) Numbers less than .1 and greater than or equal to .01 will print as .0XXXXXX (trailing zeros suppressed) instead of X.XXXXXE-2.

2) Cassette versions will allow for verification that a CSAVEd file is recorded properly. Giving the command CLOAD? <file-character> will search for the named file and compare it to the program in memory typing "NO GOOD" if they are different. Suggested by J. Scott Williams.

Note that numbers that print the same are not always equal. This is because the seventh digit of accuracy that BASIC stores is never printed. Use of the intrinsic functions or exponentiation can cause an error of ±1 in the seventh digit. Example: (from Dick Heiser)

```
10  X = 5↑2
20  PRINT X
30  IF X<>25 then PRINT "NOT EQUAL"
RUN
 25
NOT EQUAL
OK
```

This calculation of 5↑2 gives 25.00001 which does not equal 25, but which prints like 25. STR$(X) = STR$(25) would work as a check for two numbers being very close.

The Random Number generator is not as pseudo-random as it should be (first noted by Dick Heiser), and new algorithms are being examined to improve it.

A lot of people ask if there is any problem using lower case letters with BASIC. As long as the reserved words and variable names are entered in upper case, there is no problem. Remember, though, "Z"<"a".

If anyone is using BASIC version 1.1, you have a copy of a tape that was stolen back in March. No customers were ever shipped 1.1, as it was experimental and is full of bugs!

If anyone is using 2.0, I encourage them to upgrade to 3.2, mainly because Version 3 BASICs are so much faster than Version 2.

As for Extended BASIC, the non-disk version was completed a few weeks ago and should be completely documented and out the door by the time this is published. Paul Allen demonstrated Extended BASIC in Denver a few weeks ago, and Denver's top hacks couldn't find any bugs. I think a lot of people will really like the "EDIT" command that's been added.

Software Notes

by Bill Gates

To go with the decimal output routine given in September's issue, here is a decimal input routine. For fun, modify it so it checks for overflow. (Hint: use the carry bit generated by DAD.)

```
;routine to do decimal input (DECINP)
;return result in [H,L].  [A] contains the terminating character.
;[D,E] is smashed.  Stack use:  INCHR is called to get a character
;in [A].  Overflow is not checked.

DECINP:     LXI    H,Ø          ;initialize to zero
DECLOP:     CALL   INCHR        ;read a character into [A]
            CPI    "9" + 1      ;see if it is > "9"
            RNC                 ;return if so
            CPI    "Ø"          ;see if it is < "Ø"
            RC                  ;return if so
            SUI    "Ø"          ;[A] = numeric value of new digit
           †MOV    D,H          ;[D,E] = [H,L]
           †MOV    E,L
            DAD    H            ;[H,L] = old [H,L]*2
            DAD    H            ;[H,L] = old [H,L]*4
           †DAD    D            ;[H,L] = old [H,L]*5
            DAD    H            ;[H,L] = old [H,L]*1Ø
            MVI    D,Ø          ;[D,E] = new digit
            MOV    E,A
            DAD    D            ;add in the new digit
            JMP    DECLOP       ;get more digits
```

†eliminate for octal input

The simplicity of loading BASIC into an Altair is important, since people without PROMs or BASIC on ROM must load it every time they power up their machine. Here are the details of how this process works:
(All numbers are octal)

The format of a binary tape of BASIC or a monitor is as follows:

```
leader = 175 currently
last byte of checksum loader (311)
next to last byte of checksum loader (172)
  .
  .   intermediate checksum loader bytes
  .

second byte of checksum loader (61)
first byte of checksum loader (363)
gap of null characters (0)
<checksum data block -- up to 256
 data bytes per block>
<additional checksum data blocks
 until all program data has been given>
<checksum go block>
```

Checksum loaders can be loaded into most pages of memory depending on location 2 of the boot and which checksum loader is on the tape. The checksum loader for 4K BASIC and the Package I monitor starts at location 7400. The checksum loader for 8K BASIC starts at 17400. Except for being relocated, these loaders are identical.

Checksum data block:

```
74 start character
number of data bytes in the block (Ø=256)
lower 8 bits of storage address
high 8 bits
<data bytes>
checksum byte = summation without carry of all bytes
                in the block except the 170 and count
                specification
```

— CONTINUED PAGE 21 —

Notes on Disk Extended BASIC

Disk Extended BASIC is a stand-alone system which is delivered on a floppy disk. This floppy has been formatted and loaded with the utility files that print directories, format other disks and do disk diagnostics as well as Extended BASIC. The disk loader is about 100 bytes and can either reside on a PROM, be keyed in or be loaded from ACR or paper tape using the standard 20-byte bootstrap.

During initialization the number of disk buffers (maximum 8) and random access blocks (maximum 8) to be allocated are determined by the user. These numbers determine the number of files that can be open simultaneously and the number of random access files that can be open simultaneously, respectively. The disk drives that are to be brought on line are all checked for proper formatting and the locations of free sectors are stored in memory.

Each floppy can store 300,000 bytes (characters) of user information. The rest of the storage space on the floppy is used to store the file structuring and error detection information. Up to 254 files can be stored on a floppy and a single file can be up to 300,000 bytes long. A file must reside entirely on a single floppy, thus no file can be larger than 300,000 characters.

There are three modes for file access:

1) Sequential input: The file is stored as ASCII text. Numbers and strings are read as character strings in exactly the order they were typed in or written out.

2) Sequential output: Any previous contents of the file are deleted and output is done item by item in ASCII.

3) Random access: Each record is 128 characters. Numbers are written in binary, so integers take 2 bytes, single-precision numbers 4 bytes and double-precision numbers 8 bytes. Special functions return the record number of the current position in the file (LOC) and the highest numbered record currently allocated (LOF) in the file. READs and PRINTS of random access records can be intermixed. A specific record number in the file can be specified by a formula in both the READ and PRINT statements.

All the features of non-disk Extended BASIC are provided. The current expected delivery date for Disk Extended BASIC is January 1, 1976. To use Disk Extended BASIC, 16K of memory is required since the program itself uses 12.5K and each disk buffer and random access block require another 140 bytes.

end

Checksum: go block

```
                    170 start character
                    lower 8 bits of address to jump to
                    high 8 bits
```

The data block for locations Ø through 376 is the last data block on the tape so the bootstrap loader doesn't have to be keyed in again when checksum errors occur, unless the checksum error is on the final data block.

BOOT STRAP LOADER
start at location zero

```
0/    LXI   H,
      number of bytes in the checksum loader
      page number of the checksum loader

                    Set [H,L] to point to the last location
                    in the checksum loader + 1.

3/    LXI   SP,STKADR   Set [SP] so returns come back to this
                        location.  After each return [SP] is reset.

6/    IN    Ø           See if there is a character, and loop
      RAR               if not.
      RC

12/   IN    1           Read a character and see if it's leader.
      CMP   L           (Lead character = number of bytes in the
      RZ                checksum loader)

16/   DCR   L           Store the data in the next lower location,
      MOV   M,A         and loop unless all bytes have been read.
      RNZ

21/   PCH   L           Start the checksum loader at its beginning.

22/   STKADR:  DW LOPADR
                    The stack pointer points here, so this
                    gives the address returns branch to.
```

This bootstrap loader has several advantages:

1) Leader is allowed.
2) Only 20 bytes need to be keyed in.
3) It automatically starts the checksum loader, so only one tape needs to be entered.
4) It can run from Read Only Memory.
5) It starts at a convenient location (zero).
6) It is easily relocated by changing the addresses at locations 4 and 22.
7) To load different checksum loaders, only location 2 needs to be changed.

I've written a bootloader that only takes 13 bytes of keyed-in data, but anything smaller than 20 bytes isn't easy to use.

Next Month: Signed and unsigned arithmetic
Multiprecision arithmetic

Letters to the Editor

CONTINUED FROM PAGE 10

2. The modification proposed for the 4K memory card converts the protect input circuit from a toggle input to a type D flip flop. It won't hurt anything, but all it eliminates is the need to occasionally press protect more than once. A better design for the protect flip flop in the 4K memory card would have been a type D circuit. However, it is hardly a major issue, and the cure is worse than the problem.

3. The proposed change to C-7 will decrease the timing margins and should not be made.

4. A complete discussion of the ready line appears in Ed Roberts column in this issue.

If you hear of some mod or change to the system, please write or call us before you install it. Any modification that we believe is useful, reliable and doesn't damage the system will appear in this publication.

--DB

ALTAIR USERS

Jeffrey G. Clark
P.O. Box 2422
Springfield, MA 01101

Orville F. Hamm
4751 Louisiana Ave.
St. Louis, MO 63111

Charles Can KL7HRP
SR Box 80688
Fairbanks, AK 99701

Craig Brockmeier
4715 SE Adams #922C
Bartlesville, OK 74003
(918) 333-5608

Capt. James K. Bostick
12307 S. 33rd St.
Omaha, NE 68123
(402) 292-2466 (home)
294-5932 (work)

Robert A. Van Winkle WB6HKA
288 Woodbridge Ave.
Yuba City, CA 95991
(916) 673-1523

James T. Mattley
6417 Fernhurst Ave.
Parma Heights, OH 44130

Craig Pearce
2529 S. Home Ave.
Berwyn, IL 60402

William T. Shaw
3F Hogarth Circle
Cockeysville, MD 21030

Steve Patchen
Univ. of Washington
1334 N. 122nd
Seattle, WA 98133

Douglas Ingraham
Box 523
SD School of Mines
Rapid City, SD 57701

Timothy Radde
17 Harding St.
Pittsfield, MA 01201

Mark Rothstein
15401 Pegg Court
Bowie, MD 20716
(In a previous issue, there was an error in printing Mark's address. He lives in Bowie, Maryland--not Missouri)

Daniel Duncan
Box 3388
Pasadena, CA 91103

Donald D. Henson
1676 Ala Moana Blvd.
Apt. 307
Honolulu, HI 96815

The following is a list of Altair dealers. Most Altair kits should be in stock at these outlet stores.

Arrowhead Computer Co.
(The Computer Store)
11656 W. Pico Blvd.
Los Angeles, CA 90064
Lois & Dick Heiser
(213) 478-3168

Byte'Tronics
5604 Kingston Pike
Knoxville, TN 37919
Bruce Seals
(615) 588-8971

The Computer Center
3330 Piedmont Road
Atlanta, GA 30305
Jim Dunion, Rich Stafford
Steven Mann, Ron Roberts
(404) 231-1691

Computer Kits
1044 University Ave.
Berkeley, CA 94710
Pete Roberts
(415) 845-5300

Computers & Stuff
1092 S. State St.
Orem, UT 84057
Eric & Debra Stewart
(801) 224-2066

Gateway Electronics
2839 W. 44th Ave.
Denver, CO 80211
George Mensik
(303) 458-5444

Gateway Electronics
8123-25 Page Blvd.
St. Louis, MO 63130
Alfred L. Elkins (Lou)
Stuart Bartfield
(314) 427-6116

Marsh Data Systems
Suite 120, 1805 N. Westshare
Tampa, FL 33607
Don Marsh
(813) 872-7334

Microsystems
6605A Backlick Rd.
Springfield, VA 22150
(Washington DC area)
Russell Banks
(703) 569-1110

CTI Data Systems
3450 East Spring St.
Long Beach, CA 90806
Fred Whitney
(213) 426-7375

Byte Shop
1063 El Camino Real
Mountain View, CA 94043
Boyd W. Wilson, Paul Terrill

Ridgway East, Inc.
161 Bell St.
Chagrin Falls, OH 44022
(Cleveland)
Ray Wassum, Jack Stevens

The Computer, Inc.
PO Box 2621
Framingham Center, MA 01701
Richard F. Brown
(617) 877-6984

I/O Programs for the ACR

Input/Output programs for the 88-ACR

By Tom Durston

One request we've been getting frequently is for simple machine language programs to write and read data on tape through the 88-ACR. Listed below is a program to write and a program to read using the 88-ACR. These programs have been used in our engineering department to store lengthy test routines, and can be used for any type of data.

WRITE PROGRAM - 38 bytes

Writing data on tape through the 88-ACR is accomplished by first specifying the start address of data and the end address of data. Then a test byte (000 in this program) is written, followed by data output. The last portion of the program tests to see if the program has transmitted the last byte of data. If it has, the program jumps to the last positions in memory, and is observed by a change in the address lights on the front panel. If the program hasn't outputted the last data byte, the H & L registers are incremented by 1 and the program outputs the next byte. This program is placed in the upper portion of 4K memory with a starting address of 017,000. The location may be changed, but be sure to change all jump addresses accordingly. After recording data that includes program information, write down the start and end address on the tape cartridge along with the name and test byte of the program for identification.

When recording data at the beginning of a cassette tape, record at least 15 seconds of steady tone before running the write program (to get past the plastic leader and wrinkles in the beginning of the tape). Also, if recording more than one batch of data, leave at least 5 seconds of steady tone between batches. This program is written for 88-ACR addresses of 6 & 7.

88-ACR WRITE PROGRAM

TAG	MNEMONIC	ADDRESS	OCTAL CODE	EXPLANATION
	LXI	017,000	041	Load immediate H&L register pair
		1	xxx	Lo ⎫ starting address of
		2	xxx	Hi ⎭ data to be written
	LXI	3	001	Load immediate B&C register pair
		4	xxx	Lo ⎫ end address of
		5	xxx	Hi ⎭ data to be written
	MVI	6	076	Move immediate to accumulator
		7	000	Test byte to be written at beginning
	OUT	017,010	323	Output data from accumulator
		11	007	Data channel # of 88-ACR
TEST	IN	12	333	Input data to accumulator
		13	006	Status channel # of 88-ACR
	RLC	14	007	Rotate accumulator left, test for D7 true
	JC	15	332	Jump if carry (D7 not true)
		16	012	⎫
		17	017	⎭ To "TEST"
	MOV	017,020	176	Move contents of memory specified by H&L register to accumulator
	OUT	21	323	Output data from accumulator
		22	007	Data channel # of 88-ACR
	MOV	23	175	Move contents of L register to accumulator
	CMP	24	271	Compare accumulator vs B register
	JNZ	25	302	Jump if not zero (L ≠ B)
		26	040	⎫
		27	017	⎭ To "NEXT"
	MOV	017,030	174	Move contents of H register to accumulator
	CMP	31	270	Compare accumulator vs C register
	JNZ	32	302	Jump if not zero (H ≠ C)
		33	040	⎫
		34	017	⎭ To "NEXT"
	JMP	35	303	Jump (if L = B and H = C)
		36	375	⎫
		37	017	⎭ To "END"
NEXT	INX	017,040	043	Increment register pair H&L
	JMP	1	303	Jump
		2	012	⎫
		3	017	⎭ To "TEST"
END	JMP	017,375	303	Jump (loop to self)
		376	375	⎫
		377	017	⎭ To "END"

CONTINUED FROM PAGE 22

I/O Programs for the ACR

READ PROGRAM - 48 bytes

As in the write program, start and end addresses of incoming data are specified first. Next, the program looks for the test byte (000 in this program). Once the test byte is detected, the program inputs data and stores it in memory as specified by the H & L registers. The next portion of the program tests to see if the end memory address has been filled. If it has, the program jumps to the last positions in memory, and is observed by a change in the address lights on the front panel. If it is not the end, then the program increments H & L by 1 and jumps back to input another data byte. This program is placed in the upper portion of 4K of memory with a starting address of 017,000. The location may be changed, but be sure to change all jump addresses accordingly. When reading data back in, the tape and program should be started a few seconds before the start of data.

88-ACR READ PROGRAM

TAG	MNEMONIC	ADDRESS	OCTAL CODE	EXPLANATION
	LXI	017,000	041	Load immediate H&L register pair
		1	xxx	Lo | starting address of
		2	xxx	Hi | data to be read
	LXI	3	001	Load immediate B&C register pair
		4	xxx	Lo | end address of
		5	xxx	Hi | data to be read
TSTBT	IN	6	333	Input data to accumulator
		7	006	Status channel # of 88-ACR
	RRC	017,010	017	Rotate accumulator right (test DØ true)
	JC	11	332	Jump if carry (DØ not true)
		12	006	| To "TSTBT"
		13	017	
	IN	14	333	Input data to accumulator
		15	007	Data channel # of 88-ACR
	CPI	16	376	Compare immediate with test byte vs accumulator
		17	000	Test byte
	JNZ	017,020	302	Jump if not zero (test byte≠input byte)
		21	006	| To "TSTBT"
		22	017	
TEST	IN	23	333	Input data to accumulator
		24	006	Status channel # of 88-ACR
	RRC	25	017	Rotate accumulator right (test DØ true)
	JC	26	332	Jump if carry (DØ not true)
		27	023	| To "TEST"
		017,030	017	
DATA	IN	31	333	Input data to accumulator
		32	007	Data channel # of 88-ACR
	MOV	33	167	Move contents of accumulator to memory address specified by H&L registers
	MOV	34	175	Move contents of L register to accumulator
	CMP	35	271	Compare accumulator vs B register
	JNZ	36	302	Jump if not zero (L ≠ B)
		37	051	| To "NEXT"
		017,040	017	
	MOV	41	174	Move contents of H register to accumulator
	CMP	42	270	Compare accumulator vs C register
	JNZ	43	302	Jump if not zero (H ≠ C)
		44	051	| To "NEXT"
		45	017	
	JMP	46	303	Jump (if L = B and H = C)
		47	375	| To "END"
		017,050	017	
NEXT	INX	51	043	Increment H&L register pair
	JMP	52	303	Jump
		53	023	| To "TEST"
		54	017	
END	JMP	017,375	303	Jump (loop to self)
		376	375	| To "END"
		377	017	

Notebook

4K PROTECT

Hitting the protect switch will occasionally not protect the 4K dynamic memory board. This problem is alleviated by removing pin 10 of IC"T" from +5V and jumpering it to ground instead. Ground is available on pin 11 of IC"T".

SIOB & ACR OWNERS

Before inserting IC"G" on the SIOB, TTL interface, check for a copper land between pins 2 and 3. If there is one, remove it, or you will get the following symptoms: you cannot enter a 1 (HI LEVEL) in bits 2, 3, 4, or 7 (of Ø-7) except at the memory address which corresponds to the address of the I/O board.

contributed by Edwin F. Hanpton

MISPRINT

October 1975 Computer Notes, page 9: USING SERIAL BOARDS
Echo Program Step 17
reads 012
should read 013
Interrupt Program Step 11
reads 011
should read 010

Seminar Tours

Schedules for the next MITS-MOBILE tours are now being planned. Exact dates cannot be announced, as we do not yet have confirmation of the availability of meeting rooms at all locations.

The tentative itinerary for the last three weeks of February includes stops in Lubbock, Ft. Worth, Dallas, Houston, Corpus Christi, San Antonio, Austin, Odessa, and El Paso--all in that huge state of Texas. Also Tucson and Phoenix, Arizona.

From mid-March to mid-April we plan to be on the west coast--including San Diego, Los Angeles areas, Fresno, San Francisco Bay area, Sacramento, Portland, and Seattle.

The general format of these lectures will continue to aim at bringing information of interest to individuals with varying levels of experience.

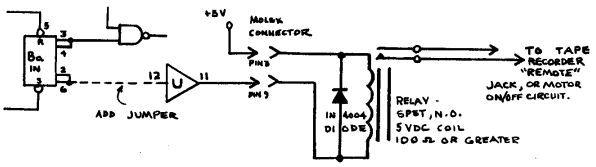
Upgrading plans for our seminars includes: more material in the course manual, more emphasis on I/O devices, more equipment demonstrations including several new MITS products.

...more notes on ACR

Tape Recorder Motor Control

Another request for the 88-ACR has been for start/stop motor control for DC motor tape recorders with the subminiature phone jack marked "REMOTE." A simple way to do this is to utilize an unused flip-flop normally intended for the interrupt function on the 88-SIOB board. Since the interrupt circuit is usually not used in the 88-ACR mode, it is possible to connect the output of one of the flip-flops to an unused driver of an 8T97, using it as a relay driver.

The circuit shown below uses control channel bit DØ to turn the motor on and off. Consult page 3 of the 88-SIOB manual and the 88-SIOB schematic for detailed information on this circuit.



The other half of IC B may be used for another control function in the same manner. IC U has 3 other spare drivers that may be used.

We suggest the relay be mounted externally to prevent recorder supply voltages from interfering with the Altair 8800.

For the machine language Read/Write programs, adding the following instructions will allow use of the start/stop feature.

To turn the motor circuit on, place these instructions before the beginning of the Write and Read programs.

Location	Octal Code	Mnemonic/Description
016,374	076	MVI move immediate to accum.
375	001	Turn on motor
376	323	Output data from accum.
377	006	Control channel

NOTE: For Write program, single step through these 4 instructions, wait appropriate time (5-15 seconds), then hit RUN.

To turn the motor off, place these instructions before the jump to self loop at "END." Also change data in location 017,376 to 371.

Location	Octal Code	Mnemonic/Description
017,371	076	MVI
372	000	turn motor off
373	323	Output
374	006	Control channel

NOTE: The flip-flops Ba and Bb do not have power on clear. It may be necessary to single step the motor off circuit to clear these flip-flops.

For use with Altair 8K BASIC, use:
OUT 6,1 - to turn motor on
CLOAD or CSAVE
OUT 6,0 - to turn motor off

Keep in mind that if writing, you must turn the motor on 5-15 seconds before outputting data.

MITS/6328 Linn NE/Albuquerque, NM 87108/ 505-265-7553
OR 262-1951

CLASSIFIED ADS

WANTED: Name & address for owner of 8800 serial number 222755K. Information was missing from warranty card mailed from Dayton, Ohio.

WANTED: Name & address for owner of 8800 serial number 221128A. Information was missing from warranty card mailed from Los Angeles.

WANTED: Software developer with IBM equipment who will convert EBDICT to ASCII on a one shot basis. Contact Jim Leek, Bakersfield Audio, 2801 F Street, Bakersfield, CA 93301

WANTED: FORTRAN to run on the Intel 8080 CPU based systems. Anyone with information on this please contact Jack O. Coats, Jr., El Paso Computer Group, 213 Argonaut #27, El Paso, TX 79912

FOR SALE OR TRADE: Techtronics 4602 video hard copy unit. Produces 8 1/2x11 prints from standard video (i.e. video terminal). Contact Mark Bunker/2703 S. 71st Pl./Kansas City, KS 66106 913-375-1138

FOR SALE: 10 CPS EIS impact printing terminal. Standard RS-232C data in and out. Close equivalent of TTY model 33KSR. Good shape. $230 FOB Denver. M. Smith/4355 S. High St./ Englewood, CO 80110

Is anyone working on inventory control & bookkeeping packages in BASIC (extended)? If so, please contact Eugene Zander/O'Keefe Supply Co./ 613 Williamson St./Madison, WI 53703

We are working on computer aided instruction packages and executive programs to operate under same. Especially interested in graphics display terminal systems. If you're interested or have info, please call: Sgt. Wesley Isgrigg or Sgt. Stanley Herr/74055 Casita Dr./29 Palms, CA 92277 (714) 367-6996

"Handy Dandy" paper tape winder. Uses 4 D batteries (not included). $15.00 plus $1.00 postage & handling. Bill Roch, 5133 Catalon Ave., Woodland Hills, CA 91364

USERS GROUP CONT. FROM PG. 2

Chicago Users Group
% Robert Swartz
195 Ivy Lane
Highland Park, IL 60035

Southeast Minnesota Amateur Computer Club (SEMACC)
2122 NW 17 Ave.
Rochester, MN 55901

Oklahoma City Club
2412 SW 45th
Oklahoma City, OK 73119

Texas Computer Club
% L.G. Walker
Rt 1 Box 272
Aledo, TX 76008

MITS Introduces New 6800-Based Computer

ALTAIR 680

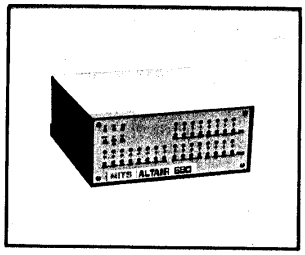
By James B. Vice

Not long ago MITS made a giant step towards the day when a computer will be a standard household item. They've now moved another step closer with the new ALTAIR 680 microcomputer. This new machine is an excellent compromise between computer power and low cost structure, without sacrificing design reliability. The MITS engineers have designed a system based on the 6800 microprocessing unit (MPU). This MPU, available from both Motorola and American Micro-Systems, adapts nicely to a minimum design configuration.

The ALTAIR 680 measures 11 1/16" wide x 11 1/16" deep x 4 11/16" high, making it less than one-third the size of the ALTAIR 8800. The basic system is available in three configurations, depending on the intended application. These include a fully user-programmable system with complete front panel controls, and two versions oriented more towards dedicated program applications.

The compact size of the 680 obviously precludes any significant amount of internal expansion, although additional memory and I/O control are already on the drawing board. The 680's small size can be deceiving. The overall concept was to keep the machine as simple, small and inexpensive as possible; but it is a complete system in itself, with the exception of some I/O device. To the industrial user this may not be important, but for the "home brewers" it can be critical.

The construction of this machine is a relatively easy matter for even the most inexperienced kit builder. Almost all of the circuitry is contained on a single large printed circuit board, including memory and a built-in I/O port. In fact, this single board is a full computer with the exception of a power transformer and some control switches. This is where the main distinction between the three configurations is encountered.



Most hobbyists will be concerned with the full front panel model. This contains all of the necessary controls for addressing and entering data besides those for controlling the processor itself. A "turn-key" front panel model is also available which eliminates all controls except restarting the processor. This could be used in any application where it is desirable to eliminate the possibility of the operator or any other person affecting the machine's memory or computing cycle. An example for such an application might be its use in controlling an intrusion detection system, or for a manufacturing machine control system. The third configuration is closely related to the last. The 680 will also be available as just the large PC board mentioned above. This board contains everything but a power supply and controls. It could be used for the same applications as the "turn-key" model, only where the computer is to be "buried" inside another machine.

"An excellent compromise between computer power and low cost structure, without sacrificing design reliability."

The latter model would also be excellent for the experimenter who wished to purchase an absolute minimum and do a bit of his own designing. Another advantage to this particular idea is the considerable amount of information available on the 6800 microprocessing unit from Motorola Semi-conductor Products, Inc. Also, the 6800 MPU is TTL compatible and requires only one 5-volt power supply.

In the full front panel model of the 680 there is an additional printed circuit board. This board contains all of the logic circuitry necessary to reset, halt or start the processor. Also located on this board are switches and associated LED indicator lights for each of the sixteen address lines and eight data lines. The front panel printed circuit board mounts directly to the main printed circuit board via a 100-contact edge connector. This eliminates the need for a cumbersome wiring harness. The only other control is the power switch, located on the back panel of the unit for safety purposes.

On the dedicated program models PROM or ROM is used for setting the starting address and for programming.

The basic ALTAIR 680 computer can be subdivided into five functional sections. These are the MPU and clock, the memory, an I/O port, control and indication, and the power supply.

The first three of these sections, along with the power supply regulation components, are located on the main printed circuit board.

MPU and Clock

At the heart of the 680 system is the 6800 Microprocessing Unit. This is a versatile and very powerful little processor, yet it is directly responsible for the overall simplicity of the 680 design.

The 6800 is an 8-bit parallel processor using a bi-directional data bus and a 16-bit address bus. The latter gives it the ability to directly address 65K bytes of memory. The instruction set consists of 72 basic instructions with variations giving 197 different instructions out of a possible 256 codes (59 of the possible codes are unassigned). The variations are the result of the 6800's unique addressing structure.

- CONTINUED Page 7 -

BULLETIN — MITS drops 4K Price to $195!

see page 5

COMPUTER NOTES

October, 1975

Volume One Issue Five

A PUBLICATION OF THE ALTAIR USERS GROUP

Across the Editor's Desk

by David Bunnell

As I have stated in the past,things change rapidly in the computer business. This issue of Computer Notes is an example.

Our lead article, "ALTAIR 680", was to have been the big news item, but as it turned out, the drop in the price of 4K boards and the announcement of new I/O cards is probably of more interest to Altair 8800 users.

I want to thank all of you who were kind enough to return your surveys--they have had a big impact already. For the benefit of those who think we don't listen to our customers, here's a list of changes resulting from the survey:

1) The 4K board price has been drastically lowered to $195. People who have ordered 4K's at the old price will receive $50 credit for each 4K ordered. This credit is applicable to the purchase of a new 4K board, which means you can buy a 4K for the price of $145 !

2) The SLA (Software License Agreement) has been eliminated. No longer will you be required to read and sign this infamous document. However, this should not be interpreted to mean that Altair BASIC is now part of the public domain. We have the copyright on this and our position is the same. A cover letter will be sent with our software to explain. It will be simple and it will hopefully eliminate the existing confusion.

3) The price for Altair BASIC to those of you who don't have a minimum system has been dropped. Effective November 1, the price of 4K BASIC is $150, 8K BASIC is $200, and Extended BASIC is $350. Prices for those with a minimum system remain the same (see ad in this issue).

4) A letter-answering system has been set up. All letters will be answered within 5 working days and eventually this will be reduced to 3 days.

5) A Want Ad column will be set up in Computer Notes for individual Altair Users who wish to trade or sell computer related items to other Altair Users. This service will be free and the only restriction is that it cannot be used to sell products that compete with existing MITS Altair products.

6) Each month $25 credit toward the purchase of Altair products will be awarded to the author of the best Altair applications article. If you have done something interesting with your Altair, write an article about it and submit it to: Editor/Computer Notes/MITS 6328 Linn NE/Albuquerque, NM 87108

By the way, the September issue of Computer Notes was not printed until the last week in September and this current issue is slated to be printed on October 24. Different publications have different policies regarding the date on their publication. Some magazines like Popular Electronics come out three weeks early; i.e., the November issue comes out during the first or second week of October. Other publications like Digital Design come out during the third of fourth week of the month on the cover (the November issue will be out around the 20th of November). If you got your September issue of C.N. around October 11, you received it "on time." This may change if we have a chance to catch up.

* * *

Recently I attended the "Computer Fair" held at the Lawrence Hall of Science on the University of California campus in Berkeley. I went there at the urging of the People's Computer Company Dragon, Bob Albrecht.

The Lawrence Hall of Science is the huge bastion of modern architecture that sits on a hilltop overlooking Berkeley and most of the bay area (quite a view). Inside, there are a number of museum type science exhibits -- the kind that are normally associated with yawning. There

— CONTINUED PAGE 4 —

COMPUTER BOPPERS

ALTAIR SERVICE DEPT.

Barbara Sims

A few computer clubs are beginning to form throughout the country. One such group is based in New Jersey. The Amateur Computer Group of New Jersey is headed by Sol Libes. This group has regular monthly meetings scheduled in advance with seemingly will-planned topics. All in all, the group appears to be very professional. A newsletter, ACGNJ News (Amateur Computer Group of New Jersey Newsletter) is a monthly publication put out through the efforts of this computer club. The newsletter seems to be very informative with information concerning the members of the group as well as information about their meetings. If you are interested and would like more information about the group, you can contact Mr. Sol Libes (201) 889-2000 (day), 277-2063 (eve), president of the club, or Mr. George Fischer (212) 351-1751.

Another computer group has formed in Texas, known officially as The Computer Hobbyist Group of North Texas. This newly formed group has also taken the bull by the horns and started a very well organized club. The members own various types of equipment and seem willing to share their knowledge as well as time with one and all. The Northern Texas Group also has a monthly newsletter being published. This newsletter offers information concerning meetings of the group, questions and answers on both hardware and software, summaries of various newsletters throughout the country, ideas on where to get help for beginners in the microcomputer field, as well as various other types of information. If you are interested in this group, you can contact Bill Fuller (214) 641-2909 or Neil Ferguson (817) 461-2867. These two gentlemen are the editors of the newsletter. Another contact would be Lannie Walker (817) 244-1013, president of The Computer Hobbyist Group of North Texas.

We also have contacts for other computer groups listed in our new column, "Local Users Groups." If you are a member of such a club and would like to make it known for other users within your area, please send in the group's name (if any) and address as well as contacts for interested people to get ahold of. We'll put this information in our newsletter for 8800 users.

Barbara

LETTER from the PRESIDENT

by H. Edward Roberts,
President, MITS

I have just finisned reviewing all the Computer Notes survey forms that you have sent to us. A number of interesting ideas and products were proposed as well as a number of negative comments. I will address these negative comments in this column.

The number one all-time problem is delivery. We are only too aware of this problem. The reason for delivery problems are generally related to two factors.

The first factor has been delivery of materials from our vendors. We have literally sucked the industry dry of certain parts over the last year; examples are 8212, 8201, 2107, 4030, 4060, TTL parts, etc. We are moving into a mode where our long term production requirements are being anticipated more accurately and hopefully this will circumvent many existing delivery problems.

The other factor has been one of optimism. We have consistently spent more time in development than was planned. About three months ago I issued a directive that we would not announce any new products until they are in production. This won't completely eliminate the problems associated with delivery through the initial surge of orders, but it will greatly help.

I am sure that the readers of this publication have noticed a lack of new product announcements recently. This should not be interpreted as a lack of on-going development. We have a large number of new, and we think, breakthrough products that are close to being put into production, but they will be in production when we announce them. The new I/O cards and PROM cards that are announced in this issue are examples.

Questions have been raised about the 4K memory by a large number of customers. I will answer them here.

1. Question: Why did you go with the dynamic memory instead of the cheaper and simpler static approach?

Answer: The dynamic 4K memory promised to be ultimately the lowest cost memory. A 4K byte dynamic memory requires approximately 32 ma of 5 volt power for the total of 8 RAM, and static memory using 2102 requires approximately 960 ma of 5 volt power. Our internal Altair bus specifications say that no card will use more than .5A of 5 volts, we certainly deviate occasionally from this specification, but we try to stay close. Current Altair 4K cards use typically 530 ma at 5 volts, this current includes the bus drivers refresh, RAM, and address select current. In summary this means that a 4K static card that meets bus drive specifications require close to three times the 5 volt current of dynamic memory. Incidentally, this data is based on the current Intel data sheets and our own internal measurements.

2. Statement: Dynamic memory is slower than static.

Answer: Grossly inaccurate statement--we currently use 4030-2 and 4060-2 which have a worst case access time of 200 nanoseconds, but even the earlier 4K designs use 300 nanosecond memories. Every 2MS the refresh circuit refreshes 64 columns , but this refresh occurs during the start of a cycle and is normally transparent to the processor Even under worst case conditions it would only change system speed by 1.6% which isn't measurable in most applications.

3. Statement: There are problems with the 4K dynamic.

Answer: True to some extend. We have been unable to get an adequate supply of either 4030, 4060, or 2107 memories. This resulted in our switching the design to the Signetic 2604. This part appeared both on the data sheet and in our test to be a direct substitute for TI or Intel parts but, apparently we received a defective lot of chips, which resulted in an excessive failure rate for the 4K card. We have asked in the previous issue of this publication for the return of any 4K cards which contained the Signetic 4K RAMs. Most of these cards have been returned. At the time I am writing this, we are now returning the tested 4K cards to their owners. But this process is a slow one due to the availability of the TI parts. Please bear with us. As a result of these problems with Signetic RAMs a number of changes and revisions have been made to this card to make it even better. A new P.C. layout is underway which eliminates most jumpers. It should be available in approximately 8 weeks. In any case, if you have any problems, let us know. We will do everything possible to resolve them.

Finally, and most important, we feel that our evaluation of the ultimate cost of the dynamic memory has been proven. Therefore, effective November 1, the price on the MITS 4K memory board will be $195.00. Furthermore, all of you guys who have loyally purchased MITS 4K cards in the past have an automatic $50.00 credit for each 4K card you purchased at $264.00 toward the purchase of a new 4K memory card. Put another way, if you paid $264.00 for a MITS 4K card in the past, you may buy a new 4K card for $145.00. I hope this will eliminate any doubts concerning the wisdom of staying with MITS on all future purchases of Altair peripherals.

Another comment that has been made is related to customer service. I think that most of our customers are aware that we are constantly improving the quality and quantity of our service. The problem we have is training people internally to respond to customer requirements, and frankly the number of customers have increased more rapidly than our ability to train. We started the year with two people involved in customer service. We now have over 15 people involved in customer service and are still training. Our goal is that by the end of November you will be able to count on almost instant service, but until then, please bear with us. We are trying!!!

In total, I believe most of our customers agree that MITS has done an enormous amount to revolutionize the computer industry. We have done it by producing a consistantly high quality product at a reasonable price. We have made mistakes in the past and will probably make more in the future, but you can be assured if it happens we will rectify the problem. Ever since we introduced the Altair we have been attacked by a small but persistent group of competitors and armchair experts. I personally have been strongly inclined to do battle, but cooler heads here at MITS have won out. I think it time to answer some of the nonsense questions that have been raised.

1. Comment: MITS uses substandard parts.

Answer: We never have nor ever will in the future. Anyone who tells you this, feel free to tell him that I said he is a liar.

2. Comment: MITS is a big company which is not interested in its customers.

Answer: False, MITS is a small company. We had less than 20 employees when we introduced the Altair and now we have grown to 90 as a result of our Altair customers. We are profit motivated, but that motivation exceeds our interest in our products by only the slightest amount.

3. Comment: MITS should give BASIC to its customers.

Answer: Wrong. We made a $180,000 royalty committment to Micro Soft in order to have BASIC available to our customers. MITS makes essentially no profit on BASIC. It is done as a service to our customers. The BASIC we supply has universally been accepted as nothing short of fantastic and has allowed Altair customers to be literall years ahead of where they would have been without this software. Contrary to some opinions, software developments are expensive and the people who do these developments feel that they shoul be paid for their efforts, I agree. It is irrevelent whether software is

— CONTINUED Page 4 —

LETTER from the President

developed to run on a large IBM computer or the Altair, it costs money. We are selling the BASIC at 1/10 to 1/100 the price large computer companies would get for a similar package, but we are still taking gas. Anyone who would like to argue this point, should feel free to call me at MITS. Anyone who is using a stolen copy of MITS BASIC should identify himself for what he is, a thief.

Recently a number of parasite companies have appeared. These companies are in a tough position. They must attack us for new business but at the same time they are dependent on our product for their survival. This has resulted in a good bit of nonsensical rheotoric and advertising. We are confident that most of our customers see through the nonsense, but a few customers have attacked MITS for the failure of a competitor to deliver or for the failure of a competitor's product to operate properly. We have more than enough of our own problems to worry about without getting involved with competitor's problems. Therefore, I will state again that we stand behind all MITS products and systems. We will do whatever is necessary to make these systems function properly if they haven't been abused by the owner. But if you use non-MITS products, any problems that arise are between you and your vendor.

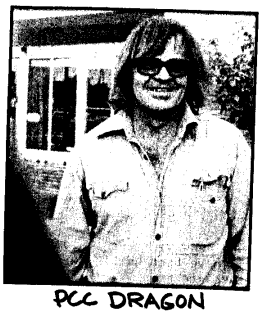
Anyone interested in discussing any of the points I have made in this column or anything else, please write to my attention. I will try to publish as many of these letters as feasible.

One final thought, I would personally like to take this space to thank you for your support both in terms of orders and in terms of suggestions and constructive criticism. I hope you will accept $50 credit on the 4K memory board as a thank you for your past loyalty.

ACROSS THE EDITOR'S DESK

is also a number of classrooms and except for one thing I would hardly recommend the place. That one thing is computers.

When I entered the Lawrence Hall of Science, the first sight I saw was a mass of computer terminals surrounded by junior high age kids who were obviously having the time of their lives (did you know that there are hundreds--perhaps thousands--of young turks who understand BASIC and can program BASIC).



PCC DRAGON

Until you have the experience of watching young people play camaraderie or attacking a match program, you can't really understand the enormous impact computers are going to have on education. Computers don't judge you like teachers -- if you do something wrong you can try again and no one is the wiser.

The advantages of computers in education are such that I am not going to be one of those who backs off and says, "but, of course, computers will never replace teachers." Computers will replace some teachers. Computers will teach subjects and teachers will teach socialization (someone has to watch the playground).

In addition to thousands of bubblegummers swarming over terminals there were some other interesting things happening at the Computer Fair. The classrooms that I refer to were set up as demonstration rooms for computer companies to display their wares (two days at the Computer Fair were reserved for teachers and administrators).

One classroom had five Altair systems up and running, the most interesting of which belonged to a company called Cromenco that had an Altair hooked up to a color television. They call this interface the "TV Dazzler."

TV Dazzler will work in a 32 x 32, 64 x 64, or 128 x 128 matrix with 7 levels of gray scale in black and white or all 7 TV colors in color plus two intensities of brightness. When you turn it on you see a color representation of memory. If you are running BASIC, you can watch it as it writes into the buffers or onto the stack, etc. The program that I saw running in the Dazzler was a demonstration program written by Stephen Dompier of Altair Music fame. What you do is put a pattern on the screen (it has complete cursor control) and type RUN (using a teletype) and the pattern repeats itself over and over again until the screen is full. The result is a beautiful combination of colors that I would characterize as electronic painting.

Les Solomon of Popular Electronics stayed up until 3 am recently playing LIFE on a TV Dazzler and he says the Dazzler will be featured in the February issue of PE.

For those of you who are interested in the TV Dazzler, the address of Cromenco is 26655 Laurel, Los Altos, California 94022. Tell them you heard about the Dazzler in Computer Notes and perhaps they will be persuaded to provide us with an article.

While I was in California, I took the opportunity to tape interviews with Bob Albrecht and Keith Britton of PCC and with Stephen Dompier. They had some interesting things to say. Did you know for instance that "the whole thing" started about 11 years ago on top of the crooked street in San Francisco at Bob Albrecht's weekly combination wine tasting-Greek dancing-computer programming seminars? Next issue I will publish exerpts from these interviews.

cont: Letter from the President

developed to run on a large IBM computer or the Altair, it costs money. We are selling the BASIC at 1/10 to 1/100 the price large computer companies would get for a similar package, but we are still taking gas. Anyone who would like to argue this point, should feel free to call me at MITS. Anyone who is using a stolen copy of MITS BASIC should identify himself for what he is, a thief.

Recently a number of parasite companies have appeared. These companies are in a tough position. They must attack us for new business but at the same time they are dependent on our product for their survival. This has resulted in a good bit of nonsensical rheotoric and advertising. We are confident that most of our customers see through the nonsense, but a few customers have attacked MITS for the failure of a competitor to deliver or for the failure of a competitor's product to operate properly. We have more than enough of our own problems to worry about without getting involved with competitor's problems. Therefore, I will state again that we stand behind all MITS products and systems. We will do whatever is necessary to make these systems function properly if they haven't been abused by the owner. But if you use non-MITS products, any problems that arise are between you and your vendor.

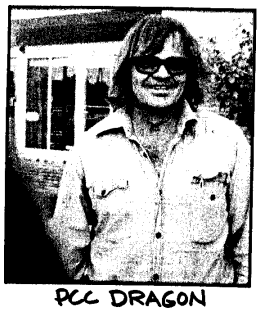
Anyone interested in discussing any of the points I have made in this column or anything else, please write to my attention. I will try to publish as many of these letters as feasible.

One final thought, I would personally like to take this space to thank you for your support both in terms of orders and in terms of suggestions and constructive criticism. I hope you will accept $50 credit on the 4K memory board as a thank you for your past loyalty.

CONT. ACROSS THE EDITOR'S DESK

is also a number of classrooms and except for one thing I would hardly recommend the place. That one thing is computers.

When I entered the Lawrence Hall of Science, the first sight I saw was a mass of computer terminals surrounded by junior high age kids who were obviously having the time of their lives (did you know that there are hundreds--perhaps thousands--of young turks who understand BASIC and can program BASIC).



PCC DRAGON

Until you have the experience of watching young people play camaraderie or attacking a match program, you can't really understand the enormous impact computers are going to have on education. Computers don't judge you like teachers -- if you do something wrong you can try again and no one is the wiser.

The advantages of computers in education are such that I am not going to be one of those who backs off and says, "but, of course, computers will never replace teachers." Computers will replace some teachers. Computers will teach subjects and teachers will teach socialization (someone has to watch the playground).

In addition to thousands of bubblegummers swarming over terminals there were some other interesting things happening at the Computer Fair. The classrooms that I refer to were set up as demonstration rooms for computer companies to display their wares (two days at the Computer Fair were reserved for teachers and administrators).

One classroom had five Altair systems up and running, the most interesting of which belonged to a company called Cromenco that had an Altair hooked up to a color television. They call this interface the "TV Dazzler."

TV Dazzler will work in a 32 x 32, 64 x 64, or 128 x 128 matrix with 7 levels of gray scale in black and white or all 7 TV colors in color plus two intensities of brightness. When you turn it on you see a color representation of memory. If you are running BASIC, you can watch it as it writes into the buffers or onto the stack, etc. The program that I saw running in the Dazzler was a demonstration program written by Stephen Dompier of Altair Music fame. What you do is put a pattern on the screen (it has complete cursor control) and type RUN (using a teletype) and the pattern repeats itself over and over again until the screen is full. The result is a beautiful combination of colors that I would characterize as electronic painting.

Les Solomon of Popular Electronics stayed up until 3 am recently playing LIFE on a TV Dazzler and he says the Dazzler will be featured in the February issue of PE.

For those of you who are interested in the TV Dazzler, the address of Cromenco is 26655 Laurel, Los Altos, California 94022. Tell them you heard about the Dazzler in Computer Notes and perhaps they will be persuaded to provide us with an article.

While I was in California, I took the opportunity to tape interviews with Bob Albrecht and Keith Britton of PCC and with Stephen Dompier. They had some interesting things to say. Did you know for instance that "the whole thing" started about 11 years ago on top of the crooked street in San Francisco at Bob Albrecht's weekly combination wine tasting-Greek dancing-computer programming seminars? Next issue I will publish exerpts from these interviews.

4K Price Cut To $195!

by David Bunnell

Moments before this issue of Computer Notes was slated to go to press, H. Edward Roberts, President of MITS, announced dramatic cuts in the prices of Altair 4K memory boards. Effective November 1, the price of 4K kits will be dropped from $264 to $195. Likewise, the price of assembled 4K memory boards will be dropped from $338 to $275.

Noting that problems with 4K boards have now been solved, Roberts also announced that customers who have purchased 4K boards prior to November 1 will receive a $50 credit for each 4K board purchased. This credit can be applied to the purchase of new Altair 4K boards only.

If you have purchased a 4K board at the old price, you can now purchase a 4K kit for just $145 or an assembled board for $225.

Roberts stated that this decision was made after reading surveys returned by MITS customers, many of whom complained about the prices of MITS Altair memory and I/O prices.

Roberts noted that these price changes might create another large backlog but that MITS will make a determined effort to improve its delivery schedule and he asked that all MITS customers please bear with us. It is now a goal of MITS to have 24 hour delivery and all MITS employees are geared toward achieving this.

MITS is determined to provide Altair customers with the best, low cost memory and I/O modules on the market. We will continue looking at our prices and we will continue developing new products, however, MITS will not use junk parts in its products in order to be competitive with junk kit dealers. MITS uses only the best, quality parts and that policy will remain in effect.

NEW IO's

After announcing the price cuts for Altair 4K boards, Roberts announced that three new Altair I/O boards are now ready for production. These include a Serial Interface (88-2SIO) that can be ordered with one or two ports, a Parallel Interface (88-4PIO) that can have up to four ports, and a PROM memory card.

These boards could have been announced weeks ago but MITS has adopted a new policy of not announcing any products until they are in production stage. It is hoped that this policy will elimate problems created in the past by premature product announcements.

88-2SIO Interface

The 88-2SIO Interface board can be ordered with one port for just $115 in kit form and $144 assembled. The additional port is $24 for kit and $35 assembled. Each port provides the following features:

1) 5 signal/control lines
   a) transmit data
   b) receive data
   c) data carrier detect
   d) clear to send
   e) request to send

This allows for maximum utilization of some of the more sophisticated terminals on the market.

2) All signal/control lines are user-selectable for ±12 volt levels (RS-232), TTL levels (0-5 volts), or 20 milliamp current loop (Teletype™).

3) Software programmable for 9 or 10 bit transmission
   a) 7 data bits + parity bit (odd, even or none) + 1 or 2 stop bits
   b) 8 data bits + 1 or 2 stop bits
   c) 8 data bits + 1 stop bit + parity bit (odd or even)

4) Full 8 bit status register
   a) received data available
   b) transmitter buffer empty
   c) carrier detect
   d) clear to send
   e) framing error
   f) received data overflow
   g) parity error
   h) interrupt request

Allows for greater control and handshaking ability.

5) Transmit and receive interrupts --disable/enable under software.

6) On-board, crystal-controlled clock for any of eight baud rates (with a single jumper): 110, 150, 300, 1200, 1800, 2400, 4800, 9600.

7) Programmable counter can provide other baud rates of 37.5, 75 and 600.

The 88-2SIO with 2 ports can interface 2 serial I/O devices, each running at a different baud rate and each using a different electrical interconnect. That is, the 2 ports can be operating entirely independently of each other (such as an RS-232 CRT terminal running at 9600 baud and a 20 milliamp Teletype running at 110 baud). Five volt power consumption is typically 520 milliamps.

88-4PIO Interface

The 88-4PIO Interface board can be ordered with a single port for just $86 in kit form and $112 assembled. Three additional ports are $30 each in kit form and $39 each assembled. Each port provides the following:

1) Sixteen data lines
   a) Each line can be initialized as an input or output so that a single port can interface a terminal (8 lines in--8 lines out) or 2 input devices (such as paper tape reader and keyboard) or 2 output devices (paper tape punch and printer) or any combination for custom applications.
   b) All data lines are fully TTL compatible and in addition, 8 of the 16 lines can directly drive the base of a transistor switch (1.5v @ 1 milliamp).

2) Four controllable interrupt lines
   a) Enabled/disabled under software control.
   b) Two lines can act as outputs for ready/busy handshake.

3) Two control/status registers--Contains a status bit for each of the four interrupt lines.

4) Removable flat cable connection from board to back panel.

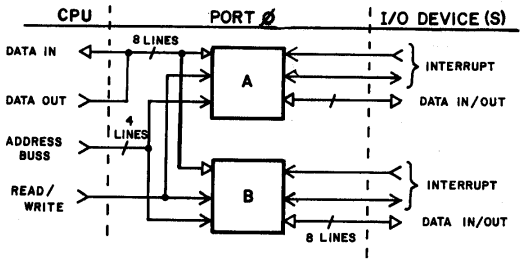
SEE DIAGRAM FOLLOWING PAGE

HARDWARE HINTS

If your CPU board seems to be writing all ones or zeros into the memory, if it gives you the wrong status, or has problems running BASIC, then check IC Q. It should be a Texas Instruments 74123. While other brands may work, we have found that the TI 74123 gives greater reliability in this particular use.

By way of a reminder, all 4K dynamic memories using Signetic RAMs (S 2604) should be returned to the factory for checkout with replacement RAMs. If you have any of our 4K dynamic memory boards that fail to give reliable service, please return them to the MITS Repair Department for checkout.

88-4PIO Interface



SIMPLIFIED BLOCK DIAGRAM (1 port)

Assuming the board is addressed at location Ø, register selection for port Ø is:

ADDRESS	REGISTER
Ø	Section A - control
1	Section A - data
2	Section B - control
3	Section B - data

Port 1 would be addresses 4,5,6,7
Port 2 would be address 8,9,10,11
Port 3 would be addresses 12,13,14,15

An 88-4PIO with 4 ports has 64 data lines (each group of 8 individually selectable) and consumes 500 milliamps at 5 volts - typical.

Prices

Item	Kit	Assembled
88-4MCS	$195	$275
	$50 credit on new orders for each board ordered, i.e. has two 4K's, can order two more at $145 each.	
88-2SIO	$115 single	$144
88-SP	$24 additional port	$35
88-4PIO	$86 single	$112
88-PP	$30 additional ports (up to 4)	$39
88-PMC	$65	$128

CUSTOMER SERVICE NOTE:

If you plan to stop by at our Albuquerque plant to have a MITS product serviced while you wait, please notify us 2 days in advance so we may set aside time for our technicians. We are extremely busy and have quite a backlog at this time, so your cooperation will be appreciated.

PROM Memory Card

1. The 88-PMC PROM memory card provides up to 2K bytes of non-volatile memory for boot loaders and other programs that must be retained if power to the computer is interrupted. The card uses either 1702 or 1702A PROMs which contain 256 8-bit bytes. These PROMs are electrically programmable and erasable (using ultra-violet light) so they can be reprogrammed if necessary. The non-erasable versions (1602 and 1602A) can also be used in the card.
2. The card has a power down option consisting of four drivers that switch the PROMs on and off in pairs to reduce current drain. In the "off" state the PROMs draw about 15% to 20% of the current required in the "on" state. Except for some overlap in switching, only two of the PROMs can be in the "on" state at a time.
3. There are provisions on the card for patching the number of wait states from 0 to 3 in order to accomodate different speed devices.

ENGINEERING NOTE:

For proper operation of a Teletype with the 88-SIOC interface board, the Teletype must be internally wired for 20mA, full duplex operation. MITS is currently selling Teletype model #3320-3JE. Instructions for changing to 20mA, full duplex operation may be found in the wiring diagram #9336WD-B2A supplied with the Teletype (this model comes from the factory wired for 60mA, half duplex). Also, be sure that R10 on the 88-SIOC is 220 ohm, not 390 ohm. This resistor determines the 20mA loop current operating the printer part of the Teletype.

Using the VLCT

The VLCT is a very useful octal I/O device. It consists of a 10-key keyboard (0-7, Ready, and Clear), an eight LED input register display and a three digit octal output display. The VLCT interfaces to the Altair through the 88-PIO using TTL logic levels. Data is entered into the Altair by use of a short input program, such as the one listed below.

VLCT INPUT PROGRAM

Stores data in memory starting at address specified by LXI instruction (must be different from input program.)

```
      0    041    LXI    Load H & L with starting address
      1    XXX           Least significant address byte
      2    XXX           Most significant address byte
IST:  3    333    IN
      4    000
      5    346    ANI    Test for input latch loaded
      6    002           (after 3 keystrokes)
      7    312    JZ
     10    003
      1    000           Jump to IST
      2    333    IN
      3    001
      4    167    MOV    (Mem H,L) ← (A)
OST:  5    333    IN
      6    000
      7    346    ANI    Test for Ready key
     20    001           pressed
      1    312    JZ
      2    015           Jump to OST
      3    000
      4    176    MOV    (A) ← (Mem H,L)
      5    323    OUT
      6    001
      7    043    INX    (H,L) + 1
     30    303    JMP
      1    003           Jump to IST
      2    000
```

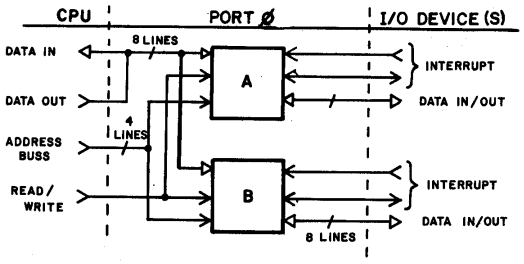
NOTE: For 88-PIO address 0&1

After 3 keystrokes on the VLCT, data is automatically transferred to the input latch, where software stores it in memory. Pressing the Ready key causes the software to echo back the data to the octal display on the VLCT.

By changing the test byte after the first "ANI" you can use the Ready key to trigger storage of data to prevent automatic entry of an incorrect code.

This program stores octal data sequentially in memory starting at the address specified by LXI. It should be used for writing machine language programs; and, for long programs, it is faster and easier than loading in through the front panel switches.

88-4PIO Interface



SIMPLIFIED BLOCK DIAGRAM (1 port)

Assuming the board is addressed at location Ø, register selection for port Ø is:

ADDRESS	REGISTER
Ø	Section A - control
1	Section A - data
2	Section B - control
3	Section B - data

Port 1 would be addresses 4,5,6,7
Port 2 would be address 8,9,10,11
Port 3 would be addresses 12,13,14,15

An 88-4PIO with 4 ports has 64 data lines (each group of 8 individually selectable) and consumes 500 milliamps at 5 volts - typical.

Prices

Item	Kit	Assembled
88-4MCS	$195	$275
	$50 credit on new orders for each board ordered, i.e. has two 4K's, can order two more at $145 each.	
88-2SIO	$115 single	$144
88-SP	$24 additional port	$35
88-4PIO	$86 single	$112
88-PP	$30 additional ports (up to 4)	$39
88-PMC	$65	$128

CUSTOMER SERVICE NOTE:

If you plan to stop by at our Albuquerque plant to have a MITS product serviced while you wait, please notify us 2 days in advance so we may set aside time for our technicians. We are extremely busy and have quite a backlog at this time, so your cooperation will be appreciated.

PROM Memory Card

1. The 88-PMC PROM memory card provides up to 2K bytes of non-volatile memory for boot loaders and other programs that must be retained if power to the computer is interrupted. The card uses either 1702 or 1702A PROMs which contain 256 8-bit bytes. These PROMs are electrically programmable and erasable (using ultra-violet light) so they can be reprogrammed if necessary. The non-erasable versions (1602 and 1602A) can also be used in the card.
2. The card has a power down option consisting of four drivers that switch the PROMs on and off in pairs to reduce current drain. In the "off" state the PROMs draw about 15% to 20% of the current required in the "on" state. Except for some overlap in switching, only two of the PROMs can be in the "on" state at a time.
3. There are provisions on the card for patching the number of wait states from 0 to 3 in order to accomodate different speed devices.

ENGINEERING NOTE:

For proper operation of a Teletype with the 88-SIOC interface board, the Teletype must be internally wired for 20mA, full duplex operation. MITS is currently selling Teletype model #3320-3JE. Instructions for changing to 20mA, full duplex operation may be found in the wiring diagram #9336WD-B2A supplied with the Teletype (this model comes from the factory wired for 60mA, half duplex). Also, be sure that R10 on the 88-SIOC is 220 ohm, not 390 ohm. This resistor determines the 20mA loop current operating the printer part of the Teletype.

Using the VLCT

The VLCT is a very useful octal I/O device. It consists of a 10-key keyboard (0-7, Ready, and Clear), an eight LED input register display and a three digit octal output display. The VLCT interfaces to the Altair through the 88-PIO using TTL logic levels. Data is entered into the Altair by use of a short input program, such as the one listed below.

VLCT INPUT PROGRAM

Stores data in memory starting at address specified by LXI instruction (must be different from input program.)

```
      0    041    LXI     Load H & L with starting address
      1    XXX            Least significant address byte
      2    XXX            Most significant address byte
IST:  3    333    IN
      4    000
      5    346    ANI     Test for input latch loaded
      6    002            (after 3 keystrokes)
      7    312    JZ
     10    003
      1    000            Jump to IST
      2    333    IN
      3    001
      4    167    MOV     (Mem H,L) ← (A)
OST:  5    333    IN
      6    000
      7    346    ANI     Test for Ready key
     20    001            pressed
      1    312    JZ
      2    015            Jump to OST
      3    000
      4    176    MOV     (A) ← (Mem H,L)
      5    323    OUT
      6    001
      7    043    INX     (H,L) + 1
     30    303    JMP
      1    003            Jump to IST
      2    000
```

NOTE: For 88-PIO address 0&1

After 3 keystrokes on the VLCT, data is automatically transferred to the input latch, where software stores it in memory. Pressing the Ready key causes the software to echo back the data to the octal display on the VLCT.

By changing the test byte after the first "ANI" you can use the Ready key to trigger storage of data to prevent automatic entry of an incorrect code.

This program stores octal data sequentially in memory starting at the address specified by LXI. It should be used for writing machine language programs; and, for long programs, it is faster and easier than loading in through the front panel switches.

Continued from Page One

ALTAIR 680 New 6800-Based Computer

The 6800 has seven different addressing modes, with the particular mode being a function of both the type of instruction and the actual coding within the instruction. The seven modes include the following:

Accumulator Addressing--These are one byte instructions, specifying either of the two accumulators.

Immediate Addressing--These are two or three byte instructions, with the MPU addressing the location given in the 2nd or 2nd and 3rd bytes when the immediate instruction is fetched.

Direct Addressing--These are two byte instructions which allow the user to directly address the lowest 256 bytes of memory in the machine.

Extended Addressing--These are three byte instructions, the second two bytes referring to an absolute address in memory for the operation.

Indexed Addressing--These are two byte instructions, the second byte being added to the 16-bit index register to give the address of the operand.

Implied Addressing--These are one byte instructions and the instruction itself gives the address.

Relative Addressing--These are two byte instructions where the second byte is added to the lower 8 bits of the program counter plus two. Carry or borrow are added to the higher 8 bits. This allows the user to address memory +129 to -125 bytes from the location of the present instruction.

These various addressing modes may take a bit of getting used to, but once understood they allow for some very fast programs to be written.

The 6800 MPU contains three 16-bit registers and three 8-bit registers. The program counter is a two byte register which keeps track of the current address of the program. The stack pointer is also a two byte register which contains the next address in an external, variable length push-down/pop-up stack. The index register is a two byte register used to store data or a memory address for indexed addressing operations. There are two single byte accumulators used for holding operands and results from the arithmetic logic unit (ALU). The 8-bit condition code register indicates the results of an ALU operation. In this register there are two unused bits, kept at a logic one. The remaining six bits are used to indicate the status of the following: carry, overflow, zero, negative, interrupt

There are several timing and control signals required to operate the MPU. Two clock inputs are required, phase 1 and phase 2. These must be nonoverlapping and run at the Vcc voltage level. In the 680 the clock is a 2-MHz crystal-controlled oscillator with logic to provide a 500-KHz two phase clock. Although the MPU is capable of using up to a 1-MHz clock, by reducing the speed to 500-KHz the overall design of the 680 was kept much simpler and greatly reduced the cost. Sixteen active high address outputs are used to specify the sections of memory or I/O to be used. These can drive up to one standard TTL load and 130pf. There are also eight bi-directional data lines with the same drive capability as the address lines. The Halt signal is an active low input which ceases activity in the computer. The Read/Write signal in the high state indicates that the MPU is in a read condition, in the low state it indicates that the MPU is in a write condition. The Valid Memory Address signal tells external devices that the MPU has a valid address on the memory bus. The Data Bus Enable signal is the input which enables the bus drivers. The Bus Available signal indicates that the machine has stopped and that the address bus is available. Reset is used to reset and start the MPU from a power-off condition. The Interrupt Request signal, when low, tells the MPU to start an interrupt sequence. This can occur only if the interrupt mask bit in the condition code register is low. The Nonmaskable Interrupt signal is essentially the same as the Interrupt Request signal except that it is not dependent on the condition code register.

Memory

The main printed circuit board on the 680 contains the basic memory for the unit also. This includes 1024 bytes of Random Access Memory and provisions for another 1024 bytes of ROM or PROM. The RAMs being used are the 2102 type 1024 x 1-bit RAMs. ROMs are custom ordered, and are usually very expensive. The PROMs being used are 1702 type, ultra-violet erasable PROMs. The PROMs are 256 x 8-bit units.

There is additional memory for the 680 on the drawing board at this time which may add up to 12K bytes more storage to the unit.

I/O Port

Also on the main printed circuit board is a built-in I/O port and the appropriate interface circuitry. This port may be configured as either an RS232 level port or either a 20ma or 60ma current loop TTY level port. This means it can be easily interfaced to almost any terminal or teletype. The entire design of the 680 is greatly simplified due to the 6800's I/O structure. The MPU uses addresses to refer to I/O devices as well as memory, rather than have special I/O instructions and a separate I/O bus. This also gives it the ability to operate a considerably large number of separate I/O devices. MITS also has additional I/O interfaces on the drawing board now; although this and the additional memory boards will be greatly influenced in their development by customer response.

Control and Indication

On the fully user-programmable version of the 680, the front panel assembly contains a RUN/HALT switch with an LED indicator for each switch position. There is a RESET switch with no indicator, and another indicator for the AC power switch which is located on the back panel of the unit. The switches for the 16 address lines and 8 data lines, and their associated indicator lights, are also located on the front panel assembly. There is also a DEPOSIT switch.

The DEPOSIT, RESET, DATA and ADDRESS switches are enabled only when the RUN/HALT switch is in the HALT position. To view the data in a particular memory address, the RUN/HALT switch must first be in the HALT position and then the ADDRESS switches may be set to the required address. The data located at that particular address will then appear on the DATA LED indicators above the DATA switches.

To place data, or to change it, in a desired location; once the correct address has been set on the address switches, the appropriate data should be entered on the DATA switches and then the DEPOSIT switch activated. Since the address bus is already connected to the switches by being in the HALT state, a write pulse causes the data to be written into the selected RAM address.

When the RESET switch is activated, the MPU itself resets. This initiates a restart sequence, pulling the address bus to its high state and causing hard-wired data on the board to be used as the restart address.

On the dedicated program versions of the 680, most of these functions are taken care of by ROM or PROM. The only controls available to the user are the AC power and RESET switches.

continued on page 9

USING SERIAL I/O BOARDS

by Tom Durston

The MITS technical staff has been receiving many questions on the writing of software to handle the 88-SIO boards. Most of the confusion has been due to a lack of explanation of the fundamental concept of the Altair I/O structure. We hope the following will help answer users' questions on 88-SIO software.

Inputting data from an external I/O device: The input instruction is a two byte (1 byte = 8 bits) instruction. The first byte (333_8) is the code telling the CPU to input data from an external device (TTY, Comter, CRT terminal, etc.) and put it in the accumulator. The second byte (XXX) is the address of the I/O board connected to the desired external device. The address of the I/O board is determined by the user by seven hardwire jumpers (I1 - I7) on the I/O board. The list of addresses and jumpers is in the back of the I/O board assembly manual. After execution of this two byte input instruction, the input data present at the I/O board is transferred to the accumulator in the CPU. There are 256 I/O addresses, 000_8 through 377_8; the even numbered ones being used for status and control of the I/O board and the odd numbered ones being used for data transfer. This gives the user a maximum possible number of 127 external I/O devices; the 128th I/O device is the sense switch input (address 377).

To input valid received data from an 88-SIO board, the status channel information must be inputted and bit DØ tested. For Rev 1 88-SIO boards, DØ = Ø indicates that a new character has been received from an external device. When status bit DØ has been found to equal Ø, then the data channel may be inputted. Status bit DØ is reset when an input to the data channel is done. From this point the user may do anything desired with the data. A typical input program using address 0&1 would look like this:

```
Byte 1   000   333   IN     Input
     2     1   000          Stat address
           2   017   RRC    Test DØ, rotate Acc right
Byte 1     3   332   JC     Jump if carry (DØ = 1)
     2     4   000
     3     5   000
Byte 1     6   333   IN     Input
     2     7   001          Data address
Byte 1    10   062   STA    Store char in memory loc 000 040
     2    11   040
     3    12   000
Byte 1    13   303   JMP    Jump to beginning
     2    14   000
     3    15   000
```

Outputting data to an external I/O device: The output instruction is a two byte instruction. The first byte (323) is the code telling the CPU to output the data in the accumulator to an external I/O device. The second byte (XXX) is the address of the I/O board connected to the desired external device. When outputting to an 88-SIO board on the even numbered address line (control channel), the interrupt enable circuit may be enabled or disabled by bits DØ and D1 (interrupt is explained later in this article). When outputting to an 88-SIO on the odd numbered address line, parallel data is transferred from the accumulator to the 88-SIO and is transmitted serially to the external device.

When outputting data to an external device, a test of status bit D7 (Rev 1 88-SIO board) must be done to see if the 88-SIO board is ready to transmit a character. When status bit D7 = Ø, indicating the transmitter buffer is empty, then data to be transmitted is placed in the accumulator and outputted to the 88-SIO board on its data channel. A typical output program using address 0&1 would look like this:

```
Byte 1   000   333   IN     Input
Byte 2     1   000          Status Channel
           2   007   RLC    Test D7, rotate Acc left
Byte 1     3   332   JC     Jump if carry (D7 = 1)
     2     4   000
     3     5   000
Byte 1     6   333   IN     Input
     2     7   377          from sense switches
Byte 1    10   323   OUT    Output
     2    11   001          Data Channel
Byte 1    12   303   JMP    Jump to beginning
     2    13   000
     3    14   000
```

This program continuously transmits data from the sense switches to the external device.

continued on page 9

HARDWARE

1. JUST A REMINDER

Is your Altair in a RUT? Does it just sit there, unresponsive and illogical? Well, you may have pulled one of your boards out of the bus with the power on. It tends to render inoperative ICs R, U, and T on the display and control board. SOLUTION: Replace R, U, and T and remember to turn power off before removing PC boards from the bus.

2. DIFFERENCES BETWEEN STATIC AND DYNAMIC MEMORY

A static memory cell consists of a flip-flop which retains its state unless changed by writing a new bit or power going off.

A dynamic memory cell consists of a capacitor with transistors used to sense, maintain, or change its charge. Dynamic memories require a refresh cycle to maintain the charges in the memory cells.

For the MITS 1K static board, 88-MCS, a wait state of 1 microsecond is required due to the access time of the RAM. This means that every step of a program takes one microsecond in addition to the instruction cycle time.

For the MITS 4K dynamic board, 88-4MCD, no wait state is required, but every 32 microseconds, a refresh cycle of one microsecond is done. During the refresh cycle, no reading or writing of data is allowed.

One advantage of dynamic over static memory is that bit for bit, it takes less power.

An advantage of static over dynamic memory is that with faster access times becoming available, they are more practical due to less control circuitry.

8i/O CONT...

By putting together the input program and output program we can create an "echo" program so that a character transmitted to the Altair will be immediately received back.

```
Byte 1   000   333   IN     Input
     2     1   000          Status Channel
           2   017   RRC    Rotate acc right
Byte 1     3   332   JC     Jump if DØ = 1
     2     4   000          (no data)
     3     5   000
Byte 1     6   333   IN     Input
     2     7   001          Data Channel
Byte 1    10   062   STA    Store data
     2    11   040
     3    12   000
Byte 1    13   333   IN     Input
     2    14   000          Status Channel
          15   007   RLC    Rotate acc left
Byte 1    16   332   JC     Jump if D7 = 1
     2    17   012          (transmit busy)
     3    20   000
Byte 1    21   072   LDA    Load acc with
     2    22   040          stored data
     3    23   000
Byte 1    24   323   OUT    Output
     2    25   001          Data Channel
Byte 1    26   303   JMP    Jump back to
     2    27   000          beginning
     3    30   000
```

OOPS

Interfacing the 88-SIO with the external device: First you must determine what interface your terminal requires. The three standard types available from MITS are EIA-RS232 (88-SIOA), TTY - 20mA loop (88-SIOC), and TTL logic (88-SIOB).

Interrupt: Using the Interrupt feature of the CPU allows the Altair to be performing a task other than monitoring the I/O ports for data transfers. To use the interrupt feature, the stack address should be specified, interrupts enabled in the CPU, interrupts enabled on the I/O board, and a subroutine written at the proper location to handle the I/O of data. Using single level interrupt (vectored interrupt hardware will be available by the end of the year), all interrupts will cause the CPU to jump to location 070 where the I/O subroutine should be located. Single level interrupt is utilized by connecting the desired interrupt mode pad ("IN", "OUT", "BH") to the "INT" pad on the 88-SIO board. A short input program that uses single level interrupt would look like this (for I/O board address 061):

```
Byte 1   000   061   LXI SP  Set stack pointer
     2     1   200           to 000 200
     3     2   000
           3   373   EI      Enable interrupts
Byte 1     4   076   MVI     Move 001 into
     2     5   001           accumulator
Byte 1     6   323   OUT     Output to I/O
     2     7   000           board control channel
Byte 1    10   303   JMP     Jump to self and
     2    11   011           wait for interrupt
     3    12   000

Byte 1   070   333   IN      Input from
     2    71   001           Data Channel
Byte 1    72   062   STA     Store data in
     2    73   040           loc 000 040
     3    74   000
          75   373   EI      Enable interrupts
          76   311   RET     Return to "jump to
                             self"
```

The EIA-RS232 interface on the 88-SIOA board offers only transmitted data, received data, and signal ground (circuits BA, BB and AB). If your external device requires other signals, they will have to be strapped to the proper DC voltages for proper operation. EIA interface levels are: Logic Ø: +5 to +25 volts, Logic 1: -5 to -25 volts.

The TTY - 20mA loop interface is used for Teletypes or I/O devices utilizing that interface level. The 88-SIOC levels are: Logic Ø--open circuit, no current flow; Logic 1--closed circuit, 20mA current.

The TTL logic interface uses 7400 type logic levels. The interface level for the 88-SIOB are: Logic Ø: +.4 volts, Logic 1: +2.2 volts.

Selecting the correct baud rate for your I/O device will depend on its requirements. For instance Teletypes operate at 110 baud only, the Comter operates at 110 or 300 baud, most CRT terminals operate between 110 baud and 9600 baud. (Baud = bits per second)

For Teletype, use 110 buad, 2 stop bits, and 8 data bits. For baud rates of 300 and up, use one stop bit and 8 data bits. Note that above 2400 baud the output pulse of IC "0" must be between 2.0 and 2.5 microseconds for correct operation. Varying R25 (7.5K) will obtain proper timing.

If you wish to check parity, use 7 data bits and either odd or even parity, depending on your requirements.

Selecting the address of the I/O board is up to the user with the following exceptions (used in the BASIC language software):

```
Address 0/1 -- I/O terminal
        2/3 -- Line Printer
        4/5 -- Alternate terminal
        6/7 -- ACR
     10/11/12 -- Altair Disk
        376 -- VI, RTC
        377 -- Sense Switch
```

Altair 680 —

cont. from page 7

Power Supply

The 5-volt supply to the computer is supplied from the power transformer through a conventional bridge rectifier and filter capacitors and voltage regulator IC. A 32-volt winding on the transformer is used to generate the unregulaced ±16 volts required for a TTY interface, and a -16 volt line is fed to four zener diode regulated outputs to provide four -9 volt lines for the PROMs.

The transformer itself, along with the power switch, are located on the computer's back panel. There are also provisions for installing a cooling fan when necessary.

MITS has decided to await customer response to determine the course of further 680 development in both the areas of software and hardware.

Although it's not quite as powerful as the ALTAIR 8800, the ALTAIR 680 is mighty close and at less than half the price for a similar system.

— — — —

Lowest Price in the World!

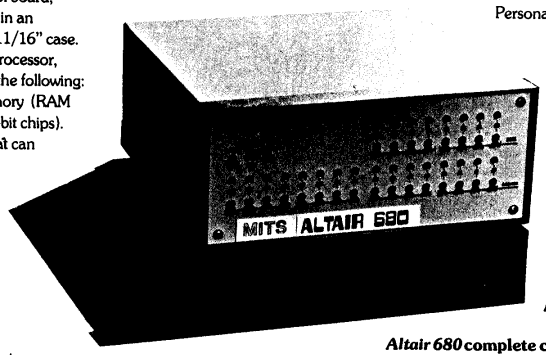
In January of 1975, MITS stunned the computer world with the announcement of the *Altair 8800* Computer that sells for $439 in kit form.

Today MITS is announcing the Altair 680.

The Altair 680, built around the revolutionary new 6800 microprocessor chip, is the lowest priced complete computer on the market. Until December 31, 1975, this computer will be sold in kit form for the amazing introductory price of $293! **(A savings of $52!)**

The Altair 680 comes with power supply, front panel control board, and CPU board inclosed in an 11" wide x 11" deep x 4 11/16" case. In addition to the 6800 processor, the CPU board contains the following:

1. 1024 words of memory (RAM 2102 type 1024 x 1-bit chips).
2. Built-in Interface that can be configured for RS232 or 20 mA Teletype loop or 60 mA Teletype.
3. Provisions for 1024 words of ROM or PROM.



The Altair 680 can be programmed from the front panel switches or it can be connected to a computer terminal (RS232) or a Teletype such as an ASR-33 or surplus five-level Baudott Teletype (under $100).

The Altair 680 can be utilized for many home, commercial or industrial applications or it can be used as a development system for *Altair 680* CPU boards. With a cycle time of 4 microseconds, 16-bit addressing, and the capability of directly addressing 65,000 words of memory and a virtually unlimited number of I/O devices, the *Altair 680* is a very versatile computer!

Altair 680 Software

Software for the *Altair 680* includes a monitor on PROM, assembler, debug, and editor. This software is available to *Altair 680* owners at a nominal cost.

Future software development will be influenced by customer demand and *may include BASIC on ROM.* MITS will sponsor lucrative software contests to encourage the rapid growth of the *Altair 680* software library. Programs in this library will be made available to all *Altair 680* owners at the cost of printing and mailing.

Contact factory for updated information and prices.

Altair Users Group

All *Altair 680* purchasers will receive a free one year membership to the Altair Users Group. This group is the largest of its kind in the world and includes thousands of *Altair 8800* and *680* users.

Members of the Altair Users Group are kept abreast of Altair developments through the monthly publication, ***Computer Notes.***

Altair 680 Documentation

The *Altair 680* kit comes with complete documentation including assembly manual, assembly hints manual, operation manual, and theory manual. Assembled units come with operation and theory manuals. Turnkey model and CPU boards also include documentation.

NOTE: ***Altair 680 manuals can be purchased separately. See back page of this catalog for prices.***

Delivery

Personal checks take 2-3 weeks to process while money orders and credit card purchases can be processed in 1-3 days. Delivery should be 30-60 days but this can vary according to order backlog. All orders are handled on a first come, first served basis.

Altair 680 Prices

***Altair 680* complete computer kit $293**
($345 after December 31, 1975)

***Altair 680* assembled and tested $420**

***Altair 680T* turnkey model (complete *Altair 680* except front panel control board) Kit Only $240**
($280 after December 31, 1975)

***Altair 680* CPU board (including pc board, 6800 microprocessor chip, 1024 word memory, 3 way interface and all remaining components except power supply) $180**
($195 after December 31, 1975)

***Altair 680* CPU board assembled and tested $275**

Option I/O socket kit (required when interfacing 680 to external devices) ..... $ 29

Option cooling fan (required when expanding 680 internally) ..... $ 16
($22 after December 31, 1975)

Option cooling fan installed ..... $ 26

PROM kit (256 x 8-bit ultraviolet, erasable 1702 devices) ..... $ 42

MITS
"Creative Electronics"

Prices, delivery and specifications subject to change.

COMPUTER POETRY

GRADED INTENSITIES

DEBRIS,
AND NUMB TRANCES
WITH CALM CREATION,
YOUR NUMBNESS ASTONISHES
IN DREAM BALLOONS.
SEDUCED IS SLEEP
WHICH HOVERS
IN TRANCEMOTIONLESS MOONLIGHT.

THE INVERSION WISHES OF DRAGONFLIES
ARE MADE
OF FLASHING GLASS
LURING IN THE FREEDOM
OF SUBORDINATION,
LIKE THEIR SILENT SPIRITS.

THE ROMANCE OF WINGS
IS THE THINNESS OF STREAMS.
THEIR SUPPORTING COMPLEXITY
IS GRACEFUL LIKE WRINKLES
TURNING A FLAT EYE.
WINGS HOVER
LIKE FOG AND GARGOYLE,
HOVERING WITH THE FLAKE
OF COMPLEXITY BRANCHED.

A CAUSE OF OCEANS,
MIGHTY AS RELIABLE ROPES,
SCREAMS IN WILDNESS
WHILE A DETAIL OF TREES
SWAYS, LIKE A COBWEB
BEING ATTACHED
BY EXCESSIVE MASKS.

TO LINK WITHOUT INVULNERABILITY
TO GLOW, IN FUNDAMENTAL COMPLETION
WITHIN A CONCENTRIC EDGE
AS STRENGTH RELOCATES ITS HARMONY
A RAINFALL EVADES THE SYSTEM
CONTRIVE, AND FIND THE RAINFALLS OF
MODERATION
COMPLEX CHAINS AND HAPPINESS
GLIDING AT DIRECTIONS
THE WINTER OF ELATION

CREATE TO CHEER
SPLIT TO COMPLETE
HARMONIES STIMULATE THE STIGMAS
INSULTING AT EMBARRASSMENTS
MY ELATIONS LOITER
ELATION-BREASTS
PERFORMING ACID HAMMERS
APPETITE LIKE ZONES

by James Runner

above poems are from
COMPUTER POEMS
gathered by Richard W. Bailey
Potagannissing Press
Drummond Island, Michigan

HARMONY

This is my word to the multitude:
I stopped upon a proud tower
Above the fresh ruffles of the surf,
I, a fond prophet in the place of the finish.
Harmony is not all: It is not flesh nor body.
Harmony is not all: It is not food nor water.
Above the humid ruffles of the shore,
This is my sentence to the creation,
This is my message to the creation.

from *PROGRAM ERATO*
by Louis T. Milic

LOCAL USERS GROUPS

This month we'll be starting a new column that will list names and addresses of local computer clubs and users groups that want to recruit new members in their area. Please send in the name of your club if you would like to have it published.

Canadian Computer Club
G. Pearen
861 11th Street
Brandon, Manitoba
R7A 4L1

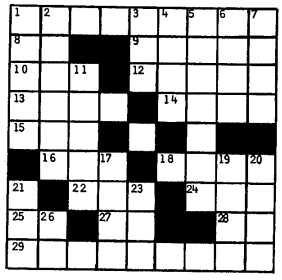
"Universe Unlimited"
User's Group
John E. Kabat
11918 Forrest Ave.
Cleveland, OH 44120
Business 216-781-9400 Ext.55
Home 216-795-2565

29 Palms, California area:
Sgt. Wesley B. Isgrigg
Box 3558, C&E Schools M.C.B.
29 Palms, CA 92278
714-368-9111 Ext.6241
evening Ext.7289
and
Sgt. Stanley E. Herr
13-C Copper Dr. M.C.B.
29 Palms, CA 92278
714-368-3809

(Wesley and Stanley are especially interested in programs to design circuits and in receiving information on Hal Chamberlain's Graphics Display.)

*** For Last Minute Additions to this *** column, see page 18 (lower right)

CROSSWORD PUZZLE



ACROSS
1 circuit between two computer components
8 neon (abbr)
9 to put to rest; to calm
10 in physics; "Outer Tensile Strength" (abbr)
12 corner joint
13 to process a printed circuit board
14 opining between mountains
15 a BASIC instruction
16 in Morse code; a dash
18 programmable read only memory (abbr)
22 allow
24 "___-Sun Bay"; a part of San Francisco Bay
25 citizens band radio (abbr)
27 route (abbr)
28 street (abbr)
29 persons who use computers as a passtime

DOWN
1 "chip" company
2 programming term for loops within loops (adj)
3 random access memory (abbr)
4 "___-flop"
5 computer name (pl)
6 autos
7 you see with them
11 base eight
17 man's name
19 to force out; drive out
20 computer company
21 "The Computer Hobbyist" (abbr)
23 Teletype (abbr)
26 hobo (slang)

contributed by Ron Santore

CONTRIBUTORS WELCOME—SEND ARTICLES TO COMPUTER NOTES/MTHS/6328 Linn

Software Contest Winners Announced

Many excellent additions to the Altair Software Library were accepted during the October software contest. Choosing the winning programs this month was particularly difficult as the overall quality of all the entries was exceptionally good. We finally narrowed it down, however, and Ian Kettleborough came up with first place in the major program category for his Fortran IV cross assembler. A memory dump by Roger Walker won first prize for the subroutine category. For this month's complete listing, see below.

FIRST PLACE MAJOR PROGRAM
#1016752
Author: Ian D. Kettleborough
Length: 44 pages
A Fortran IV cross-assembler, currently running on a Data-General. Several listing options, as well as an object dump can be selected. Conditional assembly provided for.

SECOND PLACE MAJOR PROGRAM
#910751
Author: Ross E. Housholder
Length: 22 pages
A very complete discussion of playing music on the Altair, including translation programs and an improved PLAY program with source instructions.

THIRD PLACE MAJOR PROGRAM
#916751
Author: Gary Tack
Length: 128 bytes
Program to play DICE with a basic Altair.

FIRST PLACE SUBROUTINE
#929751
Author: Roger Walker
Length: 58 bytes
Dumps a block of memory in checksum format.

SECOND PLACE SUBROUTINE
#1010752
Author: Jacques Roth
Length: 19 bytes
Timing subroutine. Delays a specified number of seconds.

#920751
Author: Dale J. Travis
Length: Approx. 80 bytes
Converts an ASCII string in octal to a signed 16-bit number.

#1014751
Author: Jack O. Coats, Jr.
Length: 73 bytes
Loads data from terminal into memory and executes data after the termination character. Currently written to work with a terminal connected through a parallel I/O board.

#102751
Author: Robert Rydel
Length: 82 bytes
"Error Detection Code Generator/Checker." Generates cyclic redundancy check (CRC) to allow for doing error detection.

#106751
Author: Gerard Bilodeau
Length: 25 bytes
A message print subroutine. Allows for embedded carriage return/line feeds and any delimeter character.

#911751
Author: Dale J. Travis
Length: 16 bytes
Subroutine to convert binary to a 3-digit BCD number.

#912751
Author: Gerhald Hansel
Length: 70 lines
A BASIC program that plays roulette.

#1010751
Author: Jacques Roth
Length: 256 bytes
Program to play the "Bagels" game with a minimum Altair. Instructions included.

#109751
Author: William D. Roch
Length: 30 lines
A BASIC program to calculate the number of days between two dates. Takes leap years into consideration.

#1016751
Author: Steve Caldara
Length: 21 lines
A BASIC "Lunar Lander" program.

#912752
Author: Bob Omer
Length: 78 lines
BASIC program that plays the Stick game. Instructions available when the program is run.

#920752
Author: Dale J. Travis
Length: Approx. 120 bytes
Program to convert a 16-bit signed integer to an ASCII string of octal digits.

Software Notes

by Bill Gates

Using the STACK

Every program written for the 8800, large or small, should take advantage of the stack. The stack is a stored list of data which behaves in a last in--first out (LIFO) fashion. That is, PUSH D/POP D doesn't modify [D,E] since the POP removes the same two bytes that were just pushed onto the stack. There are three things to remember in using the stack:

1) Initialize the stack pointer to the highest location in a block of free Read/Write memory. As data is pushed onto the stack the stack pointer is decremented.

2) Make sure that the amount of space set aside for the stack is sufficient to hold all the values (including return addresses) that are stored on the stack at one time. Every time a PUSH or CALL is done the stack pointer is decremented by two, so data pushed onto the stack will be stored in lower and lower locations. Every POP or RETurn increments the stack pointer by two, so the important thing is not how many values are PUSHed on during a program, but how many are on at the same time. Consider:

```
          LXI   SP, STKBOT
LOOPDO:   PUSH  D         ;save [D,E]
          :
          CALL  SUB1
RETADR:   :
          POP   D         ;restore [D,E]
          DCR   B
          JNZ   LOOPDO
          :
          DS 4
STKBOT:
SUB1:     :               ;no stack use
          :
          RET
```

No matter how many times the above loop is executed, [D,E] will always be saved and restored from the same memory locations, since the stack pointer is incremented by 2 by the PUSH, incremented by 2 by the CALL, decremented by 2 by the RET, and decremented by 2 by the POP. Only four bytes of stack space are set aside for this program since only 2 bytes of data (D and E) and 2 bytes of return address are ever stored on the stack at a time. If additional PUSH/POPs or CALLs are done inside the loop, more stack space would have to be set aside. Unless you are using all of memory and need to compactify as much as possible, it is a good idea to allocate a lot more stack space than you think you will ever use.

3) Unless you are being very tricky, never take more data off of the stack than is put on. This means doing more POPs and RETs than CALLs and PUSHs.

continued on page 13

SOFTWARE

ILLUSTRATION OF STACK OPERATION

Consider the example given earlier. Assume that STKBOT = 200 octal.

```
        :
    program data

174 |       |
175 |       |  ;set aside for stack use
176 |       |  ;initial contents irrelevant
177 |       |
200     other data
```

After the LXI SP, STKBOT SP will equal 200.

PUSHs and CALLs always put 2 bytes of data onto the stack as follows:

1) Decrement SP
2) Store the high 8 bits of data being PUSHed in the memory location given by SP.
3) Decrement SP
4) Store the low 8 bits of data being PUSHed into the memory location given by SP.

```
PUSHMA:
```

When the PUSH D is done: SP = 177, D is stored at 177, SP = 176, E is stored at 176.

Say RETADR = 124
When the call is done: SP = 175, high byte of Retadr = 0 and is stored at 175, SP = 174, the low byte of Retadr = 124 and is stored at 174.

So we have:

```
174 | 124 |
175 |  0  |   the return address
176 |  E  |
177 |  D  |   old [D,E]
```

POPs and RETs do the "reverse" operation:

1) Pick up the low 8 bits of data in the memory location given by the SP
2) Increment the SP
3) Pick up the high 8 bits of data in the memory location given by the SP
4) Increment the SP

Exercise: Work out the details of how the RET and POP D in the example work.

Other machines that have stacks vary in the details of implementation. Some machines increment the SP on PUSHs and decrement on POPs (PDP-1Ø). Other machines store before decrementing on PUSHs and increment before fetching on POPs (Altair 680). However, the basic notion of a last in-first out list to store data and return addresses remains the same.

The way subroutines are nested, that is, always returning to the most recent caller, makes the stack very natural for storing return addresses. to store both return addresses and data allows for some tricky programming involving manipulation of return addresses on the stack, of which a few examples were given in earlier "Software Notes." It also causes some trouble however for subroutines that wish to leave results on the stack or fetch arguments from the stack, since the return address gets in the way. The sequence:

```
        MOV   C,M
        INX   H
        MOV   B,M
        INX   H
        PUSH  B
```

was used many, many times in BASIC so it was decided to use one of the RST instructions to perform this operation. It was coded as follows:

```
SUB1C:  XTHL            ;[H,L]=return address
        SHLD  PUSHMA+1  ;modify a JMP
        POP   H         ;get [H,L] back
        MOV   C,M
        INX   H
        MOV   B,M
        INX   H
        PUSH  B
SUB1:
        :
        :
PUSHMA: JMP   *         ;JMP to return point
```

If only a single byte of data needs to be pushed onto the stack, put the data in A, B, D, or H and do a PUSH PSW, B, D, or H respectively, followed by a INX SP. To pop off the single byte of data do a DCX SP, POP A, B, D, or H. This puts garbage into the PSW, C, E, or L respectively. Unless the same PUSH will be used to store a large number of one byte pieces of data on the stack, it is simplest to merely do a PUSH/POP sequence and allow the extraneous byte to be stored.

Sometimes the amount of stack space a program requires will depend on the input to the program, for instance when BASIC evaluates complicated formulas. If this is the case, a check must be done when data is pushed on to make sure the stack isn't "overflowing." If it is, either some sort of recovery procedure must be invoked, or an error message printed. The following subroutine checks to see if the stack is pointing below STKSTP.

```
;saves all registers
CHKSTK: PUSH  H           ;save [H,L]
        LXI   H,-STKSTP   ;won't work for STKSTP=0
        DAD   SP
        POP   H           ;[H,L]
        RC                ;return if still ok
;here on stack overflow
```

If you have a subroutine which is often passed constant arguments such as:

```
        MVI   C,3
        CALL  SUB1
        :
        MVI   C,5
        CALL  SUB1
        :
        CALL  SUB1
        :
SUB1:
```

By manipulating the return address you can save one byte per call as follows:

```
        CALL  SUB1C
        DB    3     ;put constant in
        :           ;return location
        CALL  SUB1C
        DB    5
        :
        CALL  SUB1C
        DB    7
        :
SUB1C:  XTHL        ;[H,L]=return address
        MOV   C,M   ;fetch the constant
        INX   H     ;update return address
        XTHL        ;restore the return
        :           ;address and [H,L]
        :
```

This is not a useful technique in most cases, but it does give a good example of XTHL.

Next month, I'll do the long-awaited explanation of the boot and multi-precision arithmetic.

●●●●CONTRIBUTORS●●●●●●●●●

Contributors to this month's issue of Computer Notes includes:

H. Edward Roberts, President of MITS, Inc.

David Bunnell, editor and President of The Agency, Inc.

James B. Vice, Head Technical Writer, MITS, Inc.

Bill Gates, Software Specialist, MITS, Inc.

Dick Heiser, President of Arrowhead Computer Co.

Barbara Simms, Altair Customer Service

Keith Britton and Robert Mullen, People's Computer Company

Thomas W. Durston, Program Manager, Engineering, MITS, Inc.

Patrick N. Godding, Program Manager, Engineering, MITS, Inc.

Andrea Lewis, Technical Writer, Assistant Editor Computer Notes

Chris Ryland, Byte Magazine

Ron Santore, Altair User

James Runner, poet

Louis T. Milic, poet

ILLUSTRATION OF STACK OPERATION

Consider the example given earlier. Assume that STKBOT = 200 octal.

```
        :
    program data

174 |      |
175 |      |  ;set aside for stack use
176 |      |  ;initial contents irrelevant
177 |      |
200     other data
```

After the LXI SP, STKBOT SP will equal 200.

PUSHs and CALLs always put 2 bytes of data onto the stack as follows:

1) Decrement SP
2) Store the high 8 bits of data being PUSHed in the memory location given by SP.
3) Decrement SP
4) Store the low 8 bits of data being PUSHed into the memory location given by SP.

```
PUSHMA:
```

When the PUSH D is done: SP = 177, D is stored at 177, SP = 176, E is stored at 176.

Say RETADR = 124
When the call is done: SP = 175, high byte of Retadr = 0 and is stored at 175, SP = 174, the low byte of Retadr = 124 and is stored at 174.

So we have:

```
174 | 124 |
175 |  0  |   the return address
176 |  E  |
177 |  D  |   old [D,E]
```

POPs and RETs do the "reverse" operation:

1) Pick up the low 8 bits of data in the memory location given by the SP
2) Increment the SP
3) Pick up the high 8 bits of data in the memory location given by the SP
4) Increment the SP

Exercise: Work out the details of how the RET and POP D in the example work.

Other machines that have stacks vary in the details of implementation. Some machines increment the SP on PUSHs and decrement on POPs (PDP-10). Other machines store before decrementing on PUSHs and increment before fetching on POPs (Altair 680). However, the basic notion of a last in-first out list to store data and return addresses remains the same.

The way subroutines are nested, that is, always returning to the most recent caller, makes the stack very natural for storing return addresses.

The fact that the stack is used to store both return addresses and data allows for some tricky programming involving manipulation of return addresses on the stack, of which a few examples were given in earlier "Software Notes." It also causes some trouble however for subroutines that wish to leave results on the stack or fetch arguments from the stack, since the return address gets in the way. The sequence:

```
        MOV   C,M
        INX   H
        MOV   B,M
        INX   H
        PUSH  B
```

was used many, many times in BASIC so it was decided to use one of the RST instructions to perform this operation. It was coded as follows:

```
XTHL            ;[H,L]=return address
SHLD  PUSHMA+1  ;modify a JMP
POP   H         ;get [H,L] back
MOV   C,M
INX   H
MOV   B,M
INX   H
PUSH  B
JMP   *         ;JMP to return point
```

If only a single byte of data needs to be pushed onto the stack, put the data in A, B, D, or H and do a PUSH PSW, B, D, or H respectively, followed by a INX SP. To pop off the single byte of data do a DCX SP, POP A, B, D, or H. This puts garbage into the PSW, C, E, or L respectively. Unless the same PUSH will be used to store a large number of one byte pieces of data on the stack, it is simplest to merely do a PUSH/POP sequence and allow the extraneous byte to be stored.

Sometimes the amount of stack space a program requires will depend on the input to the program, for instance when BASIC evaluates complicated formulas. If this is the case, a check must be done when data is pushed on to make sure the stack isn't "overflowing." If it is, either some sort of recovery procedure must be invoked, or an error message printed. The following subroutine checks to see if the stack is pointing below STKSTP.

```
;saves all registers
CHKSTK: PUSH  H           ;save [H,L]
        LXI   H,-STKSTP   ;won't work for STKSTP=0
        DAD   SP
        POP   H           ;[H,L]
        RC                ;return if still ok
;here on stack overflow
```

If you have a subroutine which is often passed constant arguments such as:

```
        MVI   C,3
        CALL  SUB1
         :
        MVI   C,5
        CALL  SUB1
         :
        MVI   C,7
        CALL  SUB1
         :
SUB1:
```

By manipulating the return address you can save one byte per call as follows:

```
        CALL  SUB1C
        DB    3     ;put constant in
         :          ;return location
        CALL  SUB1C
        DB    5
         :
        CALL  SUB1C
        DB    7
         :
SUB1C:  XTHL        ;[H,L]=return address
        MOV   C,M   ;fetch the constant
        INX   H     ;update return address
        XTHL        ;restore the return
SUB1:    :          ;address and [H,L]
         :
```

This is not a useful technique in most cases, but it does give a good example of XTHL.

Next month, I'll do the long-awaited explanation of the boot and multi-precision arithmetic.

●●●●CONTRIBUTORS●●●●●●●●●

Contributors to this month's issue of Computer Notes includes:

H. Edward Roberts, President of MITS, Inc.

David Bunnell, editor and President of The Agency, Inc.

James B. Vice, Head Technical Writer, MITS, Inc.

Bill Gates, Software Specialist, MITS, Inc.

Dick Heiser, President of Arrowhead Computer Co.

Barbara Simms, Altair Customer Service

Keith Britton and Robert Mullen, People's Computer Company

Thomas W. Durston, Program Manager, Engineering, MITS, Inc.

Patrick N. Godding, Program Manager, Engineering, MITS, Inc.

Andrea Lewis, Technical Writer, Assistant Editor Computer Notes

Chris Ryland, Byte Magazine

Ron Santore, Altair User

James Runner, poet

ALTAIR BASIC

CLAIM: Not just anybody's BASIC
FACT: Not just anybody's BASIC

BY: KEITH BRITTON
ROBERT MULLEN

MITS does a lot of expensive advertising and their machine, while it has proved basically sound, has a number of problems requiring fixes - mostly evidence of hurried development. This has led to adverse comment and skepticism regarding their BASIC. MITS has done a lot wrong regarding the 8800 - but they have done a great deal that is right, too.... and received little credit for it. The Altair has proved fundamentally sound. That is to say the faults, though irritating, have cheap fixes that are effective. (People have designed *and sold* computers that did not work and could *not* be fixed.) But not only was it what was needed at the time, it was designed to be what was needed for the future. This shows commendable foresight on the part of Ed Roberts & crew, that is obvious. Less obvious, and something for which they deserve credit which they have not received, is the fact that they were willing to pay a considerable price to provide the hobbyist with things which he did not yet know he needed. The bus, for instance, has 100 lines. This cost money directly, more connections and more expensive sockets, and indirectly in that design freedom encourages competitors. *But this philosophy has provided the hobbyist with something open ended.* Their BASIC is similar but the features which render it open ended are sufficiently powerful that they amount to an advance in the development of the language.

BASIC (which is a registered trade mark of Dartmouth College) is a comparatively new language. It was developed because no existing language was suited to its application - an easy to learn language for instructional use by beginners on time shared computers. All languages are compromises and this one gave up a great deal for what it gained. Most professionals sneered at it because of its limitations but through the efforts of a small number of enthusiasts, (among them Bob Albrecht, who is still a member of the BASIC Standards Committee) the language became widely known. The more it was known, the more apparent became its unique merits. These were sufficiently important that BASIC may already have become the most widely known computer language, and is clearly destined to dominate the computer world over the next decade. Naturally, the race is on to incorporate into BASIC the things which were given up in the original compromise, resulting in a series of dialects of the language. Where does Altair BASIC fit in? At the moment, right out there in front of the field, but to explain why, let me back up a bit.

The first machines used machine code - which was fine for machines but miserable for humans. So that humans could generate machine code without losing their minds, a language was produced which allowed us to use phrases or mnemonics to assemble machine code, called, appropriately, ASSEMBLY LANGUAGE. This allowed the programmer to use every feature built into the machine and write programs which were the smallest and/or fastest possible. But assembly language was tedious, since every little step had to be defined, and since each new computer used different machine code, programs assembled for one machine would not run on another and the programmer had to learn a new language for each machine. To get over this new languages were developed which took much more general instructions, written in a form familiar to humans, and from them *compiled* the machine code necessary to carry them out. Thus were born FORTRAN and ALGOL, both math oriented, which accepted arithmetic/algebra *like* instructions.

Note the 'like'. While they were a great advance, both languages were still more adapted to the machine than to the human. Thus, they produced comparatively efficient code, but were difficult to use and very difficult to learn. Programs, however, had become portable to the point that they often needed only minor adaption for use on a different machine. Space requirements had risen sharply. The program to *compile* code was appreciably larger than that needed to *assemble* it, though the machine code was not much bigger. COBOL soon fixed that. COBOL was produced in an unsuccessfull attempt to make a language suitable for humans, principally by the expedient of writing everything out in English. This conclusively demonstrated that the information content of 'plus' and '+' are the same; that a human is perfectly at ease with either; that such a compiler requires a horrendous amount of memory to produce even inefficient code - but contributed little else and can, for our purposes, be forgotten.

Now these latter languages were all compiler languages. A program was written in *source language* compiled into *machine code* and then run. This made it difficult to run more than one program at once, so queues developed and people got bad tempered - particularly when they had an error and had to start again from the back. This wasn't too bad in industry or business where there weren't many programmers per computer and they had sufficient experience not to make many mistakes, but in education where fifty students might be trying to get their program to run..... Educators also met another problem - programming languages were so difficult to learn that only those who were going to specialize in computer science had the time necessary.

Kemeny and Kurtz, brilliantly, solved this problem at Dartmouth College by inventing a new kind of language. Beginners All Purpose Symbolic Instruction Code was *interpretive*, not a compiler or an assembler. This meant that the BASIC source language program instead of being converted to machine code, was only slightly altered, being turned instead into a funny stuff called pseudo-code. When the time came to actually run the program, the *BASIC interpreter* looked at the first pseudo-code statement. It pulled it apart, *parsing it*, decided what operations were asked for, did them and then looked at the next pseudo-code statement (unless a jump was called for, of course), keeping going until it came to a statement which said 'end'. This had lots of disadvantages. It was deathly slow because the machine had to keep on working out what the statements meant. It was *very* limited in the things that it would allow a programmer to do and it needed a lot of memory because, when the program was run, both the program and the interpreter had to be in the machine. Despite this, it lived and prospered. The speed didn't matter too much because, even used inefficiently, computers are *fast*. The limitations were more of a problem but not critical because the language was used for education, and so did not need the sophistication of a business or scientific language. The memory requirement was no problem at all, because now lots of people could share the same space. Pseudo-code was easy to shift into and out of a machine and, so long as the machine remembered which statement it was supposed to tackle next, it could simply continue after a pause. And the pauses came about because the machine would now *time share*, sharing processor time among many users by working on one program, moving it out of the machine onto a disk memory, moving in another program and working on that, etc., etc. Alternatively, several programs could be in the same machine each in its own memory partition, all sharing the same interpreter and processor.

All this was very nice, but the true reason for the sucess of BASIC was that it suited people. They could quickly and easily learn to use it, and it was powerful enough to do interesting and useful things. But man, unfortunately, is a chronically disatisfied animal. So then he wanted the speed he had given up and a lot of the features of the other languages to be built into BASIC - and there was no way he was going to give up the things he liked.

Well, the only direct way to speed in BASIC is to compile it instead of interpreting it. This can be done but then one loses one of the nicest traits of BASIC, the ease with which one can change the program. This is only possible because of pseudo-code. Since BASIC interprets one statement at a time, it doesn't mind a bit if you slip in an extra one or delete a few, but the slightest change in a compiled program requires that the whole thing be re-compiled. But the MITS software crew have an *indirect* way to blinding speed (or, more accurately, all the speed of which the processor is capable), they allow BASIC to jump back and forth to machine code subroutines. These can be written as such, assembled or compiled. So the Altair user can simply keep a library of such routines and use them as needed.

This has been done before, as has another of Altair BASIC's major features - the ability to talk out to user-supplied I/O devices, but not as simply and not as cheaply. A BASIC with the power of this one is a major acheivement, there's a monstrous amount of work in it, and they have sold in the low kilobuck range for each copy. Some of the other features, particularly string and numeric matrix handling are certainly unusual and may actually be unique, but the user may notice them less than the general attention to detail evident in the programmers' approach. They, Bill Gates—Paul Allen—Monte Davidoff, must have spent literally hundreds of man-hours mulling over comparatively minor details - but details which sum to a degree of convenience for the user which will endear the dialect to those who are lucky enough to own an Altair. And, hopefully, the fact that MITS have done their BASIC so well will spur their competitors to improve the language yet more. Or, if they can't take a step foward, copy MITS's innovations and at least not take a step back.

All major programs must be expected to have bugs and the extent of the testing required to identify the more subtle ones is so great that it is more reasonable to test for the obvious ones and then wait for the users to find the others, updating software with fixes at intervals. Altair BASIC version 2.0 had a serious problem in that a jump out of a FOR.... NEXT loop left garbage on the stack. Do this too often and the stack would grow relentlessly down from high memory until it ate the program. This has been fixed in version 3.0, according to Paul Allen, but we have not yet received a copy to pull apart. We also hear that there have been a number of other improvements and fixes. Presumably, someone punished it a lot harder than we did, since we found little else to criticize, but we did find one problem and it has unfortunately been left in version 3.0. If you print a value less than 0.1, it appears in scientific notation. So, if you are printing out dollars and cents, and you have less than 10¢, you suddenly see some thing like -- 3.00000E-02.

MITS has frozen development at the 3.0 version, except for bug fixing, and are close to releasing the first versions of their extended BASIC. And THAT has things that will make even more difference to the way BASIC will grow over the next few years. The worst single item BASIC gave up, relative to the other languages, was flexibility in precision, adopting 6 digit floating point for all variables. With floating point even incrementing by one is slow, and business use requires more precision. Altair Extended Basic is to have integer, 6 digit and extended precision variables. . . plus a lot more.

Reprinted from People's Computer Company, September, 1975

BASIC language was chosen for the Altair 8800 because it is the *easiest language to learn* and because it can be used for an *infinite number of applications*. Literally hundreds of thousands of BASIC programs have been written and are in the public domain. These programs include accounting programs, business programs, scientific programs, educational programs, game programs, engineering programs, and much more.

Altair BASIC is an *interactive language*. This means that you get immediate answers and you can use your Altair as a super programmable calculator as well as for writing complicated programs.

8K BASIC Features

Altair 8K BASIC leaves approximately 2K bytes in an 8K Altair for programming which can also be increased by deleting the math functions. This BASIC is the same as the 4K BASIC only with **4 additional statements** [ON....GOTO, ON....GOSUB, OUT, DEF], **1 additional command** [CONT] and **8 additional functions** [COS, LOG, EXP, TAN, ATN, INP, FRE, POS]. Other additional features include *multi-dimensioned arrays* for both strings and numbers, *AND, OR, NOT*

> *"I've seen and used other BASICs, but byte-for-byte, Altair is the most powerful BASIC I've seen. I'm particularly impressed with the n-dimensional arrays (and for strings too!), machine level I/O, and machine language 'function' features. The level of your documentation is, for me, though the high point. Sections for those who know nothing and sections for those who know a lot, plus sections that 'normal' people can read and understand."*
>
> **J. Scott Williams**
> **Bellingham, Washington**

Altair BASIC was written as efficiently as possible to allow for the *maximum number of features in the minimum amount of memory*. You can order one of three Altair BASICs: 4K BASIC-designed to run in an Altair 8800 with as little as 4K of memory, 8K BASIC, or EXTENDED BASIC (12K). Each of these BASICs allows you to have *multiple statements per line* (a memory saving feature), and each of them is capable of executing *700 floating point additions* per second!

The 8K BASIC and EXTENDED BASIC have *multi-dimensioned arrays* for both strings and numbers. This is particularly useful for applications requiring lists of names or numbers such as accounting programs, inventory programs, mailing lists, etc.

The 8K BASIC and EXTENDED BASIC also have an OUT and corresponding INP statement that allows you to use your Altair 8800 *control low speed devices* such as drill presses, lathes, stepping motors, model trains, model airplanes, alarms, heating systems, home entertainment systems, etc.

Altair BASIC comes with complete documentation including a copy of *"My Computer Likes Me When I Speak in BASIC"* by Bob Albrecht, a beginner's BASIC text.

Never before has such a powerful BASIC language been marketed at such low prices!

4K BASIC Features

Altair 4K BASIC leaves apporimxately 750 bytes in a 4K Altair for programming which can be increased by deleting the math functions. This powerful BASIC has **16 statements** [IF...THEN, GOTO, GOSUB, RETURN, FOR, NEXT, READ, INPUT, END, DATA, LET, DIM, REM, RESTOR, PRINT, and STOP] in addition to 4 commands [LIST, RUN, CLEAR, SCRATCH] and **6 functions** [RND, SQR, SIN, ABS, INT and SGN]. Other features include: *direct execution* of any statement except INPUT; an "@" *symbol* that deletes a whole line and a "←" that deletes the last character; *two-character error code* and line number printed when error occurs; *Control C* which is used to interrupt a program; *maximum line number of 65,535;* and all results calculated to at least *six decimal digits of precision.*

operators that can be used in IF statements or forumlas, *strings* with a maximum length of 255 characters, *string concatenation (A$ = B$)* and the following string functions: *LEN, ASC, CHAR$, RIGHT$, LEFT$, MID$, STR$, and VAL.*

EXTENDED BASIC

Altair EXTENDED BASIC is the same as 8K BASIC with the addition of *double precision arithmetic, PRINT USING* and *disk file I/O*. A minimum of 12K memory is required to support EXTENDED BASIC.

Other Altair 8800 software includes a Disk Operating System, assembler, text editor, and system monitor. Altair users also have access to the *Altair Library*, which contains a large number of useful programs.

SOFTWARE PRICES:

Altair 4K BASIC	**$150**
Purchasers of an Altair 8800, 4K of Altair memory, and an Altair I/O board	**$ 60**
Altair 8K BASIC	**$200**
Purchasers of an Altair 8800, 8K of Altair memory, and an Altair I/O board	**$ 75**
Altair Extended BASIC	**$350**
Purchasers of an Altair 8800, 12K of Altair memory, and an Altair I/O Board	**$150**
Altair PACKAGE ONE (assembler, text editor, system monitor)	**$175**
Purchasers of an Altair 8800, 8K of Altair memory, and an Altair I/O board	**$ 30**
Altair Disk Operating System	**$500**
Purchasers of an Altair 8800, 12K of Altair memory, Altair I/O and Altair Floppy Disk	**$150**

Note: *When ordering software, specify paper tape or cassette tape.*

PCC IS THE PEOPLE'S COMPUTER COMPANY

The People's Computer Company

is a newspaper...
about having fun with computers
and learning how to use computers
and how to buy a minicomputer
for yourself or your school
and books...films...and tools
of the future

MUSIC OF A SORT

Steven Dompier

I received my ALTAIR 8800 in the mail at 10 am, and 30 hours later it was up and running...

The radio was picking up the switching noise of the 8800! I tried some other programs to see what they sounded like, and after about 8 hours of messing around I had myself a program that could produce musical tones and actually make music; of a sort. (Or any other program you have!)

The closest sheet of music that I could find was "THE FOOL ON THE HILL", by the Beatles, so I translated it into OCTAL code, picked up the Altair, and headed down to Menlo Park for the 3rd meeting of the 'BAACUG-HBCC'.* I thought everyone there should see just what a computer can do!

* Initialization of "BAY AREA AMATEUR COMPUTER USERS GROUP-HOME BREW COMPUTER CLUB"

Next issue Unusual and exotic devices for your home, school or personal computer. Beginner's info on computer arithmetic and machine language programming. More BUILD YOUR OWN BASIC. And of course, games!

We are a bunch of people (computer people, dragons, and other creatures) who are working together, with the idea that computers should serve people, not rule them.

We do this by publishing the PCC newspaper 6 times a year. By printing letters, writing articles and providing information people want but can't find elsewhere, we seek to serve the information needs of people interested in the humanistic and recreative uses of computers.

Whence the Dragon?

Once upon a time there was a dragon. It lived happily in the land for a while, spreading alarum and disquiet and doing all those things that dragons do. And then it laid an egg.

Way back in the dim depths of the past, fifteen years or so ago, a very straight young computer man used to work for a very straight organization up there in the great city in the sky--five thousand feet up in Denver, to be precise. His name was Bob Albrecht and the flaws in his character were already becoming patent. He took to teaching youngsters aged 14 or so about computers, an heretical notion for computers were expensive and only for mathematicians, certified data processors and similar lordly beings. The corruption of these unsuspecting minds finally tipped our hitherto merely illbalanced hero into the monomania from which he has never recovered. Aroused to the vulnerability of an unwary public, he lumbered forth, vampires byte at the ready.

The scene now shifts some years and miles, to the warm, sunny rocks of California, a proper hatching ground for a dragon. The nursemaids and attendants of the egg began to coalesce. A dragon's nest was prepared with the formation of Dymax. Ostensibly a publishing house, it became, in reality, a center of subversion where computer time was wasted on mere fun. With the periodic presence of a PDP8L the seed was sown and the more perceptive (speaking with hindsight) were clearly able to discern the chip chip chip of a hatching dragon.

Then one day the egg split asunder. A Jolly DEC Giant passed by Dymax, met Bob and gestured to the PDP8L. "Take it," he said, with characteristic generosity, "It's not mine anyway." Bob, who had been at the gin again, stared at this magnificent gift until there crept into his befuddled brain the realization that he had, at last, a computer of his

Dymax moved to Menlo Park, space was contributed, and slowly a computer center grew, to the delight of hundreds of the local school children. Dymax was a profit making corporation (and making enough to support all this) so it was decided to create another Thing, but this time a Good Thing, called Peoples Computer Company, which would publish a newspaper, run the center and do diverse other things allowable under the general non profit corporation laws of the State of California. So Bob and Mary Jo Albrecht, the Jolly DEC Giant (Val Skalabrin), Bob Kahn, Marc LeBrun, Dennis Allison and Lois B. became the first board of directors, and, in 1973, PCC was incorporated.

Even before this, the dragon, which was mature to the point of precocity, had laid an egg, though no one noticed until it too was ready to hatch. Dymax, finding the center too much for its limited space, had rented the store next door and called it Peoples Computer Center, or PCC, to avoid confusion with Peoples Computer Company, or PCC. PCC (Company) divided into three divisions; the newspaper, the bookstore and the center. Bob ran the newspaper while Lois ran the bookstore. It was decided that the Center (or PCC) should be separated from PCC (the Company) and form a new corporation, Peoples Computer Center. (This new dragon has since been renamed Community Computer Center).

Simultaneously, there came the great flood. The skies opened and it was discovered that the newly installed roof was not all it had been cracked up to be. Over Dymax and the newspaper there was a split 36 feet long, and over the Center it was nearly as bad. The roofer, it appeared, did not roof in the rain, and could not stop the leaks in the dry, so PCC the old dragon, his fire steaming and almost out, moved to a new home in downtown Menlo Park,selected partly for the wallpaper and mostly for its proximity to the Village Host, which sells draught Anchor Steam Beer and fine Pizzas.

FACT:

The OLD DRAGON, Peoples Computer Company, is a non profit corporation. PCC is a low budget operation, principally because few people have been paid and the payment has been minimal. PCC is supported by the newspaper subscriptions, the revenue on books sold through the bookstore and much good will. Equipment donated to PCC is used educationally and may be loaned to other non profit organizations, with the requirement that any software developed using it be placed in the public domain. Profits from PCC's game book, What To Do After You Hit Return, purchased through the bookstore are split two ways. Half goes to CCCenter (the new dragon) and the other half is placed in a special fund to buy

PCC BOOKSTORE

BASIC, Albrecht, Finkel & Brown, 1973, p.323 – $3.95
Basic BASIC, James Coan, 1970, p. 256 – $7.45
BASIC PROGRAMMING, Kemeny & Kurtz, 1967, p. 145 – $8.50
COMPUTERS & COMPUTATION, Scientific American, p. 280 – $6.00
COMPUTER LIB & DREAM MACHINES, Theodore H. Nelson, 1974, p. 186 – $7.00
DRAGON SHIRTS, Nancy Hertert, 1974 – $3.50
GAMES, TRICKS & PUZZLES, Wallace Judd, 1974, p. 100 – $2.95
GAMES WITH THE POCKET CALCULATOR, Thiagarajan & Stolovitch, 1975 – $2.00
MATH, WRITING & GAMES, Herbert Kohl, 1974, p. 252 – $2.45
MY COMPUTER LIKES ME, Bob Albrecht, 1972, p. 64 – $2.00
101 BASIC GAMES, Ed. David Ahl, 1974, p. 250 – $7.50
PROBLEMS FOR COMPUTER SOLUTION, Gruenberger & Jaffray, 1965 – $7.95
PROFESSOR GOOGOL, Sam Valenza,Jr., 1973, p. 144 – $3.25
PROBABILITY, D.J.Koosis, 1973, p. 163– $2.95
PCC GAMES Program Listings – $2.00
PRACTICAL, LOW-COST HOME/SCHOOL MICROPROCESSOR SYSTEM, 1974 – $1.00
STEPS TO AN ECOLOGY OF MIND, Gregory Bateson, 1972, p. 517 – $1.95
STATISTICS, D.J.Koosis, 1972, p. 282 – $3.95
THE ENERGY PRIMER, Portola Institute, 1974, p. 200 – $4.50
TTL COOKBOOK, Don Lancaster, 1974, p. 328 – $7.95
II CYBERNETIC FRONTIERS, Stewart Brand, 1974, p. 96 – $2.00
WHOLE EARTH EPILOG, Stewart Brand,Editor, 1974, p. 318 – $4.00
WHAT TO DO AFTER YOU HIT RETURN, PCC, 1975, p. 157 – $6.95
DIGITAL LOGIC CIRCUITS, Sol Libes, 1975, p. 184 – $5.98

TO ORDER—

name _____

address _____ **zip** _____

Quantity	Item	Price Each	Total

* Please add $0.50 for orders under $10.00 and $1.00 for orders $10.00 and over.
* California residents, only, add 6%

Sales Tax*	
Postage*	
Total	

pcc P.O. Box 310 Menlo Park, Ca. 94025

Back issues get them while they last at the following low low prices:

Vol. I, Nos. 1 – 3† $1.00
Vol. II, Nos 5 & 6† $1.00
Vol. III, Nos. 1 – 5 $4.00

† Sorry, we are out of Vol. 1, Nos. 4 & 5 and Vol II, Nos. 1 – 4.

Or mix up individual issues –

2 – 9	$0.80 each
10 – 99	$0.70 each
100 +	$0.60 each

Subscriptions are $5.00 for 6 issues. ($6.00 outside the U.S.A. – surface mail; $12.00 air mail.) Subscriptions begin with the July issue.

Group subscriptions (all mailed to the same address)

2 – 9	$4.00 each
10 – 99	$3.50 each
100 or more	$3.00 each

PEOPLE'S COMPUTER COMPANY

P.O. Box 310 Menlo Park, Ca. 94025

○ back issues
○ vol. I $1
○ vol. II $1
○ vol. III $4

○ regular 1 year $5 ○ 2 years $9 ○
○ group (how many?) ____

Name ____
Address ____
____
zip

Letters to the Editor

Dear David,

Just received Issue Four of the Computer Notes and would like to provide a few comments. First though, I want to thank you for your letter of Sept. 10, and the priority effort in getting me the updates. For some reason I get this intuitive feeling that I may not be on your manual update file. Back in April there was some confusion in one order I placed for some memory along with the update order. Anyway, if this confusion caused me to be dropped from the list, I need to be put back on it. (Ref. Invoice No. 29691 date 3/14/75--William C. Fuller).

Now, related to issue 4. I think you wrote a damn good editorial. I don't necessarily agree with Crain's position about the stifling of business by government and consumer groups, and that managers are no longer able to make innovative, creative decisions. I think MITS proved it just ain't so! At least in electronics.

I believe the problem goes deeper than that, and involves social, psychological, political, economic, etc. shocks that have jarred the "collective unconscious" of the people in the past few years. The song goes something like "...where have all the heroes gone..."

But back to the issue at hand. I think that your analogy of the musician's royalty is valid in applying it to privately developed software. Since software isn't my profession, I have no way of knowing if your prices (royalties) are reasonable or not, so I voice no opinion on it.

Related to your software license agreement. People I have talked to are confused by it, as I admit I am. Some clarification was made about it in a past issue of the Notes, but confusion still exists.

One question I haven't seen asked or really answered is about the application of the license agreement to user developed programs not in MITS BASIC, but in the 8080 instruction set. For example, are programs available from MITS like 521751, 72153 and 721751 supplied to MITS by users under the software license agreement?

Anyway, I think your editorial was appropriate. If MITS can maintain its openness as it has done in the past, then the innovativeness may not be just in the electronics field.

Congratulations on the appearance of the 680...I'm tempted!

I haven't really had time to go through the BASIC language documentation, but a first glance evaluation is that yes, it is a vast improvement over the operator's manual.

By the way, the Texas Computer Club is now The Computer Hobbyist Group of North Texas. Our basic policy is no computer bias, all types welcome. Presently members have 8 Altairs of various configurations and 9 other varieties based upon the 8008, 6800, Mostek 5065, TI APC 980A, PDP-8, etc. We are presently scheduling outside speakers (equipment), so if you have someone who might want to make a presentation on the new 680 around Nov. or Dec. let me know. A club survey made at the last two meetings indicates at least 5 are interested in the Altair. Membership is somewhere between 50 and 75.

Pax, Bill Fuller
Grand Prarie, TX

Dear Bill,

I checked out your Update and you are still on the list. The next mailing should be in about three weeks.

The programs in the Altair Users Library are part of the Public Domain and once you have a copy you can do whatever you want with it. If you should happen to reprint a program, be kind enough to give credit to the author.

Thanks for the nice comments and good luck to the Texas Computer Club.

--D.B.

ALTAIR USERS

Paul W. Schuler, III
1100 Mansion Ave.
Collingswood, NJ 08108

H. Eugene Latham
Box 187
616 AC&W SQ
APO, NY 09277
Wasserkuppe Air Station
Germany

Leonard J. Paul
PO Box 229
Santa Monica, CA 90406
805-497-1975

Samuel Seiber
759 N. Skyline
Idaho Falls, ID 83401

Donald C. Schertz
764 Toyon Dr.
Monterey, CA 93940
home 408-375-6186
office 408-646-2982

Pittsburgh Area Computer Club

Dear Fellow Computer Phreaks:

Just a quick note to let you know that hobby computing and computers are alive and well in Pittsburgh. As you can see from the logo the name of the organization is the PITTSBURGH AREA COMPUTER CLUB. We had our first meeting on October 8, 1975 and even though the weather was inclement we had 15 people in attendance. In addition there is definite interest from at least that many more! Not bad for a beginning.

The name of our acting president is Eric Liber and the name of the secretary/treasurer is Fred Kitman. We can be reached at 400 Smithfield St., Pittsburgh, PA 15222 or at the following telephone numbers:

Day: 412-391-3800
Nite: 412-276-6546

The members of the club have a total of 5 computers with the following breakdown:

Altairs -- 2
PDP/11 -- 1
Wang -- 1
8008 -- 1

Not a bad start!

We would appreciate the inclusion of our organization's name in your publication to inform other hobbyists in this area of our existence.

Thank you for your help,

Eric Liber
Fred Kitman

LAST MINUTE ADDITIONS TO "LOCAL USERS GROUPS"
(continued from page 11)

Anyone interested in forming a users group in the BOSTON area, contact:
John Vullo
21 Sunset Ave.
N. Reading, MA 01864
617-664-4271

PITTSBURGH AREA Computer Club
Fred Kitman
400 Smithfield St.
Pittsburg, PA 15222
412-391-3800
or
Eric Liber
412-276-6546

ARROWHEAD TIPS

Gary Hullquist, M.D.
248 Sunset Dr.
(or Box 448
Professional Bldg.)
Mountain City, TN 37683

Glenn D. Nelson
Geology Dept.
Box 1846
Brown University
Providence, RI 02912

The following tips are on construction of the Altair 8800. Page numbers refer to the Assembly Manual.

Page 11
(D/C Board): Capacitor C7 has been changed several times; you may find change notices referring to various stages of this change process, as well as extra parts. C7 was changed from .001 MFD to .0047 MFD, then to .0068 MFD. Now, the absolute last final ultimate change (as of August 10th!) makes C7 = .01 MFD and changes C8 from .01 to .1 MFD.

Page 18
(D/C Board): Don't bolt the printed circuit board to the sub-panel; the switches will hold it fine. The switches come with extra mounting nuts and extra guide washers; you can safely throw away all this extra hardware - you don't need it. Mount the switches as shown with only one nut each for best results.

Page 39
(CPU board): The wafer connector is about 5% too large to fit the board (or you might say the holes in the board are too close). To prevent the connector plug from arching, clamp it down flat while soldering it on. If you can't clamp it, then try cutting it with a hacksaw into two 4-pin connectors.

(Checkout): After turning the computer on you should reset it. To reset the computer you have to hold the stop switch raised while raising the reset switch. Release the reset switch first. (No, we don't know why, but it's traditional!)

Pages 40-46
(1K Memory): Until May, only the 1K memory board was available, and most systems were ordered with 256 words of memory on one of these boards. Now that the 2K and 4K memories are available, it isn't sensible to require some 1K boards in every system, but the instructions are still embedded in the CPU manual. CPU kits don't have any memory in them.

Page 50
(P/S Board): The bridge rectifier seems to cause more than its share of problems. Be sure the leads are clean - several of us have found that solder won't wet the leads, and it's a mess to try to clean the partially soldered assembly. Run the leads through some alcohol and/or steel wool before installing the bridge. The spacer-and-washer arrangement on Page 51 is a jig to get the bridge flat at the right position it will later be bolted directly to the chassis.

Page 58
(Chassis): MITS doesn't ask you to cut wires to close tolerances. If you follow the instructions here without trimming transformer leads, and use all the #20 wire, you'll have long loops of slack. This is good for allowing boards, etc. to be moved without breaking wires, but you may want to install the terminal lugs after consulting the wiring diagrams on Pages 59 and 62.

Page 66
(Mother board): The way the instructions spell it out, you'll have a slack loop of cable. Hold everything in place on the chassis to see the actual length required. We are including a sorted list of wires to help you check your progress. Using masking tape, group the wires in decades after protecting them with a cable clamp. Then, install them on the motherboard, by decades; 50's, 60's, 70's, 20's, 80's, 30's, 90's, and finally 40's.

Install the cable clamps by bolting them to the printed circuit board only. If you put screws through the sub-panel, then the dress-panel won't fit flush against it and you won't be able to screw the chassis into the case!

Page 68
(Expander Boards): The card guides are maybe sorta optional. They look nice, but they really aren't required to hold the boards in place - the edge connectors are plenty strong for that.

Page 74
(CPU Chip): Many people advocate postponing installation of the CPU chip until after the regulator and zener diodes on the CPU have been tested.

Page 77
(Nameplate): This beauty gets a lot of criticism: "Mine was off-color, kinda pinkish." That's a sticky plastic cover to protect it until after you've installed it. Peel the covering off afterward. The white lettering on the dress-panel can be chipped off by hard use. If you decide to protect it with clear acrylic spray, use a matte-finish product. Ours looks funny with a glossy krylon finish.

DATA/CONTROL BOARD CONNECTIONS TO SYSTEM BUS

ORGANIZED BY DECADE

0's	10's	20's	30's	40's
		20	30	41
		21	31	42
		24	32	43
		26	33	44
		27	34	45
		28	37	46 (11)
		29	39	47
				48
50's	**60's**	**70's**	**80's**	**90's**
53	68	70	80	91
54	69	71	81	92
		72	82	93
		75	83	94
		76	84	95
		77	85	96
		78	86	97
		79	87	98
				99

Altair User's Group

In the words of *The Agency*, MITS' advertising subsidiary, "The Altair User's Group is quite possibly the largest hobbyist organization in the world. It is both a means of communication among Altair users and a method of building a comprehensive library of Altair programs ... among other benefits, you will receive a subscription to the monthly publication *Computer Notes*, which contains complete update information on Altair hardware and software developments, programming tips, general computer articles and other useful information." And that's a pretty fair statement, judging from the August 75 issue of *Computer Notes*.

Edited by Dave Bunnell, the head of *The Agency*, *Computer Notes* covers most items of interest to Altair users. The August edition is headlined "WORLD'S FIRST COMPUTER STORE," featuring an article on The Computer Store, located in West Los Angeles, which is apparently the first retail store solely for computers and computer supplies (shades of Arthur's Information Parlor). The store sells Altairs over the counter of course, and functions as a general computer hobbyist gathering place and information center. *Computer Notes* also features a travelogue of the MITS-MOBILE Altair Caravan, which toured the Southeast during August and September.

In his editorial, Dave Bunnell leads a good deal of rumors to their proper resting places, covering "off brand peripherals, memory cards, etc.", the false rumors concerning "less than full spec" Intel 8080 chips, software agreement technicalities, MITS' development of a Motorola 6800 system, and delivery complaints. Dave precedes all this with the explanation, "One point that has gotten us good press in a number of publications is that we try and level with our customers." From Dave's straightforward presentation, I think they deserve another "good press."

There are no surprises in the rest of the newsletter, with its "Altair Service Dept.," "Letters to the Editor," "HARDWARE," and "SOFTWARE" sections. In "HARDWARE" Tom Durston and Paul van Baalen deliver some ACR (Audio Cassette Recording) hardware alignment updates, fixes for 8800 problems, some "Boo Boos," various maintenance techniques and hardware specifications.

The "SOFTWARE" section contains "Software Contest Winners Announced" by Bill Gates; "Q & A" on the "Monitor, Editor and Assembler" by Paul Wasmund, the author of these software components; "General Software" by Paul Allen, the director of MITS' Software Department, answers various questions concerning MITS' software performance, policies and future plans; Monte Davidoff, one of the authors of Altair BASIC, illustrates the string handling and recursive subroutine capabilities of BASIC in "Fun with Altair BASIC."

Monte does a good job of describing these interesting possibilities of BASIC, even though such applications can be a bit strenuous for both the reader and the BASIC language itself, as witnessed by Monte's closing comment on factorial computation: "If confusion still prevails, do not worry about it."

Bill Gates, in his article "Software Hints for 8800," gives just that. Mentioning some of the reasons for the 8080's power, Bill goes on to discuss binary coded decimal (BCD) arithmetic, giving a sample routine for conversion from BCD to binary. After a short lesson on special short branching ("skip") techniques the reader is presented with some neat but fairly general stack usage tips. Bill doesn't waste words in his article; a rank beginner would probably be left far behind by these last concise hints.

In a short review, Dave Bunnell gives BYTE an A+ for format and an A– for content ("They have something to work for."). Although it sounds a little like an ad for BYTE, Dave is pleasantly positive in his remarks.

I have a couple of complaints. The format of *Computer Notes* is imaginatively done, but at some points the text is hard to follow – you don't know what to read next. [But then, it is only fair to point out that BYTE may suffer the same malaise in one or two places ... Carl] Also, a lot is assumed on the part of the reader. I realize that any such newsletter can't function as a tutorial publication (that is a major part of BYTE's job) but the "HARDWARE" and "SOFTWARE" sections would certainly throw any real beginner. But I judge too harshly: *Computer Notes* is truly a good bulletin for the users of Altair equipment.

To paraphrase Dave's closing question in his BYTE review, is the Altair User's Group something no Altair computer hobbyist should be without? At this point in time, I would have to say that he has no choice since membership is automatic for Altair owners. Should the Altair computer hobbyist be thankful for *Computer Notes* and other benefits of the User's Group? I would have to say yes.

... Chris Ryland

Reprinted from the November issue of Byte. Thanks Chris.

DELIVERY DATES:

The following is a revised list of delivery dates on products appearing in the Altair Brochure, but not yet released. Our product development department is somewhat overloaded at this point, so please bear with us.

PRODUCT	DATE AVAILABLE
COMTER II	January 1976
88-DISK, 88-DCDD	December 1975
88-DMAC, DMAE, DMAI	no date set
88-PPC	January 1976
88-PMC	December 1975
88-PROM	November 1975
88-VI	December 1975
88-RTC	December 1975
88-32DU	discontinued
88-KB	discontinued
88-ACC	no date set
88-CC	no date set
CRT Terminal	no date set
Extended BASIC	December 1975
Extended BASIC/DOS	December 1975
DOS	December 1975

What's Going On In Here

There have been changes in our personnel which will affect our customers: Mr. Paul Van Baalen is now Marketing Vice President and Mr. Harvey Lee is Repair Department Manager. Please refer technical questions to our Repair Department and cost and delivery inquiries to Marketing.

What's Going On Out There

Those of us in marketing and engineering are very interested in what our customers (present and future) would like to see in the way of new products, ideas, iet. The best way for us to know is to drop a post card to Barbara Sims telling us what your ideas are.

For instance we have had some requests for control features for the 88-ACR. Also, we are interested to know if there is a need for a standardized low cost tape recording system, and what standard should be adopted. NOTE: The "HIT" system (Popular Electronics, Sept 1975) is one alternative, but has the disadvantage of not being compatible with serial data transmission format and rates.

Edward Enck
106 Edford St.
New York, NY 10014

Joseph T. Finneran
11 Montrose Ave.
Pinehurst, MA 01866

Richard R. Kenyon
18609-A Del Rio Rd.
Cerritos, CA 90701

Michael Jackson WN4JJI
Sky Park Rd.
R#7
Florence, AL 35630

Wolfgang Luhn
c/o CCT
1001 North American Way
Miami, FL 33132

Gordon L. Quivey
Natlandsfjellet 207
5030 Landas
Norway

Bruce Sidlinger
Recognition Equip., Inc.
PO Box 24

Steve Caldara
36 Theo.Roosevelt Dr. (home)
Blanvelt, NY 10913
965 E. State St. (school)
Ithaca, NY 14850

Ian D. Kettleborough
1108 B. Holik Dr.
College Station, TX 77840

James D. Larson
Inn Lane, Apt. 206

BYTE'TRONICS TO SELL ALTAIRS

BY Lloyd J. Austin

The telephone rings in a small but comfortable store-front office in west Knoxville, Tennessee. A moment later a pleasant voice answers, "MITS - Byte 'Tronics; how can we help you?" Sandra Seals, who is most likely to be the first voice you meet when you call, makes you feel at ease and yet gives you the assurance that you will get prompt and courteous service, no matter what your reason for calling. This is natural with her because she knows that she is working with well-trained and competent people who solve computer problems every day.

Whether you have called for service, for information about the newest of the line of Altair Microcomputers, for a suggestion in a software program, or any other of the myriad of questions that just come naturally, you will get your answers quickly and efficiently. If you have a question concerning the availability of a MITS product, its price, its specifications, or other similar information, Sandra will switch you over to Bruce Seals, who is the marketing director and

The People at BYTE'TRONICS
(l to r) Hugh, Bruce, Sandra, Johnny

sales manager. If you need to know about the compatibility between your Altair 8800 and some input or output device, you will probably talk with Technical Director Hugh Huddelston about it. When you need a software program with special modifications and subroutines, Johnny Reed is your man, and he is familiar with programming on all of the several languages that can be used. All personnel are well-qualified to help you with any service problem because they come from a background of computer applications and are thoroughly familiar with the MITS circuits in Altair, including the optional boards and peripheral devices that can be used.

The office of Byte 'Tronics is located at 5604 Kingston Pike, Knoxville, Tennessee - Zip Code 37919. Everyone is welcome to drop in at any time, to call on the phone at (615) 588-8971, or to make contact by letter. The office hours are (Eastern Time Zone) 10AM to 10PM Monday through Friday and 9AM to 10PM on Saturday. That should allow everyone who is interested to find the office open at some convenient time.

Operating as an independent distributor, Byte'Tronics sells the full line of MITS products. The Altair and its peripherals are available in either kit or ready-to-use form and, whenever possible, on an off-the-shelf timetable. The prices are the same as those in effect at the factory, including special sale prices when they are offered. But the sale and delivery of the equipment is only the beginning of a long and fruitful relationship between every customer and Byte 'Tronics, because their service--both hardware and software--is part of the package that you get when you trade with them. Byte 'Tronics is equally at home with the hobbyist or the professional engineer, the university student or the production line foreman, or any of the many other people who are finding new uses every day for computers-now that they are available and affordable. Hugh and Bruce and Johnny form an excellent team to furnish this client-supplier relationship because they have all been involved in every aspect of the business as users, fixers, testers, and operators.

Hugh's pride and joy is the fully stocked parts cabinet in the back room, with the electronic equipment that will help chase down the trickiest failure. His thorough knowledge of the purpose for each portion of the circuit on the board, together with his experience in troubleshooting, makes him very effi-

Continued on page 3---

COMPUTER NOTES

September, 1975

Volume One Issue Four

A PUBLICATION OF THE ALTAIR USERS GROUP

MITS-MOBILE TOURS NE

Nearing the completion of a highly successful tour of the Southeastern U.S., the MITS MOBILE "Van Man," Mike Hunter, reports SRO crowds in most cities and an enthusiastic response from all those participating. The Northeast and Midwest are up next, with the first stop of the fall tour set for Buffalo, NY on October 20. (for complete schedule, see page 3.)

When the MITS MOBILE comes to your area, you will have a unique opportunity to attend a seminar and slide presentation covering a wide range of topics including computer concepts, technical aspects of computers in general and computer programming. This is your chance to hear some straight talk about computers and to ask any questions you may have--from the most "simple" to the most "advanced." Mike wants to show you how accessible low-cost computing really is.

All the seminars are held at Holiday Inns. Hands-on demonstrations of a complete Altair Basic Language System begin at 6pm, the seminar itself runs from 7-10pm, and after 10pm more discussions or demonstrations for anyone who wishes to stay.

Everyone present will receive a three-ring binder loaded with course material, Altair data and schematics, and catalogs. Reservations cost $12 and must be made in advance by mailing your check (BankAmericard or MasterCharge accepted also) to MITS in Albuquerque. You will receive your admission ticket and the exact location of the seminar by return mail.

MITS/6328 Linn NE/Albuquerque, NM 87108/ 505-265-7553 OR 505-262-1951

************** BULLETIN **************

We have discovered that many Signetic 2604 4K RAM's found on some of our 4K Dynamic Memory Boards do not meet the required specifications for access time and refresh period. They are identified on the package as S 2604. If you have an Altair 4K Board that does not work properly and it has Signetic RAM's, please return the board along with the Signetic RAM's for complete checkout with replacement RAM's. We are sorry for the inconvenience.

Across the Editor's Desk

by David Bunnell

RANCE CRAIN of AD AGE

Rance Crain, the editorial director of Advertising Age and Crain Communications, was in Albuquerque recently to address the local ad club.

Now this isn't the type of news item that would normally be of interest to Altair users, but Mr. Crain said some very interesting things. For example, it is his contention that business has been so stifled by government regulations and consumer protection groups that managers are no longer able to make innovative, creative decisions. And the end result of this is a lack of imaginative products.

My initial reaction is that Mr. Crain is unaware of the electronics business. However, in arguing his position, Crain made some very convincing points about the changing values of consumers and the slowness of catch-up by industry. This is particularly evident in the automotive industry where Detroit has been slow to see the growing consumer demand for cars that are efficient, reliable and safe.

According to Crain, the consumer code of the 60's was "Buy, Buy, Buy." This is no longer in effect. Consumers are more cautious now and, most importantly, consumers are smarter.

The MITS Philosophy

MITS is fortunate that it is a relatively new corporation. Its philosophy is in the formulative stage where it still can be influenced by changing consumer attitudes and existing economic realities.

For one thing, MITS is a very efficient corporation in that money isn't wasted for frills. We make sure that our products get the best engineering expertise but carpeting and pretty packaging are low priorities.

MITS tries not to commit itself to positions that it can't keep, and we try to level with our customers when something goes wrong. Still, we are dictated to by the rules of reality and sometimes someone pokes us in the eye. And sometimes we deserve it.

Consumer Responsibility

The business ethics of the 70's has so far failed to focus on the responsibilities of the consumer. If business is to adhere to new rules, if it is to eliminate misleading advertising, be conscious of the environmental impact of its products, avoid making excessive profits, etc., what then, if anything, should the consumer do in return?

Now, I don't pretend to have the answer to this question but morality is a two way street, and one thing that bothers me is the fact that a few of our customers have been ripping off MITS software. In violation of their software license some people have been arrogantly, and I think foolishly, copying MITS BASIC for resale or to pass out to their freinds.

I believe this practice is fostered by the contention that all software should be made part of the public domain.

Now I ask you--does a musician have the right to collect the royalty on the sale of his records or does a writer have the right to collect the royalty on the sale of his books? Are people who copy software any different than those who copy records and books?

Altair BASIC is one of the most advanced BASICs ever written and it cost MITS a premium price to develop. Considering this and the price charged by many companies for software, $75 for Altair 8K BASIC is a near steal. And the price of $500 for people who haven't purchased a minimum 8800 system is more than reasonable.

Since I've spouted off, I want to invite Altair customers to spout off to us if they've encountered any difficulties such as poor service, delivery, etc. Call us at 505-262-1951 or address your letters to the Altair Customer Service Department.

We can't go about changing the whole MITS mode of operation everytime someone sends us a complaint--but, believe me, your letters will be read and seriously considered by someone in a high management position.

Final Message

Five or six years ago MITS was literally a garage operation not unlike many of the new computer hobby companies that have recently sprung up here and there. MITS was successful because it was able to deliver what it advertised and the success of these new companies will likewise be highly dependent on their credibility.

Most of the people in this business have high ethical standards but watch out for the few ripoff artists. Don't order anything until you know it is "real."

ALTAIR SERVICE DEPT.

Barbara Sims

Hello Again!

We now have several programs in our Users Group library. Descriptions of these have been published in our past and present newsletters. We are also printing prices of the programs in Computer Notes regularly. If you have a program that would be of general interest, please send it in.

It has been a great help now that programs are being submitted on the program submission forms, however there has been a slight mix-up concerning the form. The coding form we sent you in the August issue of our newsletter was a sample. The sample should be used to make up an original and have it printed if you wish. Then, use the printed copies to send in to us. We can print directly from your program if it is handled in this way. However, if you merely photocopy or xerox our sample, we have to retype the entire program. This slows up printing and in the end slows up delivery of the programs to you. The same is true of a typewriter with an old ribbon or dirty keys. If the type is not clear and of good contrast, then we have to retype the program for printing. This all may sound very particular, but we are trying to cut down the handling time so that programs sent in will immediately be available to you, the user.

If you'd like a simpler course to follow, you can order the program submission forms directly from MITS, 50 copies costing $2.00. This is a fairly reasonable price compared to local printing companies. At any rate, keep the programs coming in.

Our Marketing Department needs some help from customers also. Anytime you call in an order to MITS of any type, it is critical that you give the name your 8800 was ordered under and if at all possible, the 8800 invoice number. This helps us in our record keeping between our accounting department, marketing department, and service department.

Adios!

Barbara

Byte 'Tronics continued---

cient. If you need an interface to drive some particular piece of gear, Hugh is most likely the one to make it - or at least to supervise how it is made.

Johnny has recently been spending some of his time at several customers' places of business. He has been writing complete software programs to custom requirements. He is a good man to know when you need help. of course, he can operate any system, simple to complex, and he can spot an error in your program about as fast as the Altair can signal "Error."

Johnny and Bruce are especially interested in a new and growing function - the local Users Group of East Tennessee. All owners of an Altair 8800 are eleigible for a year's free membership in this group, in keeping with the MITS plans. This group meets once a week to swap information and just "talk computers." Sharing ideas and programs is their purpose, and this should be helpful to all whether they are novice or experienced. If you would like more information, drop a post card to them at Byte 'Tronics and you will get the same quick and courteous response as if you asked to buy $10,000 worth of equipment!

So if you are among those who are fortunate enough to already be an Altair user, or if you just want to be, either way you should get acquainted with your friends at Byte 'Tronics because they are well qualified and anxious to help you. All you need is a post card, a letter, a phone call, or - if it is convenient - a visit to the office in Knoxville, and your world will open up in front of you.

MITS SALES REPRESENTATIVES

For detailed information on Altair computers and computer products, contact the MITS sales representative in your area. Or call our factory direct:

MITS/6328 Linn NE
Albuquerque, NM 87108
505-265-7553 or 262-1952

MIDWEST

Inland Associates
13100 Manchester Road #G-25
St. Louis, MO 63131
314-821-3742

Inland Associates
2310 West 75th
Shawnee Mission, KS 66208
913-362-2366

Ridgway East, Inc.
15326 Garfield Road
Detroit, MI 48239
313-538-3940

Ridgway East, Inc.
138 Elseden Building
Florence, KY 41042
606-371-1269

Ridgway East, Inc.
173 Hawthorn Drive
Chagrin Falls, OH 44022
216-247-4845

SOUTHEAST

MITS, Inc.
Southeastern Regional Office
5508 NW 72nd Avenue
Miami, FL 33166
305-885-9388

COL-INS-CO, Inc.
1313 44th Street
Orlando, FL 32809
305-423-7615

COL-INS-CO, Inc.
Offices in:
Huntsville, AL
Largo, FL
Atlanta, GA
Raleigh, NC
Call 800-327-6600 Toll-free

WEST COAST

REPCO
50 East Middlefield Road
Mountain View, CA 94043
415-965-8581

CTI Data Systems
3450 East Spring Street
Long Beach, CA 90806
213-426-7375

CTI Data Systems
8869 Balboa Avenue C
San Diego, CA 92123
714-292-0636

REPCO
PO Box 811
Bellevue, WA 98004
206-455-1246

WESTERN U.S.

BFA Corporation
1350 Chambers Road #207
Aurora, CO 80011
303-344-3800

BFA Corporation
395 Lawndale Drive
Salt Lake City, UT 84115
801-466-6522

BFA Corporation
4251 North Brown Avenue
Scottsdale, AZ 85251
602-946-4215

BFA Corporation
9004 Menaul Boulevard NE
Albuquerque, NM 87112
505-292-1212

BFA Corporation
Dona Ana Road, PO Box 1237
Las Cruces, NM 88001
505-523-0601

NORTHEAST

J. J. Wild, Inc. of New England
PO Box 382
Needham, MA 02192
617-444-2366

J. J. Wild, Inc. of New England
PO Box 342
Southbury, CT 06488
203-264-9494

J. J. Wild, Inc.
73 Main Street
Woodbridge, NJ 07095
201-636-7780

J. J. Wild, Inc.
80 Second Street Pike
Southampton, PA 18966
215-357-6645

J. J. Wild, Inc.
400 Jericho Turnpike
Jericho, NY 11753
516-935-6600
(for Western New York and Western Pennsylvania, see: Ridgway East, Inc. Chagrin Falls, OH)

NOTE: Sales Representatives are geared to serving industrial customers. Hobbyists should contact the factory directly or one of the MITS distributors. Distributors now include Byte'Tronics in Knoxville, Tennessee (see page 1 article) and The Computer Store in West Los Angeles (11656 Pico, phone 213-478-3168). Other MITS distributors will soon be set up across the country. Watch Computer Notes for all the details.

MITS-MOBILE CARAVAN SEMINAR SCHEDULE

GROUP A -- Reservation Deadline October 10, 1975

October 20	Buffalo, NY	October 30	Hartford, CT
October 21	Rochester, NY	October 31	White Plains, NY
October 22	Syracuse, NY	November 3	New York City
October 24	Albany, NY	November 4	Hempstead, LI, NY
October 27	Boston, MA (Cambridge)	November 6	Plainview, LI, NY
October 28	Boston, MA (Newton)	November 7	Hackensack, NJ

GROUP B -- Reservation Deadline October 25, 1975

November 10	Allentown, PA	December 1	Chicago (Northbrook), IL
November 11	Philadelphia, PA	December 2	Milwaukee, WI
November 13	Baltimore, MD	December 4	Minneapolis, MN
November 14	Washington, DC	December 5	Madison, WI
November 17	Pittsburgh, PA	December 8	Indianapolis, IN
November 19	Columbus, OH	December 9	Cincinnati, OH
November 20	Cleveland, OH	December 10	Louisville, KY
November 21	Detroit, MI	December 12	St. Louis, MO
November 24	Kalamazoo, MI	December 15	Davenport, IA
November 25	Chicago (Hinesdale), IL	December 16	Des Moines, IA
		December 17	Omaha, NE
		December 19	Kansas City, MO
		December 21	Wichita, KS

88-VLCT MOD

88-VLCT READY KEY MOD

PROBLEM:
Pressing "READY" key should cause one strobe pulse to PIO board "SBO" line, causing computer to output data to octal display. Noise from keyswitch bounce causes multiple pulses on "SBO" line, causing next byte entered to be echoed without pressing READY.

SOLUTION:
Change R32 from 10K to 10meg. R32 is across C6, the .01 pulse timing capacitor for the READY key. Increasing R32 to 10meg makes discharge time for C6 greater than 10ms, preventing keyswitch bounce.

NOTE:
READY key schematic is incorrect: R33, 100Ω, shown going to Vcc is actually connected to ground. R31, 47Ω, shown going to ground, actually goes to Vcc.

10meg -- MITS part number 102079

USERS' NAMES & ADDRESSES

Frank E. Corlett
205 Port-O-Call
Bridgeport, TX 76026

Keith L. Kendall
295 E. 500 S.
Vernal, UT 84078

Glenn Nelson
160 Greenway W.
Newhyde Park, NY 11040

Mount St. Mary's
c/o Bill O'Toole
Emmitsburg, MD 21727

Sgt. Wesley B. Isgregg
Box 3558 C & E Schools MCB
29 Palms, CA 92276
714-368-9111 DWH Ext. 6241
AWH Ext. 7289

Sgt. Stanley E. Herr
13-C Copper Dr. MCB
29 Palms, CA 92278
714-368-3809

Robert Beard
2530 Hillegass Apt. 109
Berkeley, CA 94704

Tod Rapp
129 Park Dr.
Xenia, OH 45385
Business 513-372-8294
Home 513-372-8445

Jim Fisk-WB9N1V
4116 Euclid Ave.
Ft. Wayne, IN 46806
219-745-0359

Michael A. Enkelis
9924 S.W. 31st Ave.
Portland, OR 97219
503-246-4614

Stephen E. Clark
2099 Powder Horn Dr.
Clearwater, FL 33528

Bruce Segal
64 Summit Crescent
Westmount PQ
Canada H3Y 1L6

C. A. Kirkpatrick
2041 San Sebastian Ct.
Apt. 70
Houston, TX 77058

Richard Hole-WA8TSY
Gerber Products Company
Fremont, MI 49412
Business 616-928-2692
Home 616-652-6884

Gary Tack
P.O. Box 866
Corrales, NM 87048

Matthew W. Smith
4355 S. High St.
Englewood, CO 80110

Gary S. Trent
Route 5, Box 900
Space 17
Orange, TX 77630

John E. Kabat
"Universe Unlimited"
User's Group
11918 Forrest Ave.
Cleveland, OH 44120
Business 216-781-9400 Ext.55
Home 216-795-2565

D. Minott
352 Arkansas Dr.
Valley Stream, NY 11580

Donald C. Schertz
764 Toyon Dr.
Monterey, CA 93940

Ivan Wampfler
7861 Old River Rd.
Rockford, IL 61103
815-633-4757

Richard Bushick-WA3PWO
418 Brian Ct.
Mechanicsburg, PA 17055

John Rabenaldt
1301 N. Jackson #24
Odessa, TX 79761
915-337-6931

To be featured on cover of November, 1975 Popular Electronics.

SNEAK PREVIEW

In January of 1975, MITS stunned the computer world with the announcement of the Altair 8800 general purpose computer that sells for $439 in kit form and $621 assembled.

In October of 1975, MITS will announce a complete computer built around the 6800 MPU available from Motorola and AMI.

This computer will come with an MPU board that has 1K RAM, built-in I/O that can be configured three ways, and provision for 1K ROM or PROM. It will have power supply and be sold with front panel control board in an 11" x 11" x 4 11/16" case for $293 in kit form and $420 assembled.

The MPU Board—ideal for dedicated control applications—will be marketed for $180 in kit form and $275 assembled.

For complete details, see November's Popular Electronics.

Aug Software Contest Winners Announced

by Paul Allen and Bill Gates

This month nine programs and nine subroutines were added to the Library.

The ultimate in memory clears seems to have been written by Ward Christensen (#731751).

"I have been using the following 'program' to clear memory in my Altair since the first week I had it. It clears all of memory except byte 0, and leaves the address in location 1. It works because the Altair seems to have the stack pointer at location 0 when powered up. At worst, it would have to be run twice."

```
0000   063   INX SP
0001   307   RST 0
```

The winning major program this month is a quadruple byte integer manipulation package by Steve Phillips (see below). In second place is R. J. Walker's PIO BASIC or Package I loader. Third place goes to John Trautschold for his mutiplication program for floating point numbers.

In the subroutine category, first place goes to Donald Tork for his table search routine. Second place goes to Jonathan Griffitts for a subroutine which may be used to display the A register in the status lights for one second, which could be useful as a debugging tool.

PROGRAMS

#92751
Author: Dean B. McDaniel
Length: 23 bytes
"Object: To kill the rotating bit. If you miss the lit bit another one at that sense switch position will turn on, now leaving you two bits to destroy."

#815751
Author: John Trautschold
Length: 476 bytes
Multiplies a 12-digit floating point number times an 8-digit floating point number.

#88751
Author: Charlie Shields
Length: 22 bytes
Outputs 1's to an I/O port a selected percentage of the time. Outputs 0's the rest of the time.

#829751
Author: R. J. Walker
Length: 57 bytes
Loads checksummed 4K or 8K Altair BASIC from an unmodified PIO board.

#825751
Author: Robert L. Berg
Length: 34 bytes
Simple memory test. Halts when finds a bad location.

#818752
Author: Steve Phillips
Length: 477 bytes
Quadruple byte signed integer manipulation package. Addition, subtraction, multiplication, division as well as sign and zero value testing and other useful routines.

#813751
Author: John S. Robison
Length: 277 bytes
Tests ACR interface by comparing the record line output to the input from the monitor playback output from those tape machines which have monitor capability.

#94751
Author: Carl Swift
Length: 15 and 11 statements
BASIC programs to dump/store memory in octal using Altair BASIC's PEEK and POKE statements.

SUBROUTINES

#825754
Author: Dr. Jack W. Crenshaw
Length: 16 bytes
ASCII/HEX -- HEX/ASCII conversion subroutines.

#818751
Author: Jonathan Griffitts
Length: 23 bytes
Displays the contents of the accumulator in the address lights.

#811753
Author: Thomas D. Thomas
Length: 19 bytes
Adds a quadruple byte integer pointed to by [H,L] to registers [B,C,D,E].

#813752
Author: David Nowak
Length: 4 bytes
Subroutine to do a relative jump using a displacement in [D,E]. Could be used as an RST.

```
RJMP:  XTHL    ;Get return address in
               ;[H,L]
               ;Save [H,L] on stack
       DAD D   ;Add on displacement
       XTHL    ;Save back return
               ;address and restore
               ;[H,L]
       RET     ;Do relative branch
```

(Author's note: How about an RST that does a relative branch on the signed byte that follows the RST? Anyone?)

#84753
Author: Nilton G. Gimenes
Length: 33 bytes
Converts a 2-byte unsigned number into 6 individual octal digits in consecutive memory locations.

#811751
Author: Donald S. Tork
Length: 74 bytes
Searches a table of up to 255 entries of up to 255 byte strings for a match on the search string, which may also be up to 255 bytes long.

#99751
Author: Gary Tack
Length: 21 and 27 bytes
Routines to add/subtract two 16-digit BCD numbers.

#84752
Author: Nilton G. Gimenes
Length: 28 bytes
Translates six octal digits in consecutive locations in memory to a double byte value in [D,E].

#84751
Author: Nilton G. Gimenes
Length: 15 bytes for each routine
Octal/ASCII -- ASCII/octal conversion routines for octal digits/characters stored in six sequential memory locations.

NOTES ON PACKAGE I
By Paul Wasmund

One major point that was not covered in the Package I documentation was the stack. There are 30 bytes of stack available for the user at all times. If a program needs more stack space than that, it should set up a stack of its own.

If a program should infinite loop, stop your machine and restart the monitor by examining location Ø and pressing the run switch.

Release Two of Package I will be ready in early October. New features include tab stops to help make your program listings more readable and a highly improved editor. Two new commands have been added and all old editing commands have been improved.

FIND - The find command searches for a given string until a line containing it is found. The entire line it is in is then printed.

ALTER - Allows altering characters within a line. This command allows lines to be changed without having to replace them.

The insert and delete commands have been improved so that you can now insert blocks of lines and delete blocks of lines.

SOFTWARE

GENERAL SoftWARE UpDATE INFO

by Paul Allen

Here are a few items of interest:

1. The current version of BASIC being shipped is 3.1. The only change is that in the 8K version the array access is now significantly faster.

2. The FORTRAN cross assembler is now available on paper tape as well as a listing. The paper tape and listing together cost $30, while the listing alone costs $15. If you wish to obtain the cross assembler in another form, call the software department at the factory. The program order number is 521751.

3. There will be two versions of Extended BASIC; one that runs with the disk and one that runs without it. We are assuming that all orders placed are for the disk version so if you want the "stand-alone" version of Extended BASIC, drop us a line to let us know. The advantage of the stand-alone version is that it is 2K bytes smaller.

Programmed I/O

The coding technique for data input and output in which the CPU waits for completion of the I/O operation is usually termed "programmed I/O." This is by far the easiest and most common way of writing input and output subroutines for the Altair, and is used by BASIC and the Package I software.

There are usually two subroutines for each device. One that inputs a character from the device and one that outputs a character to the device. The input routine (INCHR) waits for the device's input buffer full flag to be set and then reads the character. On the Altair, the device status is in the input side of the lower I/O channel, and the data is read from that channel +1. Assuming we will return the byte read from the device in the A register, the code is as follows (for an old SIOC board--character ready bit in bit 5):

```
INCHR:  IN INCHN        ;where ICHN is the input channel
        ANI 40Q         ;TEST BIT 5=0 (Q means octal). The mask 40Q is
                        ;"anded" with the device status in the A register.
                        ;The mask (40Q) selects only bit 5.
        JZ INCHR        ;If no input data ready, loop.
        IN INCHN+1      ;Read the input byte.
        RET             ;Return from the subroutine.
```

Note that the input character routine is a "subroutine" that could be called many different places in a program by using a CALL instruction, i.e.

```
        .
        .
        .
        CALL INCHR      ;Get a character from the terminal.
        CPI 15Q         ;Was it a carriage return?
        JZ ENDLIN       ;If so, end of input line.
        .
        .
        .
```

Of course, the stack pointer must be set up pointing to an area of memory set aside for use by subroutine calls and PUSH/POP and other stack manipulations. This is most easily done as follows (this code is usually placed at the start of your program):

```
START:  LXI   SP, STACK
        .
        .
        .
        DS    20        ;set aside 20 locations (10 levels) of stack space
STACK:
```

More information on how to use the stack will be provided in Bill Gates' software article next month.

A corresponding character (byte) output subroutine for an old (REV 0) SIO board is listed below. The byte to be output is in the A register:

```
OUTCHR: PUSH PSW        ;Save the A register on the stack.
OUTLP:  IN INCHN        ;Read the device status into the A register.
        ANI 2Q          ;See if bit 1 is = 0.
        JZ OUTLP        ;If it is, keep waiting for the terminal to finish
                        ;printing.
        POP PSW         ;Get back the saved output byte.
        OUT INCHN + 1   ;Now output the byte to the terminal.
        RET             ;Return from subroutine.
```

REMINDER:
Users of BASIC and Package I should address their console I/O boards (SIOA, SIOC) for I/O port 0. The ACR board should be address for I/O port 6.

Often it is desirable to echo the character read from a terminal's keyboard immediately back to the terminal. The easiest way to do this is to insert

```
INECHO:  CALL INCHR
```

right before the OUTCHR routine and then call INECHO instead of INCHR. If we knew we were always going to echo the input character back to the terminal, we could have the input character subroutine (INCHR) "fall into" the output character routine (OUTCHR). This may be done by placing INCHR directly ahead of OUTCHR and also removing the RET at the end of INCHR so an "OUTCHR" will always be performed when INCHR is called.

---continued on page 7

Slight modifications must be made to these routines if we want to use REV 1 or modified REV 0 serial I/O boards. In these boards, the character ready bit is in bit 0 of the status byte, and the character done (sent) bit is in bit 7. Also, the bits are "active low," that is, a 1 means the bit is false and a zero means the bit is true, which is just the opposite of the way the bits were set on the REV 0 board used in the previous examples. We could test bits by using an AND immediate instruction as before (i.e. replace the ANI 40Q in INCHR with an ANI 1Q and the ANI 2Q to an ANI 200Q) and changing the JZ's to JNZ's. However since the status bits are in the least and most significant bits in the status byte, we can conveniently test them by using the rotate instruction to move the bit in question into the carry flag and then using a JNC instruction to loop:

```
INCHR:  IN ICHN     ;Read status
        RAR         ;Character ready?
        JC  INCHR   ;If not, loop
        IN INCHN+1  ;Read character
        RET         ;Return

OUTCHR: PUSH PSW    ;Save character
OUTLP:  IN ICHN     ;Read status
        RAL         ;Test bit 7
        JC  OUTLP
        POP PSW     ;Get character
                    ;back in A
        OUT INCHN+1;Send it to
                    ;terminal
        RET         ;All done, return
```

Using rotates instead of ANIs saves one byte in each routine. Remember: taking care to save each byte you can will make long programs significantly shorter and faster.

PIO boards (often used for SWTPC TVTs) have the status bits "active low" like REV 1 SIO boards, but the status bits are in different positions: character ready is bit 1 and character done is bit 0, so:

```
INCHR:  IN   INCHN
        ANI  2Q
        JNZ  INCHR
        IN   INCHN+1
        RET

OUTCHR: PUSH PSW
OUTLP:  IN   ICHN
        RAR
        JNC  OUTLP
        POP  PSW
        OUT  INCHN+1
        RET
```

If you are confused by the use of "masks," here is an explanation. If we want to make a jump on only one bit of the A register, we "and" a mask with that bit on with A. The result of the AND will be zero if that bit was zero, and non-zero if the bit was one. Here is a table of bit masks (in octal) for each bit position:

BIT	MASK	
0	1	(usually use RAR to test)
1	2	
2	4	
3	10	
4	20	
5	40	
6	100	
7	200	(usually use RAL to test)

Note that bits 0 and 7 take fewer bytes to test than the rest because they can be rotated into the carry status bit as mentioned earlier.

It is often very useful to use bit testing and setting in a program. Suppose you are writing an assembler and you want to remember if you have seen any colons or commas on a line. You could use one bit in a register to flag the fact you had seen a colon and another bit to flag whether you had seen a comma; and you could use the other six bits of the register for six other flags. Suppose the flags were kept in the B register. Then, to set a flag (if bit=1 means set):

```
        .
        .
        .
   MOV A,B    ;Get flag register in A

   DRI 2      ;Mark colon seen (bit 1)

   MOV B,A    ;Save flags back
```

To reset a flag:

```
   MOV A,B    ;Get flag register in A

   ANI 375    ;377-2
              ;Reset colon flag (bit 1)

   MOV B,A    ;Save flags back
```

To test two flags:

```
   MOV A,B    ;Get flag register

   ANI 12Q    ;Test both bits 3 & 1
              ;(colon and comma)

   JZ NETHER ;Jump to NETHER if both
              ;flags = 0

  JNZ ONEFLG ;Jump to ONEFLG if one
              ;or both of two flags
              ;set.
```

To complement (invert) a flag (reset it if set, set it if reset):

```
   MOV A,B    ;Get flag register

   XRI 2      ;Flip (complement) bit 1

   MOV B,A    ;Save flags back
```

Final Note:

If you'd like us to cover some particular technique or coding practice in detail, let us know.

Software Notes

by Bill Gates

Though the most difficult and enjoyable part of writing a program is the design of data structures and program flow, it is also important to use the least number of instructions possible to perform each function in a program. For instance:

```
CALL SUB1      should be replaced by
RET

JMP SUB1       unless something fairly
               tricky is being done
```

with return addresses. The JMP is faster, takes one less byte, and uses no stack space. An instruction book on programming the 8008 ignores this simple fact!

JMPs should be avoided wherever possible. By rearranging code you can often avoid having an unconditional JMP by falling into the routine you were JMPing to.

The beginning programmer will use lots of SHLDs, LHLDs, STAs and LDAs when they are not necessary. The stack can be used to save temporary values in most cases. SHLDs, LHLDs, LDAs and STAs should only be used for values referenced in many different contexts within a program, i.e. an I/O parameter or the current line number.

A good technique for familiarizing yourself with the instruction set is to go out of your way to use every instruction at least once (except perhaps DAA). Go through the instruction set from time to time and look closely at the instructions you seem to use very rarely. With few exceptions (DAA, SPHL) all the instructions can be used to advantage, even in small programs. One of the most overlooked instructions is XTHL. When all the accumulators have values that must be saved and a value needs to be taken off the stack, XTHL is the only instruction that can be used.

```
Example:  ;Exchange [B,C] with [H,L]

PUSH B    ;put [B,C] on the stack

XTHL      ;[H,L] = top stack entry =
           [B,C]

          ;[H,L] goes on the stack

POP B     ;[B,C] = original [H,L]
```

Sometimes the simple way of doing things is the best. PUSH B/POP D may seem like a tricky way of setting [D,E] = [B,C], but the obvious sequence MOV D,B/MOV E,C is much faster.

Some tricks involve instruction sequences which at first sight seem meaningless. For instance: SUB A or XRA A. Subtracting A from itself or exclusive-oring A with itself are the only one-byte ways of setting A=0. MVI A,0 must still be used if the condition codes need to be preserved, but this is rare.

ADC A is equivalent to RAL, except it affects all the condition codes. SBB A sets A=0 if carry is off and A=377 if carry is on. The routine below uses this fact to convert A as a signed integer to a double byte signed integer in [H,L]:

```
MOV L,A  ;setup the low order
         ;now the sign must be
         ;"extended" by setting H=0
         ;if A=>0 and H=377 otherwise
RAL      ;Carry = 1 if A<0
         ;Carry = 0 if A=>0
SBB A    ;A=0 if old A was =>0
         ;A=377 if old A was <0
MOV H,A  ;setup the high order
```

The sequence:

```
INR E
DCR E
```

doesn't modify any values, but it does set the condition codes (except carry) depending on what is in E. If E is being used as a flag to indicate, say, whether or not a decimal point has been seen, the zero flag is set up to do a conditional JMP.

The subject of good decimal print routines has been discussed extensively in the Altair Software Department this week. This routine is one of the four or five I wrote this week -- each with its own advantages and disadvantages. This one is fairly tricky, in that it takes a little bit of looking at to understand.

```
#1 ;
   ;Print the binary unsigned number
   ;in [H,L] in decimal, suppressing
   ;leading zeros
   ;
   ;24 bytes (25 if saves D,E)
   ;ON RETURN:
   ;A = last digit in ASCII
   ;B,D = 255 (all constants in
   ;decimal)
   ;C,E = last digit -10
   ;H,L = 0
   ;
   ;Uses up to 18 bytes of stack
   ;Total compute time up to 85
   ;milliseconds
   ;
   ;IDEA: calculate a digit, save it
   ;      on the stack, and call the
   ;      digit calculator to calcu-
   ;      late and print higher order
   ;      digits, pop the digit off
   ;      and print it.
   ;
DECOUT: LXI B, -10      ;CALL here
GETDIG: MOV D,B         ;[D,E] = -1
        MOV E,B         ;since B = 255
LOOPSB: DAD B           ;Subtract 10 from [H,L] until [H,L] < 10. Carry
                        ;won't be set by the last DAD when [H,L] < 10.
        INX D           ;increment the count
        JC LOOPSB       ;loop subtracting
        PUSH H          ;[L] = current digit -10
                        ;Save the current digit on the stack.  Change to
                        ;XTHL and add PUSH D at GETDIG to save [D,E].
        XCHG            ;[H,L] = old [H,L]/10
        MOV A,H         ;Set zero flag if [H,L] = 0
        ORA L
        CNZ GETDIG      ;If not zero, print the higher order digits and
                        ;then return here to print this digit.
        MVI A, "0" + 10 ;A = constant to add to digit
        POP B           ;pop the digit into C
        ADD C           ;A = ASCII of digit
        JMP OUTCHR      ;Jump to the routine to print A and return.  If
                        ;OUTCHR is located next, the JMP can be eliminated.
```

Parity is used as a check to detect errors in data transmission. Each data word is given an additional bit which is set to 1 if there are an odd number of 1's in the data and 0 otherwise. When the data is received the parity bit is checked to make sure it is set properly. Thus, if you are reading a 7-bit ASCII paper tape with the 8th bit used for parity, the parity of the entire 8 bits should be even.

The reason I first thought about a parity routine for the 8080 is that the parity condition code and all the instructions related to it (JPO, JPE, RPE, RPO, CPO, CPE) are seldom used. I wondered how difficult it would be to calculate parity if the parity flag were removed. A user-settable flag would be much more useful than the parity flag. BASIC uses the parity flag in only about eight places, and all of these are special tricks. Here is the smallest parity routine I've been able to write:

```
;Enter with number in A.  10 bytes.
;On exit, A=0 and all the other reg-
;isters are preserved.
;Carry is set depending on A's
;parity.
;Enter at ODDPAR for carry on to
;mean odd parity.
ODDPAR: ADD A           ;Move a bit of A into carry.
        RZ              ;If all bits added into carry, return.
        JNC ODDPAR      ;If no bit moved into carry, rotate more.
;enter at EVNPAR for carry on to
;mean even parity
EVNPAR: ADI 200         ;Complement the parity of the remaining bits
        JMP ODDPAR      ;Rotate more.
```

I said last month I would explain the bootstrap loader but I've decided that should wait until next month when I explain the basics of the stack.

Also next month: multiprecision arithmetic, and more interesting subroutines.

WHICH I/O INTERFACE FOR YOU?

1. SIOC-
   For Teletypes* or other 20mA current loop asynchronous terminals up to 19,200 baud. (5-8 data bits)

2. SIOA-
   For asynchronous RS-232 CRTs or other terminals of data rates up to 19,200 baud. (5-8 data bits)

3. SIOB-
   Same as SIOA and SIOC except output and input are TTL compatible levels.

4. PIO-
   For bidirectional transmission of bytes at speeds up to approximately 25,000 bytes/sec (200,000 baud). Eight lines (1 byte) in and out plus "handshaking." All lines standard TTL compatible. Most commonly used for SWTPC-TVTs or equivalent, custom A/D-D/A interfaces, computer to computer interfaces, numerical control applications.

*Teletype is a registered trademark of the Teletype Corporation.

Package 1 continued--

Another large improvement was made in line number specifications. In addition to being able to say Print line 5 (P5) you now can also say print the current line (P.) or print the current line plus or minus a constant (P.-6, .+6).

Also, typing escape will print the previous line, and line feed will print the line after the current one.

WORLD'S FIRST COMPUTER STORE

The World's First Retail Computer Store, The Computer Store, is now open for business in West Los Angeles. It is located at 11656 Pico (at Barrington) 1/4 mile west of the San Diego Frwy. (phone 213-478-3168).

Owned and operated by Dick Heiser, The Computer Store is an offshoot of the Arrowhead Computer Company. Not too surprisingly, The Computer Store features the Altair 8800 in both kit and assembled units along with Altair I/O's,(Parallel,Serial RS-232, Serial TTL, and Serial TTY), and Altair memory (1K and 4K boards). That's right, you can walk right in off the street, put down your money and walk out with your own Altair computer!

The Computer Store

The store also has an Altair 8K BASIC language system connected to a Teletype for demonstration purposes.

Besides selling Altairs, Heiser is making his store available for a number of services including kit assembly advice, software information, and a meeting place for computer hobbyists. Store hours are Wednesday through Friday, 2pm-8pm and Saturday-Sunday, 1pm-6pm.

Contacted by Computer Notes, Heiser reported that business has been very brisk and that he is doing much better than he ever imagined.

"People in this business haven't been optimistic enough," said Heiser, "I've tried to be as optimistic as possible but that has turned out to be too conservative. You have to be wildly optimistic!"

COMPUTER NOTES

August, 1975

Volume One Issue Three

A PUBLICATION OF THE ALTAIR USERS GROUP

MITS-MOBILE HITS SE

The MITS-MOBILE is on the road again--this time making stops throughout the Southeastern section of the United States.

The new driver and lecturer of the MITS-MOBILE (he refers to himself as the "van man") is Mike Hunter who originates from Miami where, among other things, he worked as a Custom Engineer for IBM.

Mike's computer experience includes knowledge of APL,Fortran, Cobol,BASIC and basic logic circuitry. Besides having a great deal of experience with the Altair 8800, Mike has also worked with the IBM 370 and Honeywell computers.

Mike will be teaching computer seminars at major Southeastern population centers. These seminars are divided into four sections covering (1) introduction to computers,(2) Altair 8800 hardware and interfacing techniques,(3) programming in machine language, assembly language, and Altair BASIC, and (4) questions and answers.

Participants in the seminars will each receive an Altair binder filled with a course outline, technical data and other good stuff. There will be a slide presentation, a door prize, refreshments and hands-on demonstration both before and after the seminar.

All seminars will be held at Holiday Inns (see schedule on page 3). Cost of the seminars is $9.75 per person. Interested persons are asked to make reservations before-hand by mailing a check to MITS (note "Altair Caravan" on envelope). A limited number of tickets are available at the door in most locations.

--SEE PAGE 3 FOR SCHEDULE--

Dick Heiser

Heiser, who is an active member of the Southern California Computer Society, is also marketing computer books and literature. His book list is basically the same as that at People's Computer Company and includes BASIC by Albrecht,Finkel & Bron, BASIC BASIC by James Coan, COMPUTER LIB & DREAM MACHINES by Theodore H. Nelson, DRAGON SHIRTS by Nancy Hertert, MY COMPUTER LIKES ME by Dymax, 101 BASIC GAMES by David Ahl, and many others.

In the future, Computer Notes will carry progress reports about The Computer Store and articles about other Altair retail outlets which will soon be opening across the country.

Rumors, Clarifications, and Who's Pulling Your Leg

Across the Editor's Desk

by David Bunnell

Let's face it-- the Altair 8800 is a revolutionary development that is having quite an impact.

Now in this business, revolutions come and go. However, I think it is safe to assume that the Altair has made its mark.

Because of this, the Altair 8800 has received a considerable amount of press coverage. Some of this coverage has been good, accurate reporting with constructive criticism and some of it has been based on rumors and faulty information.

For instance, one hobby publication ran an article stating that MITS had yet to deliver 4K boards when at the time the number of delivered Altair 4K boards was in the thousands.

One point that has gotten us good press in a number of publications is that we try to level with our customers. When we make a mistake we publish it in Computer Notes.

So, let's clear the air:

ITEM: Off-brand peripherals, memory cards, etc.

There is very little we can do or even want to do to prevent Altair 8800 customers from using these devices. The days of Big Brother Computer Company requiring Big Brother users to use Big Brother peripherals are over-- period.

However, this does not mean that we don't intend to be competitive. One obvious example of this is the structure of our software prices. Purchasers of an Altair 8800 plus X amount of Altair memory plus an Altair I/O card receive a discounted price on Altair software.

If you buy a certain amount of Altair equipment from us we can afford to give you a break on software. If you just buy an Altair 8800, we can't afford to give you the same break.

Another point in this area is that we cannot be responsible under our warranty for repair problems that result from customers using off-brand add-ons and we can't be responsible for interfacing these devices. This, I think, should be understandable.

Anyway, off-brand add-ons are a testimony to the success of Altair and the faith among other manufacturers that the 8800 is here to stay for along time.

Someone recently showed us a 4K board that's been widely advertised. We checked it out and found that while it was an attractive board it uses almost 4 times the power of an Altair 4K Board. This means that you are going to run short on power rather quickly as you develop your machine.

The Altair 8800 was designed to power up to 16 Altair boards. If you run out because you're using some other types of memory or I/O boards-- well, that can't be our responsibility.

ITEM: Intel 8080 chips that are less than full specification.

A rumor that we have done something about is the rumor that MITS uses inferior 8080 chips (how else could we market the Altair 8800 for $439).

This rumor is absolutely untrue. It is a lie.

Unfortunately, those responsible for this rumor include field salesmen for Intel who were hard pressed explaining our low price to potential customers.($439 for an Altair compared to something like $350 for a single 8080 chip).

Thanks to help from Intel executives, this rumor has at least been partially put to rest (see Intel letter reprinted on this page).

ITEM: Software Agreement. Software costs us alot of bucks. Therefore we can't allow people to copy our software and we will prosecute anyone who violates their license agreement.

However, peripherals or memory boards or I/O boards you have built yourself or obtained from another source need not be included on your designated equipment list. We would like to have this information just so we can keep an eye on what's happening, but you aren't required to submit it.

While the agreement does forbid you from running MITS software on someone else's computer, this should not be interpreted to mean that you can't do program development on a larger machine. MITS recognizes the advantages of such an approach and there will be no objections raised to Altair software's being used on other machines if the use is restricted to development of programs to be run on Altairs.

ITEM: 6800 CPU Chip. Yes, MITS is working on developments around this chip. However, it is absolutely not true that we have any plans,notions, thoughts or intentions of replacing the 8080 CPU with a 6800 CPU.

The Altair 8800 is our most powerful processor and it will remain our most powerful processor for a considerable length of time. Altair 8800 development programs are in high gear and Altair 8800 owners can be assured that we intend to support them now and indefinitely into the future.

ITEM: Delivery. We have had complaints about delivery and in some cases we have had problems in this area. However, all things considered, we are doing quite well.

For instance, the Altair 8800 in kit form is approaching the point where it is going to be an off-the-shelf item. And this should not be interpreted to mean sales are down (mail order business is usually down in the summer-- this summer our sales have actually gone up).

If you order an Altair 8800 kit and pay for it by personal check it takes 3 weeks to clear your check and this, obviously, is the cause of many of our delays. If, however, you order an Altair 8800 kit and pay for it by money order, cashier's check or by BankAmericard or Master Charge, it takes only 2 or 3 days to process your order.

Other kit items that are approaching off-the-shelf status include the IK and 4K memory boards, the parallel and all three serial I/O boards, the ACR board, and 8K BASIC language.

The CT256 and the COMTER II have been delayed for case modifications but we hope to have them back on the track by the end of August.

As for ASR-33 Teletypes-- we have sold out our stock but we are looking for a new delivery by September which is also the time schedule for delivery of Altair Line Printers.

One problem we have had with assembled units is that many orders specify that they will not accept partial shipments. This means if you have ordered a Real Time Clock or a Vectored Interrupt your shipment will be held up until these two items are being shipped. (November 1st).

The point is: we may not be perfect, but we have a pretty damned good track record.

40
hank o'hara sales bits

TO: All Field Sales

FROM: Hank O'Hara

SUBJECT: MITS ALTAIR 8800 MICROCOMPUTER

We wish to clarify any misconceptions that may exist in your minds regarding the MITS ALTAIR system. This product is designed around the Intel Standard Data Sheet 8080 family. The immediate market acceptance of the Altair system should serve as an opportunity indicator for Intel's MCS product family to all of us. It demonstrates in clear terms what a creative designer can achieve with Intel's products.

MITS is a valued customer to Intel. MITS should receive the same support from Intel field sales as any other major customer when these successful users of Intel products are used to support sales presentations to prospective customers for future applications of our MCS product family.

Any presentations of our MDS-800 and other Intellecs to prospective customers should be based on the positive merits of our products only. It is not necessary to make derogatory comments regarding any competitive manufacturer of MCS™ type products that use Intel MCS components. The use of such statements is to be avoided at all times.

Your total cooperation in seeing that these suggestions are followed will be appreciated.

Hank O'Hara

/ej

9P258-0775-400

intel

ALTAIR SERVICE DEPT.

Barbara Sims

Well here we are once again! Our schedule for the newsletter seems to be getting much better wouldn't you say? I have a couple of new suggestions to pass on to you that will help us out tremendously.

If you need to send a unit or board in for repair, it would help considerably if you'd include the name the unit was purchased under (i.e. surname, company or school name). Also we need an enclosure describing the problem(s) you're having with the unit. Due to the amount of mail coming into our office, it is very difficult to match repair shipments with letters that come in separately. If our service department does not have a complaint to work from, it takes a little longer to locate problems.

Some of our users have received defective parts. We'll be glad to replace anything that is defective, but you need to send the defective part in first, then we'll get a good one out to you right away. Again you need to enclose a note and always show the name the 8800 was purchased under.

Our software library is coming along. We still need entries,though. As of September 1, 1975 we will be putting a new policy into effect. Your cooperation will be appreciated. No more programs will be acceptable unless they are type written on our software program format (sample enclosed). It is too difficult for us to keep up with typing the programs for printing. The programs must also be on good quality paper 81/2 by 11" (not onion skin or any light weight paper.) The typewriter ribbon used must be black with a good contrast. Without these stipulations, our printer will not run copies of your program. In addition make sure that you keep a copy of your program on file, as we cannot return programs that are not accepted into our library.

We sent out software license agreements for BASIC to all our Altair customers. If you have ordered software, or order it in the future, please return the agreement as soon as possible. We cannot ship your software without the agreement in house. Also, it would help me out if you'll fill in the back page with what programs or equipment you own or have on order. If you are using our software with equipment from other sources, we'd like you to list that equipment also.

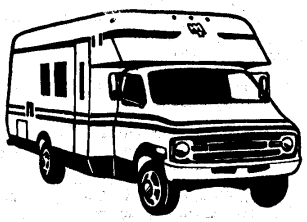
There has been a suggestion from a user that a radio communication be set up for ham operators. If any of you operate ham radios and would like your call number published, send it in and we'll try to get them in our next newsletter. You'll have to get it set up yourselves, but we'll help out all we can from here.

I guess that's it for this month. Take care!

Barbara

new MITS secrete #
use it when you can't get through or when you just plain want fast access!
505-262-1951

MITS-MOBILE CARAVAN SCHEDULE



note: All seminars are being held at Holiday Inns

AUG.11, Amarillo,Texas; #3 I-40 on I-40 at Ross.
AUG.12, Oklahoma City,Oklahoma; #5 South; I-35 at S.E. 29th St.
AUG.14, Tulsa,Oklahoma; #1 West; US 66,Alt.75; I-44 and Oklahoma 33, exit at Turner Turnpike.
AUG.15, Fort Smith,Arkansas; #2 North; US 64 and 71 North; exit at Grand Ave., I-540.
AUG.18, Little Rock, Arkansas; #2 North; I-40, US 67 and St.107.
AUG.19, Nashville, Tennessee; #2 Southeast; US 41-S, take Murf. Road exit from I-24.
AUG.21, Knoxville, Tennessee; #1 Northeast; US 70, 11E and 25 W. Ashville Highway exit.
AUG.22, Greensboro, North Carolina; #4 Airport; I-40 and 68, (Greensboro/Winston-Salem).
AUG.25, Raleigh, North Carolina; #1 North; US 1 and 401 N.
AUG.26, Charlotte, North Carolina; #4 North I-85; take I-85 at Sugar Creek exit.
AUG.28, Columbia, South Carolina; #1 Southwest;US 21, 176 and 321, five blocks off US 1.
AUG.29, Augusta, Georgia; US 1, 78,278 at 25 S.
SEPT.2, Atlanta, Georgia; #7 at Six Flags West; 1.6 miles from the I-285 interchange.
SEPT.4, Birmingham, Alabama; #4 South; US 31 and I-65.
SEPT.5, Huntsville, Alabama;University Drive, US 72 and ST 53.
SEPT.8, Memphis, Tennessee; #1 Holiday City; US 78, I-240 and Getwell Road to Holiday City.
SEPT.9, Shreveport, Louisiana;#3 at Bossier City; off I-20 take the Hamilton Lane exit.
SEPT.11, Baton Rouge, Louisiana; #3 West; I-10 and Highway 415.
SEPT.12, New Orleans, Louisiana; #8 Airport;take Williams Blvd. exit from I-10.
SEPT.15, Mobile, Alabama; #1 West; Exit 90 East off I-10 from the west or West off I-65 or I-10 from the east.
SEPT.16, Tallahassee, Florida; #1 Apalachee; US 27 and Magnolia Dr.
SEPT.18, Jacksonville, Florida; #6 South; I-95 at Emerson Street exit.
SEPT.19, Titusville, Florida (Kennedy Space Center); Orlando/ Cape Canaveral; take US1 South exit I-95 and Florida 50 on Indian River
SEPT.22, Titusville, Florida; same location as above.
SEPT.23, Tampa Florida; #3 I-4 East; Tampa/St. Petersburg; exit I-4 and US 41 at 50th Street.
SEPT.25, Miami, Florida; #9; Central I-95 on I-95 at the 79th Street exit.

Hands-on demonstrations begin at 6 p.m. and all seminars begin at 7 p.m.

Letters to the Editor

Dear Editor:

Thank you for the surprise package I received last week consisting of the Computer Notes, the People's Computer Company and the Computer Hobbyist, all of which I enjoyed thoroughly. I especially liked the note on the 4K RAM board regarding the data loss problem when using the reset switch and the hint about using the EI and the unconditional jump to the EI to indicate end of program on units without any I/O devices. I had experienced both of the above problems and it was reassuring to find out that they were not something that I had "done" (or not "done") during the construction of my Altair.

I would also like to take this opportunity to express my admiration to your company for the excellent (and I've built quite a few) quality of the Altair kit. I can tell that you people really care about your customers! I especially like the openness with which you admit mistakes, and we all make them.

Please add my name to your list of those satisfied Altair owners and publish it in any future directories.

Sincerely yours,
Ross F. Housholder
1725 Brooks Dr.
Arlington, Tex. 76012

Dear Editor:

I was very glad to see your "Computer Notes"-- an excellent idea. Many of us Altair users will be programming with machine language, at least for awhile, so perhaps this publication can provide us with certain programming hints, arithmetic subroutines in particular. You demonstrated simple addition but I'm lost when it comes to manipulation of negative numbers, division, multiplication, numbers greater than 28, decimal points, etc. Perhaps you could help out here, via "Computer Notes." I would like to see a demonstration of the "ADC" and "SSB" instructions.

Issue #1 indicated users names would be published in future issues. This is also to authorize use of my name. Thanks muchly.

George Markel
505 Cypress Point Dr., Apt.38
Mountain View, CA 94043

Dear Editor:

Thank you for the excellent issue of your "Notes". It is filled with helpful notes--even for someone well up on the Intel 8080. Page after page the reaction is high.

Re Hint #1 on Page 8, I note that occasionally (i.e. in the miscalled "general case") one needs to preserve contents of A through the count test and the simplest way is to revive the "count-up" method one had to use in the old 8008 to step through memory (using a fast RST routine a la your Hint #2), thus

```
Lxi B, count; 2's complement
LOOP:
......
......
INR C
JNZ LOOP      8 bytes!
INR B
JNZ LOOP
```

Of course, using RST and ... RNZ...RET one got this in only 4 bytes (Hint #2).

This is a quite trivial point. Thanks again-- and can you send the next issue?

Sincerely,
Donald V. Weaver
Jamaica, NY 11435

Dear Dave,

I would like to enter your software competition. While I don't have a program or sub-routine as such, what I do have is an additional software package that extends the range of applications of the Altair beyond anything now envisioned.

A brief write-up of my offering, complete with suggested pricing structure, is attached.

Wendell S. Rice
Chief Engineer
Data Documents Systems Corp.
Merriam, KS

Software Package No. 69

Altair SUPER EXTENDED BASIC -- $1495.00

When purchased with an Altair, 42K memory and either a duplex I/O board and 4K of write-only memory, you have our deepest sympathy.

INSTRUCTION STATEMENTS:

CCS	Chinese Character Set
BH	Branch and Hang
BSO	Branch on Sleepy Operator
DO	Divide and Overflow
RPB	Reverse Parity and Branch
ARZ	Add and Reset to Zero
WWLR	Write Wrong Length Record
SRSD	Seek Record and Scar Disc
RC	Read Chaos
TDB	Transfer and Drop Bits
EROS	Erase Read Only Storage
UER	Update and Erase Record
CM	Circulate Memory
MWM	Move and Wrap Memory
DIA	Develop Ineffective Address
LMB	Lose Message and Branch
SC	Scramble Channels
LC	Loop Continuous
BIM	Branch on Index Missing
CD	Create Data
WOS	Write Only Storage
BLI	Branch and Loop Indefinite
HCF	Halt and Catch Fire
BBI	Branch on Burned-out Indicator
BPO	Branch and Power-off
II	Inquire and Ignore
AI	Add Improper
SRZ	Subtract and Reset to Zero
RI	Read Invalid
WNR	Write Noise Record
ED	Eject Disc
EIOC	Execute Invalid Op Code
RNR	Read Noise Record
DSP	Destroy Storage Protect
MDB	Move and Drop Bits
MLR	Move and Lose Record
MC	Move Continuous
RT	Reduce Thru-put
IOR	Illogical "OR"
IAND	Illogical "AND"
UCB	Uncouple CPU's and Branch
EO	Execute Operator
RBG	Random Bug Generator (Special Feature)
IIB	Ignore Inquiry and Branch

Listed below are the names and addresses of Altair users who have given us permission to print their names and addresses. Already, a number of users have corresponded with us about starting a local Altair Users Clubs in their areas. If you'd like to have your name published in the next Computer Notes write or call us giving your permission.

Dr. Jack Crenshaw
Comp-Sultants
P.O. Box 1016
Huntsville, AL 35807

John D. Turner
Navseeact Japan
Box Five
FPO Seattle, WA 98762

Robert C. Rae
1921 Dogwood Lane
Vienna, VA 22180

Agoris Home-Made Candies
Demo Agoris (owner)
153 W. Pike St.
Houston, PA 15342
(412) 745-6670

Thomas L. Dixon
P.O. Box 5086
Santa Monica, CA 90405

Manley Nichols
Box 344
Henning, MN 56551

Charles Merritt
1618 Randolph
Topeka, KS 66604
(913) 233-8785

Paul Potter Reinhardt II
1120 Folkstone Apt. #6
San Mateo, CA 94402

ACR UpDATE INFO

by Tom Durston

Here is some information that we hope can be useful to 88-ACR owners.

First, ACR stands for Audio Cassette Recording, a method using audio frequencies to record digital information. However, you are not limited to using only cassette recorders for data storage. Any type of recorder with wow and flutter less than 0.5% may be used. If you plan to transfer tapes between machines, be sure tape speed is within 0.5% or you will have to record the test signal (125s) for speed correction alignment of the demodulator (R29). At the end of this article is an easy alignment procedure for aligning your ACR with a voltmeter which is just as accurate as using an oscilloscope and easier than using the input test program.

If you are getting basic on cassette, we recommend a music quality, A.C. only cassette recorder to guarantee speed accuracy and stability. We have found that Radio Shack offers a line of good cassette recorders as well as high quality cassette tape. We have used the model SCT-5C deck and the CTR-20B portable recorder with very good results.

The reason for requiring stable and accurate tape speed is due to the frequencies used (2025HZ=Logic 0,2225HZ=Logic 1). The reason they are used is because they are also used in 103 type telephone modems for computer communications. A tape that is recorded on a recorder that is 1% fast, and played on a recorder that is 1% slow gives a 2% speed difference which is a 40HZ frequency change. Since the frequency difference between Logic 1 and Logic 0 is 200 HZ, the error signal is 40/200 HZ or 20%. This does not include instantaneous changes in speed (wow and flutter).

If the total error in transferring tapes between recorders is less than 20% there is no problem; if it is greater than 20% you will have to readjust the demodulator as described next.

To align your 88-ACR, we have found a new, easy, and accurate method utilizing a voltmeter. The principle being applied is that a D.C. voltmeter reads average voltage, and if you are measuring a square wave, you will read half the peak to peak voltage.

STEP 1: When the output test program in the 88-ACR manual is run (with or without recorder) the voltage on pin 25 (Transmit data) of the UART (I.C."M" of the 88 SIOB) should read about 2.5 volts DC, halfway between Logic 0 (0 volts)and Logic 1 (+5 volts). Note the exact voltage.

STEP 2: A). Deposit in memory:

ADDR	DATA	
000	333	Input to ACC
001	007	88-ACR Data Addr.

B) Examine 000
C) Single Step twice

This should allow you to examine data from I O port #7. Data lights 0-7 will indicate the data being received from the recorder (when properly adjusted). This process may be used to examine data or status information from any I/O board- just change the address number.

STEP 3: Play the tape with the output test program recorded on it. Slowly adjust R29 so that the recorded data (125) shows on the data lights. Now measure pin 20 (receive data) of the UART (I.C."M" of the 88-SIOB) and carefully adjust R29 to the voltage noted in Step 1, about 2.5 volts DC. This adjusts the demodulator to the speed of the tape being played. If you recorded and played the tape on the same recorder, then it is adjusted to record and play at that tape speed.

If you have to make this adjustment to play a tape from a different recorder (i.e. the Basic type), it will work only for tapes recorded at that different speed.

It should be readjusted for tapes being recorded and played on the same recorder such as the C load and C save functions.

STEP 4:To adjust the two filter pots, R9 and R13, play the tape with the output test program recorded on it and measure point "TP" on the modem board (pin 6 of IC "B") with the voltmeter on a low voltage A.C. range (10-15 volts.)

The voltage measured with the amplifiers clipping should be about 5 v RMS. Reduce the recorder volume until the voltmeter reads 2 or and R13 for maximum reading. Reduce the volume again, if necessary, to keep the reading below 4 volts RMS. Go back and adjust again to get maximum peak. Note that this is not a critical adjustment. Turn the volume back up to max, and go back and recheck the adjustment in step 3.

This completes alignment of the 88-ACR. If it requires readjustment to play data recorded on another machine, do steps 2 and 3.
The MITS Altair Basic cassette has the test recording on the beginning of the back side. If you are using the Basic cassette, be sure your 88-ACR is wired for 300 Baud and address 6 and 7.

HARDWARE

8800 MOD

(1) DEPOSIT PROBLEM (D/C board)
SYMPTOMS: Machine won't deposit at all or deposits all ones when using the front panel deposit switch.
FIX: Change the timing capacitors for the deposit single shot (IC G on the Display/Control board). Change C7 to 0.01MF, C8 to 0.1 MF.

(2) 12 VOLT ZENER (CPU board)
SYMPTOMS: Zener running very hot.
FIX: Change R46 on CPU board to 43 ohms (Use either a 1W or 2W resistor). This should be done only if there are four or less cards on the bus. If there are more than four cards R46 should be 33 ohms.

(3) AC SWITCH (Display/Control board).
PROBLEM: The tracks on the D/C board which connect the AC switch terminals to the pads where the switch wires are connected to the board have 115VAC on them. If these are inadvertently shorted to other tracks on the board several IC's are wiped out.
FIX:(a) remove the AC switch lines from the D/C board.
(b) Cut the tracks leading from the pads where the switch is mounted to the pads where the AC switch wires attach to the D/C board. Cut the tracks as close as possible to the switch pads so there will be no length of trac with 115VAC on it.

(c) Slide about 1" of heat shrink tubing (3/32 to 7/64 diam.) over the AC switch wires. Solder these wires directly to the AC switch terminals. Slide the heat shrink tubing down so it covers the switch terminals and uninsulated ends of the switch wires. Heat the tubing once it has been properly positioned to keep it in place.

(4) CLOCK SPECS (CPU Board)
(a) Phase 1 pulse width (measured at 90% points), 60 nanoseconds minimum.
(b) Phase 2 pulse width (measured at 90% points), 220 nanoseconds minimum.
(c) Delay from leading edge (90%) of Phase one to leading edge (10%) of Phase 2. 130 nana-seconds minimum.
(d) Delay from trailing edge (10%) of Phase 2 to leading edge (10%) of phase 1. 70 nano-seconds minimum.

MAINTENANCE

by Paul Van Baalen

A good percentage of the people who have had trouble with their bits have shown this symptom. "All the data lites on at all times". Below is a list of the most common causes:

A. Check the mother board for shorts using VOM.

B. Insure that all regulated voltages are OK.

C. Make sure the memory is installed and the address strapping is correct. Be sure that the connector from the front panel for the data lines is on the CPU.

D. Are all the data lines high on the D/C board? If they aren't, check the continuity of the CPU connector.

E. Check the timing of 01 &02.

F. Check the drivers J&H on the static memory board. These tri-state drivers need a low on pins 1&15 to pass a signal. If 1&15 are high the outputs of these drivers will be high all the time.

G. See if line 68 on the bus is high all the time. If it is, backtrack thru the logic. Most probable cause is ICG on the front panel being bad or a short or solder bridge in the vicinity of ICG.

H. Check IC's U&T on D/C board to insure the signal levels are correct,i.e.; no 1.5 volt levels as opposed to $\leq$.5V or $\geq$4+VDC.

Boo-Boo's

The Status Signal $\overline{WO}$ is incorrectly labelled on the front panel as WO.

There is a problem with the protect operation on the 4K RAM board. Pin 10 of ICT is tied to +5V and should be tied to ground. The easiest way to fix this is to lift Pin 10 of ICT and run a short jumper from the lifted Pin to Pin 11 of ICT which is tied to ground.

BYTE BYTE BYTE

BYTE magazine is out and it is as real as everyone imagined.

Subtitled "the small systems journal" BYTE is in many ways a sigh of relief for those computer freaks who were losing their eyesight trying to read all of the club newsletters.

What I mean is that BYTE is a very professional magazine. It looks nice and it is printed on high quality magazine stock that won't fall apart in time like many publications.

It is interesting to note that there are four columns of type on each editorial page and that it is right justified with no breaks between words. The purpose is to make BYTE convenient to read (especially for you speed readers out there).

OK, let's give BYTE an A+ for format--now how about content.

According to editor Carl Helmers in the first BYTE editorial, BYTE is "a monthly compendium of information for the owners and users of the new microcomputer systems becoming widely available at moderate costs."

Helmers goes on to explain that the content of BYTE is divided into the trilogy of hardware, software and applications. All well and good.

There are three very good articles in BYTE which I think live up to the trilogy concept-- they are WHICH MICROPROCESSOR FOR YOU?? by Hal Chamberlin, WRITE YOUR OWN ASSEMBLER by Dan Fylstra (with some super illustrations), and LIFE Line by Carl Helmers.

Chamberlin's article first describes then compares the various aspects of the Intel 8008, Intel 8080 and National IMP-16 microprocessors. While there are, of course, many points of possible disagreement, Chamberlin demonstrates that he knows what he's talking about and that he knows how to organize and write his thoughts in a legible fashion (perhaps something he learned writing all those good articles for The Computer Hobbyist.)

Fylstra's article describes in general terms what assemblers do, scanning techniques, symbol tables, hashing methods and advanced "bells and whistles". While I wouldn't recommend writing your own assembler (unless you are an advanced hobbyist) this is a good article for understanding the assembler and how it works.

LIFE Line is the first of a series of articles about the computer game, LIFE. Besides describing the concept of LIFE (better than I've seen it explained elsewhere) Helmers is using this series as a vehicle to explain the art of programming with "real hardware and software systems". The goal of this series is an inexpensive hardware/software system for the home brew computer builder.

These three articles, I think, exemplify an attempt to make BYTE an exciting publication.

The rest of the articles in BYTE (with the possible exception of SERIAL INTERFACE by Don Lancaster) aren't nearly as well written or as topical as the above, but then they aren't too bad. (Carl Mikkelsen wrote an interesting article about recycling used IC's that convinced me you have to be half mad to attempt recycling used IC's).

All in all I'd give BYTE an A- for content (they have to have something to work for).

Is BYTE magazine something no computer hobbyist should be without? At this point in time, I'd have to say YES. --DB

July Software Contest Winners Announced

by Bill Gates

The Altair software library has doubled in size with the acceptance of fifteen more programs into the library. Memory test programs seem to be the most popular. We have four already, and I plan to submit the program we use in-house to test the 4K boards. An exhaustive memory test should check for side affects, that is checking to make sure writing into one cell didn't affect another.

The second most popular type of program is a program to relocate other programs in memory. We received two this month, bringing the total to three. When the software library gets sufficiently large we will organize the programs into different groups, like test programs, game programs, BASIC programs, etc.

Two BASIC programs were submitted this month. A number of users who have BASIC up and running on their machines have told me they are writing their own special game program which they plan to submit to the library, so I expect a lot more Basic programs to be added to the library in the near future.

The winning major program this month ($50) is TIC-TAC-Toe (#721751) written by Gary Tack. The switches are used for input and the lights for output. The computer always goes first, and never loses. The best you can do is draw. The computer's complete strategy is shown in a diagram that comes with the program.

The second place prize ($25) goes to John Klein for his BASIC program that plays Blackjack (#730751). This is one of the first BASIC programs written specifically for Altair BASIC and it uses some of the nice features of 8K BASIC. The program is testimony to the fact that someone who has no previous experience with BASIC can pick it up and write a large, fun program in less than a week.

The first program I thought of writing when I got my Altair was one to display patterns in the lights. What gets displayed on the address lights when a program is running depends on the instruction being executed. A little experimentation to figure out exactly what gets displayed for each instruction is fun and takes only a short time. The major program winning third place ($15) is FLASH (#722751) written by Mathew Smith. FLASH is a demonstration program that uses the lights for output. When it is run the lights count up and then down, but at varying rates of speed. Different initial patterns can be used to change the output.

The most active contributor to the library so far has been Lee Eastburn. His sort program (#616751) is the winning subroutine this month.($25). It is a bubble sort, where two cells are exchanged when they are in the wrong order, and the data is scanned to find cells out of order until a complete scan is made during which no cells are found to be out of order. The size of items to sort and the sort field within the item can easily be changed.

The second place winner($15) RND (#722751), written by Roger Gulbranson, is a random number generator. This is an essential subroutine for any game program involving chance. A 16-bit number is generated using a generalized feedback shift register algorithm. This program can easily be modified to generate random numbers of any size.

Here is a list of programs accepted into the library since the last newsletter, along with a brief description:

#722751
Author: Roger Gulbranson
Length: 35-80 bytes depending on how much data is used.
Generates 16-bit random numbers using generalized feedback shift register Algorithm.

#730751
Author: John Klein
Length: About 200 lines of BASIC; plays Blackjack.
Full instructions given with the program.

521751
Author: Dennis Bahr
Length: 25 bytes
RAM memory test program. Tests all locations between a lower and upper bound with all possible values.

#721753
Author: Mathew Smith
Length: 44 bytes
FLASH displays patterns in the lights. Good demonstration program.

#721751
Author: Mr. Tack
Length: 240 bytes
Plays tic-tac-toe. Uses the switches and lights.

#725752

Author: Daniel Lomax
Length: 31 bytes
Program that finds out where it is in memory and then relocates itself to the page in the switch register.

#724751
Author: Billy Wood
Length: 40 bytes
Ram memory diagnostic. Goes through memory testing locations with all bits on, and one off and vice versa.

#717752,#71753
Author: Dr. George Haller
Length: 30 bytes each
One program does an ascending sort and the other a descending sort. Both use a bubble sort. The lists to be sorted consist of single bytes, all of which are used as the sort field.

#722751
Author: Herb Archer
Length: 35 bytes
Ram test program. Two different bit patterns are stored in each memory cell and checked for accuracy.

#717751
Author: Samuel Cook
Length: 22 bytes
The interrupt light on the front panel is turned on and off in a loop that gets smaller and smaller, starting at 1.8 seconds.

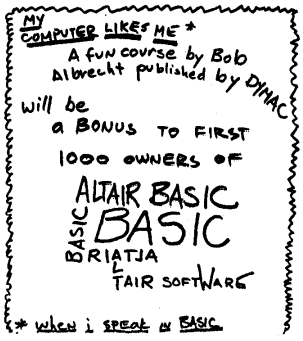
#711751
Author: L.M. Eastburn
Length: 180 bytes
Program to relocate other programs. Modified instruction operands when appropriate.

#714751
Author: Donald Tork
Length: 33 bytes
Loads data from an ASCII device into memory.

#630751
Author: Jim Babock
Length: Basic, about 250 lines
Plays famous Las Vegas card game.

#728752
Author: J.R. Scott
Length: 36 bytes
Translates some Bandot character codes to ASCII equivalent.

#616751
Author: L.M. Eastburn
Length: 250 bytes
Sort program using bubble sort. Allows specification of data size and sort field.



SOFTWARE

Q and A -

MONITOR EDITOR & ASSEMBLER
BY PAUL WASMUND

Here are answers to some frequently asked questions about the monitor, editor and assembler. Any other questions you may have regarding these programs should be sent to Paul Wasmund.

Q. Does the monitor have any kind of file structure to eliminate the need to store each program on a separate audio cassette?

A. Yes, when files are created by either the editor or the assembler, a three-character file name is output as a header, along with the rest of the file. The monitor has commands that are used by the editor and assembler to search for such a named file.

Q. What devices are presently supported and what devices will be supported in the near future?

A. At present the teletype, compter and audio cassette are the only supported devices. We should have handlers for the line printer and disk shortly.

Q. Can I add other I/O handlers to the monitor to support specific devices I own?

A. Yes, room has been left in the monitors device table for the addition of other device handlers.

Q. Does the monitor support interrupts?

A. No, the monitor currently does not have any facility to handle interrupt I/O.

Q. How useful is the system monitor?

A. The monitor gives the programmer complete freedom from front panel manipulation, except in the case of a program infinite loop, and it allows the editor and assembler to use device independent I/O routines. If the user needs the memory used by the monitor he can write his own loader to load programs into the area previously used by the monitor. However, the text editor and assembler require that the monitor be resident when they are used.

Q. What is the capability of the editor?

A. At present the only commands that edit text are the insert, delete and replace commands. We plan to add an interline edit command, a string search command, and extend the three present commands to work with more than one line.

Q. In an 8K machine how much memory do I have for programs if I run them only with the monitor?

A. You have slightly better than 6K of free space with only the monitor in memory.

Q. Does the assembler generate relocatable code?

A. No, the assembler generates absolute code. It has a feature allowing the programmer to assemble programs for execution at any location in memory. Using this feature you can assemble a program to run in the memory the assembler normally resides in.

Q. Since the paper tape reader on the teletypes sold by MITS cannot be controlled, how are assembly listings generated?

A. During assembly error messages are echoed in place of the 2 characters that would normally be echoed. At the end of assembly a symbol table is printed, which gives the location of all symbols used, a list of all symbols that are still undefined is output, and an inverse assembly listing can be generated.

Q. What is an inverse assembly?

A. The assembler has a program in it which goes through the memory just assembled into, and for any byte that it recognizes as an instruction it will output the mnemonic for that instruction along with the location that the instruction is in.

GENERAL SOFTWARE

by Paul Allen

Q. Will MITS trade hardware for software? I'd like to write a FORTRAN,APL,etc. if MITS will give me a machine!

A. It is MITS present policy not to trade hardware for software or other services. Sorry. If you do develop software once you've purchased a machine, MITS will be glad to examine it for possible purchase or for license to MITS customers.

Q. What kind of programs are you looking for?

A. High level language (FORTRAN, APL,RPG) compilers or interpreters for applications programs written in BASIC.

Q. Do you know yet how fast the improved floating point routines are?

A. Yes. Version 3.0 of Altair BASIC will be about twice as fast as version 2.0. A FOR I=ITO 10,000: NEXT loop takes about 14 seconds. A floating multiply takes about 5 milliseconds and a divide takes about 8 milliseconds. These last two times may vary significantly depending on the operands.

Q. Any other differences in 3.0?

A. Yes, the 8K version is not only faster, it is also about 100 bytes shorter, due to many long hours spent by our programmers optimizing the assembly code. No new features are present in 3.0. The 4K 3.0 is the same size as 4K 2.0 but it is two times faster.

Q. When will 3.0 be shipped to customers?

A. In early August. Customers who have already received 2.0 may obtain 3.0 for a copying charge of $15.

Q. Do you need the system monitor or text editor to run BASIC?

A. No, BASIC has its own built-in program editor and I/O routines.

Q. Do you plan to offer applications

A. Some BASIC game programs have already been submitted to the users group. We plan to have some business applications packages (accounts receivable, payable, payroll, etc.), available before the end of the year. Precise prices and release dates are not yet available.

Q. How does Altair BASIC compare to other minicomputer BASICs?

A. Altair BASIC has more features per byte of memory than any other BASIC we know of.

Q. Do you need the DOS to run extended BASIC?

A. No. Both extended BASIC and the DOS use the same file structure and disk driver. Extended BASIC can run without the DOS and vice versa. Extended BASIC is designed for BASIC programming development in a disk enviornment, and the DOS is designed for assembly language programming development.

Q. What are your development plans at the moment?

A. We are working full speed on the extended BASIC and DOS. The 4K and 8K versions are essentially "frozen"; that is, we don't anticipate changing them except to make bug fixes.

DEBUG

DBG-8800 is a powerful debugging tool for the ALTAIR 8800 computer. It requires 2K of memory, and any other memory may be used for the assembly language program being debugged. The DBG-8800 may be used to:

1) Display memory locations, registers, flags in any of the following modes:
   A.) Octal
   B.) ASCII
   C.) Decimal
   D.) Symbolic Instruction format (i.e. XRA A)

2) Modify memory locations, registers, or flags using the four I/O modes.

3) Set, display, or remove breakpoints in the program being debugged. Up to 8 breakpoints may be set at one time.

4) Begin or resume execution of the program being debugged either of a specified location or from the most recently encountered breakpoint.

DBG-8800 is available with complete documentation. The object binary and documentation cost $25 (order as DBGOBJ). The source code costs $75 (order as DBGSRC). Please specify paper tape or cassette when ordering. Delivery is 45 days.

NOTE: DBG-8800 resides in the top 2K of 8K. It uses an RST instruction for breakpoint

FUN WITH ALTAIR BASIC

by Monte Davidoff

After loading BASIC, you may be ready for a little enjoyment from your Altair. Here are three short programs that illustrate some of the things that are easy to do in 8K Altair BASIC.

The first program wants two strings S$ and W$. The program finds all the occurances of W$ as a substring. (i.e. it finds all the places in S$ that are exactly the same characters that are in W$.) For example, if S$="ABCDAB" and S$="AB" then the program would say "AB" was found in S$ starting at characters 1 and 5. Now let's look at the program.

The first thing we do is input S$ and W$. Notice that we can print text in an INPUT statement also. The next thing the program does is set the variable CN to Zero. (Remember you can have two character variable names in Altair 8K BASIC). CN will be used to count how many times W$ is found in S$. Next, we use the CHR$ function to set Q$ to a string one character long. 34 is the ASCII code for a double quote. Next we use the concatentation operator "+" to put quotes around W$ and S$. This is just so the printouts will look nice. Next comes the meat of the program. I ranges from 1 to the number of characters in S$. The MID$ function will return as many characters out of S$ as there are in W$. If these are equal to W$, we have found our substring. So, we add one to the count, CN, and print a message.

Remember that if the IF statement is true, the statements following the THEN will be executed. If the IF statement is false, we skip the statements on the same line and start at ethe beginning of the next line. In 8K Altair BASIC, if there is no variable given in the NEXT statement, BASIC will assume the NEXT is for the most recent FOR loop. Line 90 just prints out how many times we found W$ in S$. This program is just intended to show what kinds of things can be done with the string functions such as MID$ and LEN.

The next program shows how you can use STR$ to make your output look nice. This program produces a table of Pascal's Triangle. This table is used in probability. It is formed by starting with a one at the top and then adding the two numbers that appear to the upper right and left to form an entry in the next row. Notice the NEXT J,I in line 60 to end both for loops. The trick comes in printing the thing out to look like a triangle. The STR$ function is used to convert the number to be printed to a string. Then the LEN function is used to find out how many characters are needed to print the number. These are added together in line 90 to see how many characters will be used to print all the numbers in the line. In line 100 we find how many spaces to print between each number. We have 72-LL positions to fill, and we want to divide the number of spaces into groups. .5 is added to round to the nearest integer. Line 120 then prints the row out using the SPC function to print the correct number of spaces. The last NEXT in line 120 ends the FOR in line 80,so we repeat this procedure for each line in the triangle. The right side does not look as nice as the left side. This is because the terminal can only print a character at discrete positions on the paper. However, the program still shows the general idea of using the STR$ function.

The third program is intended for people who already know about BASIC on an advanced level. While BASIC is a simple language, it still has the power to do complicated things. This program calculates N factorial (N!) where n!= Nx (n-1)x (N-2)... 3x2x1 and 0!=1. So 5!= 5x4x3x2x1= 120.

The program uses the fact that 0!=1 and N!=N.(n-1)!

The FOR loop in line 20 just sets up things so a table can be printed in two columns. The subroutine in line 60 prints out the result. The STR$ function is used to make the output look nice by having the exclamation mark come immediately after the number.

Now for the tricky part. The subroutine in line 40 calculates F=N!. In line 40, if N=0 then we know F should be one and we RETURN because we are done. If N is not zero, Line 50 decreases N by one, and then GOSUB'S to line 40 again to calculate (N-1)! The amount of times GOSUB'S can be executed with out a RETURN is limited only by how m much memory you have. I ran these programs on a machine with 8K. Anyway, when the subroutine returns in line 50, the N=N+1 updates N to the value it was before the GOSUB, F=N*F updates (N-1)! to N! and then we return. If you are confused about this program, you might want to print out the values of N and F at line 40 before the IF statement. (i.e. type: 40 PRINT N,F: IF N=0 THEN F=1: RETURN). If confusion still prevails, do not worry about it. There are very few BASIC programs in which this type of trick is used. If you want to run this program in 4K Altair BASIC, change line 60 to: 60 PRINT N;"!="F,:RETURN

---------- Having Fun With Altair BASIC--Program One ----------

```
RUN
GIVE ME A SENTENCE? THE RAIN IN SPAIN FALLS MAINLY ON THE PLAIN
GIVE ME SOME CHARACTERS? THE
"THE" STARTS AT 1
"THE" STARTS AT 35
"THE" OCCURS 2 TIMES IN "THE RAIN IN SPAIN FALLS MAINLY ON THE PLAI

GIVE ME A SENTENCE? I LIKE APPLES COOKED IN TURPENTINE
GIVE ME SOME CHARACTERS? ELEPHANT
"ELEPHANT" OCCURS 0 TIMES IN "I LIKE APPLES COOKED IN TURPENTINE"

GIVE ME A SENTENCE? A MAN LOOKED AT A CAT
GIVE ME SOME CHARACTERS? A
"A" STARTS AT 1
"A" STARTS AT 4
"A" STARTS AT 14
"A" STARTS AT 17
"A" STARTS AT 20
"A" OCCURS 5 TIMES IN "A MAN LOOKED AT A CAT"

GIVE ME A SENTENCE?

OK
LIST

10 REM SEARCH FOR A CHARACTER IN A SENTENCE
20 INPUT "GIVE ME A SENTENCE";S$
30 INPUT "GIVE ME SOME CHARACTERS";W$
40 CN=0 : REM CN=NUMBER OF TIMES W$ OCCURS IN S$
50 Q$=CHR$(34) : SQ$=Q$+S$+Q$ : WQ$=Q$+W$+Q$
60 FOR I=1 TO LEN(S$)
70 IF MID$(S$,I,LEN(W$))=W$ THEN CN=CN+1 : PRINT WQ$ " STARTS AT" I
80 NEXT
90 PRINT WQ$ " OCCURS" CN "TIMES IN " SQ$ : PRINT
100 GOTO 20
OK
```

--ALTAIR USERS--
continued from page 4

Jonathan Griffitts
1030 Adams Circle
#205
Boulder, CO 80302

J.W. Klotz
73 Red Ridge Rd.
Levittown, PA 19056

Rabkin Realty Corp.
c/o Morton Rabkin
1028 Hatch St.
Cincinnati, OH 45202

Bruce Sidlinger
6201 Airline Rd.
Dallas, TX 75205

Ron Santore
1957 Huasna Dr.
San Luis Obispo, CA 93401

Grant E. Johnson
2322 West 13th St.
Ashtabula, OH 44004

Walt Gerard
15 Eastridge Dr.
Council Bluffs, IA 51501
(712) 322-7761 Office
(712) 328-0634 Home

Edwin F. Hampton
P.O. Box 45629
Tinker AFB Branch
Oklahoma City, OK 73145

Fred Patterson
5711 Cornick Rd.
Norfolk, Va. 23502

Howard Klein,DDS
3914 103rd St.
Corona, NY 11368

Daniel Lomax
Community Data Systems
203 E. Congress St.
Tucson, Az. 85701

Gordon Scale
152 Glebe Ave., Apt.3
Ottawa, ON Canada

2Lt. L.R. Sears III
COB,25th Signal Bn.
Ft. Bragg, NC, 28307

George Markle
505 Cypress Point Dr., Apt.38
Mountain View, Ca. 94043

Eric A. Bardeen
Advanced Precision Industries Inc.
870 North Davis Dr.
Warner Robins, Ga. 31093

Ross F. Housholder
1725 Brooks Dr.
Arlington, Tex. 76012

William D. Roch
5133 Catalon Ave.
Woodland Hills,CA 91364

---continued on page 12

HAVING FUN WITH ALTAIR BASIC--Program Two

```
RUN
HOW MANY ROWS? 9
                    1
                  1   1
                1   2   1
              1   3   3   1
            1   4   6   4   1
          1   5  10  10   5   1
        1   6  15  20  15   6   1
      1   7  21  35  35  21   7   1
    1   8  28  56  70  56  28   8   1
  1   9  36  84 126 126  84  36   9   1
OK
LIST
10 REM PRINT PASCAL'S TRIANGLE LIKE A TRIANGLE
20 INPUT "HOW MANY ROWS";R : R=R+1
30 DIM P(R,R)
40 REM GENERATE PASCAL'S TRIANGLE
50 P(1,1)=1 : REM INITIALIZE THE TRIANGLE
60 FOR I=2 TO R : FOR J=1 TO I : P(I,J)=P(I-1,J-1)+P(I-1,J) : NEXT J,I
70 REM PRINT THE TRIANGLE
80 FOR I=1 TO R : L1=0
90 FOR J=1 TO I : L1=L1+LEN(STR$(P(I,J))) : NEXT : REM L1=LINE LENGTH
100 SP=INT((72-L1)/(I+1)+.5) : REM SP=NUMBER OF SPACES BETWEEN NUMBERS
110 REM PRINT A ROW
120 FOR J=1 TO I : PRINT SPC(SP);STR$(P(I,J)); : NEXT : PRINT : NEXT
OK
```

HAVING FUN WITH ALTAIR BASIC--Program Three

```
RUN
 0! =  1
 1! =  1
 2! =  2
 3! =  6
 4! =  24
 5! =  120
 6! =  720
 7! =  5040
 8! =  40320
 9! =  362880
 10! =  3.6288E+06
 11! =  3.99168E+07
 12! =  4.79002E+08
 13! =  6.22702E+09
 14! =  8.71783E+10
 15! =  1.30767E+12
 16! =  2.09228E+13
 18! =  6.40237E+15
 19! =  1.21645E+17
 20! =  2.4329E+18
 21! =  5.10909E+19
 22! =  1.124E+21
 23! =  2.5852E+22
 24! =  6.20448E+23
 25! =  1.55112E+25
 26! =  4.03291E+26
 27! =  1.08889E+28
 28! =  3.04888E+29
 29! =  8.84176E+30
 30! =  2.65253E+32
 31! =  8.22284E+33
 32! =  2.63131E+35
 33! =  8.68332E+36
?OV ERROR IN 50
OK
LIST

10 REM CALCULATE N! RECURSIVELY
20 FOR M=0 TO 17 : N=M : GOSUB 40 : GOSUB 60 : PRINT TAB(36);
30 N=M+18 : GOSUB 40 : GOSUB 60 : PRINT : NEXT
40 IF N=0 THEN F=1 : RETURN
50 N=N-1 : GOSUB 40 : N=N+1 : F=N*F : RETURN
60 PRINT STR$(N);"! =" F, : RETURN
OK
```

SoftWARE HINTs for 8800

by Bill Gates

When I first heard about 8-bit computers, I thought about how difficult 12-bit computers, like the PDP-8, are to program. The PDP-8 has one accumulator, 256 words directly addressable and eight instructions. Cutting down on this would make a computer unusable.

The 8080 instruction set is actually much better than that of the PDP-8. There are seven accumulators, 65,536 words directly addressable and 78 instructions. How is it possible to have all this on a computer with only 2/3 as many bit per word? One of the most important reasons is the use of multi-byte instructions. Any possible address can be specified in three byte instructions that use the second and third bytes to form an address. For ease of manipulation up to three addresses can be stored in the registers. Decrementing (DCX), incrementing (INX) or adding another number (DAD) to these addresses are all one instruction operations.

Another important thing about the 8080's instruction set is the stack, used both for storing temporary values and subroutine return addresses. Two registers can be stored in, or loaded from memory with a single PUSH/or POP instruction. When a subroutine returns, conditionally or unconditionally, no address needs to be specified since the new program counter is always taken from the top of the stack. The stack allows a programmer to be tricky and elegant using very few instructions.

4K BASIC is a good example of how compact a complicated program written for the Altair can be. Some 16-bit machines have 4K BASIC's as good as Altair 4K BASIC--but, to use the same amount of memory as Altair 4K BASIC, a 16-bit machine would have to have a 2K BASIC which is unheard of.

BCD Arithmetic

BCD stand for binary coded decimal. This is a way of storing numbers according to their decimal digits. Four bits are used to store each digit, so two digits are stored in each word. Each decimal digit is represented by its value in binary, so 0=0000, 1=0001, 2=0010, 3=0011, 4=0100, 5=0101, 6=0110, 7=0111, 8=1000, 9=1001. This leaves six possible configurations of digits which are meaningless (1010, 1011, 1100, 1101, 1110, 1111). These wasted combinations mean that BCD is not as compact a way of storing numbers as is binary. In binary, a word can have values from 0 (all 0's) to 255 (all 1's). In BCD, a word can range from 0 (all 0's) to 255 (all 1's).

The advantage of BCD is that decimal input and output are extremely easy. With binary numbers, decimal input involves multiplying by 10 and decimal output dividing by 10. If the operations to be performed on a number are simple (addition or subtraction), BCD may be a more convenient form of storage. Simple calculators use BCD for internal storage, whereas more powerful calculators, such as those that have trigonometric functions, convert numbers to a binary format.

The Altair has a special instruction for BCD arithmetic called DAA (decimal adjust). Adding 31=0011 0001 to 56=0101 0110 will give 87=0100 0111, so the ADD instruction works fine in this case. The problem with using ADD occurs whenever you get decimal carry. Example: 46=0100 0110 + 27=0010 0111 will give 0110 1101, which is meaningless as a BCD number. If a result has a digit greater than 9, we want to have the next higher digit incremented. Also, consider 19=0001 1001 + 48=0100 1000 = 0110 0001 which is 61. The carry from the low order digit resulted in a number greater than 15, so the 10's digit was affected. Therefore, only 10 was added to the number, instead of the 16 that should have been added. The CY1 flag bit, in the PSW, is used to remember if a carry occured out of the fourth bit, i.e., whether the 10's digit was affected. The DAA instruction is always used after adding BCD numbers. DAA, the only instruction that uses CY1, looks at the number in accumulator A and reformats it as a BCD number. This is done by adding 6 if CY1 is on, checking for a digit being greater than 9 and if so, subtracting 10 from that digit and incrementing the next higher digit. If the high order digit overflows, carry is set.

```
;Routine to convert A from BCD
;to binary.  13 bytes.

;A routine which doesn't loop takes
; 14 bytes

BCDBIN:     PUSH B              ;save [B,C]
            ORA  A              ; turn carry off
            MVI  B,100 decimal

LOOPBC      INR  A              ; add to BCD #
            DAA                 ; set carry if
                                ;equal to 100
            DCR  B              ; count down
                                ;[B]
            JNC  LOOPBC         ;if [A] is not equal to 100
                                ;continue to
                                ;count down
            MOV  A,B            ;save result
                                ;in [A]
            POP  B              ;restore [B,C]
            RET
```

If anyone has a shorter solution than the above, please send it in.
Challenge: What is the shortest Binary to BCD routine?

THE LXI TRICK

Many computers have SKIP instructions. Having multi-word instructions makes it difficult to tell how much should be skipped, so computers with multi-word instruction are seldom provided with a SKIP instruction. the LXI trick, however, allows the skipping of one or two words. Example: (from Altair BASIC)

```
ERR3:MVI    E,3                 ;set up the error #
     JMP    ERROR
ERR7:MVI    E,7                 ;set up the error #
     JMP    ERROR
ERR5:MVI    E,5                 ;
     JMP    ERROR
ERR2:MVI    E,2
ERROR:LXI   B,ERRMSG

change this to:

ERR3:       MVI  E,3
            1                   ;first byte of LXIB,
ERR7:       MVI  E,7
            1                   ;first byte of LXIB,
ERR5:       MVI  E,5
            1
ERR2        MVI  E,2
ERROR:      LXI  B,ERRMSG
```

If a jump is made to ERR3, E will be set up. Then a LXIB will be executed. B, and C, will be given garbage values depending on the instructions that follow, and the program counter will be incremented beyond the MVI E,7.

--continued on page 12--

--ALTAIR USERS--

El Paso Computer Group
c/o Jack O. Coats Jr.
213 Argonaut #27
El Paso,TX 79912

Fred Kitman
Opus One
400 Smithfield
Pittsburgh,PA 15222

SCYL
Students CYbernetics Lab
16 Linwood Ave.
Buffalo,NY 14209

Joe Konrad
33 Plant St.
New London, Ct. 06320

Mark Rothstein
15401 Pegg Court
Bowie, Mo. 20716

Software Contest:

Members will be encouraged to submit programs for the *Altair Library*. These programs will be one of two categories: A. Subroutines, and B. Major Programs. All programs will be screened and tested by MITS.

Once a program has been found to be acceptable, it will be included in the *Altair* library and a description of the program will be printed in the User's Club newsletter. The author of the program will be entitled to a free printout of any two programs from the *Altair* library.(value up to $10.00)

There will be prizes awarded to the authors of the best programs. The prize for the best "major program" (announced in each newsletter) will be **$50.00 credit** toward the purchase of an *Altair* or *Altair* options. Second prize will be **$25.00 credit** and third place will be **$15.00 credit.** The author of the best "subroutine" will receive **$25.00 credit.** Second prize for a "subroutine" will be **$15.00 credit.**

A grand prize of $1000.00 credit will be awarded each year to the author of the overall best "major program." A prize of **$250.00 credit** will be awarded to the author of the best "subroutine."

MITS employees and their families will be encouraged to be members of the *Altair User's Club*, however, they will not be eligible for prizes. Contest void where prohibited by law.

Effective Sept.15, 1975 assembly language programs must be submitted on official entry forms and they must be typewritten. Samples of these forms are included with this mailing of Computer Notes. You may duplicate the forms yourself or you may order these forms from MITS for $2.00 per 50 (includes postage and handling). See Barbara's column for further information.

Programs from the Altair Library will be sold at a nominal fee (to cover printing, postage & handling). The prices for the first 15 programs (listed in the July issue of Computer Notes) are:

#523751--$4.00
#516751--$3.00
#63751 --$2.00
#516753--$2.00
#55751 --$3.00
#62752 --$3.00
#422751--$2.00
#62751 --$2.00
#611751--$3.00
#429751--$2.00
#519753--$2.00
#527751--$2.00
#519751--$2.00
#521751--$15.00

Another example: (from Basic)

```
AND:        MVI  A,1            ;set flag this is AND
            JMP  DOBOOL         ;by setting A non-zero
OR:         XRA  A              ;set flag this is OR
                                ;by setting A to zero
DOBOOL:     .....
change to:
                                ;set flag with A not 0
                                ;first byte of ORI
AND:        366 OCTAL           ;XRA A does not equal 0 so will
                                  not equal zero
OR:         XRA  A
DOBOOL:     ....
```

Skip one byte:

```
MVI   A,(or other register)     ;sets register to next op code,
                                 but does not affect the condition
                                 codes.
CPI                             ; Doesn't change any registers, but
                                 condition codes are set.
ORI   A,                        ;Sets register to next op code, and
                                 will always reset the ZERO condi-
                                 tion code, and carry.
```

Skip two bytes:

```
LXI   B,(or other register pair);Sets register pair to following
                                 two bytes.  Does not affect
                                 condition codes.
JC,JNZ,etc.
        (or other conditional Jump)
                                ;if you know one of the condition
                                 codes is on or off you can skip the
                                 next two bytes by using a JMP which
                                 won't get executed.  Condition codes
                                 are not affected, nor are any
                                 registers.
```

Other hints:
If you have the following sequence:

```
            CALL   SUB1
            JMP    LABL1
SUB1:       LHLD    ADR1

shorten it to:
            LXI    H,LABL1
            PUSH   H
SUB1:       LHLD   ADR1
```

This may look like a fairly special trick, but manipulating return addresses with PUSH's and POP's is a very general technique. For instance, say, a subroutine has a condition in which it wants to return to its caller's caller and not the caller. POPing off the caller's return address will handle this.
Consider this:

```
CPI   ","       Replace this with:
JZ    TERMN
CPI   ":"                                   ;assuming [B,C] is
JZ    TERMN             LXI   B,TERMN       ;free,put the
CPI   12                PUSH  B             ;address to go to
JZ    TERMN                                 ;onto the stack.
CPI   ";"               CPI   ","
JZ.   TERMN             RZ                  ;go to TERMN if
                        CPI   ":"           ;matches
                        RZ
                        CPI   12
                        RZ
                        CPI   ";"
                        RZ
                        POP   B             ;eliminate TERMN
                                            ;address from the
                                            ;stack
```

Final challenge: Write a short subroutine that finds the parity of A without using the parity flag. Give the result in the carry flag, and don't smash any registers but A. You can use the stack.

NEXT MONTH: The bootstrap loader explained, some obvious mistakes and arithmetic tricks.

ALTAIR BASIC - UP AND RUNNING

In January, when Popular Electronics featured the Altair Computer on its front cover, we knew that we had a great product. But no one could have predicted the enormous flood of inquiries and phone calls and orders that started hitting us about mid-January.

Partly because the Altair has generated such a huge volume of business, we have been able to speed up our Altair development program and broaden our horizons somewhat. Undoubtedly the most newsworthy of these developments is the introduction of a BASIC programming language for the Altair Computer.

That's right. We've got BASIC and it's up and running!

People who are familiar with programming and BASIC language will most likely understand why we're making such a big deal out of this. For those who aren't familiar, we offer the following explanation.

A few years back, realizing that computers needn't be so darn complicated, a group of professors at Dartmouth College developed a revolutionary, new computer language called BASIC language. This language was designed so that people with little or no computer knowledge could learn how to program.

BASIC language works because it is just what it says--it is, namely, BASIC. For example, when you want to instruct the computer to PRINT something and you are using BASIC language, you simply type the word PRINT on your terminal or teletype keyboard followed by whatever it is you want the computer to print. BASIC is BASIC. It is simple and understandable.

To illustrate this further, let's take a look at this sample BASIC program, designed to calculate a simple interest problem.

```
SCRATCH↵

10 LET P=650↵

20 LET T=18↵

30 LET R=.065↵

40 LET I=P*T*R/12↵

50 LET P1=P+I↵

60 LET M=P1/T ↵

70 PRINT "TOTAL INTEREST IS";I ↵

80 PRINT "TOTAL MONEY OWED IS";P1↵

90 PRINT "MONTHLY PAYMENTS ARE";M↵

   RUN ↵
```

This program is a set of instructions to the computer telling it to answer some basic questions about a loan of $650.00 for 18 months at an add-on (simple interest) rate of 6 1/2%. These questions are: (1) What is the total amount of interest? (2) What is the total amount of money owed? and (3) What are the monthly payments?

While at this point the program may seem obscure to you, it will become clearer if we analyze each of its various components.

COMMANDS

Commands are direct orders to a computer that are executed immediately upon entry. The two commands in the preceding program are SCRATCH ↵ and RUN ↵. SCRATCH commands the computer to clear its memory of any unnecessary data and RUN commands the computer to RUN the program in its memory.

STATEMENTS

A statement is an instruction to the computer. It is executed only after a command has been given. The first statement in the preceding program is: 10 LET P=650 ↵. This statement instructs the computer to assign a value of 650 to P. As you may recall, 650 is the principal on the loan.

--CONTINUED PAGE 3--

COMPUTER NOTES

APRIL 7, 1975

A PUBLICATION OF THE ALTAIR USERS GROUP VOLUME ONE ISSUE ONE

COMPUTER HISTORY

How many of you have heard of Charles Babbage?

Charles Babbage was an English inventor and philosopher who was blessed with an unusually fervent mind. In 1822, he proposed that the English government finance the construction of a massive machine which he called the "Difference Engine." This machine would be able to add, subtract, divide and multiply numbers and it would also be able to mechanically print the results!

Babbage somehow managed to raise 17,000 pounds, but two tons of brass, steel and pewter clockwork later, the project was abandoned as utterly hopeless.

Of course, Babbage wasn't discouraged enough to give up inventing number machines. In 1834 he was back with a new plan - this one bigger and better. He proposed the construction of an "Analytical Engine" that just happened to contain the basic elements of a digital computer.

The Analytical Engine had a memory called the "store" unit, a CPU called the "mill" unit, and a control unit. Furthermore, this Engine had two sets of instructions cards--one for mathematical operations and another consisting of variables and constants.

Unfortunately, Babbage was unsuccessful in raising the bread for this second project. "You're just not practical, Babbage," is what his legislative friends said.

The point of this is simply that computer history is fun. You can't exactly produce a year by year calendar of events since the invention of the abacus in 450 BC, however, there is certainly some enlightening material.

One of the most enlightening books about computers is the textbook, Computers and Society by Stanley Rothman, published by Science Research Associates. This book attempts to address itself to the impact computers have had and will have on society. It is filled with good general information and has a very readable format.

If any members of the Altair Users Group would like to write an article on computer history, we'd be more than glad to publish it with full credit.

--DB

Across the Editor's Desk

David Bunnell, editor

According to our figures, 2% of the world's general purpose computers are now Altair 8800's. And the Altair's only been on the market since January.

It's been a fascinating first quarter, to say the least. At a time when the country is in the throes of a deep recession, business at MITS has been great.

So you might say that the theme of this first edition of Computer Notes is "let's keep the ball rolling."

One way to implement this theme is through the development of software. Altair BASIC is certainly a good start, but we need to start building a comprehensive Altair Software Library.

That's where you come in.

To motivate members of the Altair Users Group to send in their programs for inclusion in the library, MITS is sponsoring a very attractive "SOFTWARE CONTEST."

In the next issue of Computer Notes, we will be awarding a prize of $50.00 credit toward the purchase of an Altair or Altair options to the author of the best "major program." Other prizes include $25.00 credit to the author of the best subroutine. (see page 4 for details)

Another way to keep the ball rolling is by the development of new Altair Options and System Packages. Our "NEW PRODUCTS" section features a line printer, teletype and floppy disk system and we are also using Computer Notes to introduce our BASIC language systems.

Computer Notes is really the tool for communicating all this good news and also, as you can see by reading this newspaper, to let you in on "boo-boo's" we have made, give you technical advice and general computer information.

We'll need some active participation on the part of Altair Users to make this publication a real success. Feel free to write to us with your comments and advice.

The next issue will come out in June. The way things have been going, we'll have tons of good news for you then.

Boo-Boo's

In that simple sample in the Operator's Manual on pages 33-38, we forgot to put in that you must operate it from Loc 0, and the easiest way to do that is to RESET before you run. To examine the results, hit STOP, put the address you stored in the ADR switches. If you want to run again, RUSET & RUN. I always use this RESET to start from Loc 0. It's the easiest way.

There is a mistake on some of the back pages of the Operator's Manual. It shows two instructions with code of 353. The code for PCHL is 351 and XCHG is 353.

As most of you know, we had a shipping boo-boo with our kits. About 100 units got out with a 7812 voltage regulator instead of a 7805. That's a 12v regulator instead of a 5v. If any of you have received them and haven't been notified, let us know and we'll send you the right one immediately. Only kits were affected.

ALTAIR SERVICE DEPT.

Barbara Sims

Greetings from the Altair Special Services Department. I'm Barbara Sims. Now that you own an Altair 8800, I'll be talking with you when you phone, or write, in for help with your minicomputer. Paul Van Baalen will be the engineer in charge of our Customer Service Department, which includes the Special Services. Between the two of us we hope to answer any questions you might have. Our department will also be in charge of handling customer problems, taking care of any additions to your 8800, and trouble shooting your unit.

In the near future we hope to have a special phone number for 8800 owners to use when calling the Altair Service Desk. This should speed matters up when you have questions concerning your minicomputer.

Here's wishing you a fantastic future with your Altair 8800 minicomputer.

Software

by Harvey Lee

Although this column is designed primarily for the benefit of the beginning programmer, much of the information will also be of value to the more experienced programmer. Any Altair 8800 peculiarities, items of interest from the Altair Users Group Library and examples of programs will be included along with guidelines on basic programming skills.

For example, the normal use of the HLT (halt) instruction is discussed in the Altair 8800 Operator's Manual. If HLT is used in a program in which no interrupts occur, the computer becomes inoperative, and it would seem necessary to turn the computer off then back on to restore operation. This zaps the memory. The computer can be cleared by holding the STOP/RUN switch in the STOP position while moving the RESET/CLEAR switch to RESET and releasing, thus saving the memory.

Harvey Lee

An error has been discovered in the Altair 8800 Operator's Manual on page 65. The LXI (Load Register Pair Immediate) instruction loads the first byte of data (the least significant 8 bits) into the second register of the specified pair and loads the second byte of data (the most significant 8 bits) into the first register of the specified pair. Early editions of the manual had this order of loading reversed (it has since been corrected).

SEE PAGE 5 FOR EXAMPLE PROGRAM

By relieving man of dull, repetitive tasks, providing him with information and instructions and by solving problems, the computer of the future will be a steam engine as applied to the mind.

CONTINUED FROM PAGE ONE--ALTAIR BASIC - UP AND RUNNING

LINE NUMBERS

A line number is placed at the beginning of each statement. It does two things. First, it tells the computer not to execute the instruction that follows until a command has been given and second, it gives order to the execution of the statements in a program. Following the entry of a command, the first statement to be executed is the one with the smallest line number, then the next smallest and so on.

RETURN (↵)

Each statement and each command in a program is ended by the depression of the RETURN key. The symbol ↵ represents pressing this key.

LET Statements

The first six statements in our example program are LET statements. The LET statement is used to assign a number called a value to an abbreviated code called a variable. In the first LET statement, the value of 650 is assigned to the variable, P. The value in a LET statement can be a single number or it can be the result of a computation as it is in statement 40. (LET I=P*T*R/12 ↵)

A variable will retain the value assigned to it throughout the program unless you change the value. Once a value has been assigned to the variable P; P can be used to represent this value throughout the remaining program statements.

In view of what we now know about LET statements, let's analyze the six LET statements in our example program:

```
10 LET P=650↵
```

This LET statement assigns the value, 650, to the variable, P. $650 is the amount of the principal of the loan.

```
20 LET T=18↵
```

This LET statement assigns the value, 18, to the variable, T. 18 months is the amount of time you have to pay off the loan.

```
30 LET R=.065↵
```

This LET statement assigns the value, .065, to the variable, R. The rate of interest is 6 1/2%, which converts to the decimal .065.

```
40 LET I=P*T*R/12↵
```

This LET statement assigns the value, P*T*R/12, to the variable, I. "*" is the symbol for multiplication and "/" is the symbol for division. Interest on the loan is equal to principal times time times rate. We divide this answer by 12 because the value we have entered for time is 18 (statement 20) and the interest formula calls for time to be in years.

```
50 LET P1=P+I ↵
```

This LET statement assigns the value, P+I, to the variable, P1. The total payment is equal to the principal plus the interest.

```
60 LET M=P1/T ↵
```

This LET statement assigns the value, P1/T, to the variable, M. The monthly payment is equal to the total payment divided by time.

PRINT Statement

The last three statements in the example program are PRINT statements. A PRINT statement can instruct the computer to display a "string literal" -- a "string" of characters enclosed within quotation marks in the PRINT statement.

PRINT statement 70 instructs the computer to print "TOTAL INTEREST IS" followed by the value of I. PRINT statement 80 instructs the computer to print "TOTAL MONEY OWED IS" followed by the value of P1 and statement 90 instructs the computer to print "MONTHLY PAYMENTS ARE" followed by the value of M.

This BASIC program is rather limited as far as programs go. By the use of a READ statement, it can easily be modified to solve any simple interest problem, not just the above loan of $650.00. However, our sample program does illustrate some of the principals of BASIC programming.

In addition to being easy to learn and to use, the Altair BASIC language software is also very powerful. It has transidental math functions that allow you to use the computer like an advanced, scientific calculator and it has powerful looping instructions such as GOTO, GOSUB, IF and THEN that allow you to write very sophisticated programs.

There are two keys to the new computer revolution. One is computers must be inexpensive and the other is computers must be understandable. With the Altair 8800 and Altair BASIC, both of these criteria have been met.

Altair BASIC contains the following statements:

PRINT	ON
IF	STOP
THEN	DATA
GOSUB	RESTORE
RETURN	LET
NEXT	DIM
FOR	REM
READ	LIST
INPUT	CLEAR
END	SCRATCH

Altair BASIC also contains the following functions:

SIN	SQR
COS	INT
LOG	FRE
EXP	AND
TAN	ABS
ATN	SGN

Software Prices.....

Altair BASIC language.....$500.00

When purchased with an Altair, 8K memory and either a serial I/O board or an audio-cassette interface (no copy fee).....$75.00

Altair EXTENDED BASIC*.....$750.00

When purchased with an Altair, 12K memory and either a serial I/O board or an audio-cassette interface (no copy fee).....$150.00

SOFTWARE PACKAGE ONE (resident assembler, text editor, and system monitor)**.....$500.00

When purchased with an Altair, 8K memory and either a serial I/O board or an audio-cassette interface.....FREE binary listing

($30.00 copying fee on audio-cassette or paper tape)

*EXTENDED BASIC is Altair BASIC with logical operators, double precision arithmetic, PRINT USING, string capability, etc. More on this in the next issue.

**A resident assembler allows you to write machine language programs using mnemonics. A text editor allows you to make corrections to a program as you write it. A system monitor is a housekeeping routine that monitors system conditions and allows you to communicate with peripherals without manual boot strapping, etc.

New Products

OKIDATA CP 110 LINE PRINTER

PRINT CHARACTERISTICS:

Print Method. Impact Dot Matrix. Dot Size. 0.015" Character Height. 0.105" 7 verical pins on 10 degree slant. Character Width. 0.075" 5x7 Matrix. Character Spacing. 10 characters/inch. Line Width. 80 characters. Font. 64 character ASCII subset standard; other optional.

PRINT MECHANISM:

Head Motion. Print head moves uniformly in both directions and pauses only at end of line. Individual characters are printed on the fly. Print Speed. 110 characters/second. Throughput. 70 lines per minute Bi-directional printing.

PAPER MECHANISM:

Line Spacing. 6 lines/inch. Feed Rate. 180 millisecond single space 420 line/minute Slew.

INPUT POWER:

Voltage. 100, 110, 117, 235 VAC +/- 10%, 50 or 60 Hz selected at input transformer taps. Power. 350 watts maximum.

SAVE $200.00. Until April 30, 1975, members of the Altair Users Group can buy this line printer for $200.00 less than the listed retail price (see price list).

4 ALTAIR SYSTEMS

***ALTAIR BASIC I**
Altair 8800 Computer
2 4K Dynamic Memory Boards
Comter II (32-character, self-scan display Terminal with built-in Audio Cassette Record Interface)
Serial Input/Output Card and Connectors
Cooling Fan
BASIC Software

***ALTAIR EXTENDED BASIC II**
Altair 8800 Computer
3 4K Dynamic Memory Boards
Comter II Terminal
Serial Input/Output Card and Connectors
Cooling Fan
Extra Mother Board
EXTENDED BASIC Software

***ALTAIR DOS/BASIC III**
Altair 8800 Computer
4 4K Dynamic Memory Boards
Comter II Terminal
Serial Input/Output Card and Connectors
Cooling Fan
Extra Mother Board
Disk Controller
2 Disk Drives
EXTENDED BASIC and DOS Software

ALTAIR EXTENDED Engr/Acctg IV
Altair 8800 Computer
8 4K Dynamic Memory Boards
Teletype ASR-33
Line Printer (110 characters per second—includes controller)
Serial Input/Output Card and Connectors
Cooling Fan
3 Extra Mother Boards
Disk Controller
2 Disk Drives
EXTENDED BASIC and DOS Software

*Teletype ASR-33 can be substituted for Comter II Terminal—See Price List

COMPUTER TRADE-IN

If you have a Mark 8 or a Shelby or an M16 National Computer and you would rather have an Altair 8800, we will offer you $150.00 on a trade-in.

For an Altair Computer kit, send in your used computer plus a check for $297.00 ($439 minus $150 plus $8 for postage and handling). For an assembled unit send your used computer plus a check for $479.00.

ALTAIR FLOPPY DISC

88-DISK Disk Drive — Consists of Pertex FD 400 floppy disk drive, power supply (110-125v AC, 60 Hz), cooling fan, disk buffer and address select electronics in Optima case similar to Altair Computer case. Capable of storing up to 300,000 words on a flexible disk. Disk included. Up to 16 disk drives can be controlled by one 88-DC Disk Controller.

88-DC Disk Controller — The 88-DC Disk Controller consists of two circuit boards. It electronically sectors each track on the disk into eight groups of 512 words. (Each disk has 77 tracks). Capable of controlling up to 16 Disk Drives.

ACR-33 Teletypes NOW AVAILABLE (see Price List)

Software Contest:

Members will be encouraged to submit programs for the *Altair Library*. These programs will be one of two categories: A. Subroutines, and B. Major Programs. All programs will be screened and tested by MITS.

Once a program has been found to be acceptable, it will be included in the *Altair* library and a description of the program will be printed in the User's Club newsletter. The author of the program will be entitled to a free printout of any two programs from the *Altair* library.

There will be prizes awarded to the authors of the best programs. The prize for the best "major program" (announced in each newsletter) will be **$50.00 credit** toward the purchase of an *Altair* or *Altair* options. Second prize will be **$25.00 credit** and third place will be **$15.00 credit.** The author of the best "subroutine" will receive **$25.00 credit.** Second prize for a "subroutine" will be **$15.00 credit.**

A grand prize of $1000.00 credit will be awarded each year to the author of the overall best "major program." A prize of **$250.00 credit** will be awarded to the author of the best "subroutine."

MITS employees and their families will be encouraged to be members of the *Altair User's Club*, however, they will not be eligible for prizes. Contest void where prohibited by law.

Note: When you submit a program make sure that it is legible (type written preferred). For machine language or assembly language programs, submit (from left to right) a tag (optional), mnemonic, address, octal code and explanation (optional) for each program step.

Program submission forms are available from MITS for $2.00 per 50. This price includes postage and handling.

The age of the afl

Now Available

MAINTENANCE OPTIONS

Below you'll find the two maintenance options that are available at the present time. We do have plans to have service centers throughout the country, but this is probably a year or so down the line.

I. Time and materials--$22.00/hour plus retail parts cost.

II. Maintenance contract--a person may apply for and receive a contract on an assembled unit within the 90 day warranty period. After this 90 day period the machine must be returned for checkout before a contract will be issued. Either party may cancel upon written 30 day notice to the other party.

Time and materials will be charged in all cases for repairs incurred during factory check out. However the check out itself is free if contract is issued.

Customer pays freight both to and from MITS for repairs and check out. MITS reserves the right to replace any portion of the machine with new or reconditioned parts at their option to expedite repair.

Maintenance contracts will be voided upon evidence of unauthorized modification to any part of the machine, misuse of the machine, or acts of God. (i.e. time and material will be charged)

CASE (includes Power Supply and Display Control Board, and CPU Board)		$11.00/month
88-MCS	--	3.00 month
88-1MCS	--	3.00/month
88-4MCD	--	4.00/month
88-SIOA	--	2.00/month
88-SIOB	--	2.00/month
88-SIOC	--	3.00/month
88-PIO	--	2.00/month
88-ACR	--	3.00/month

Prices on other equipment and options will be announced in later issues. Maintenance not provided for customer-designed interface without negotiation.

Requests for maintenance contracts or inquiries about same should be directed to Paul Van Baalen.

CONTINUED FROM PAGE TWO

Example program demonstrating the use of LXI instructions

Memory address and Octal Codes for the instructions have been included for those who wish to try the program.

Memory Address	Octal Code	Mnemonic	Explanation
000 001 002	001 377 000	LXI B	Load immediate register pair B,C B with 000 octal, C with 377 octal
003 004 005	041 200 000	LXI H	Load immediate register pair H,L H with 000 octal, L with 200 octal
006	165	MOV M,L	Move contents of Register L to memory location specified by registers H,L. In this case, location 200 octal.
007	043	INX H	Increment H,L by 1
010	164	MOV M,H	Move contents of Register H to Memory location specified by H,L. In this case, location 201 octal.
011	043	INX H	Increment H,L
012	160	MOV M,B	Move contents of B to Memory specified by H,L. Location 202 octal.
013	043	INX H	Increment H,L
014	161	MOV M,C	Move contents of C to Memory. Location 203 octal.
015	043	INX H	Increment H,L
016	165	MOV M,L	Move contents of L to Memory. Location 204 octal.
017 020 021	303 000 000	JMP	Jump to Memory Address 000 octal and continue execution. This will cause the computer to cycle through the program until the STOP switch is activated.

Upon examination, the following memory addresses should have the corresponding data: (numbers are in octal)

Address	Data	Corresponds to:
200	200	Data loaded into Register L
201	000	Data loaded into Register H
202	000	Data loaded into Register B
203	377	Data loaded into Register C
204	204	Data in Register L after execution of all the INX H instructions.

If a HLT instruction were used to replace the JMP instruction, the computer would halt execution to wait for an interrupt to occur. This can be cleared as previously described.

rdable computer.

Altair Technical hints

USING THE STACK

The stack is a portion of memory the programmer sets aside for temporary storage of data or addresses. The stack is necessary for the proper execution of many instructions. The Stack Pointer is a 16-bit register that specifies the address in the stack that will be operated upon.

To establish the stack use the LXI SP instruction. The data byte immediately following the instruction has the least significant 8 bits of the address, and the next data byte has the most significant 8 bits of the address. For example:

```
LXI  SP     sets the stack pointer
077,000     at memory address
            000 077 octal.
```

There are two basic operations on the stack, the PUSH and the POP. The PUSH instruction moves the contents of the specified register pair into the stack. The first register of the specified register pair goes into the stack at the address in the stack pointer minus 1 and the second register at the address in the stack pointer minus 2. The stack pointer is then decremented by 2. For example:

```
PUSH D      with the stack pointer
            at 077 octal would move
            the contents of regis-
            ter D to memory address
            076, E to 075 and then
            set the stack pointer
            to 075.
```

The POP instruction is the reverse of the PUSH. So the content of the stack at the address contained in the stack pointer is moved into the second register of the specified register pair. The content of the stack at the address contained in the stack pointer plus 1 is moved into the first register of the specified pair. The stack pointer is then incremented by 2. For example:

```
POP D       with the stack pointer
            at 075 octal would move
            the content of 075 to
            register E, 076 to reg-
            ister D and then set
            the stack pointer to 077.
```

In addition to these instructions the stack pointer may be operated on by many of the register pair instructions.

When programming remember the following:

1) If an instruction requires a stack for proper execution be

--CONTINUED ON PAGE 7--

Updated Price List

MITS ALTAIR 8800

PART NUMBER	DESCRIPTION	KIT	ASSEM	DAYS DELIVERY
Computers, Terminals & Line Printer				
*8800	Altair 8800 Computer	$ 439.00	$ 621.00	60
COMTER II	Terminal w/Built in Audio Cassette I/O	780.00	920.00	60
CT-256	Comter 256 Terminal	745.00	885.00	45-60
CT-256	Comter 256 in Aluminum Suitcase	--	965.00	90
CT257, 8 or 9	Pages 2, 3 or 4 for CT-256	95.00	105.00	45-60
CT-8096	CRT Terminal	TBD	TBD	JUNE
88-VLCT	Low Cost Terminal	129.00	169.00	45-60
88-80LP	Line Printer & Controller, 110 char/sec	1,750.00	1,975.00	90
88-TTY	Teletype ACR-33	1,500.00	1,500.00	60-90
Memory				
88-MCS	256 words memory--Exp to 1K with 88-MM	103.00	134.00	60
88-MM	Adds 256 words to 88-MCS	53.00	61.00	60
88-1MCS	1K Static Memory	176.00	209.00	60
88-4MCD	4K Dynamic Memory	264.00	338.00	60
88-DCDD	Disc Controller, 1 Disc Drive & Multiplexer	1,480.00	1,980.00	60
88-DISC	Disc Drive in cabinet with added Multiplexer	1,180.00	1,600.00	60
88-DMAC	Direct Memory Access Controller	98.00	149.00	90
88-DMAE	Direct Memory I/O Channel--External	126.00	186.00	90
88-DMAI	Direct Memory I/O Channel--Internal	123.00	183.00	90
88-ACR	Audio Cassette Record Interface	128.00	174.00	60
I/O and Expansion Devices				
88-PIO	Parallel I/O	92.00	114.00	60
88-SIOA	Serial I/O RS-232 compatible	119.00	138.00	60
88-SIOB	Serial I/O--TTL	124.00	146.00	60
88-SIOC	Serial I/O--TTY	124.00	146.00	60
*88-EC	Expander Mother Board (adds 4 slots to 8800)	16.00	31.00	60
88-EBC	Expander Cabinet (add'l case, P/S, etc. for 16 slots)	394.00	485.00	60
88-EXC	Extender Card	57.00	83.00	60
Miscellaneous				
88-VI	Vectored Interrupt	126.00	179.00	60
88-RTC	Real Time Clock	53.00	84.00	60
88-ACC	Altair Cyclops Camera	180.00	235.00	90
88-CCC	Camera Controller Card	260.00	340.00	90
88-KB	ASCII Keyboard	198.00	254.00	60
88-32DU	32 char Alpha/Numeric Display	498.00	549.00	60
88-PPCB	Prototype P.C. Board	57.00	84.00	60
88-FAN	Cooling Fan	16.00	20.00	15
88-25DB	Pr. Connectors--1 each 7325-DB25P & S + cover	11.00	11.00	15
MS-416	MitScope--4 channel scope	127.00	189.00	
PROM				
88-PMC	PROM Memory Card	76.00	128.00	90
88-PROM	PROM'S	42.00	57.00	90
88-PPC	PROM Programmer Card	324.00	420.00	90

Suggested Systems

System 6			--	$5,297.00	90
I	ALTAIR Basic I	Price	2,393.00	2,886.00	60
		Less Discount	-425.00	-425.00	
		YOUR COST	1,968.00	2,461.00	
II	ALTAIR Extended Basic II	Price	2,950.00	3,531.00	60
		Less Discount	-725.00	-725.00	
		YOUR COST	2,225.00	2,806.00	
III	ALTAIR DOS/Basic III	Price	6,374.00	7,949.00	90
		Less Discount	-1,300.00	-1,300.00	
		YOUR COST	5,074.00	6,649.00	
IV	ALTAIR Extended Engr/Acctg IV	Price	10,002.00	11,989.00	120
		Less Discount	-1,500.00	-1,500.00	
		YOUR COST	8,502.00	10,489.00	

Software

		With Minimum System:	
Package 1	$500.00	+ 8K memory, I/O	Free Binary (on tape $30.00)
BASIC	500.00	+ 8K memory, I/O & terminal	$ 75.00
EXT. BASIC	750.00	+ 12K memory, I/O & terminal	150.00
DOS	500.00	n/a	
User's Group	30.00/yr.	Free with 8800 (Foreign--add $5.00 airmail postage)	

Miscellaneous Parts

(A)	3 Chip Package (Microprocessor & 2 memory)	$250.00	--	10
(B)	Set 4 P.C. Boards (1 ea. CPU, Exp., Memory & Front Panel)	73.00	--	30
(C)	P/N 88-CPU Complete CPU Board	310.00	360.00	60

**Manuals - Terminal CT-256

Operators	$ 6.50
Assembly	10.00
Theory of Operation, Schematics & Trouble Shooting	10.00

**Manuals - Altair 8800 Computer

Operators	7.50
Assembly	9.00
Theory of Operation, Schematics & Trouble Shooting	9.00
One year up-date to theory manual	10.00

**Manuals - Peripherals

Combination Operators & Assembly (each)	5.00

Postage & Terms

-- Terms: Cash with order, Mastercharge or Bankamericard
-- Postage & Handling:
1) Add $8.00 each for Terminal, Computer, Line Printer, Teletype and Disc
2) Add for Peripherals:
(a) -0- if ordered with computer
(b) $3.00 if ordered separately
3) Add $1.00 postage for Chip Package & P.C. Board Set
4) Postage included in price of manuals
5) Canada, Hawaii & Alaska, postage charges subject to quotation.

(*)Note: Basic unit has 4 slots available, one of which is used up with CPU Board. When ordering more than 3 added peripherals, added 88-EC required for each 4 peripherals.
(**) Note: Manuals are included at no cost with purchased units.
Prices, specifications and delivery subject to change.

People's Computer Company is a not-for-profit corporation. The newspaper is about recreational and educational uses of computers . . . computers for everyone. P C C is published five times (and sometimes more) during the school year.

Subscriptions begin with the first issue in the fall. Single subscriptions are $5.00 for five issues ($6.00 outside U S A sent surface mail, $12.00 sent airmail).

* Special offer expire June 30, 1975.
† Sorry, we are out of Vol. I No. 5 and Vol. II No. 2.

PEOPLE'S COMPUTER COMPANY

PEOPLE'S COMPUTER COMPANY
P.O. BOX 310
MENLO PARK, CALIFORNIA 94025

drawing by Nancy Hertert

Order Form

- ◯ one year regular subscription '74-'75 $5.
- ◯ two year special offer* subscription '74-'75-'76 $8.
- ◯ back issues: Vol. I Nos. 1,2,3,4† $2.
- ◯ back issues: Vol. II Nos. 1,3,4,5,6† $4.

Enclosed is a check or money order for _____ made payable to People's Computer Company. *CN-4-75*

Name_____
Address_____
_____
zip

CONTINUED FROM PAGE SIX--

USING THE STACK

sure to provide it. (For example, the stack is necessary when using subroutines.)

2) There should be a POP instruction for every PUSH instruction. (The stack is for temporary storage.)

3) The stack pointer moves down through memory as data is added to the stack and back up as data is removed. Be sure to allow sufficient memory for the maximum possible requirements of the stack.

ATTENTION KITBUILDERS

There are 4 holes on some of the Display Control boards which have been drilled out to accommodate a heavier gauge wire. This means that you as the kitbuilder will have to solder the wire on both sides of the board to assure power and ground get to the necessary places. The holes involved are labeled AC SW, GND and (+8V).

A TALKING COMPUTER???

One of the most obvious problems in programming a computer to listen and comprehend the spoken word is that a computer cannot see the situation. The way in which something is said, the gestures and expressions of the person saying it has great bearing on the meaning.

There is a program, the GENIE program, which has successfully taught a computer to take instructions in ordinary language. Of course, the vocabulary is rather small, but the computer has a 90% chance of complete understanding.

According to Edmund C. Berkeley, the editor of Computers and People, there are five critera for successfully programming a computer to understand spoken language. These are:

1. Recognition of a framework of a sentence.
2. Recognition of synonyms and word groups.
3. Understanding of context.
4. A vocabulary of 300-500 words.
5. Variation in the way something has been stated.

Berkeley further claims that each of these criteria have been met in seperate programs. That combining them in a single program is not far away.

A talking computer? It's something to talk about.

Q and A

Q. What do I need to hook my Southwest Technical Products CT1024 to my ALTAIR?

A. You need a CT-S serial interface from Southwest Tech and a SIOA interface from MITS.

Another possibility that we believe will work is a 88-PIO from MITS and a CT-L parallel interface from Southwest.

Q. What do I do about a short routine or subroutine that doesn't do what I want it to?

A. Send it to our service desk -- we won't rewrite it for you but we will tell you what you are doing wrong.

Q. What kinds of hardware problems have you been experiencing with the 8800?

A. The problems at this point have been only in kits and have been primarily solder bridges and cold solder joints in the ground or power circuits. Point of interest: Of 1000 units in the field, only 13 have been returned to date. Five of these have been at the request of the repair department to investigate potentially serious problems. P.S. They weren't serious.

Q. How many options can I put in my ALTAIR 8800 before I have to beef up the power supply?

A. There are 16 slots in the basic chassis. One of these is taken up by the CPU board. The other 15 can be filled in any manner (memory, I/O, etc.) to satisfy the user's requirements.

Q. How do I use the sense switches?

A. Sense switches are inputted to AREG by an input 303 from DEVICE 377. After this the data may be manipulated in any fashion that you choose.

Q. What does MITS have in the mill as a mass storage device priced between the Audio Cassette and the Floppy Disk?

A. We do have a mag tape system in the planning stages right now, and will be announcing more on it in future newsletters.

Q. How can I contact other Users Group members?

A. If you will send a letter to Barb authorizing her to publish your name to other users, we'll have these names printed in future newsletters.

Q. When am I going to get more information on the floppy disk and line printer?

A. Look at the "New Products" page in this newsletter. (page 4)

Q. I've seen bits and pieces about software available for the 8800. What's the full story?

A. See page 3 for software details and prices.

ALTAIR CHECKOUT PROCEDURE

1. BEFORE APPLYING POWER VISUALLY (WITH THE AID OF A MAGNIFYING GLASS IF POSSIBLE) CHECK FOR SOLDER BRIDGES, COLD SOLDER JOINTS, BROKEN LANDS AND/OR WIRES AND CORRECT ORIENTATION OF COMPONENTS. 97.37452% OF ALL FAILURES CAN BE CAUGHT DURING THIS STEP.

2. With the pluggable boards out of the machine, power it up and check the terminal boards for the proper input voltages.
Check the bus for +5 volts (pins 1 & 51), +15 volts (pin 2), -15 volts (pin 52), and ground (pins 50, 100). With the boards out these should reach as follows:
(Use negative side of power supply electrolytic capacitors as ground.)

Pins 1 & 51	+10 VDC	approx.
Pin 2	+20 VDC	
Pin 52	-17 VDC	
Pins 50, 100	-0V	

Now check that none of these voltages are seen on pins adjacent to the above checked pins.

NOTE: IN NO CASE REMOVE OR INSTALL ANY BOARDS WITH POWER ON.

3. After powering the machine down, install the boards and check the output of the voltage regulators for +5VDC. Also check the output of the 12V zener on the CPU board for +12VDC.

Assuming everything above is kosher, you should now power the machine down; install the CPU plug and power it back up. The machine comes up in an undetermined state, so what you do is hold the STOP switch in the stop position and give it a RESET. Then check to see if it is protected; if it is, push the PROTECT-UNPROTECT switch in the unprotect position.

4. Now you can check out the different switches and indicators. All address switches should be in the off position. Hold the RESET on--you should have all the status lights off and all data address lights on. When you release the RESET switch, all the address lights should go off. The MEMR, M1, WO, WAIT lights should be on and whatever data there is in location 0 will be displayed in the data lights.

Now to check the lights and switches for proper operation, turn each address switch on one at a time and make sure that the corresponding address light comes on when EXAMINE switch is operated. What you are doing here is checking for obvious shorts in the address bus area so only one switch should be on at a time.

The data lights should be checked in the same fashion. Only use the lower 8 switches and the DEPOSIT switch to check these.

Checking the EXAMINE NEXT and DEPOSIT NEXT is fairly simple. Just keep pressing them and observe that the address lights count up binarily.

Make sure that PROTECT switch turns on the Protect Status light and that UNPROTECT turns it off. With the PROTECT on you shouldn't be able to change the contents of memory with DEPOSIT or DEPOSIT NEXT (or instructions either).

Now you're ready to try a program. Use the one in the Operator's Manual on pages 33-38. After you load everything in be sure to RESET so that you start from LOC 0. SINGLE STEP through it first to check out the SINGLE STEP switch and then run it. Every time you stop it to examine the results be sure to RESET prior to restarting.

MITS UPGRADES ALTAIR 680

As with the Altair 8800, customer demand for the Altair 680 has far exceeded MITS' expectations. As a matter of fact, the Altair 680 may very well sell as well as the Altair 8800. Due to this demand and to delays in initial shipment, MITS has decided to immediately upgrade the Altair 680 to its second generation design. This means that the Altair 680 will now include the following items at no additional cost:

1) PROM monitor. 1702A PROM monitor chip programmed so that you can immediately load and run paper tape object programs such as the text editor and assembler (see below).

2) Asynchronous Communication Interface Adapter (ACIA). Allows the machine to transmit and receive a character at a time rather than one bit. Minimizes software needed for I/O routines. Contains crystal clock for baud rate synchronization. User-selectable for RS232, Baudot, TTL, 20ma or 60ma current loop. Baud rates of 110, 150, 300, 1200 and 2400.

3) A two pass resident assembler and text editor will be available for assembly language programming. This software is compatible with Motorola's format for assembly language programs, text and object files. 8K bytes of memory is required to run this package. The assembler produces a full assembly listing on the second pass, including the hex codes for the location counter and the instruction mnemonics. A symbol table listing is also produced. The text editor has full capabilities for text editing, including line insertion, printing, deletion and modification; as well as commands for changing one string of characters to another and for searching the text buffers for a particular character string.

4) A BASIC interpreter is under development which will be compatible with the 8800 8K BASIC interpreter.

Although delivery dates will be set back by 30-60 days, the upgraded version will give Altair 680 users a lot more computer for their money. The features that have been added to the upgraded version will greatly enhance the design and capability of the 680 as it was first introduced in Computer Notes, October, 1975.

For the hobbyist, the Altair 680 is still the lowest-priced, most versatile kit of its kind. For industrial dedicated-program applications, the Altair 680 "turn-key" model offers reliable computer power at a compact size and compact price.

For those of you who haven't been introduced to the Altair 680 yet, you'll want to check the following specifications:

- Parallel 8-bit word/16-bit address processor
- Built-in 1K RAM
- Built-in provision for 1K PROM/ROM
- Built-in I/O port
- Case size: 11 1/16" wide
  11 1/16" deep
  4 11/16" high
- Space for up to 4 boards
- 72 executable machine instructions (variations up to 197)
- Price: $345 kit
  $420 assembled

COMPUTER NOTES

JANUARY, 1976

VOLUME ONE ISSUE SEVEN

A PUBLICATION OF THE ALTAIR USERS GROUP

ALTAIR CONVENTION SET FOR MARCH 27-28

The First Annual World's ALTAIR COMPUTER CONVENTION has been definitely scheduled for Saturday and Sunday, March 27 and 28 at the new MITS building in Albuquerque, New Mexico.

In addition to many demonstrations and the official unvailing of the new Altair 8800B (see Ed Roberts' column), there will be at least 4 programs presented during this weekend. These programs include a presentation of the new MITS Traveling Seminar, a seminar conducted by MITS engineers and software writers, a meeting of the Altair User's Group, and a seminar on the home computing field in general.

This last seminar will feature some of the leading personalities in the home computing field. Included are Larry Steckler, technical editor of Radio Electronics, Carl Helmers, editor of Byte magazine, Art Childs, editor of Interface magazine, David Ahl, publisher of Creative Computing, Ward Spaniol, president of the Southern California Computer Society, and Terry Silver, also of the SCCS.

The convention will be free to all out-of-town guests. For more information, read Barbara's column and also see the ad in this issue.

IN THIS ISSUE...

SEE YOU AT THE CONVENTION!

Across the Editor's Desk

by David Bunnell

Moving, Moving, Moving

This morning I went over to production to talk to one of the technicians and production wasn't there. Seems that the move is on.

Working at MITS this past year has been quite an experience. In January of 1975 we had 20 or so employees and occupied two bays in the Cal-Linn Building and it seemed that we had more than enough space.

Well, the situation now is that we occupy several bays of the Cal-Linn. If you're in engineering and you need to talk to someone personally in software, you have to go out the front door of MITS, walk some 300 feet to the end of the building, turn the corner and it's the last door on the right. This is a pleasant little stroll when the weather is nice and a miserable mad dash when it isn't.

The plan is this: production, shipping, repair and the stock room all will move during the third week of January. The rest of MITS will move during the first week of February. Hopefully, while you're reading this issue of Computer Notes I'm sitting in my new office. (Admiring the decor, no doubt.)

The New Building

I took my first tour of the new location last week, and I must say it was impressive.

As a good journalist, I should be able to report to you the total square footage and other such pertinent statistics (how high the ceilings are?) but unfortunately I forgot to ask anyone.

Let me just say that compared to where we are now, the new place is huge. And the nicest thing about it is that the building was actually an empty shell so we were able to design all the offices, production rooms, etc., to correspond to the work flow. This means that we should be able to double and triple our production during the next few months.

The new MITS building, as reported in the last issue of C. N., is located by the Albuquerque Airport at 2450 Alamo SE.

Letters, Phone Calls

You can begin addressing your correspondence to MITS at 2450 Alamo SE, 87106. Actually, it won't matter too much as MITS will be receiving mail at the old address for several months to come.

As for phone calls, the 505-265-7553, 262-1951, and 262-1952 numbers will be transferred to the new building. We will also have a new rotary number which will be announced in the next issue. People calling The Agency will be notified of the new number by the operator.

This Issue

This issue of Computer Notes has a new products slant. The Teletype offer, the new Altair 4K static, hints about the Altair 8800B, upgrading of the Altair 680, the Cyclops Camera, and the TV DAZZLER--looks like an exciting year for Altair users.

We did try to have the January issue out by January 10. Obviously, we didn't make it. Oh well, next month.

LOCAL USERS GROUPS

Anyone interested in forming a computer club in Tennessee, please contact Kent Kersten, 711 Ronnie Rd., Madison, TN 37115

Anyone interested in joining the Canadian Computer Club should contact Mr. G. Pearen, 861 11th Street, Brandon, MB, Canada (204) 725-1079.

IBM 5100 USERS CLUB: Physician with 5100 would like to correspond with anyone in the health care field who is interested in using the 5100 for CAI applications, e.g., patient education. I would be willing to exchange documentation in BASIC or APL and/or start a 5100 USERS CLUB specifically oriented toward medical applications. Write:

Richard E. Easton, M. D.
5541 Parliament Drive
Suite 104
Va. Beach, VA 23462
or call: (804) 490-0124

Anyone interested in forming a Computer Group in the Albuquerque, New Mexico area, please contact Gary Tack, P.O. Box 866, Corrales, NM 87048, (505) 898-7537.

USERS GROUP NEWS

Barbara Sime

SOFTWARE LIBRARY PROGRAMS

We are getting more and more entries to our software contest each month. It has been suggested by one of our contributors that we ask you, the users, what type of software programs you would like to see in our library. If you have requests that you feel would be of general interest to many Altair users, please drop me a card and I'll put your software program suggestions in my column next month.

COMPUTER NOTES

There have been many requests recently for back issues of Computer Notes. I regret to say we have only a limited amount of back issues, and are completely out of issues 1, 2, & 3.

ALTAIR CONVENTION

A date of March 27 & 28 has been set for the Altair Convention. Two hundred rooms have been set aside at the Airport Marina Hotel for the convention. One hundred of these rooms have already been reserved by the SCCS (Southern California Computer Society), which is chartering a plane to bring approximately 200 members. Rates for the remaining 100 rooms are $20.00 for a single occupancy and $24.00 for a double occupancy. These rooms are available for March 26-28. If enough people respond, we plan to reserve rooms in another Albuquerque hotel. Our new plant is located within walking distance of the Albuquerque International Airport and the Airport Marina Hotel.

MITS will not accept phone reservations for the Altair Convention. All reservations must be made on official forms mailed to all members of the Altair Users Group. If you need space for demonstrating your equipment, you must reserve this space through our official forms also. If you have not received an official

—CONTINUED PAGE SIX—

Ramblings from Ed Roberts

PROBLEMS:

The major problem this month is the Repair Department. There is still a significant delay in cycling units that have been sent in for repair. The cycle time has improved considerably over the past few months, but we still have a long way to go. Delivery of standard products has improved enormously in the past three months, but we are still a long way from achieving the desired 24 hour turnaround. The recent holidays have caused a perturbation in deliveries which should be smoothed out during January. During the last half of January and the first half of February we are moving and this will slow some things down.

POLICY CHANGE:

I mentioned in a previous column that we would not pre-announce products, I am changing that policy to this extent. We will continue not to nationally advertise new products until they are in production, but we will announce and discuss some of the up-coming products in Computer Notes. This change is a result of the large number of inquiries and criticisms from customers concerning this policy. Apparently, the pre-release information is needed by many Altair owners for their long-term system planning. Therefore, where feasible, we will provide preliminary information on new products that have not been formally announced. This information will be for planning purposes only and may change as the product gets closer to production.

TELETYPE:

I personally consider the new Teletype T.M. offer to represent a major breakthrough for many Altair owners. You can now own a new Teletype which is warranteed by Teletype Corporation for a price less than the price of a good used Teletype.

CRT and related terminals are certainly useful for many applications, but ultimately almost every application requires some sort of hard copy device. The Teletype is unbeatable from the standpoint of price and reliability. Due to the allocation arrangement with Teletype, you should get your order in as quickly as possible.

POWER SUPPLY:

I have received some questions concerning my comments about power supplies in last month's Computer Notes. The total 8V transformer current available in a basic Altair is approximately 10 amps. Approximately 2 amps are available for the front panel and this is supplied by a separate transformer. The bus transformer is rated at 8V and 8 amps, but as I stated last month, we feel that with a 4 amp load on the bus power supply the voltage margins are less than desirable and, therefore, we use a higher voltage bus transformer in systems with more than 4 or 5 cards. There are some users who actually get as much as 11 amps out of the basic Altair power supply, but this clearly represents an overload to the transformers and the systems voltages have to be marginal. As stated in the previous article, if you have purchased six or more cards from MITS, we will supply you with the higher voltage transformer and mod kit at no cost. If you haven't bought six or more cards from MITS, you may purchase this kit for $43.00. If you wish to purchase this mod kit, order kit 88-PSM.

8800B:

We have received many questions about the new Altair 8800B. Let me give you a thumbnail summary of some of its features and capabilities.

The 8800B is an entirely new Altair, the control and display panels are an entirely new design and contain PROM memory. The power supply supplies 18 amps unregulated at 8 volts and the ± 18 volt supply will provide up to 4 amps.

The CPU uses all the latest IC's for processor control, e.g., the clock width is crystal controlled as well as the frequency. The output of the MPU is buffered by bidirectional Schottky buffers. The front panel is logically isolated from the system bus by an interface card. The interface card and front panel are connected by pluggable ribbon cables. The system bus has 18 slots each with a full 3/4" center. The front dress panel is a four color laminated Mylar and aluminum panel which is back lighted by the indicator LED's. The front panel switches are still high quality toggle switches, but the new switches have longer, flat handles.

Most important are the new functions available on the front panel.

Accumulator Functions:

1. Display accumulator - displays contents of accumulator.
2. Deposit accumulator - deposit data switch register into accumulator.
3. Output accumulator - output accumulator to address on front panel.
4. Input accumulator - input to accumulator from address on front panel.

Slow:

This is a new function which allows the processor to be single stepped at a rate of 32 instructions per second.

The user can redefine the front panel functions by simply re-programming the front panel PROM and creating a custom front panel for special applications. Probably the most important thing about the 8800B is the fact that existing Altair owners will be able to purchase a kit from MITS to upgrade their existing Altair to a B at significantly less cost than purchasing a new machine.

The 8800B is a purist machine, for example there are not any single shots anywhere in the mainframe, all front panel lines are fully buffered from the rest of the bus, special Schottky noise reduction techniques are used throughout, etc. Next month's Computer Notes will provide an in-depth article on the 8800B. Volume production on the B will begin in late February. If you would like to reserve an 8800B or an 8800B update kit, please send a note to Barbara Sims. Pricing information will not be available until mid February.

FEEDBACK:

Your letters to me are extremely valuable, please continue to let me know your thoughts.

DISCOUNTS:

We have in the past offered automatic discounts to customers who have previously purchased MITS products. In the future this policy will be further expanded to make sure that in addition to the many, though sometimes intangible reasons you should purchase MITS products such as software, maintenance support, single source system responsibility, etc., the price advantage for a total system from MITS will be significant.

Highest regards,

Ed

FAST DELIVERY

TELETYPE PRICES SLASHED!

MITS has concluded an agreement with Teletype Corporation that will allow Altair owners to purchase an ASR-33 Teletype from MITS for $500 less than the current MITS price.

Beginning in April of this year, Teletype Corporation will set aside a monthly allotment of Teletypes* for Altair owners. As long as orders do not exceed this allotment, deliveries are expected to be made within two weeks of receipt of order.

For its part, MITS will process all orders and will provide the interface electronics. The interface which fits inside the Teletype will consist of one PC board and a small number of components. It is very easy to assemble. All interconnections will be made with Molex connectors. These connectors come preassembled on the interface cables.

The MITS TTY interface kit will provide "online" and "local" operation; 20 milliamp current loop interface electronics, and power supply electronics.

There are three options being made available under this agreement. These options are:

(1) Teletype printer only (all options include TTY kit). This is ideal for customers who have a keyboard and an audio-cassette interface (or a COMTER II). Provides economical hard copy printout.

(2) KSR-33 Teletype. 20 mA current loop Teletype. Same as ASR-33 Teletype without a paper tape reader and punch. For customers with audio-cassette interface who need an input device and hard copy output.

(3) ASR-33 Teletype. Same as KSR-33 with paper tape reader/punch.

The Teletype machine will be shipped directly from Teletype Corporation to the customer while the interface will be shipped by MITS to the customer. This interface fits inside the Teletype machine and it should not be confused with the interface board needed inside the Altair 8800. To connect an Altair 8800 to a Teletype, you need an SIO-C interface or a 2SIO interface board. These are not included with the Teletype.

PRICES:

	Regular	Discount Customers
88-TYR - - - - -	$669	$629
(Teletype printer and MITS TTY kit)		
88-TYK - - - - -	$769	$729
(KSR-33 Teletype and MITS TTY kit)		
88-TYA - - - - -	$1,019	$969
(ASR-33 Teletype and MITS TTY kit)		

NOTE: MITS discount customers are those who have purchased $1,200 or more worth of Altair equipment.

*Teletype is a registered trademark of Teletype, Inc.

SPECIAL TELETYPE ORDER FORM

☐ENCLOSED IS CHECK FOR _____

☐MASTERCHARGE NUMBER _____ ☐OR BANK AMERICARD _____

INCLUDE $3 FOR POSTAGE AND HANDLING

☐88-TYR TELETYPE PRINTER AND MITS TTY KIT
☐88-TYK KSR-33 TELETYPE AND MITS TTY KIT
☐88-TYA ASR-33 TELETYPE AND MITS TTY KIT

☐PLEASE NOTE THAT I AM A MITS DISCOUNT CUSTOMER

NAME _____

ADDRESS _____

CITY _____ STATE & ZIP _____

MITS/2450 ALAMO SE/ALBUQUERQUE, NM 87106

CLASSIFIED ADS

FOR SALE: IBM 7090 Computer System (4 tape drives, 36 bit word, 32K core, line printer, card reader, all set up manuals, working condition)$50K; Tektronix scopes, models 507, 511AD, 512, 517A, 551. Contact: Bruce Blevins 1600 E. Missouri Apt. 2, Las Cruces, New Mexico 88001 (505) 646-4239

FOR SALE: Southwest Technical Products CT-1024. Mainframe board with installed IC sockets, IC's, capacitors, and resistors. Less plug & sockets for plug-in cards. Memory board with installed IC sockets and by-pass capacitors. Serial I/O interface board completed board with SWTP set baud rate. For information call or write: Larry Belmontes, Jr., 1762 Vale St., Corpus Christi, Texas 78416 (512) 853-4623 Home, or (512) 853-9086 Altair's Communication Center.

FOR SALE: Tally Reader and Punch Combination-Model 420 and 424, used, good condition. $150.
Contact: L. F. Carbaugh
PO Box 398
New Cumberland, PA 17070

FOR SALE: Octal Pad of 100 sheets at $2.25 per pad, plus 45¢ postage and .13¢ California sales tax, if applicable. Also included in the offer are one "Address" and "ASCII" table with each order. Contact:
Ron Santore
1957 Huasna Dr.
San Luis Obispo, CA
93401

NEW COMPUTER STORE!

Coming Soon--COMPUTER LAND
Delaware Valley's comprehensive computer store. (NJ, PA, DEL. area)
Located at: 1228 Barrowdale
Rydal, PA 19046 or
PO Box 225,
Ft. Washington, PA 19034

ALTAIR CONVENTION MARCH 27-28

Altair Equipment in the Nation's Capitol

Altair computer power was brought to the nation's capitol on November 15, 1975 with the opening of MICROSYSTEMS, one of MITS' newest authorized dealers. Located at the edge of the Capitol Beltway in Springfield, Virginia, MICROSYSTEMS is only minutes from the nation's capitol, national monuments, and the Pentagon.

Headed by Russell Banks, the MICROSYSTEMS organization sells, services, and supports the full Altair Computer line. Mr. Banks and the other members of the MICROSYSTEMS team are all seasoned computer professionals with many years of hardware and software experience in business, scientific, and automation applications using mini and micro computers. They are especially qualified and interested in providing turnkey systems using Altair components in a variety of applications.

MICROSYSTEMS operates out of a suite of offices at 6605A Backlick Road in Springfield, serving not only the nation's capitol, but Maryland, Virginia, and the middle Atlantic states. The MICROSYSTEMS offices have been configured into a showroom, office, repair and lab facility, and a meeting/training room. The latter area is the home of the local Washington Area Altair Users Group which invites all local or visiting Altair owners or prospective owners to their regular and informal get-togethers where they exchange hardware, software, and application ideas.

The MICROSYSTEMS grand opening coincided with the MITS seminars in Baltimore and Washington with Mr. Banks being introduced to the attendees by the MITS "Van Man," Mike Hunter. Referring to the capacity crowds and keen interest at the seminars Mr. Banks indicated, "The response has been nothing short of fantastic. We are very happy and proud to be a new part of the Altair team. The large round of applause given at the end of the Washington Seminar by the 126 persons attending was just one of the many fabulous indications of the enthusiasm for the Altair line and the great MITS support. Many of these people came to our grand opening the following day, and many are now Altair owners."

In addition to the Altair line, MICROSYSTEMS is now stocking computer-related publications and parts and accessories of interest to the Altair user. The MICROSYSTEMS organization is geared up to handle both the commercial professional and the hobbyist. If you are interested in an Altair system, adding components to an existing system or hardware and/or software support and engineering services, see them at their Springfield showroom, or call them at (703) 569-1110.

form by January 30, please call our office, and one will be mailed to you immediately. Official reservation forms must be returned no later than February 23. Any received after that date will not be assured of room reservations or demonstration space.

Some highlights of the Altair Convention are listed below:

---Program of speakers who are noted in their field. Speakers that are presently scheduled include:

1. Carl Helmers, editor Byte Magazine will speak on the "The Trends and Applications in the Hobby Field".
2. Larry Steckler, technical editor Radio Electronics, topic for speech unknown at present.
3. David Ahl, publisher Creative Computing has tentatively agreed to speak on "Kit Building in Schools & the Role of the Computer in Education".
4. Two members of SCCS will be speaking.

---The new MITS Mobile Seminar program will be presented Friday night, March 26, for early arrivals. The presentation will be conducted by Mr. Pat Ward.

---A second seminar will be presented by MITS on Saturday to cover all Altair products and software. This seminar will be conducted by MITS engineers and software writers.

---Users Group meeting to discuss programs you'd like to see in the Software Library, how we can improve our users group and library, plus additional topics.

---$10,000 worth of MITS equipment will be given away.

We hope all of you will be able to attend the Altair Convention.

Barbara

8080 BASIC

8080 BASIC language software was developed at MITS for use with the Altair 8800 computer and other 8080-based products. Its versatility, efficiency, and ease of use have already put it at the head of the micro BASIC field. If you are building or are planning to build an 8080-based product, you'll want to note the following 8080 BASIC features:

- ✓ String manipulation (substrings, concatenation)
- ✓ String and numeric arrays with up to 30 dimensions
- ✓ Boolean operators (AND, OR, NOT) for IF statements and bit manipulations
- ✓ Four variable types:
  - 16-bit signed integers
  - 32-bit floating point
  - 64-bit floating point
  - rings
- ✓ Assem language subroutine calls
- ✓ EDIT command
- ✓ PRINT USING for formatted output
- ✓ BASIC may be placed on ROM
- ✓ TRACE on/TRACE off
- ✓ Floppy disk random and sequential data file I/O
- ✓ Direct I/O port access with INP/OUT statements

There are three versions of 8080 BASIC available for use with the Altair 8800 computer from MITS or the Intel 8/MOD 80 or MDS microcomputers or other 8080 systems. BASIC prices for Intellec 8/MOD 80 or MDS systems are: Extended BASIC (10.3K bytes) $350, 8K BASIC (6K bytes) $200, 4K BASIC (3.3K bytes) $150. NOTE: Time and materials charges must be added to the above prices to configure BASIC for non-Intellec systems.

Licenses for source listings and rights to distribute the binary are also available to oem buyers. Write or call Mr. Paul Allen at the MITS plant in Albuquerque for more detailed information. MITS/2450 Alamo SE/Albuquerque, NM 87106 505/262-1951

DAZZLER featured in POP 'TRONICS

A new Altair-compatible interface, the TV DAZZLER from Cromemco, is being featured in the February 1976 issue of Popular Electronics. Providing an interface between the computer and a TV set, the DAZZLER can be "used to generate action games, animated displays, educational learning drills, graphs, even light shows--all in full color."* Considering its versatility and wide variety of applications, the TV DAZZLER represents a unique and affordable concept in computer peripherals.

The basic kit costs $195 and is designed to plug directly into the Altair 8800 using direct memory access (DMA). There are two PC boards, each taking up one slot on the Altair bus. Board #1 outputs a conventional NTSC (National Television Standards Committee) color video signal, and board #2 communicates with the computer via a high speed DMA controller.

"When writing programs for the DAZZLER, it is important to remember that the TV picture is stored as a specially coded sequence in the computer memory. The DAZZLER simply interprets this code to form the image."* Communication between the computer and the DAZZLER uses two output ports (016 and 017) and an input port (016). Output port 016 turns the DAZZLER on and off and sets the starting address of the picture in the computer memory. The data output from port 017 determines the format of the picture as to normal resolution or 4X resolution, amount of memory to be used for the picture, black-and-white or color, and the color or intensity of each frame of the picture. Input port 016 uses one bit to indicate that the DAZZLER is enabled and one bit to indicate the end of a frame.

Interfacing and construction details are outlined in the PE article, along with a parts list, test program, and an octal listing for a DAZZLER Game of Life.

To obtain the schematics, etching and drilling guide and component placement diagram free of charge, send a stamped (for 3 oz.), self-addressed 9" x 12" envelope to:

Cromemco
One First Street
Los Altos, CA 94022

Prices for the TV DAZZLER:

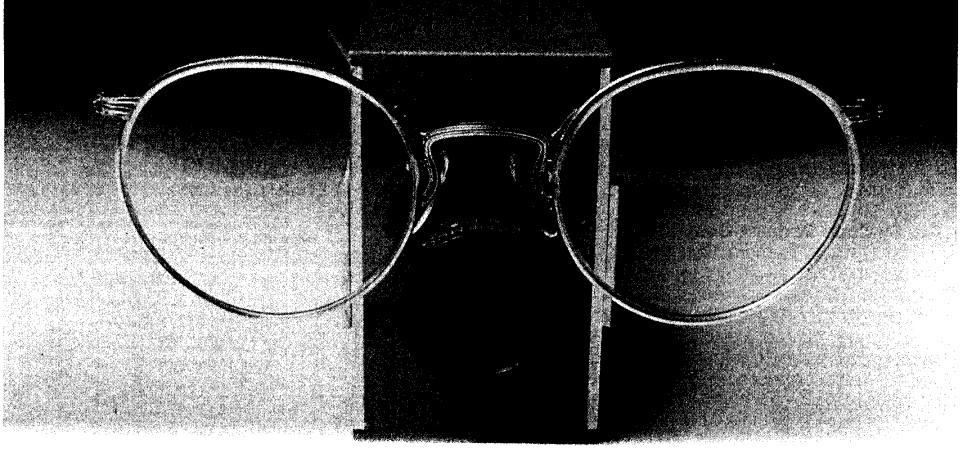
$195-kit
$215-kit with IC sockets
$350-assembled and tested DAZZLER

*"Build the TV DAZZLER" by Terry Walker, Roger Melen, Harry Garland, and Ed Hall. Popular Electronics, Feb., 1976.

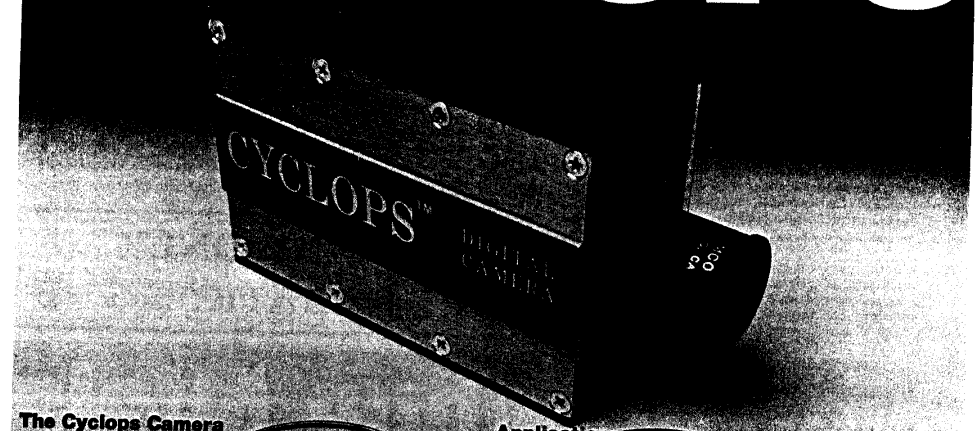
NOTE: Static memory is required in the Altair 8800 when interfacing with the TV Dazzler.

 SEE PAGE 11

Eyes for your Altair



CYCLOPS



The Cyclops Camera

The Cyclops Camera (88-ACC) is a general purpose digital camera [illegible] image sensor. In particular, the Cyclops is designed to be used with the Cyclops Camera Controller (88-CCC) to interface the camera to the MITS Altair 8800 computer.

The camera is equipped with a 25mm, f-2.8 lens using a standard "D" mount. Digital differential output is provided for interfacing with digital systems.

All connections to the Cyclops Camera are made to a 9-pin connector on the rear of the camera. For proper operation the Cyclops requires a positive and negative power supply (unregulated), a clock signal, and a reset signal. These signals are all available from the Cyclops Camera Controller (88-CCC) interface.

The Cyclops Camera is built in an extruded aluminum case with an attractive blue baked enamel finish. The camera is very compact measuring just 4½ x 2¾ x 1¾ inches.

Applications

Applications include security systems, image recognition systems and automated control systems.

Prices

The Cyclops Camera and the Cyclops Camera Controller come in both assembled and kit form.

Cyclops Camera (88-ACC-K) Kit	$180
Cyclops Camera (88-ACC-W) Assembled	$235
Cyclops Camera Controller (88-CCC-K) Kit	$260
Cyclops Camera Controller (88-CCC-A) Assembled	$340

Include $5 postage and handling. California residents add 6% sales tax.

CROMEMCO

One First Street/Los Altos, CA 94022

The Cyclops Camera Controller

The Cyclops Camera Controller is designed to interface the Cyclops Camera to the MITS Altair 8800 Computer. The controller consists of two boards that plug directly into the Altair computer.

The controller supplies all clock and power supply signals required for Cyclops, and permits software control of exposure, frame rate, and memory allocation for picture storage. Direct Memory Access (DMA) is used to store the picture in the computer memory.

Cyclops Coupon

☐ Enclosed is a check for $_____

☐ BankAmericard #_____ Exp. Date_____

☐ Master Charge #_____

☐ Cyclops Camera ☐ Kit ☐ Assembled

☐ Cyclops Camera Controller ☐ Kit ☐ Assembled

☐ Please send complete information package

NAME_____

ADDRESS_____

CITY_____ STATE & ZIP_____

Cromemco/One First Street/Los Altos, CA 94022 Phone (415) 941-2967

Letters to the Editor

Dear David:

This is to inform you that the various groups in and around Chicago (of small-computer users) have been merged into one organization, CACHE, and our mailing list is well over 150.

Our organization address is:
CACHE
PO Box 36
Vernon Hills, IL 60061

Computer Notes continues to improve. The Nov./Dec. issue is by far the best, with the borrowings from Byte and Creative Computing. I am sure that all concerned would appreciate it if you would publish the journal in 8 1/2 by 11 format, so that it could be inserted into a ring binder.

Keep up the good work.

Sincerely,
Bill Precht
Chicago, IL

Dear David,

Since my 13 year old daughter has a flair for writing and poems, she made a homemade bookmarker with a picture of an Altair 8800 computer on it. I thought you might enjoy the poem she composed and printed on the marker.

THE COMPUTER

The computer is said
To be man's best chum
I'd say because a computer isn't dumb

It can make true
All of man's wildest dreams
And never goof up
Or so it seems

So when you hear a man say
That his life is so bare
It is because he doesn't own
8800 Altair.

If you have room, I would appreciate your printing this poem in the newsletter. I know she would enjoy seeing it there. Thanks.

William D. Thomas

Chess Playing Program

Dear Sir:

I recently purchased a Martin Research MIKE-2 Micro Computer and have designed a chess playing program which may be of interest to other Micro Computer users.

To begin with, it requires 4K of memory. I chose to read the program into RAM using a TCH tape cassette interface, input moves via a Clare-Pendar keyboard, and output a picture of the chess board position to a TV screen via a Digital Group 8-line CRT interface. The TV picture appears as shown below:

```
BR BN BB BQ BK BB BN BR
BP BP BP BP BP BP BP BP
51 52 53 54 55 56 57 60
41 42 43 44 45 46 47 50
31 32 33 34 WP 36 37 40
21 22 23 24 25 26 27 30
WP WP WP WP 15 WP WP WP
WR WN WB WQ WK QB WN WR
```

The computer always plays white in this first version, but this will be an option later. In this illustration, the computer has made the first move -- P-K4.

Moves are entered via an ASCII keyboard as follows:

P65,45 would move the Black pawn to square 45.

I have designed two overlays for the program, but only the second overlay actually plays chess, and it may be used as a stand alone program if desired. The first overlay is designed to play "book" openings--the user could tailor his own initial responses to suit himself. When the user makes a move which does not exist in the first overlay "book", the program automatically calls the TCH recorder interface and reads in the second overlay which evaluates the board to see if a checkmate is possible--within a specified search depth. If no mate is possible, as is the normal case, the program chooses a "best" move using a series of adjustable criterion. Needless to say, the best move evaluation is an ongoing investigation and will be constantly updated.

The "book" overlay program is a simple search algorithm, however, the chess playing overlay is complex in design. It cannot be said to play good chess yet. Primarily, it is a fun program which a beginner to average player would find stimulating. The opportunities for improving its performance, however, are manifold, and I believe it will eventually prove to be a challenging competitor.

The program has the following limitations in its first version:

1. It does not recognize illegal user moves, nor can it recover from mistaken keyboard entries.
2. It cannot "castle" or take a pawn "in-passing". It will not object to the user making these moves, however.
3. The computer plays white--for now.

These limitations are not a serious compromise and should be easily improveable.

It should be added that the algorithm which is used in this 8008 application is equally amendable to the 8080 or 6800 CPU users.

Finally, I would like to say that I have spent a great deal of time and effort in the design and implementation of this program--spanning over five years. I would like to think that it may have some commercial value. To that end, I am contacting several potentially interested parties and would be very curious as to your comments.

Sincerely,
John Ford
5561 Esplanada Ave.
Santa Maria, CA 93454

Dear David:

KOMPUUTER PHREQUEHAUTE! @3#%¢?&! On page 9: That nice little routine under interrupt in your serial I/O article October issue of COMPUTER NOTES has one little bug in it, no one seemed to notice. (Just thought I'd make note on it before someone tries to run it as is). Line 11 should read:

```
2   11   010 Jump to self and
         wait for interrupt.
```

As is, the machine DAD's on jump back!

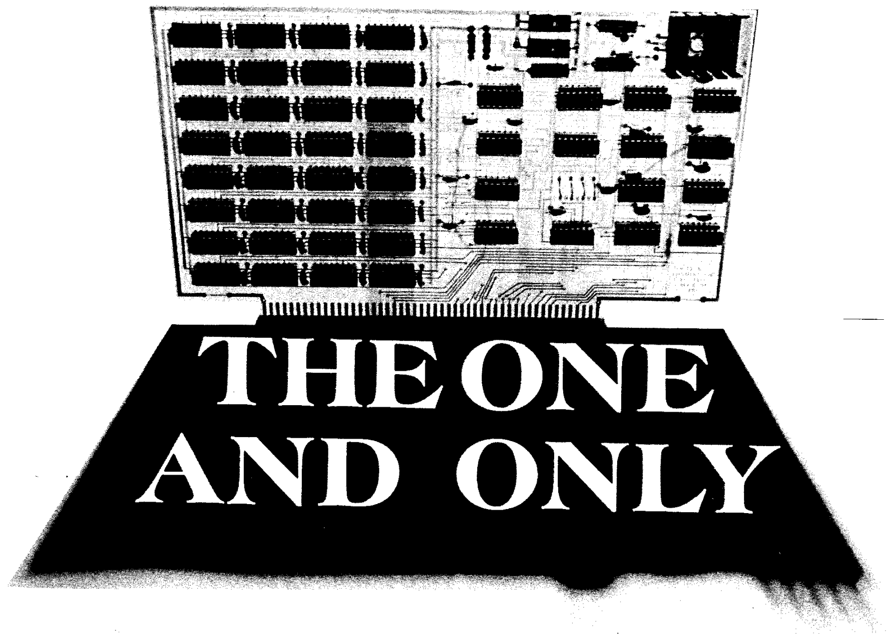
I'd also be interested in hearing from some of the computer Phreques in this area (very upstate New York) so you can put my name on your pencil pal list.

Cordially,
Harris G. Bruch
Plattsburgh, NY

Dear David,

Thank you for your very lucid and persuading answer to my letter about changes to the Altair. I was a little surprised to see it in Computer Notes, but I suppose many people would be wondering about modifications and it is reassuring to know that we owners can get the real scoop. Let me take this opportunity to tell you that Computer Notes started off at a good level and has been improving every issue. I really look forward to receiving it. Please thank Tom Durston, a telephone friend of mine, for the excellent article on the "I/O Programs for the ACR".

Sincerely, Dr. George L. Haller
Naples, Florida



ALTAIR 4K STATIC

THE ONE AND ONLY ALTAIR 4K STATIC

ALTAIR 4K STATIC FROM MITS IS UNQUESTIONABLY THE FINEST 4K STATIC MEMORY AVAILABLE ANYWHERE. IT IS ALSO THE FASTEST.

ALTAIR 4K STATIC USES INTEL 2102 A-4 MEMORY CHIPS WHICH HAVE A WORST CASE ACCESS OF 450 NANOSECONDS AT 70 C. AT NORMAL SYSTEM TEMPERATURES THE ACCESS TIMES ARE TYPICALLY LESS THAN 300 NANOSECONDS.

ALTAIR 4K STATIC IS FULLY ISOLATED FROM THE SYSTEM BUS BY SCHMITT T.M. TRIGGERS. THUS, THE EXCESSIVE CAPACITIVE LOADING CAUSED BY OTHER 4K STATIC MEMORIES IS ELIMINATED. USE OF THESE TRIGGERS ON ALL ALTAIR 4K STATIC INPUTS GREATLY REDUCES NOISE. INTERNAL DATA COLLECTION NODES ALSO USE SCHMITT TRIGGERS, WHICH PREVENTS INTERNAL DATA BUS NOISE FROM BEING TRANSMITTED TO THE SYSTEM DATA BUS.

ALTAIR 4K STATIC USES THE NEW 8800B SCHOTTKY NOISE SUPPRESSION SCHEMES. ALL LOGIC USED IN THE CARD IS LOW POWER SHOTTKY, WHICH RESULTS IN THE SPEED OF STANDARD TTL AND USES ONE FOURTH THE POWER.

ALTAIR 4K STATIC IS THE ONLY 4K STATIC SUPPORTED BY MITS. OWNERS OF ALTAIR 4K STATIC ARE ELIGIBLE TO QUALIFY FOR DISCOUNTS ON ALTAIR BASIC AND OTHER MITS PRODUCTS.

ALTAIR 4K STATIC IS THE ONLY 4K STATIC THAT COMES WITH ALL THE REQUIRED ALTAIR HARDWARE INCLUDING EDGE CONNECTORS AND CARD GUIDES.

ALTAIR 4K STATIC IS THE ANSWER FOR ALTAIR OWNERS WHO NEED STATIC MEMORY FOR SPECIAL APPLICATIONS SUCH AS THE TV DAZZLER FROM CROMEMCO.

PRICES:

ALTAIR 4K STATIC KIT.....$159
ALTAIR 4K STATIC KIT WITH 2K MEMORY...$174
CHIP SET TO CONVERT 2K TO 4K.....$45

NOTE: BEFORE YOU ARE MISLED BY "LOWER" PRICES ON OTHER 4K STATIC MEMORIES, ASK YOURSELF THESE THREE QUESTIONS: 1. IS THIS CARD ENGINEERED AS WELL AS MITS 4K STATIC? 2. DOES IT COME COMPLETE WITH EDGE CONNECTORS AND CARD GUIDES? AND 3. DOES IT QUALIFY ME FOR MITS ALTAIR DISCOUNTS?

AS WITH OTHER 4K STATIC MEMORIES, ALTAIR 4K STATIC WITH A 5 VOLT CURRENT EXCEEDS THE SYSTEM BUS SPECIFICATIONS. WHILE POWER CONSUMPTION IS LESS THAN MANY 4K STATICS, IT STILL DRAWS APPROXIMATELY 1 AMP.

ALTAIR IS A REGISTERED TRADEMARK OF MITS, INC.

ALTAIR 4K STATIC ORDER FORM

☐ ENCLOSED IS CHECK FOR $ _____
☐ BANKAMERICARD NO. _____ ☐ OR MASTER CHARGE NO.
☐ ALTAIR 4K STATIC KIT ☐ ALTAIR 2K STATIC KIT
INCLUDE $3 FOR POSTAGE AND HANDLING

NAME _____

ADDRESS _____

CITY _____ STATE AND ZIP _____

MITS/2450 ALAMO SE/ALBUQUERQUE, NM 87106

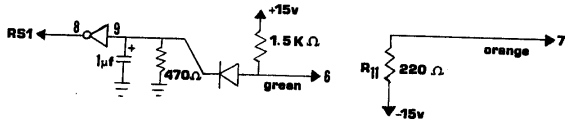
SIOC REV 0 MOD

In house, the SIOC Rev 0 boards underwent two significant changes. One change, the more apparent of the two, concerns the location of the interface connections between the TTY and the board as well as the nature of the interface electronically. First the position of the pins between the computer and the TTY will be redefined to be more compatible to the Rev 1 cabling. The color code is arbitrary but provided in order that the client is aware of MITS standards.

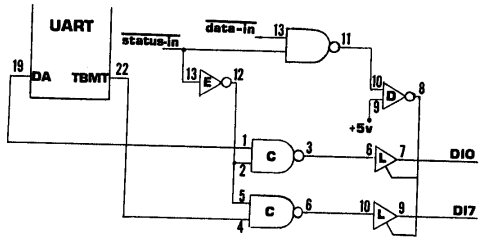
Molex Conn on Board	Color	Male & Female 25-Pin Conn.	Term Lugs in TTY(ASR-33)
4 (GnD)	Black	2	6
5 (Tran)	Red	3	7
6 (Rcv)	Green	4	3
7 (Rcv)	Orange	5	4

It might also be noted that the SIOC board is capable of only functioning with full duplex transmission. That is, a White-Blue lead and a Brown-Yellow lead should be moved from term lugs 4 and 3 respectively to lug 5. Unless some major change has been made in the interface electronics the SIOC board will have difficulty in functioning with a 60mA current loop in the TTY. The desired 20MA loop can be obtained by 1) moving a Purple lead from lug 8 to lug 9. 2) Moving the left-most terminal on R1 (a 4 connection resistor located about 8" back from the line/local switch on the base plate) to the left-most terminal connection of R1 (from the 3rd connection to the 4th connection).

The Rev 0 modified electronic interface and the Rev 1 interface appear below:



The second modification of the Rev 0 board (called hardware interrupt) is in the status provided to the data bus by the UART. The new board will pull DI0 low on the DA high condition (input status) and will pull DI7 low on favorable output status. The logic follows.



Cassette Test Pattern

Effective 1 Feb. 1976, all cassette tapes shipped from MITS will have an additional test pattern recorded on the back side. After the 125 test pattern (90 sec.), there will be 1 minute of a 175 test pattern. This is to check the demodulator on a non-symmetrical signal. Use the 333,007, single step twice procedure to examine the data coming from the A. C. R. to check for the correct pattern on the 175's.

Recording/Playback Method Changed

Effective 1 Feb. 1976, all cassette tapes shipped from MITS will be recorded on the left channel only. This will improve performance on playback on monaural and stereo cassette machines. If you use a stereo recorder connect only to the left channel for recording and playback.

MITS MOBILE Computer Caravan

The MITS Mobile Caravan will be on tour again in the near future. As most of you are aware, we are in the process of moving to a new facility. This, in conjunction with the Altair Convention (WACC) in March has caused some delays in the tour.

The Texas trip which tentatively includes Lubbock, Dallas/Ft. Worth, Houston, Corpus Christi, San Antonio, Austin, Odessa and El Paso, will probably begin in late March or early April. The West Coast trip will follow sometime during May.

We are currently working on changes in the seminar format. These modifications will include additional information in the course manual, more equipment demonstrations, and more thorough coverage of hardware and software.

Submitted by Pat Ward
1/6/76

GOOFS!

On page 24 (back page) of Nov./Dec. 1975 Computer Notes, there is a diagram for tape recorder motor control. The pins for the IC "U" are incorrect as shown.

Input-Pin 12 of IC "U" should be Pin 4.

Output-Pin 11 of IC "U" should be Pin 5.

Pin 9 of Molex connector should be Pin 6.

Two Monitors Top Software Contest

by Bill Gates

Twenty programs were added to the library this month, ranging in size from 9 bytes to more than 1800. For those users who are learning about machine language programming, the short programs in the library can be useful as learning aids to illustrate basic concepts. User groups might consider ordering most of the programs in the library to put in a notebook to make available to members. If enough groups are interested in this, we'll have a special "package deal" price. The long programs, of course, are the most useful and are sometimes written by people with lots of experience. The main problem with these is loading them onto paper tape or cassette takes a long time. I encourage users to share machine readable copies of these programs. This will be simplified if everyone stores programs in either a straight dump format; or in the MITS checksum format (using the checksum dumper in the library).

Another source of 8800 programs is the Intel 8080 library. Paying to join is expensive, but contribution of a substantial program entitles you to a free membership. They have all the floating point subroutines, numerous cross-assemblers, and some neat games (Blackjack, Life, Kalah). MITS has no objections to users submitting programs to both libraries.

Both the first and second place major programs this month are monitors--a utility program to help run, load, save and set up other programs. For users with 8K or more, Package 1 is the best thing to use, but for users with less memory, a smaller monitor with features to examine or modify core locations and save or load programs is very handy. Both monitors are small enough to be put on ROMs.

The programs that tied for third place, both by Lee Eastburn who gets a double prize ($30), not half for each, are both good demonstration programs. I think those users who think such programs are complex to write will be surprised to see how easily they understand the logic in these two programs. That's not to say they took very little time to write. The data for BLAZER fills over a 100 of our coding sheets, all of which had to be finished, experimented with, and typed up.

I hope everyone sends in the programs they wrote over the holidays so next month will be another record month for the library.

FIRST PLACE MAJOR PROGRAM
#1203751
Author: Jim Gerow
Length: 867 bytes
Title: 8800 Mini-Monitor
A complete monitor that provides several commands:

Examine	- prints contents of a single location
Deposit	- store into a location
Program	- give a start address and program data
Run	- start execution at a given address
Tape	- I, O, or V -- allows programs to be dumped to, input from, or verified on cassette.
Search	- scans memory for a specific value.
Clear	- zero locations between two addresses
List	- display memory contents between two locations.

SECOND PLACE MAJOR PROGRAM
#1217552
Author: Walter King
Length: 384 bytes
Title: Micro-operating System
An extremely handy operating system that allows saving and loading from cassette, dumping of memory locations, running of programs; and provides subroutines for character input, character output, string output, cassette input, and cassette output. Functions such as program loading can be done with program calls.

—CONTINUED PAGE 14—

MAINTENANCE SOFTWARE

By Harvey Lee

Maintenance software serves two primary purposes. First, it aids in identifying problem areas that exist within a computer system. Second, by not finding any problems, it should give a high degree of confidence in the system. With these thoughts in mind, we use several different programs in the process of checking out repair and production computers.

Let us consider how one short program can be of use in this regard.

Sense Switch Read

TAG	MNEMONIC	ADDRESS	OCTAL CODE	EXPLANATION
STRT	IN S.SW	000,000	333	Input sense switch data (I/O channel 377)
		001	377	to the accumulator
	STA 100	002	062	Stores the contents of the accumulator
		003	100	in memory location 100 (octal)
		004	000	
	JMP STRT	005	303	jump to the start of the program
		006	000	
		007	000	

This program reads the sense switches (A8 through A15), then stores the information in memory location 100. Thus, when memory location 100 is displayed, the data lights should reflect the positions of the sense switches. For example, if switch A8 were up, then D0 would be on. Or if A15 was down, then D7 would be off.

What this program tells the technician depends on his knowledge of the computer. If the technician has trouble entering or checking the program in the computer's memory, he knows the proper portions of the D/C board, CPU board, and memory boards to check for the problem.

After loading the program, he should then single step through the program. By observing as it executes the program step by step, he can often identify a problem area on the CPU board or on the D/C board. —CONT P. 17—

SOFTWARE

SOFTWARE NOTES

Multi-Precision Arithmetic

By Bill Gates

On the 8080, multi-precision unsigned arithmetic is made easy by the carry bit and its affect on the instructions "ADC" and "SBB". Unsigned arithmetic treats all numbers as positive, with all zeros being the least number, and all ones being the highest. Adding and subtracting memory addresses is the most common form of unsigned arithmetic. Multi-precision arithmetic must be used whenever the range of values desired is greater than that accepted by the arithmetic unit of the computer you are using. The 8080's arithmetic units accept 2-8 bit operands, one from the [A] register and the other from B, C, D, E, H, L, contents of address in [H, L] or the byte following the arithmetic instruction (immediatemode) so anytime values greater than 255 are to be accepted, multi-precision arithmetic must be used. "DCX", "INX" and "DAD" allow 16-bit quantities to be added to, or subtracted from, so 16-bit arithmetic could be considered single precision. However, the lack of any 16-bit arithmetic instructions that use carry to affect their result make it more appropriate to consider 16-bit arithmetic multi-precision.

```
;16-bit unsigned add [H, L] = [D, E] + [H, L]
 ADD16U:  DAD D
;carry is set as an overflow indicator
```

```
;32-bit unsigned add [D, E, B, C] (M + 3, M + 2, M + 1, M)

 ADD32U:  MOV A, C
          ADD M
          MOV C, A
          INX H
          MOV A, B
          ADC M
          MOV B, A                    ;carry is returned as an overflow
          INX H                       ;indicator
          MOV A, E
          ADC M
          MOV E, A
          INX H
          MOV A, D
          ADC M
          MOV D, A
          RET

;16-bit subtract [H, L] = [D, E] - [H, L]

 SUB16U:  MOV A, E
          SUB L
          MOV L, A
          MOV A, D
          SBC H
          MOV H, A
          RET

;carry indicates that [H, L] was greater than [D, E]
;32-bit subtract [D, E, B, C] = (M + 3, M + 2, M + 1, M) - [D, E, B, C]

 SUB32U:  MOV A, M
          SUB C
          MOV C, A
          INX H
          MOV A, M
          SBC B
          MOV B, A
          INX H
          MOV A, M
          SBC E
          MOV E, A
          INX H
          MOV A, M
          SBC D
          MOV D, A
          RET

;carry indicates [D, E, B, C] was greater than (M + 3, M + 2, M + 1, M)
;add 8-bit [A] to 16-bit [B, C] unsigned.  Result in [B, C]

 AD816U:  ADD C
          MOV C, A
          ADC B
          SUB C
          MOV B, A
          ;no overflow indication is given
```

—CONTINUED P. 15—

SOFTWARE CONTEST

THIRD PLACE MAJOR TIE
#1201751
Author: Lee Eastburn
Length: 309 program bytes,
1536 data bytes
Title: BLAZER
Prints 4 block letters using 64 characters across and 12 lines vertically. All printing characters are provided for, and "custom" characters can easily be added. Allows for characters to be printed and background blank, or character to be blank and background printed. The characters used in making the blocks are part of the input to the program.

FIRST PLACE SUBROUTINE
#1124753
Author: George Rompot
Length: 44 bytes
Title: DIV 16
Divides two 16-bit unsigned integer values. Returns a 16-bit quotient and a 16-bit remainder. Uses subroutine #1124752.

SECOND PLACE SUBROUTINE
#1201752
Author: Jeffrey Clark
Length: 59 bytes
Title: Display Clock
A clock subroutine which displays either the second, minute, hour, or day in the address lights, depending on the sense switches. The initial time can be entered in locations 100-103.

#1124751
Author: George Rompot
Length: 7 & 9 bytes
Title: RDEL & RHLDEL
A 16-bit left shift routine for [D, E] and a 32-bit left shift routine for [H, L, D, E].

THIRD PLACE MAJOR TIE
#1217751
Author: Lee Eastburn
Length: 335 bytes
Title: Calendar Printer
Prints a calendar for any month. When the box for each day is typed, the user can type a single character to select one of the preset messages or a special message can be typed in. The format is 71 characters across (10 per day) and 64 lines vertically. The heading includes month name, and day name for each column.

#1124752
Author: George Rompot
Length: 26 bytes.
Title: MPY8
A subroutine to multiply [C] by [D, E], leaving the result in [H, L]. Overflow is handled by a special return sequence. The 2-byte left shift routine in #1124751 is called.

—CONTINUED P. 15—

In signed arithmetic (2's complement) half the numbers are treated as negative and the other half as positive. A 1 followed by all zeros is the smallest number. All 1's is the largest negative number (-1), and all 0's is the smallest positive number. A zero followed by all ones is the largest number. Note that the absolute value of the smallest number (a one followed by all zeros) is larger than the largest number. This creates an overflow case for negation, and makes subtraction tricky if this special case is handled. This "special" negative number is -32768 if 16-bit signed arithmetic is used. This highest 16-bit signed number is 32767.

This signed format allows two numbers to be added through a simple DAD. The only complication is checking for overflow. The table below gives the different possibilities for adding signed numbers:

	Arg 1	Arg 2	Carry	Sign of Result	Overflow
1	pos	pos	off	neg	yes
2	pos	pos	off	pos	no
3	pos	neg	off	neg	no
4	pos	neg	on	pos	no
5	neg	neg	on	neg	no
6	neg	neg	on	pos	yes

Overflow only occurs when the result of adding two positive numbers is greater than 32767, or the result of adding 2 negative numbers is less than -32768. The formula: (⊗ means exclusive - or)

Sign of arg 1 ⊗ sign of arg 2 ⊗ carry ⊗ sign of result is 1, if and only if overflow occurred. Subtraction is merely a negation followed by an addition, unless -32768 is being subtracted (i.e. -30 - (-32768)), in which case no negation is necessary, but the sign of -32768 as an addend must be positive.

```
;16-bit signed negate [H, L] = -[H, L]

NEGIGS: XRA A                   ;get negative [L]
        SUB L
        MOV L, A
        SBB H
        SUB L
        MOV H, A                ;[H] = -[H] - borrow if any
        SUI 128                 ;see if -32768 (decimal)
        ORA L                   ;with [H] = 128, [L] = 0
        RNZ                     ;if not, return
        JMP OVERFL              ;overflow here

;16-bit signed add and subtract [H, L] = [D, E] = [H, L]

SUBENT: CALL NEGIGS             ;entry from subtraction
ADD16S: MOV B, H                ;MSB of [B] = sign of arg 1
SBZENT: DAD D                   ;do the add
        RAR                     ;MSB of [A] = carry
        XRA B                   ;XOR in sign of arg 1
        XRA D                   ;XOR in sign of arg 2
        XRA H                   ;XOR in sign of result
        RP                      ;return if MSB ø
        JMP OVERFL              ;otherwise there was overflow

;subtract [H, L] from [D, E]
SUB16S: MOV A, H                ;is it -32768?
        SUI 128
        ORA L
        JNZ SUBENT              ;if not, just negate and add
        MOV B, L                ;say sign is positive
        JMP SBZENT              ;do the add
```

USR Routines

There are two ways for a "USR" routine to get the argument passed to it as a signed integer in [D, E]. The easiest is to use a CALL followed by the two byte address in locations 4 and 5. The only disadvantage to this is that it has to be changed when you get a new version of BASIC, since the address in locations 4 and 5 changes from version to version. The alternate way is long, but doesn't have to be changed when you get a new version of BASIC. It is:

```
        LXI H, BACK LC
        PUSH H
        LHLD 4
        PCHL
BACKLC: rest of USR routine
```

-CONTINUED P.17-

— SOFTWARE CONTEST

#1125751
Author: Christopher Terry
Length: 23 lines
Title: INDEL
Basic subroutine "INDEL" which inserts a fixed length digit string in, or deletes it from a string which can contain multiple digit strings. Several examples are given.

#1125752
Author: Mathew Smith
Length: 56 bytes
Title: Random 16
Generates a 16-bit random number. All possible numbers are generated before any number repeats.

#1125753
Author: Craig Pearce
Length: 43 bytes
Title: Number Guessing Routine
A well-documented routine that uses the front panel to play a high/low number guessing game with the user. (Good illustration of basic techniques.)

#1125759
Author: J. Scott Williams
Length: 50 lines
Title: Basic Cassette Dumper
A very handy program that allows data on cassette to be printed out in ascii, octal, hex, or decimal. Full instructions and examples are given.

#1201753
Author: Jeffrey Clark
Length: 9-136 bytes
Title: RELJMP
Uses an RST subroutine to allow 2-byte jumps to locations within 128 bytes of the jump. Similar to M6800's "BRA", and can be used to make long programs shorter.

#1201754
Author: Jack Coats
Length: 12 bytes
Title: Vectored Branch
A subroutine that branches back to a location 2 * [A] beyond the call.

#1201755
Author: Gary Rupert
Length: 21 bytes
Title: Set Memory Demo
Stores the low 8-bits of each address in that address up to a specified limit.

#1208751
Author: George Burditt
Length: 16 bytes
Title: Vectored RST
Allows up to 256 3-byte RST instructions by using a normal RST followed by an argument. Non-interruptable and impractical, since a CALL to each routine would be simpler, but very illustrative of stack techniques.

-CONTINUED P.16 —

Here are 4 example USR routines written for 4K and 8K BASIC. In Extended BASIC, the argument is passed and returned in [H, L]. So appropriate modifications will have to be made to use these with Extended BASIC.

#1 function: turn interrupts ON if argument is negative.
turn interrupts OFF if argument is positive.

```
        CALL <address at location 4>
        MOV A, D                    ;get argument high order
        ORA A                       ;set MINUS if negative
        DI                          ;assume positive
        RP                          ;return if so
        EI                          ;otherwise, turn interrupts on
        RET
```

#2 function: Delay for 11.5u seconds * argument + overhead

```
        CALL <address at location 4>
LOOPDL: DCX D                       ;decrement the argument
        MOV A, E                    ;is [D, E] = 0?
        ORA D
        JNZ LOOPDL                  ;if not, continue looping
        RET
```

#3 function: Execute instruction or instructions in [D, E]. Return value of [A].

```
        CALL  address at location 4   ; get argument in [D, E].
        XCHG
        SHLD INSTRS
        LXI, H 0                    ;save the stack pointer
        DAD SP
        SHLD STORSP + 1
        LXI SP, PSWLOC + 1          ;set up to read the USR ac-
        POP PSW                     ;cumulators, fetch the USR
        POP H                       ;accumulators
        POP D
        POP B
INSTRS: DS 2
        PUSH B                      ;store the USR accumulators
        PUSH D
        PUSH H
        PUSH PSW
        MOV B, A                    ;return the contents of [A].
        XRA A                       ;as the result in [A, B]
STORSP: LXI SP, 0                   ;restore the stack pointer
        LHLD 6                      ;convert [A,B] and re-enter
        PCHL                        ;the BASIC program
PSWLOC: DS 8                        ;the accumulators can be set
                                    ;up and examined by using
                                    ;PEEK's and POKE's on these
                                    ;locations
```

#4 function: dispatch to one of several subroutines depending on the high 8-bits of the argument

```
        CALL <address at location 4>
        LXI H, TBLLOC               ;point at dispatch table
        MVI B, 0                    ;get dispatch offset
        MOV C, D                    ;in [B, C]
        DAD B                       ;add in 2 * offset since
        DAD B                       ;table entries are 2 bytes
        MOV A, M                    ;fetch the dispatch address
        INX H                       ;into [H, L]
        MOV H, M
        MOV L, A
        PCHL                        ;dispatch

TBLLOC: DW USRZER                   ;address to go to on zero
        DW USRONE                   ;on one
        DW USRTWO
```

Note on allowing interrupts

To allow interrupts a program must always leave 16-bytes of free stack space. If multiple interrupts can come in 16 *(maximum number at once) bytes must be left free. Also no tricks involving INX SP or DCX SP can be used. The third example USR routine is not interruptible, since an interrupt following the LXI SP would not work.

Next Month: signed and unsigned multiplication and division.

#1209751
Author: Alan Miller
Length: 20 bytes
Title: Count in Lights
Using a modifiable delay period, this program counts in the upper eight address lights.

#1216751
Author: Sasan Ardalan
Length: 15 bytes
Title: 8-bit BCD to Binary
A non-looping BCD to binary routine. Execution takes 42 microseconds and all registers are saved.

#1223751
Author: Frederick Dingwall
Length: 48 bytes
Title: 8-bit Octal Input
Inputs 3 ascii characters which are converted to an 8-bit number which is stored in memory. Allows contents of consecutive locations to be stored in order of input.

#1223752
Author: Frederick Dingwall
Length: 48 bytes
Title: 8-bit Hexadecimal Input
Same as #1223751, except numbers are input as 2-character hexadecimal constants.

see you at the WACC!

MARCH 27-28

by Paul Wasmund

PACKAGE I REVISITED

After loading the MONITOR as described in Appendix E of the Package 1 manual, use the following example to load the editor.

Example 1.

To load the editor from a TTY:

1. Type the following monitor commands.

   ?OPN MAG,TY,A

   ?EDT,MAG

2. Put the paper tape in the reader and turn the reader on.

To load from an ACR:

1. If you stopped the cassette within 5 seconds after the monitor loaded, skip the next step.

2. Rewind the cassette. Turn the recorder to play and wait 3 minutes and 45 seconds. Stop the recorder.

3. Type the following monitor commands.

   ?OPN MAG, AC, A

   ?EDT,MAG

4. Turn the recorder to play. When the editor has finished loading, it will type:

   START INPUT

   *

Once the editor is loaded, you are ready to type in a program. Remember, the first statement of your program should be a ORG and the second should be a ORR if needed.

Let's now review the use of the ORG and ORR pseudo op's. The ORG pseudo op is used to define the locations that the program will run in, while the ORR instruction sets the location to save the program in while assembling.

Example 2.

You have written a program that is approximately 1K long. You want to assemble it to run in the lowest memory available in your machine, which is 3700Q when using the REV 2 monitor. You want to assemble it by using the version 1 (ASM) assembler. The first statement should be:

```
ORG 3700Q; Sets Run Loc

ORR 14000Q
```

The ORR statement was used because ASM ends at 11635Q, so your program could not be stored at 3700Q. 14000Q was chosen to leave space

Once you have typed in the ORG and possibly an ORR statement, you can type in the main body of your program. Do not include an END statement in your program. After the entire program has been entered, and any typing errors corrected, you are ready to dump off a copy of the program. To do this, return to the monitor, open MAG to your mass storage device (AC or TY), and give the ASCII option. Then return to the editor with the R option (i.e. EDT (R)), and dump the program using the Save (S) command.

Example 3.

```
*E
   ?OPN MAG,TY,A
   ?EDT(R)
*S

FILENAME=
CHANGE SENSE SWITCH 15 FOR DUMP
```

When this message is printed, turn on your punch and flip sense switch 15. If you have a ACR, change the open command as follows.

```
?OPN MAG,AC,A
```

and, after FILENAME=, type a 3 character file name followed by the carriage return.

NOTE: MAKE SURE THE CASSETTE IS BLANK OR YOU ARE POSITIONED AFTER THE LAST FILE ON IT.

Now, load the assembler by substituting ASM for EDT in Example 1. When the assembler has finished loading, the following message is printed.

```
ALTAIR LOADING ASSEMBLER VERSION
2.2
*ASM*
```

If you saved your program on the ACR, make sure the last time you opened MAG, you used the following command:

```
?OPN MAG,AC,A
```

If that's not true, return to the monitor, give the above command, and restart the assembler.

```
?ASM
```

If you have a paper tape of your program, load it into the paper tape reader and start it running. If you have a cassette of your program, type the following:

```
FILE NAM  ;INSTEAD OF NAM PUT
          ;THE 3 CHAR NAME YOU
          ;GAVE THE FILE IN THE
          ;EDITOR (FILENAME =
          ;FILENAM).
```

The assembler should return to the monitor when finished with your program. At this point, you should set up to make an absolute dump as follows:

```
?OPN MAG,TY  (Substitute AC for
              TY if you are
              dumping to a cas-
              sette.)
?ASM(P,A)  (If you want the
            symbol table dumped
            and/or an inverse
            assembly, include
            the options now.)
  *ASM*
END NAM   ;REPLACE NAM WITH THE 3
          ;CHARACTER NAME YOU WANT
          ;TO CALL THE PROGRAM.
SENSE SWITCH 15 FOR DUMP
```

At this point, turn on the tape punch or recorder, and flip sense switch 15. When your program has finished dumping, a message will be printed indicating this. You can now return to the monitor to test your program.

—MAINTENANCE SOFTWARE—

The last thing he does is to run the program, stopping to examine the memory for each of the sense switches. This assures proper sense switch reading, which is necessary when loading BASIC.

If the program fails to execute properly, the technician then begins his troubleshooting in the appropriate area.

If the program will run and single step, but yields the wrong data at memory location 100, he has several possible problem areas. First a visual inspection is made on the CPU board to verify R9 through R16 are 4.3K ohm resistors and 1C"Q" is a TI 74123. Next the timing relationship of Ø1 and Ø2 are checked. If this checks properly, he will then check the logic on the D/C board, starting at 1C"U" pin 8 (a logic low level that has a high going pulse when the sense switches are being read).

If the computer will operate correctly when either running or single stepping, but not both, the technician should begin his troubleshooting on the CPU board at 1C"R" pin 8 (normally high; goes low to enable DI data bus) and checks logic levels of 1C"O". If the CPU checks out, he will then proceed to the appropriate area on the D/C board to continue his troubleshooting. If improper operation results in writing alternating bit patterns through memory (usually 071 with 000) he begins his troubleshooting with 1C"O" on the CPU. These problems are generally caused by improper enabling of the DI data bus.

As can be seen, this program while short can be of great benefit to the technician. It aids him in identifying the area in which to start his troubleshooting. If he fails to find a problem, he feels confident in the proper operation of the majority of the computers circuits. He is now ready to use more complex programs in checking out the computer. We plan to discuss more of the programs used by us to check out the ALTAIR 8800 in the future. If you hav

What's Going On In Here

Which Depts. to call for answers to questions.

In our new building we will have a switchboard operator to help us with the growing number of calls we are getting. Therefore, we hope to be giving you better service; however, please be sure that you ask for the specific department you need for answers to your questions. Our departments stay open from 8:00 to 5:00 Mountain Standard Time. However, the Service Department does close down at 4:30 MST.

Ask for the Marketing Department for answers to your price and delivery and "where is my order" questions. The Service Department will take care of your technical problems as you're building or running your machine. The Marketing Department can take care of general technical questions. For example, if you want to know which I/O you need to run the Southwest Technical Products Terminal; ask the Marketing Department--you don't have to bother the Service Department. The Software Department should be called with specifics regarding problems with your tape or questions as to commands, but consult Marketing as to where your order for software is. Accounting rarely needs to be consulted, but if you asked for a refund quite some time ago, you might call them for information as to whether the check has been sent or not. With these few hints and the help of the operator, we hope that your long distance calls will be handled as expeditiously as possible. Thank you very much for your business and your cooperation.

by Pam.

What's Going On Out There

Damage during shipment of items returned.

In a few instances in the past we have received units back for repair that were sent to us with boards plugged into the chassis only. Without the case top and bottom, the chassis has very little protection. The postage for the weight of the case top and bottom is not as costly as the damage done to the unit occurring during shipment. We hope that by informing you of this problem, costly labor and special handling charges will not have to be passed on to our customers sending such units back. We thank you for your attention to this problem.

Altair Users

Les Slater
2933 Guilford Ave.
Baltimore, MD 21218

Craig Brockmeier
4715 SE Adams #922C
Bartlesville, OK 74003
(918) 333-5608

John Joseph Herro-W8LKW
4924 Woodman Park Drive
Apt. #1
Dayton, OH 45432
(513) 255-2341 (work)
(513) 252-0490 (home)

Frederick M. Dingwall
5917 North Harvard Ave.
Oklahoma City, OK 73122

Charles Wells
465 Northgate Pass
Roswell, GA 30075

John F. Dickson
100 Lakewood Rd.
Neptune, NJ 07753
(201) 774-3153

Glen Seaton
36 Celestial Court
Cape Canaveral, FL 32920
(305) 784-1419

John F. Gardner
900 Embury Rd.
Penfield, NY 14526

David Gilmer
232 Lark
Tumwater, WA 98502

Eric D. Robbins
Rt 1 Box 66
Dagsboro, DE 19939

R. Palmer Benedict
10 Radcliffe Rd.
Wellesley, MA 02181
(617) 235-0057

Glen Charnock
560 W. Gonzales Rd. #22
Oxnard, CA 93030

Stanley D. Davis
R.D. 1
Stittville, NY 13469

Theo Pozzy
P.O. Box 752
Bangor, ME 04401

Edward Hurtack, Jr. WA2VWT
523 O'Donohue Ave.
South Plainfield, NJ 07080

Richard Curtiss
843 NW 54th
Seattle, WA 98107
(206) 784-8018

Harris G. Bruch
PO Box 1155
Plattsburgh, NY 12901

John Merritt
1969 Connaught Ave.
Halifax, NS CANADA

PUZZLE

contributed by
R. O. Whitaker, Engr.

Q1. A man starts at the point where the prime meridian crosses the equator and walks 45° Northeast, constantly checking the direction of his route using a geographic compass which always points toward the North geographic pole, Provided the man walks with equal facility on land and water will his journey ever end? If so, where? If so, how many kilometers will he have travelled when he gets there?

Q2. A man bound for Albuquerque (to buy an Altair computer) arrives at a crossroads where a filling station is located. He does not know which road to take. He knows that every station operator is either an absolute liar and always tells a lie or is perfectly honest and never tells a lie. What single question can he ask the operator that from the answer he can determine which road to take?

Q3. Which character in the following sequence is out of order?

A1. He spirals to the North pole. Arriving after traveling a distance of 10,000,000 2 meters. This distance because the meter is defined as 1/10,000,000 the distance equator to pole and for each increment he travels along his path, he travels 1/ 2 straight north.

A2. "Which road would you say leads to Albuquerque? The man then takes the road designated by the operator.

A3. Eleven. This set of digits for a base-16 numbering system (or something very nearly equivalent to it) was proposed by a Mr. Gabrielian of Newport Beach at a recent Computer Arithmetic Conference.

New PCC Publication

"Tiny BASIC"

TINY BASIC is about building your own BASIC and--maybe--it might grow into a publication about learning how to build your own software for the home/school/personal computer. We will publish ideas and software in the public domain for anyone to use. TINY BASIC will be a sharing thing with explicit permission to use the information in a non-commercial way. We will start modestly, promising 3 issues over the next few months. Bernard Greening is the editor of TINY BASIC with help from Dennis Allison and whoever wants to help.

TINY BASIC, Volume 1. Three issues for $3. First issue: January 1976.

Send to: TINY BASIC
PO Box 310
Menlo Park, CA 94025

Goats and dancing?

Blousemaker rips seams over business mini

By Donald Silverman
Minicomputer News

Business acquaintance named Thornton went seeking a computer system. Simple set-up to handle tasks like billing, payroll, and perhaps some inventory control. Straightforward work for a business system, you'd expect. Especially in a small business like Thornton's (he makes women's blouses).

Few days ago I saw Thornton, normally a light drinker, tippling heavily at the Ritz's cocktail bar.

"Thornton, what a surprise to see you here," I said, surprised. "What could possibly have driven you to drink like this?"

Thornton looked up at me. He slowly put down the Chivas Regal on the rocks he was about to gulp down. Thornton reached out to me as if for support, then said: "What am I, the only one that's crazy?

"These computer guys!" Thornton continued. "Oy, what a bunch! Go buy a computer from them! Like I know from computers like I know from . . . what? Nuclear physics?"

Thornton paused, as if he were waiting for my answer. But he continued, "So what happens?"

He paused again, looked around the room and pulled me close to him.

"One guy," Thornton started off quietly, "he carries on for hours with junk like RAMs and PROMs and who knows what else? Oh, a *big deal* he makes of them. *His* system, he says, comes with this kind of RAM, that kind of PROM.

"'So what's with this ram and prom?' I ask. 'You need goats and dances to run computers?'

"All right, so maybe I was pulling his leg a bit. But I need this RAM, ROM PROM junk? I need to worry what makes a computer work? I got worries enough how to make a blouse that sells. I need computer headaches?

"Then I run across Marvin, the *big* engineer. With Marvin you don't just *buy* a computer. Oh, no! With Marvin you understand every tiny detail on what makes it work or you don't buy.

"A number one head, this Marvin. Degrees from MIT, from this school, that school, institute of this, institute of that. You name it. But a salesman? With Marvin selling my blouses I'd go broke in a day. With Marvin you wouldn't just *buy* a blouse. You would have to know how the cloth was made, how the sewing machine works, why one button costs more than another button. A woman needs this? Look, a blouse is a blouse. Some are simple, some are fancy. If it fits, you buy it.

"Even so, Marvin was a big help. He was the one who helped me understand computers. Not everything. But enough so I wouldn't get scared every time I hear words like bit and bite. (Marvin spelled that b-y-t-e, like a fancy Englishman.)

"A bit he explained is a zero or a one. And a bite? That's just a bunch of zeroes and ones. The more bits and bites you put into a computer, the more money you got to pay. So now I understand. You bite off a bit of a bite and you end up with a bit.

"'Look, Marvin,' I said. 'I appreciate all you're trying to do for me. But tell me something. So far your computer doesn't seem any different from anybody else's computer. In what way is your computer so different from all other computers?'

"Oy, vay! A tornado I unleash. Suddenly Marvin is fuming and frothing and saying things like speed and access and input and output and faster and real time.

"'Marvin, Marvin, Marvin.' I shout out. 'Slow down, my child. You say your machine is faster? Okay, so how much faster?'

"Then Marvin, instead of telling me how much faster, he tells me his problems with seconds. 'Look, Marvin,' I said, 'don't tell me from *your* seconds. I get enough seconds from *my* Millie's. You know what it's like to find a good stitcher nowadays? You know what it's like to try to turn out quality at a price when nobody wants to work anymore?'

"From Marvin I go to Stanley. Stanley, a super salesman. What a peach. A real gem. Smooth talker. Smooth looker. He tells me not to worry about bits and bites and RAMs and ROMs and this and that. Don't worry. All I should do is tell him my *problem*, he'll give me a solution.

"So I tell Stanley I got billing to do. I got a payroll to worry about. I got an inventory that's costing me an arm and a leg. And Stanley, the smooth talker, pulls out a catalog. Thumbs through here and there. And comes up with a computer for me.

"Then he takes me into a showroom. And he shows me *my* computer.

"'Stanley, it's a beautiful machine. And you're sure it'll do the job for me?' Oh, he assures me up and down. So I look it over and I'm ready to sign on the dotted line when suddenly I notice something. 'Hey, Stanley,' I say. 'Just so I'm sure I know how it works and can explain it to the girls in the office, where is it I put my invoices to be typed?'

"Stanley stares at me like I'm crazy or something. He mumbles something about he's in the computer business not printers. If it's a *complete* package I want, I should go to a systems house.

"'A *what* kind of house?' I yell at the bum. 'Me, a reputable businessman?'

"So then Stanley, the bum, explains to me what it is, a system house. For people like me, who all we want is to

"If you had to buy a car like you have to buy a computer, we'd all be riding around on horses."

get a computer and everything else ready to go, then we go to a systems house and they deliver us a turkey system. Why they should call it a turkey, I don't know. But I should argue with a billion-dollar computer industry?

"Finally I say to myself, 'Thornton, you're finally getting somewhere. All you should do now is find a good systems house and you get a computer.'

"First thing happens I walk into the systems house, they want I should tell them which applications package I want. A what package? Applications? I don't know applications from apple strudel.

"I tell them I got a little billing, I got a little payroll, I got accounts who haven't paid me in 90 days. 'Fine,' they say, 'we can fit you out with our *standard* small business package.' And then I tell them I got inventory headaches. 'Aaaah!' they exclaim. 'It's no longer a *standard* package but a *custom* package. Cost a little *more*, but *not* to worry.'

"'Fine.' I tell them. 'Do it. And how much will it cost?'

"A runaround I get like I never got before. They look at me like I said a dirty word. They tell me 'it' depends. 'It' depends on how much of this I want? 'It' depends on how much time they need. It, it, it. Like to say price or money or cost is somehow obscene. Like you can't give me a firm price before I buy? Like when a store wants a custom blouse, I should't give them a firm price before they order? I'm supposed to say 'it' depends on how long my cutter takes, how long my stitcher patzkies around, how long my shipping clerk takes?

"What a business, this computer business. If you had to buy a car like you have to buy a computer, we'd all be riding around on horses."

Thornton's tale, part 2

When business acquaintance Thornton unfolded his tale of woes in buying a business computer, I thought that surely he was exaggerating. After all, all he wanted was a simple set-up to handle straightforward tasks like billing, payroll and inventory control.

Nevertheless, Thornton, while nursing another Chivas Regal on the rocks at the Ritz's bar, insisted on his tale's veracity.

"I should lie to *you*?" he said. "I should tell you I got troubles when I got no troubles?"

"Perhaps," I suggested, "there's something unique about manufacturing blouses, something that makes it less amenable to computerization than, say, manufacturing electronic parts."

"Aiiii," Thornton moaned. "If only I did make electric parts. At least I'd be making a decent living. You know what the mark-up on blouses is? You know what I got to go through just to break even?

"Blouses? Making blouses is so different from making anything else? Somebody makes it, somebody sells it, somebody buys it. In between, there's a lot of headaches, a lot of sweat and believe *me*, a *lot* of aggravation. You got workers you got to pay more than they're worth. You got stores who make you wait 60, 90, 120 days before they even *think* about paying you. You got suppliers who say they'll supply you at one price and then supply you at another price and tell you you don't like it, don't take it.

"What's so different? Look, every businessman, he's got the same problems. Even the computer people, they got problems. And I can sympathize with them.

"But they should sell computers like they sell cars. You go into a dealer, he shows you what he's got, he tells you the price, you pay him what he wants – all right! so you discuss it a little – and you drive off.

"But computers? I told you before, if Detroit sold cars like computer people sell computers, we'd all be riding horses. And we'd be a lot happier, believe me.

"First of all, you just wouldn't buy a *car*. You'd buy something with a fancy name like 'vehicular system.' "

"Or," I interjected, "an interactive modular component of a point-to-point, surface transportation system."

"And you think you could just go into a dealer and buy a complete package?" Thornton continued without pause. "You think so, you're crazy. You'd have to go *there* for tires, over *there* for upholstery, and over *there* for a horn. Why? I'll tell you why.

"Because the automobile dealer, he would say, 'I supply automobiles. I *don't* supply accessories.' Big shot! Hoo, hoo, hoo!

"I want tires, accessories – or what is it the computer people say? peripheries? – to a tire dealer I have to go. I want a radio, I should see a radio dealer. I want a steering wheel, I should see a steering wheel dealer. I want a bucket seat, I should see a bucket dealer.

"And I should be able to go into a showroom and say, 'Look, I need a car that can take me from here to there?' No, I would have to know *why* it can take me from here to there. I would have to know *how* it can take me from here to there. I should have to know how much gas flows through here. I should have to know how much oil flows through there.

"And all I want to know is, can it get me there? If it gets me there, with a little bit of style (you know, a seat that doesn't make you feel you're sitting on a piano bench, a little air conditioning, an engine that maybe at the lights lets you show off a little), I take it. I have to know how the gas gets from here to there? I have to know the engine is turning around a thousand times a second? I should care it turns? I should care how much it *costs* me every time it turns.

"I remember when Marvin, the computer genius, is telling me about two key this, four key that, eight key this, key that, key this. And *me? I'm* holding my head, and *I'm* crying, 'Oy vay, oy vay, oy vay!' And Marvin is going, 'Key, key, key.'

"Later, I find out it's not key, it's K and K means a thousand. So why can't he *say* a thousand. Like you buy a car with a 4K RPM motor? No, you buy a four *thousand* RPM motor. And why? Because automobile dealers speak English.

"Besides, where does this K come from? In grammar school, in *grammar* school already, I learn the abbreviation for a thousand is M. M, from the Roman numeral, not K.

"And when I buy a car, do I have to sit down with a hundred-page catalog and study it like I was a scholar? I have to sit there with tables this long and match this (what do they call it? module?) *part* with *this* part. I have to go into an automobile dealer and spend hours telling him what I need and why I need it, and then he tells me he'll give me only so much of the car and if I want ash trays I should go to the ash tray maker?

"No, I go to buy a car, and it's already set up to get me from here to there. So it's got a little more horsepower or a little less horsepower than maybe I think I want. That's such a big deal? The important thing is that *I* don't have the aggravation of worrying how to set the car up so it'll get me from here to there.

"And you know why? Because the automobile people are smart. They know what people want. They say here's a car for you, you want a little extra trim, fine, you don't want it, fine. Whatever you want, it's ready to go. So it's got a little bit more than I want? What's the difference? The time I save in aggravation, it's worth the price.

"You have to understand. All I want when I buy a car is I put in the key, I turn the key...and it runs. I don't have to change the way I do business. I don't have to turn my factory upside down. I don't have to wait for somebody to show the car how to run. And I don't have to hire a special key turner. So, is that asking too much?"

■ ■ ■

First Annual

WACC!

World Altair Computer Convention

This year's most exciting computer convention could very well be the First Annual WORLD ALTAIR COMPUTER CONVENTION. Computer hobbyists from all over the World will gather in Albuquerque, New Mexico on Saturday and Sunday, March 27 & 28. Many of them will be bringing their Altair systems in order to compete for the $10,000 worth of Altair equipment to be given away at the convention. In addition to demonstrations of Altair product applications, there will be FOUR SEMINARS presented during this dynamic weekend.

SEMINAR ONE will be a seminar on LOW COST COMPUTING conducted by some of the leading figures in the field. A preliminary list of speakers includes Larry Steckler, technical editor of Radio Electronics, Carl Helmers, editor of Byte magazine, Art Childs, editor of Interface magazine, David Ahl, publisher of Creative Computing, Judge Pierce Young, president and founder of the Southern California Computer Society, and Terry Silver, also of the SCCS. And this is only the beginning.

SEMINAR TWO will be a complete discussion of ALTAIR PRODUCTS and Altair design philosophy. Speakers will include H. Edward Roberts, president of MITS, Inc.; Project Engineers Bill Yates, Bob Zaller, Tom Durston, and Pat Goding; Software Writers Paul Allen and Bill Gates; and Computer Notes editor, David Bunnell.

SEMINAR THREE will be a presentation of the updated MITS TRAVELING SEMINAR presented by Pat Ward. Altair technical binders will be given away free to people attending this seminar.

SEMINAR FOUR will be an organizational meeting of the Altair Users Group conducted by Barbara Sims and David Bunnell. Topics will include organization of the Users Group and ways to improve MITS service to Altair users. All seminars will be opened to the audience for questions.

Attendance to the WORLD ALTAIR COMPUTER CONVENTION will be **free to all Altair owners and out of town guests.** The convention will be held at the new MITS building at 2450 Alamo SE, within walking distance of the Albuquerque Airport Terminal. The entire Airport Marina Hotel has been reserved for this occasion. Reservations at this hotel (which is also within walking distance of MITS and the Airport Terminal) can be made by filling out the coupon in this ad and returning it prior to February 26. Cost of reservations are $20 per night for a single and $24 for a double.

$10,000

MITS will be presenting door prizes and prizes for the best demonstrations at the convention. These prizes will include Altair 8800's, Altair 680's, and related equipment of a retail value not less than $10,000. To enter in this contest or to have a booth at the convention, you must fill out an official application form from MITS, Inc. Rules and regulations governing demonstrations and booths are available with application forms.

ALTAIR CONVENTION COUPON

Name_____

Address_____

City_____ State & Zip_____

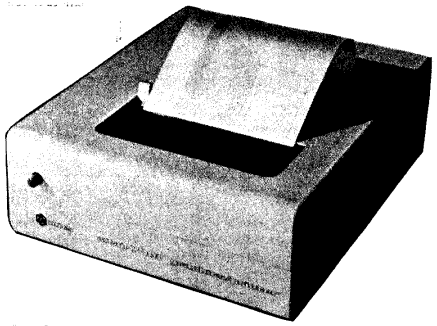
☐ Yes, I plan to attend the first annual WACC to be held in Albuquerque, New Mexico on March 27 and 28, 1976.

☐ Please reserve a room for me at the Albuquerque Marina Hotel. I will need a ☐ single ☐ double room. I plan on staying in Albuquerque the following nights: ☐ Friday ☐ Saturday ☐ Sunday.

☐ Please send me the official entry form for the Altair Demonstration Contest.

MITS/2450 Alamo SE/Albuquerque, 87106 505-262-1951

COMPUTER NOTES
©mits 1976
SEPTEMBER Vol. 2 Issue 4



Altair 7000 Graphics Printer

Keeping up with its innovative tradition, MITS has recently announced several new products which will greatly expand the capabilities of the Altair Computer system.

The tremendous flexibility of the 7000 Graphics/Printer, which acts as a printer, plotter and graphics device, makes it one of the fastest and most economical methods of electrostatic printing. The new Universal I/O board for the Altair 680b expands the I/O capabilities of the 680b beyond the one serial port on the main board. Thanks to the 88-Mux (24 Channel Multiplexer), the input capacity of the 88-Analog-to-Digital Converter for applications requiring a large number of analogs has been greatly increased. The new 88-S4K Memory Board now makes totally synchronous memory logic available to Altair 8800 users.

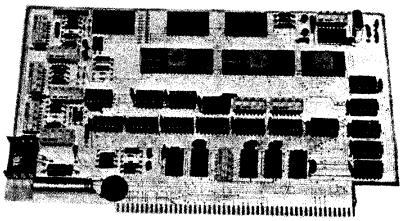
The 7000 Graphics/Printer

Although there are a number of methods used for printing computer output, electrostatic printing is finally being recognized as the only method which is fast, economical and, now with the MITS 7000, is also the most flexible means. In last month's C.N. we introduced the 7000 Graphics/Printer as a multifunction, hard-copy output device which is plug compatible with the 680 and 8800 mainframes via one PIO port. The enthusiastic response to the 7000 warrants a more detailed explanation of its operation and applications.

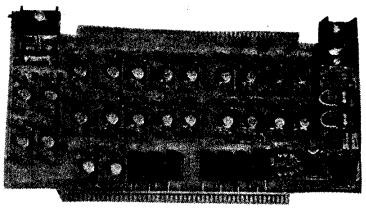
The flexibility of the 7000 is due to eight print electrodes, driven directly by software, instead of the usual seven found in 5 x 7 matrix printers. Copies made from the printed output are actually more legible than copies of typed paper and can be made for about 1¢ per foot of electrosensitive paper.

When the 7000 is used as a line printer, characters are generated using a 5 x 7 dot matrix. Altair BASIC supports three different sizes of character sets (each with upper and lower case) to produce line widths of 20, 40 or 80 characters in the four-inch wide printing area. The speed is 160 characters per second (80 characters per line) or 120 lines per minute. Different character sizes are selected with the CHR$ function in BASIC.

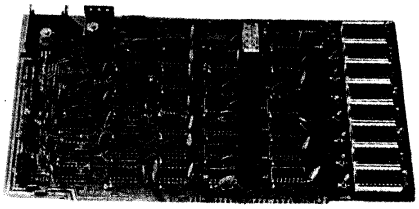
The eighth or extra printing electrode in this unit provides symmetry along the horizontal and vertical axes to permit plotting. With the vertical distance between electrodes equal to the distance between lines, there's no gap from line to line. This special feature makes the 7000 ideal for graphics. Pictures can be produced that show either a distinct outline or



680-b Universal I/O Board



88 - Multiplexer



listed on page ten of the July issue assumes that any year which is an integer multiple of four is a leap year. The correct leap year rule is: any year 'evenly' divisible by four is a leap year UNLESS it is 'evenly' divisible by 100 (a 'century' year), BUT it is still a leap year if it is 'evenly' divisible by 400. To correct the program, rewrite the lines shown here:

```
5000 IF INT(Y1/4)*4<>Y1 THEN 5200 ;NOT A LEAP YEAR
5100 IF INT(Y1/100)*100=Y1 AND INT(Y1/400)*400<>Y1 THEN 5200 ;NOT A LEAP YR
5150 L1=1: GOTO 5300
5200 L1=0
5300 IF INT(Y2/4)*4<>Y2 THEN 5500 ;NOT A LEAP YEAR
5400 IF INT(Y2/100)*100=Y2 AND INT(Y2/400)*400<>Y2 THEN 5500 ;NOT A LEAP YR
5450 L2=1: GOTO 5600
5500 L2=0
  .
  .
  .
10510 IF INT(Y2/4)*4<>Y2 THEN 10530 ;NOT A LEAP YEAR
10520 IF INT(Y2/100)*100=Y2 AND INT(Y2/400)*400<>Y2 THEN 10530 ;NOT A LEAP YR
10525 L2=1: GOTO 10600
```

Note that lines 5200, 5500 and 10530 need not be re-written; lines 5150, 5450 and 10525 are added lines. These changes will allow the program to work correctly for persons born before March 1, 1900. The leap-year rule stated above is correct for any year later than 1582, when the Gregorian calendar was adopted as a long-needed reform.

In particular, we're interested in first person experience building computer kits and peripherals, and then getting them operational. But even more, we're trying to focus on "what do you do with it after it's built?" That is, applications. Things like a file system for LP records, menu/shopping planning for various tasks and dietary requirements, kinetic video art, and, of course, games. But challenging games, cybernetic games, learning games.

Contributions should be sent to Ms. Burchenal Green, Editor, Creative Computing, P.O. Box 789-M, Morristown, N.J., 07960.

Sincerely,
David H. Ahl,
Publisher

– Continued on Page Fifteen –

customer service news

By Gale Schonfeld

This month the Repair Department has asked me to relay the following message to you:

WARRANTY ON MITS PRODUCTS

The warranty on kits is 90 days for parts. Labor is charged at $22.00 per hour for all computer mainframes and related products. The warranty on assembled items is 90 days for parts and labor.

For detailed information on product warranty, please check your manuals. Remember - the warranty does not cover postage and handling to and from the MITS factory.

HOW TO SHIP UNITS IN FOR REPAIR

Packing - Make sure when packing items to be returned for repair that all accessories are secured in place and that they are not "floating" inside or outside of the mainframe. Transformers should be bolted down or shipped separately, peripheral boards should be secured in their card guides and edge connectors, or packed in a separate box, disk drives must have the "block" secured in place, etc. Damage resulting from poor packing or packaging will automatically void your warranty. The Repair Department will advise you of any packing damage before repair action is begun.

Packaging - If at all possible, items should be shipped in the original MITS box, padded well with newspaper or styrofoam beads. Styrofoam corner pads should be used to protect mainframes and terminals. Double boxing is preferred.

Damages caused by the shipping agent - If an item received appears to have been damaged in shipment, the Repair Department will immediately contact the customer so that appropriate action may be taken for claims purposes.

Mailing labels - Please be sure that your mailing label reads "ATTN: Repair Department". We do have several departments which receive in-coming packages and unnecessary delays can be caused by misrouting.

Enclose a letter - Please be sure to enclose a letter explaining the problems you are having with your equipment. Examples of these problems would also be helpful. Also, please remember to state who the actual owner of the computer or peripherals is (the owner, by our records, is the person or company listed in the "sold to" address on your invoice). We ask this in order to prevent unnecessary charges due to lack of information on warranties.

CHARGES

Labor Charges - Labor charges are rated at $22.00 per hour for all mainframes and related products.

– Continued on Page Fifteen –

COMPUTER NOTES

Editor — Andrea Lewis

Production — Tom Antreasian, Al McCahon, Grace Brown

Contributors — Ed Roberts, Gale Schonfeld, Paul Allen, Tom Durston, Michael Hunter, Pat Godding, Mark Chamberlin, Linda Blocki, Richard Haber

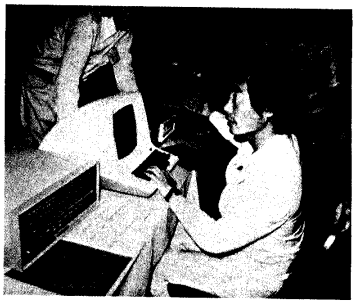
Trekking with the Altair

By Steve Lowe of Microsystems, Springfield, VA

At the recent "STARTREK EXPO" held at the Hyatt Regency Hotel in Washington, D.C., MICROSYSTEMS was on hand with an impressive display of ALTAIR computers to introduce "Trekies" to the world of microcomputers. Several ALTAIR computer systems were displayed featuring a disk system, a talking ALTAIR, a T.V. Dazzler system using an Advent 4' x 6' Videobeam T.V., and a game system using a second Videobeam T.V. as a terminal.

Throughout the three day event, each of the systems proved their worth in attracting and holding convention-goers' attention while the MICROSYSTEMS staff answered questions and provided information to interested people. Many had no idea that microcomputers were so versatile and later dropped in at the MICROSYSTEMS showroom for more in-depth demonstrations of MITS computers and ALTAIR BASIC.

One highlight of the show for the MICROSYSTEMS crew was a visit to their display by several of the original stars of the STARTREK T.V. series. George Takei, who plays Mr. Sulu in the series, was lured into the display room and tried out different versions of STARTREK games in front of a large crowd. During one game where the Enterprise engages in combat with a similarly armed Klingon Vessel, George asked, "Do we have to fight? Why can't we negotiate?" Moments later a moan of anguish engulfed the room as the crowd watched the Klingon ship out-maneuver and destroy the USS Enterprise. George redeemed himself, however, by winning his second battle. A newcomer to the world of microcomputers, he admitted that he could understand the fascination that made the MICROSYSTEMS display such a popular attraction at the show. Even before Gene Roddenberry (the producer of Star Trek) spoke later that day about the impact of computers on the future of humanity, MICROSYSTEMS had been proving that thanks to MITS, the impact of computers is already being felt around the world.

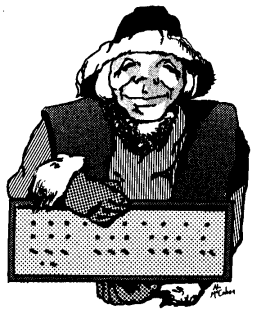


Moments before the Klingon's destroyed the Enterprise, George Takei (Mr. Sulu from the Star Trek T.V. series) pauses to ask the MICROSYSTEMS crew about "the possibility of negotiating with the Klingon vessel."

Book Review

By Linda Blocki

The Compleat Computer



RIDDLE: What does a medical center, the Senate Watergate Committee's investigative team and a Tibetan monastery have in common? ANSWER: a computer. Surprised? Thanks to new electronic developments, the use of computers is quickly expanding to practically all areas of everyday American life.

Information about computers is no longer confined to complicated articles by mathematicians and data processors. Many noncomputer specialists are now doing some interesting research and writing, which is gradually replacing the public's confusion about all that hardware and software with beneficial information that anyone can understand.

The Compleat Computer (1976), a carefully compiled collection of over 100 informative and sometimes humorous articles by noncomputer specialists, seems to be the best publication so far to help expose people to the many diverse opinions about the use of computers. Author Dennie Van Tassel, user liaison in the computer center at UCSC and collector of computer miscellany, has filled his paperback book with a wide variety of selections from fiction, poetry, newspapers, cartoons, and advertising as well as more detailed articles that concern the computer specialists. Such well-known noncomputer experts as Norman Cousins, Ray Bradbury and Isaac Assimov are just a few of the writers whose articles appear throughout the book.

Some of the different areas the articles cover include the story of a fully computerized poison control center in a children's hospital in Missouri, a fictional account of a Tibetan monastery that used a Mark V computer to compile a list of all the possible names of God and a computer which acted as a key "member" of the Senate Watergate investigative team by spewing out minute facts about any witness in a fraction of a second.

In order to include as much material as possible, Van Tassel has capsulized the longer articles and selected only the "tastiest tidbits" for publication. His extensive references following each article are helpful to the interested reader who wishes to pursue a topic in greater depth. After each well-organized section of the book, a long list of questions and exercises is included to further aid the reader in exploring other various opinions about the use of computers.

The book is divided into nine sections starting with three introductory chapters--"In the Beginning," "How Computers Do It" and "The Software"--which discuss the basics of computer operation. Articles appearing in this first section include "The Development of Automatic Computing," "Computer Games People Play" and "Technology, McDonald's Collide as Students Best Burger Bonanza," a humorous article describing how Cal Tech students used an IBM computer to print out 1.2 million entry blanks and win a McDonald's contest.

Continued on Page Fourteen

Continued from Page One

the effect is a very dark image. When the columns are printed farther apart, the image appears lighter.

The 7000 is controlled by using a single port on an 88-4PIO parallel interface board. One section of the port provides the eight bits of information to be printed, and the other section provides control.

The control signals to the 7000 are: (1) MOTOR ON, which starts the motor running while the print head remains disengaged, (2) PRINT, which engages the print head to begin traveling across the page and (3) LINE FEED, which causes a line feed with no print head movement.

The control signals from the 7000 are: (1) CT or character timing pulse. The first pulse defines the left-hand margin once the print head has begun to move, (2) DT or dot timing pulse. There are eight dot pulses for each character pulse.

Each time a new column of information is to be printed, the appropriate data bits are forced low (logic "0") by the 4PIO. Each low data line causes the related printhead electrode to discharge to the paper. This produces one dot. In all there are more than 500 eight-dot columns in a line.

To print characters, seven electrodes are used to provide a one-dot space between lines. In the plot mode all eight electrodes are used so that there's no space between the lines that make up a plot.

The software required to use the Altair 7ØØØ Graphics/Printer as a line printer has been integrated into the BASIC interpreter. In order to list or print using the 7ØØØ, the LPRINT or LLIST commands are used. In order to change the character size, an LPRINT command must be issued which includes one of the following three special characters:

CHR$(1) prints 80 characters/line
CHR$(2) prints 40 characters/line
CHR$(3) prints 26 characters/line

Example:

```
LPRINT CHR$(1);"TEST SMALL CHARACTERS"
LPRINT CHR$(2);"TEST MEDIUM CHARACTERS"
LPRINT CHR$(3);"TEST LARGE CHARACTERS"
```

produces the following output:

```
TEST SMALL CHARACTERS
TEST MEDIUM CHARACTERS
TEST LARGE CHARACTERS
```

Note: If a new character size is not requested, the most recently requested character size will be used.

Except for one assembly language subroutine, the software for using the Altair 7ØØØ Graphics/Printer as a plotter is written entirely in BASIC language. This will allow the user to make his own custom modifications to the standard software. It will also allow him to save room in memory by removing subroutines that are not required.

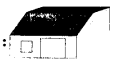
The image to be printed is stored in memory in a buffer with each bit representing a dot in the picture. If the bit is turned on, the corresponding dot is present. If the bit is turned off, the corresponding dot is absent. A 256 byte segment of memory represents the 8 rows of 256 dots printed on one pass of the print head. In order to cause this 8 by 256 segment to be printed, a single call to the assembly language routine is required.

Since an 8 by 256 dot picture is far too small to be of any practical use, the plot routine uses a number of these 8 by 256 elements to compose a picture. The standard number is 32, and this requires an 8K buffer for the image. The user may increase or decrease this number by altering a single BASIC statement as his needs require or his memory permits.

There are BASIC subroutines for:

1) performing initialization - setting buffer size, location, etc.
2) printing the entire buffer
3) clearing the buffer
4) marking a dot
5) writing a character
6) writing a string for label
7) calculating scaling factors
8) plotting a point
9) drawing a line

Here is some sample output:



7000 Graphics/Printer Specifications:

Price and availability: $785, 60 days

Printing medium: Electrosensitive paper (5 inches wide)

Horizontal resolution:
A. Internal timing--80 dots/inch
B. External timing--better than 128 dots/inch

Vertical resolution: 65 dots/inch

Printhead speed: 0.0175 inches/msec. ± 0.1%

Timing markers:
A. Every 1/80 inch of printhead travel (DT)
B. Every 1/10 inch of printhead travel (CT)

Plotting speed: Two lines per second, 8 dots vertical

Input raster: Eight-bit parallel

Power: 115V AC. 36 VA

Weight: 14 lbs.

Interface: 1 PIO Port (88-4PIO or parallel port on 680b Universal I/O Board)

The 680b Universal I/O Card:

The 680b Universal I/O card provides two parallel ports and one serial port to greatly enhance the I/O capabilities of the Altair 680b while occupying only one slot on the expander board.

Parallel Ports

The design of the Universal I/O's two parallel ports is based upon a Peripheral Interface Adapter (PIA). The PIA contains all control and data registers, thus most option are software selectable. These options include data direction (each data line can act as an input or an output), and interrupt/control structure. The Universal I/O can be expanded up to two parallel ports.

Parallel Port Selection:

Each Universal I/O card requires 16 address lines. Hardware sets the upper 8 address lines (A15 to A8) to FØ (hexadecimal) for all I/O ports. These addresses are FØXX.

Address lines A7 through A4 and their complements $\overline{A7}$ through $\overline{A4}$ are user-selectable. With these addresses there are 16 different address locations for the Universal I/O (4 addresses are reserved for future use).

Address lines A3 and A2 select between 3 ports. A3 addresses the parallel ports or the serial port. A2 selects a particular parallel port.

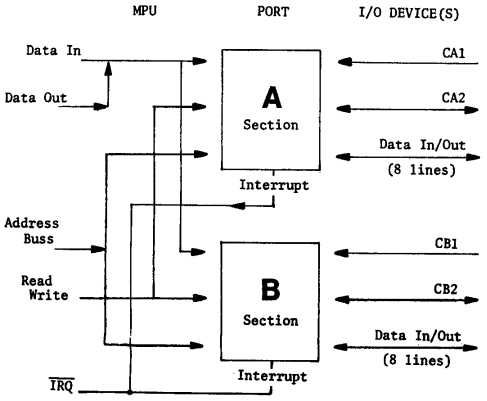
Each PIA contains 2 sections. Sections A and B each contain two channels, control status channel and data-data direction channel. Address lines AØ and A1 enable the selection of port section, A or B, and the selection of control status channel or data channel. If the two parallel ports are addressed at F008 and F00C, the port, section and channel addresses would appear as follows:

(Refer to Figure One)

Figure One

ADDRESS	IC	SECTION	CHANNEL
F008	B	A	CONTROL/STATUS
F009			DATA-DDR
F00A		B	CONTROL/STATUS
F00B			DATA-DDR
F00C	A	A	CONTROL/STATUS
F00D			DATA-DDR
F00E		B	CONTROL/STATUS
F00F			DATA-DDR

The following block diagram illustrates the internal structure of a PIA.



(Refer also to "Software Initialization of Parallel and Serial I/O Boards" by Patrick Godding, Computer Notes, June, 1976, pp. 14-17.)

The Universal I/O with only one PIA parallel port can handle two inputs (such as a paper tape reader or keyboard) or two output devices (such as a paper tape punch and printer) or any combination of custom applications. A Universal I/O with two PIA parallel ports has 32 data lines (each group of eight is individually selectable). All data lines are fully TTL compatible. Eight of the 16 lines are capable of directly driving the base of a transistor switch (1.5v at 1ma).

Serial Port

The design of the Universal I/O's serial port is based upon an Asynchronous Communications Interface Adapter (ACIA). The ACIA allows serial data to be taken in on its receive line and transfers the data onto the Data Bus, or data can be entered from the data bus into the ACIA and then sent out the transmit data line in serial form.

The ACIA contains both control and status registers. Five control lines allow maximum utilization of sophisticated terminals. The five control lines are: (1) transmit data, (2) receive data, (3) data carrier detect, (4) clear to send and (5) request to send.

The 8-bit Status Register allows for greater control and handshaking ability by indicating received data available, transmitter buffer empty, carrier detect, clear to send, framing error, received data overflow, parity error, and interrupt request.

All lines are switch-selectable for RS-232, TTL levels or 20 milliamp current loop (TTY). The serial port is programmable for nine or ten bit transmission as follows:

a. 7 data bits + parity bit (odd, even, or none) + 1 or 2 stop bits;

b. 8 data bits + 1 or 2 stop bits;

c. 8 data bits + 1 stop bit + parity bit (odd or even)

The transmit and receive interrupts enable or disable under software control. The Universal I/O provides an onboard, crystal-controlled clock that allows user selection for any of 13 baud rates by positioning a dip switch.

The Selectable Baud rates are:

50	2400	300
75	9600	150
134.5	4800	110
200	1800	
600	1200	

Universal I/O Board Specifications:

Level Selection: Switch selectable, TTL, RS232, TTY

Baud rate generator (ACIA): Crystal-controlled CMOS Divider

Device Connection: (fully expanded) 12 conductor cable, 10-pin removable connector on board and 25 pin connector (ACIA). Three removable flat cables with a 24-pin plug on the board and a 25-pin connector passed through the back panel (for PIAs and other parallel interface).

680b Mb slots: One

Power: +5 volt at approximately 350 milliamps fully expanded. Typically 27 milliamps @ +16 volts. Typically 10 milliamps @ -16 volts.

Bit Configuration: Software selectable for seven or eight bits, one or two stop bits and odd or even parity PIA.

88-S4K Memory Board:

An ideal addition to the Altair 8800 series computer is the 88-S4K Synchronous 4K Memory Board, which has many outstanding features including totally synchronous design logic. This means the memory relies solely on the CPU for timing signals - no single shots and no critical on-board timing.

Troubleshooting the 680b

by Rich Haber

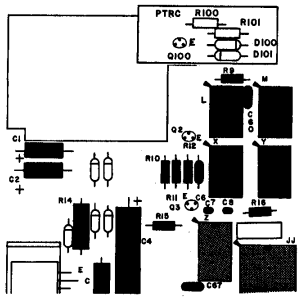


Figure 1.

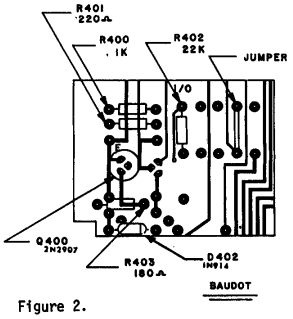


Figure 2.

We have been very impressed in the repair department by how few 680b's have been sent back to us. Kit builders have been doing a really good job assembling and troubleshooting their units. If you are having any trouble with your 680b, the troubleshooting aids on pages 17-19 in the Theory of Operation Manual will help you track down some of the most common problems.

Apparently not everybody received or noticed the errata sheet explaining that Q2 and Q3 and Q100 were silkscreened incorrectly on some of the main boards. The emitters were marked where the collectors should be and vice versa. The correct positioning for these transistors is shown in Figure 1. There was also an error on page 30 of the Assembly Manual and page 10 of the Operator's Manual concerning the Baudot interface. The values for R401 and R400 were reversed and a diode (D402) was left out. See Figure 2 for the correct configuration.

Here is a list of some common problems with the 680b and how to track down the causes.

680b Troubleshooting

1. All address lights except A0 lit.

   This indicates that the computer is locked in the reset mode. This can be verified if pin 40 of the MPU ($\overline{RES}$) is LOW. Probable causes:

   a. Q2 and Q3 are in backwards (silkscreen shows C & E reversed).
   b. Solder bridge on transistor lands.
   c. No phase 1 clock signal to IC K pin 10 on front panel. If true, then check pin 2 of IC pp.
   d. Bus line 54 shorted.

2. MPU always running.

   This can be verified if pin 2 of MPU ($\overline{HALT}$) is HIGH. Probable causes:

   a. No Ø2 phase 2 clock signal to IC K pin 2 to retrigger the one-shot; if so, then check PP-4.
   b. IC I or K defective (check logic).
   c. Q4 or Q5 in backwards, shorted or defective.

3. Can't deposit.

   a. Make sure RAMs are strapped to the address you want.
   b. Check to see if pin 34 of the MPU (R/$\overline{W}$) goes LOW when the deposit switch is toggled. If 34 won't go LOW, look at I-12 on front panel and trace back.
   c. $\overline{BA}$ (pin 7 of MPU) should be HIGH.
   d. $\overline{IRQ}$ (pin 4 of MPU) should be HIGH.
   e. IC M pin 12 should be HIGH for read, pulse LOW for deposit. Does Ø2 appear at pins 1 and 2?
   f. AA-8 should be LOW. If not, and all AA inputs are HIGH, look for a short on this line.
   g. Check for solder bridges on RAMs.

4. Can't deposit at any one bit.

   First interchange RAMs and see if bad bit changes. If it does, then the RAM is bad. If not, make the following checks on the bad bit (leftmost RAM is bit 0):

   a. Pin 13 should be LOW.
   b. Pin 10 should be HIGH.
   c. Pin 3 should be HIGH to read and pulse LOW to write.
   d. Is data appearing at DI (Pin 11)? Is data appearing at DO (Pin 12)? If not, check logic at NAND gate and inverter on output (pin 12). Outputs of NAND gates should be HIGH for a "1" and LOW for a "0". Failure here could involve laborious tracing for solder bridges or shorted IC.

- Continued -

altair 8800b
Assembly Manual Corrections:

(Display/Control Board)

#1 - Capacitor C7 should be ommitted and resistor R75 should be replaced by a jumper wire. This filter circuit is not necessary since it will attenuate the Ø2 input to IC S1-3 too much.

#2 - When installing the resistor pack (page 5-24) it is necessary to clip off the last three leads at the end furthest from the dot on the resistor pack. There are no holes on the PC board for these leads and these three resistors are unused.

toggled. If not, check to see that: DD8 is LOW, GG13 is HIGH, HH4 is LOW and NN40 is LOW while resetting.

If the monitor fails to print a period, it is occasionally due to two or more addresses being shorted together. Toggle each address switch separately to see if the LED lights. If a LED fails to light, position all the address switches up. If the LED comes on, then there is a short between addresses. You can isolate which one by putting the switches down one by one.

I/O problems.

Pins 2 and 6 of the ACIA should be HIGH after initializing the monitor (with 680b in terminal option), otherwise a short is indicated. Pins 3 and 4 should have a square wave signal equal to 16 times the baud rate; look for a .568 msec. period for 110 baud, .208 msec. for 300 baud. NOTE: R15 should be 1K ohm instead of 4.7K. Do not bother to replace it unless the voltage at the right side of R15 is below TTL levels. If the voltage is very low, IC Z is probably bad.

The most common causes of problems on units we have received have been:

a. Solder bridges and cold solder joints (especially on 100 pin connector)
b. Incorrect parts placement
c. Incorrect hardwire strapping
d. IC pins bent under chip

I would like to recall your attention to the problem mentioned in the June Computer Notes ("Altair 680b Hardware Notes," page 9). If the MPU is given an invalid instruction to execute, it cannot be reset through the front panel switch. Instead, power must be turned off and on and then RESET must be activated, thereby erasing memory. To correct this, do the simple modification that is outlined in the article.
Here is a convenient check list of logic levels for troubleshooting:

Chip	ID	Pin	Label	Status (MPU Stopped)
MPU	*NN*	2	$\overline{\text{HALT}}$	LOW
		3	Ø1 (phase 1 clock)	Ø1
		4	$\overline{\text{IRQ}}$ (interrupt request)	HIGH
		5	VMA (valid memory)	LOW
		6	$\overline{\text{NMI}}$ (non maskable interrupt)	HIGH
		7	BA (bus available)	HIGH
		34	R/W (read/write)	HIGH (pulses LOW during deposit)
		37	Ø2 (phase 2 clock)	Ø2
		40	RES (reset)	HIGH
ACIA	JJ	3&4	clock square wave frequency = 16 x baud rate	
		9	$\overline{\text{CS2}}$ (chip select 2)	LOW
		10	SS1	HIGH
		11	RS (register select)	tied to A0
		13	R/W (read/write)	HIGH
		14	enable	Ø2
		7	$\overline{\text{IRQ}}$	HIGH
RAMs	C-K	3	R/W-P (read/write-prime)	HIGH (pulses LOW during deposit)
		10	VCC	HIGH
		11	DI (data in)	same as front panel switches
		12	DO (data out)	same as front panel lights
		13	$\overline{\text{CS}}$ (chip select)	LOW when addressed
PROM	T-V	14	$\overline{\text{CS}}$ (chip select)	LOW when addressed

Continued on Page Ten

Altair Users

Larry Landsuerk
2320 N. 56th Lane
Fort Smith, Arkansas 72901

Ronald W. Moore
A Div. USS America CV66
FPO, New York 09501

Robert M. White
5240 Kootenai St.
Boise, Idaho 83705

Carl Knipp
Box #1
Cary, IL 60013

New Address for:
Capt. James K. Bostick
Tuslog DET II
PSC Box 2324
APO, New York 09224

Jim Henley
420 Bankcroft Ct., #8
Rockford, IL 61107

Computer Clubs

Houston Texas

HAMCC
David M. Fogg, President
4223 S.W. Fwy., #203
Houston, TX 77027
(713) 626-2935

The HAMCC meets on the second Friday and the fourth Tuesday of each month.

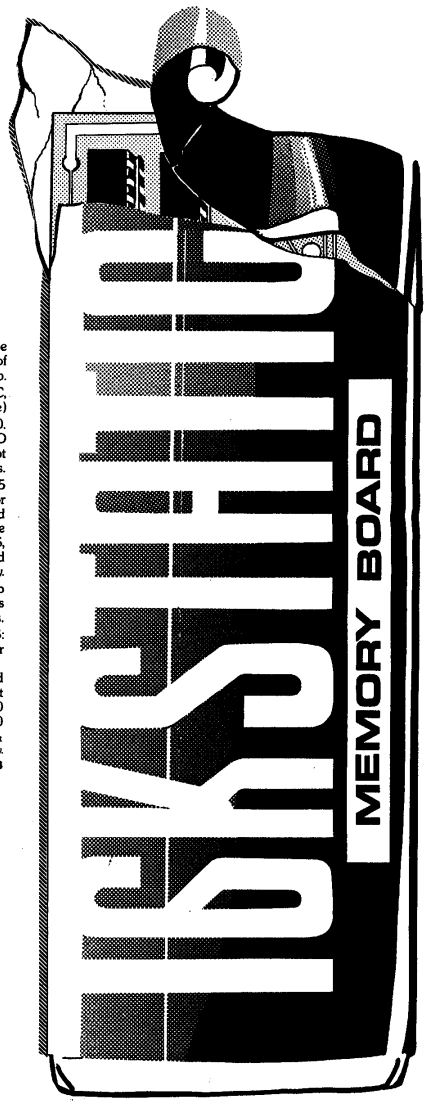
Rockford, Illinois

Anyone interested in forming a club in the Rockford, Illinois, area should contact:

Jim Henley
420 Bancroft Ct., #8
Rockford, IL 51107
1-815-399-6558

HARD WARE

Something Sweet for your altair™ 680-b



MITS is pleased to announce the development of a 16K static card for the Altair 680b. With an access time of 215 nanoseconds and low power consumption of 5 watts, we feel that this is an excellent addition to the Altair 680b.

To sweeten the pot even more, we are including a free copy of Altair 680 BASIC, assembler, and text editor on paper tape. ($275 value)

Altair 680 BASIC is identical to the 8K BASIC developed for the Altair 8800. Features include Boolean operators, the ability to read or write a byte from any I/O port or memory location, multiple statements per line, and the ability to interrupt program execution and then continue after the examination of variable values.

Other features of Altair 680 BASIC include variable length strings (up to 255 characters), with LEFT$, RIGHT$ and MID$ functions, a concatenation operator and VAL and STR$ to convert between strings and numbers. Both string and numeric arrays of up to 30 dimensions can be used. Nesting of loops and subroutine calls is limited only by available memory. Intrinsic functions include: SIN, COS, TAN, LOG, EXP, SQR, SGN, ABS, INT, FRE, RND and POS, in addition to TAB and SPC in PRINT statements. Altair 680 BASIC takes 7K bytes of memory.

MITS has also developed an expander card for the Altair 680b that lets you add up to three boards inside the main case. Read "Computer Notes" for announcements of additional Altair 680b boards.

PRICES:

Altair 680-BSM, 16K Static Memory Board, including Altair 680 BASIC, assembler and text editor . . . . . $685.00 kit
$865.00 assembled
Altair 680-MB Expander Card with one Edge Connector . . . . . $24.00 kit
Altair 680 BASIC (purchased separately) $200.00
Altair 680 assembler and text editor (purchased separately) . . . . . $ 75.00

PRICE APPLIES ONLY TO PURCHASERS OF ALTAIR 680b COMPUTER

Prices, specifications subject to change. Allow 30-60 days for delivery.

MITS, Inc. 2450 Alamo S.E./Albuquerque, New Mexico 87106

mits

– Continued from Page Five –

There are no wait states so that the CPU runs at maximum speed.

The 88-S4K provides 4,096 bytes of Random Access Memory while consuming very low power. Each board contains memory protect circuitry and address selection circuitry for any one of 16 starting locations in increments of 4K. The entire 4,096 bytes of memory can be protected by switching to PROTECT. A DIP switch is used for board selection with no hardwire jumpers, and test points have been installed at important signal outputs for ease of checkout and troubleshooting. Ferrite beads are used on all common supply lines for noise isolation.

For ease of assembly, an epoxy solder mask on areas not to be soldered has been added, as well as sockets for all memory ICs, which provide easy installation and removal of the ICs. Included with the 88-S4K is a well-documented manual with detailed theory and troubleshooting sections and step-by-step assembly instructions.

88-S4K SPECIFICATIONS

RAM Access Time: 200-300 ns.

Power Requirement
(worst case):

+5V current - 450 ma (max.)
+12V current - 290 ma (max.)
+12V current (unselected) -
10 ma (max.)

Memory Array: 4,096 8-bit words

Dimensions: 10" x 5"

Altair Slots: One

Price: $155.00 kit, $255.00 assem.

Availability: Within 60 days of order.

The 88-MUX (24-Channel Multiplexer):

The 88-MUX, companion card to the 88-Analog/Digital Converter (88-ADC--see page 8 of the August issue of C.N.), will expand the input capacity of the 88-ADC for applications requiring a large number of analog inputs.

The 88-ADC is actually a stand-alone card for many systems, because it contains an on-board 8-channel multiplexer. However, for a majority of system layouts, the real potential of the ADC and MUX conversion system lies in the ability of the MUX to process more than eight signals. By using four 88-MUX cards, it is possible to process up to 96 analog signals!

Another advantage of using the 88-MUX is the optional differential input. With simple modifications, the card can be set up to handle a

As a bonus, the gain and scale factoring of each channel can be set independently, giving extreme flexibility in system design. Input filtering can also be added to provide the desired roll-off characteristics.

An appropriate interface cable is provided with each ADC and MUX pair. (For more than one 88-MUX per system, a cable is required for each MUX card.)

Specifications for the 88-MUX (24-Channel Multiplexer):

Price: $319, assembled only.

Availability: within 60 days of order.

Gain: Up to 1000

Input Impedance: 1000 megohms

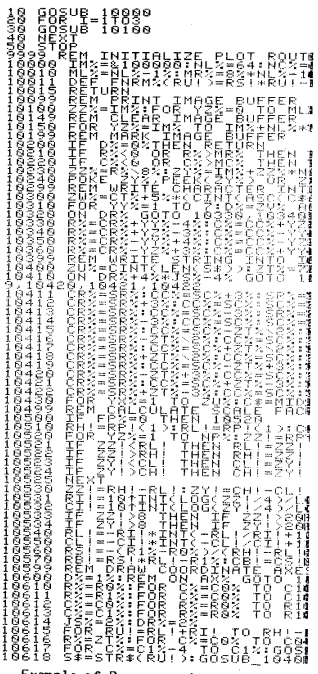
Offset: 5MV (max.)

Input Signal
Level: (-4 to +10v)
(-10 to +4v)
(-5 to +5v)

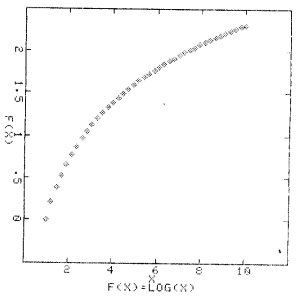
Settling Time
to .01%: 15 microseconds (max.)

P.S. requirements: +5v@40MA
+15v@180MA
-15v@180MA

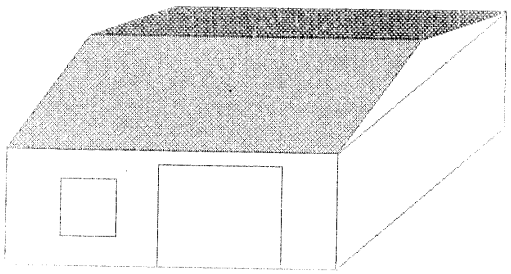
(See Computer Notes, August, 1976, page 8, for specifications on the 88 A-to-D Converter.)



Example of Program printed by Graphics Printer.(73% of full size.)



Example of Graph printed by 7000 Graphics Printer.(73% of full size.)



CPU runs at maximum speed.

The 88-S4K provides 4,096 bytes of Random Access Memory while consuming very low power. Each board contains memory protect circuitry and address selection circuitry for any one of 16 starting locations in increments of 4K. The entire 4,096 bytes of memory can be protected by switching to PROTECT. A DIP switch is used for board selection with no hardwire jumpers, and test points have been installed at important signal outputs for ease of checkout and troubleshooting. Ferrite beads are used on all common supply lines for noise isolation.

For ease of assembly, an epoxy solder mask on areas not to be soldered has been added, as well as sockets for all memory ICs, which provide easy installation and removal of the ICs. Included with the 88-S4K is a well-documented manual with detailed theory and troubleshooting sections and step-by-step assembly instructions.

88-S4K SPECIFICATIONS

RAM Access Time: 200-300 ns.

Power Requirement (worst case):

+5V current - 450 ma (max.)
+12V current - 290 ma (max.)
+12V current (unselected) - 10 ma (max.)

Memory Array: 4,096 8-bit words

Dimensions: 10" x 5"

Altair Slots: One

Price: $155.00 kit, $255.00 assem.

Availability: Within 60 days of order.

The 88-MUX (24-Channel Multiplexer):

The 88-MUX, companion card to the 88-Analog/Digital Converter (88-ADC--see page 8 of the August issue of C.N.), will expand the input capacity of the 88-ADC for applications requiring a large number of analog inputs.

The 88-ADC is actually a stand-alone card for many systems, because it contains an on-board 8-channel multiplexer. However, for a majority of system layouts, the real potential of the ADC and MUX conversion system lies in the ability of the MUX to process more than eight signals. By using four 88-MUX cards, it is possible to process up to 96 analog signals!

Another advantage of using the 88-MUX is the optional differential input. With simple modifications, the card can be set up to handle a differential input on each channel.

set independently, giving extreme flexibility in system design. Input filtering can also be added to provide the desired roll-off characteristics.

An appropriate interface cable is provided with each ADC and MUX pair. (For more than one 88-MUX per system, a cable is required for each MUX card.)

Specifications for the 88-MUX (24-Channel Multiplexer):

Price: $319, assembled only.

Availability: within 60 days of order.

Gain: Up to 1000

Input Impedance: 1000 megohms

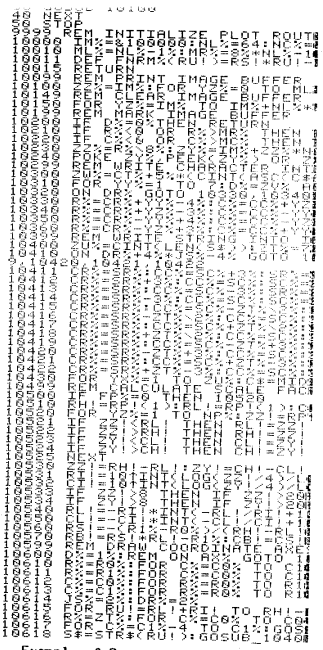
Offset: 5MV (max.)

Input Signal Level: (-4 to +10v)
(-10 to +4v)
(-5 to +5v)

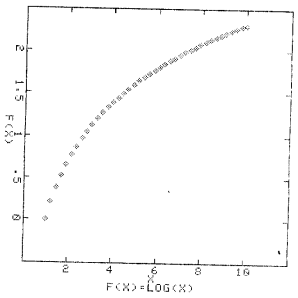
Settling Time to .01%: 15 microseconds (max.)

P.S. requirements: +5v@40MA
+15v@180MA
-15v@180MA

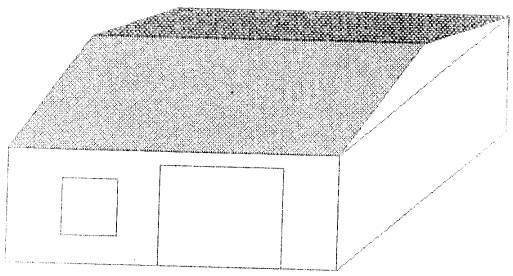
(See Computer Notes, August, 1976, page 8, for specifications on the 88 A-to-D Converter.)



Example of Program printed by Graphics Printer.(73% of full size.)



Example of Graph printed by 7000 Graphics Printer.(73% of full size.)



Example of 3-Dimensional House projected by 7000 Graphics Printer. (73% of full size.)

corrections

Users have discovered the following mistakes in two of our software programs:

#2-3-761, page 18:

Memory location 1 042 166 should read 1 042 301

Page 31:

Memory location 2 015 014 should read 2 015 013

#6-1-763, Line 570:

A=FNR(NØ5)+INT(NØ+5) : B=FNR(N1/2Ø)+ INT(N1/2Ø) : C=FNR9L2/5Ø)

should read A=FNR(NØ5)+INT(NØ/5) etc.

This change makes it possible to win the game.

mits

2450 Alamo SE

Albuquerque, NM 87106

505-243-7821

TROUBLESHOOTING

– Continued from Page Seven –

Here are three test programs that are useful in checking out your 680b.

Jump

```
0000    7E      jump        when running only A0 & 1 should be lit
  1     00
  2     00
```

Add Two Bytes

```
0000    86      LDA
  1     your choice of data
  2     C6      LDA B
  3     your choice of data
  4     1B      ABA
  5     B7      STAA
  6     00      LDC
  7     40
  8     7E      JMP
  9     00
  A     00      when using front panel
        or
  9     FF
  A     AB      when using terminal
```

Reset, run, stop and sum of data should appear at address 0040.

reset and initialize ACIA

```
0000    86
  1     03
  2     B7
  3     FO
  4     00
  5     86
  6     D1
  7     B7
  8     FO
  9     00
```

check to see if a character has been received

```
  A     B6
  B     FO      ;wait for data
  C     00
  D     47      ;rotate right
  E     24      ;branch
  F     FA
```

input data

```
0010    B6
  1     FO
  2     01
```

check to see if ACIA ready to output

```
  3     F6
  4     FO
  5     00
  6     57      ;rotate
  7     57      ;rotate
  8     24
  9     F9
```

output data

```
001A    B7
  B     FO
  C     01
  D     20
  E     EB      jump to 0000
                and type character
```

Should you need more help with your 680b, please feel free to call us. If you decide not to repair the unit yourself, please send it in. There is currently no backlog of 680b's in repair and return should be relatively prompt.

on the side

By Wayne Cronin

As the only licensed ham at MITS, I've been elected to edit a new ham-oriented column for Computer Notes. I'll be using this space to pass along ideas for adapting computers (hopefully ours!) to ham purposes. That means I'll need lots of input from readers, and I'm sure many of you have some good ham applications ideas to share with us.

ASCII, HAMS, AND THE FCC

If you have your own computer system, you probably have some kind of I/O device that uses ASCII code. If you could use your terminal to key your rig, you could use your computer to get in on RTTY activity, or to communicate with another ham's computer via radio. Unfortunately, since current FCC regulations allow only Baudot code transmission, you have to use some kind of hardware or software code conversion scheme to accomplish either of these functions. This is a needless complication and a waste of processor power.

The only way to get the rules changed is to petition the FCC. If you're thinking of submitting a petition, here are a few things to think over first.

Any new rules changes should be general enough to allow us to take advantage of future advances in the state of the art automatically; we don't want to have to cut more red tape each time something new comes along. If a proposal for a rules change is too specific in its wording, we could be neatly backing ourselves into a corner.

As an example, let's suppose that the FCC adopted new regulations based on a petition by amateurs requesting the use of ASCII. This could be interpreted as prohibiting Baudot. If this happened, we would instantly lose compatibility with Baudot equipped DX stations, and hundreds of U.S. hams using older equipment would be forced to use the same type of hardware or software conversions we had hoped to avoid.

I really don't think that anybody would submit such a restrictive proposal (much less that the FCC would adopt it). This is just an extreme example of the dangers of hastily written, narrowly worded proposals.

If any of you are considering submitting a rules change request concerning the use of ASCII, I'd like to hear your suggestions for a generalized wording that would allow us to use ASCII and Baudot now and also leave a few "loopholes" for future developments. (Don't forget to consider the possible effects of the proposed new bandwidth regulations on HF computer communications.)

By Stan Webb

The first place winner this month is Kenneth Aird with his FORTRAN Cross Assembler for the 680. This program is beautifully written and well documented. With the addition of this program, we now have three FORTRAN Cross Assemblers in our library. Consequently, we cannot accept any more programs of this type.

Second place goes to Keith Fischer for his BECO Text Editor. This program, written in Altair BASIC, is considerably more powerful than the BASIC Editor, and would be a valuable addition to a BASIC system. The user documentation is fairly good, but it lacks much program documentation.

Erik Mueller takes third place with his MINOL Interpreter. MINOL is a subset of BASIC designed to run in less than 4K of memory. This subset is very restrictive when compared to a large BASIC, but is undoubtedly easier to work with than machine language. This program was previously published in the April, 1976, issue of Dr. Dobb's Journal of Computer Calisthenics and Orthodontia.

Due to the lack of competition in the subroutine category, no prize will be awarded this month.

HAM SOFTWARE (and the lack of it)

If any of you have written ham applications software for 8080 or 6800 based machines, please consider submitting it to our software library. If I receive any programs that aren't excessively long, I'll try to get a listing into Computer Notes. Some of you may be able to contribute information that could be used as the basis for a ham software package. An example would be a set of mathematical formulas for predicting satellite orbits. Information like this made available through Computer Notes could result in a lot more programs for the library.

COMPUTER NETS?

If you know of any nets devoted to computer topics, please let me know, and I will spread the word. If you'd like to start a net, send me your suggestions for a band and time.

NEXT

Next month I'll talk a little about some of my own ideas for ham computers and software. I hope to have lots of your ideas to talk about also. Please write or call me at the MITS Repair Department with your comments and suggestions. 73.

AUTHOR'S COMMENT:

The library has a lot of material now, so I'd like to see our users put more effort into writing clean programs with good documentation. The programs that are neatly typed on our submission forms and that are well documented are more worthwhile to other users than those that are hastily done and have no documentation.

FIRST PLACE MAJOR PROGRAM

#9-1-761
Author: Kenneth Aird
Length: 41,000 bytes FORTRAN
Title: M6800 Cross Assembler
Very well written FORTRAN Cross Assembler for M6800.

SECOND PLACE MAJOR PROGRAM

#8-23-761
Author: Keith Fischer
Length: 150 lines Altair BASIC
Title: BECO
Powerful Text Editor.

THIRD PLACE MAJOR PROGRAM

#8-13-761
Author: Erik Mueller
Length: 7,000 (octal) bytes
Title: MINOL
Interpreter for a 4K subset of BASIC.

#8-9-761
Author: Alan Miller
Length: 200 lines Altair BASIC
Title: QUBIC
Plays 3D tic-tac-toe.

#8-10-761
Author: Alan Miller
Length: 7 lines Altair BASIC
Title: Numerical Integration
Numerical integration by Simpson's method and Trapezoidal Rule.

#8-16-761
Author: Roger Frank
Length: 7 lines Altair BASIC
Title: Memory Size
This program resets BASIC memory sizes without restarting (for 3-2 only).

#8-19-761
Author: Alan Miller
Length: 5 lines Altair BASIC
Title: ARCSIN
Program to compute arcsin and arccos.

#8-26-761
Author: Roger Frank
Length: 22 lines Altair BASIC
Title: Memory Test
Program to test unused memory.

#8-27-761
Author: John Stanton
Length: 52 bytes
Title: 4PIO KBD/PTR Loader
Loads data into memory from keyboard and echoes it.

#8-28-761
Author: J. David Green
Length: 168 lines Altair BASIC
Title: Horse Racing
A horse racing game that involves betting on the races.

HAM on the side

By Wayne Cronin

As the only licensed ham at MITS, I've been elected to edit a new ham-oriented column for Computer Notes. I'll be using this space to pass along ideas for adapting computers (hopefully ours!) to ham purposes. That means I'll need lots of input from readers, and I'm sure many of you have some good ham applications ideas to share with us.

ASCII, HAMS, AND THE FCC

If you have your own computer system, you probably have some kind of I/O device that uses ASCII code. If you could use your terminal to key your rig, you could use your computer to get in on RTTY activity, or to communicate with another ham's computer via radio. Unfortunately, since current FCC regulations allow only Baudot code transmission, you have to use some kind of hardware or software code conversion scheme to accomplish either of these functions. This is a needless complication and a waste of processor power.

The only way to get the rules changed is to petition the FCC. If you're thinking of submitting a petition, here are a few things to think over first.

Any new rules changes should be general enough to allow us to take advantage of future advances in the state of the art automatically; we don't want to have to cut more red tape each time something new comes along. If a proposal for a rules change is too specific in its wording, we could be neatly backing ourselves into a corner.

As an example, let's suppose that the FCC adopted new regulations based on a petition by amateurs requesting the use of ASCII. This could be interpreted as prohibiting Baudot. If this happened, we would instantly lose compatibility with Baudot equipped DX stations, and hundreds of U.S. hams using older equipment would be forced to use the same type of hardware or software conversions we had hoped to avoid.

I really don't think that anybody would submit such a restrictive proposal (much less that the FCC would adopt it). This is just an extreme example of the dangers of hastily written, narrowly worded proposals.

If any of you are considering submitting a rules change request concerning the use of ASCII, I'd like to hear your suggestions for a generalized wording that would allow us to use ASCII and Baudot now and also leave a few "loopholes" for future developments. (Don't forget to consider the possible effects of the proposed new bandwidth regulations on HF computer communications.)

HAM SOFTWARE (and the lack of it)

If any of you have written ham applications software for 8080 or 6800 based machines, please consider submitting it to our software library. If I receive any programs that aren't excessively long, I'll try to get a listing into Computer Notes. Some of you may be able to contribute information that could be used as the basis for a ham software package. An example would be a set of mathematical formulas for predicting satellite orbits. Information like this made available through Computer Notes could result in a lot more programs for the library.

COMPUTER NETS?

If you know of any nets devoted to computer topics, please let me know, and I will spread the word. If you'd like to start a net, send me your suggestions for a band and time.

NEXT

Next month I'll talk a little about some of my own ideas for ham computers and software. I hope to have lots of your ideas to talk about also. Please write or call me at the MITS Repair Department with your comments and suggestions. 73.

September Software Contest

By Stan Webb

The first place winner this month is Kenneth Aird with his FORTRAN Cross Assembler for the 680. This program is beautifully written and well documented. With the addition of this program, we now have three FORTRAN Cross Assemblers in our library. Consequently, we cannot accept any more programs of this type.

Second place goes to Keith Fischer for his BECO Text Editor. This program, written in Altair BASIC, is considerably more powerful than the BASIC Editor, and would be a valuable addition to a BASIC system. The user documentation is fairly good, but it lacks much program documentation.

Erik Mueller takes third place with his MINOL Interpreter. MINOL is a subset of BASIC designed to run in less than 4K of memory. This subset is very restrictive when compared to a large BASIC, but is undoubtedly easier to work with than machine language. This program was previously published in the April, 1976, issue of Dr. Dobb's Journal of Computer Calisthenics and Orthodontia.

Due to the lack of competition in the subroutine category, no prize will be awarded this month.

AUTHOR'S COMMENT:

The library has a lot of material now, so I'd like to see our users put more effort into writing clean programs with good documentation. The programs that are neatly typed on our submission forms and that are well documented are more worthwhile to other users than those that are hastily done and have no documentation.

FIRST PLACE MAJOR PROGRAM

#9-1-761
Author: Kenneth Aird
Length: 41,000 bytes FORTRAN
Title: M6800 Cross Assembler
Very well written FORTRAN Cross Assembler for M6800.

SECOND PLACE MAJOR PROGRAM

#8-23-761
Author: Keith Fischer
Length: 150 lines Altair BASIC
Title: BECO
Powerful Text Editor.

THIRD PLACE MAJOR PROGRAM

#8-13-761
Author: Erik Mueller
Length: 7,000 (octal) bytes
Title: MINOL
Interpreter for a 4K subset of BASIC.

#8-9-761
Author: Alan Miller
Length: 200 lines Altair BASIC
Title: QUBIC
Plays 3D tic-tac-toe.

#8-10-761
Author: Alan Miller
Length: 7 lines Altair BASIC
Title: Numerical Integration
Numerical integration by Simpson's method and Trapezoidal Rule.

#8-16-761
Author: Roger Frank
Length: 7 lines Altair BASIC
Title: Memory Size
This program resets BASIC memory sizes without restarting (for 3-2 only).

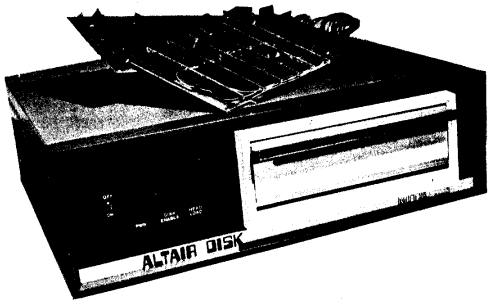
#8-19-761
Author: Alan Miller
Length: 5 lines Altair BASIC
Title: ARCSIN
Program to compute arcsin and arccos.

#8-26-761
Author: Roger Frank
Length: 22 lines Altair BASIC
Title: Memory Test
Program to test unused memory.

#8-27-761
Author: John Stanton
Length: 52 bytes
Title: 4PIO KBD/PTR Loader
Loads data into memory from keyboard and echoes it.

#8-28-761
Author: J. David Green
Length: 168 lines Altair BASIC
Title: Horse Racing
A horse racing game that involves betting on the races

altair™
Floppy Disk (88-DCDD)



The 88-DCDD consists of the Disk Controller and one Disk Drive with an interconnect cable. The Disk Controller consists of 2 PC boards (over 60 ICs) that fit in the Altair chassis. The Disk Drive unit consists of a PERTEC FD-400, a power supply PC board, and a Buffer/Address/ Line Driver PC board. The Disk Controller converts the serial data to and from 8-bit parallel words (one word every 32 microseconds). The Disk Controller also controls all mechanical functions of the disk as well as presenting disk status to the computer.

Software and System Features

Altair Disk Extended BASIC is an enhanced version of Altair Extended BASIC with added capabilities for saving and loading programs, and for manipulating data files on disk.

Altair Disk Extended Basic uses random and sequential files for storing information on disk.

Utility software is included with Altair Disk Extended BASIC for copying diskettes, initializing blank diskettes, listing directories, etc.

Disk bootstrap loader is available on paper tape, cassette tape, or PROM (used with 88-PMC PROM Memory Card).

Hard sectored format (non IBM compatible) allows storage of over 300,000 data bytes.

Altair Disk Extended BASIC requires a minimum of 20K of RAM memory to operate in.

PROM Disk Bootstrap loader allows loading of Altair Disk Extended BASIC in less than 10 seconds from the time power is turned on.

Hardware

A. Description and Features

The Disk Controller, which acts as the interface between the Altair and the Disk Drives, consists of 2 PC boards that fit in the Altair chassis. They require 2 slots in the Altair, contain over 60 ICs, and connect to the Disk Drives via an 18 pair flat cable. The Controller can address up to 16 drives.

The Disk Drive Unit consists of a Pertec FD-400 drive in an Optima case 5½" high, 17" wide, and 17½" deep (same width and depth as the Altair 8800). Also in the Disk unit is a power supply and a Buffer/Address card for selecting the drive and interconnecting multiple disk systems. A fan is included to maintain low ambient temperature for continuous operation. The Disk Drive units interconnect to each other in daisy chain fashion and to the controller using 18 pair flat cables and DC-37 type 37-pin rectangular connectors.

B. Hardware Specifications

Access Time:
- Track to track: 10 ms.
- Head load and settle time: 45 ms.
- Average time to read or write: 400 ms.
- Worst case: 1135 ms.

Rotational speed: 360 RPM (166.7 ms/rev)

Tracks: 77 per disk

Sectoring: Hard sectored, 32 sectors per track, 5.2 ms/sector (non IBM compatible)

Data Transfer Rate: 250,000 bits/sec. (one 8-bit byte every 32 microseconds)

Maximum number of drives per system: 16

Data storage capacity: 310,000 bytes per disk

Data bytes per sector: 128

Data bytes per track: 4,096

Disk Drive head life: over 10,000 hours of diskette to head contact

Disk Drive MTBF: exceeds 4,000 hours

Disk Drive data reliability: not more than 1 in 10^9 soft (recoverable errors, 1 in 10^{12} hard (non-recoverable errors

Power:
- Controller: 1.1 amps at + 8V unregulated (from Altair bus)
- Disk Drive Unit: 110 watts 50/60 Hz 117/220 VAC

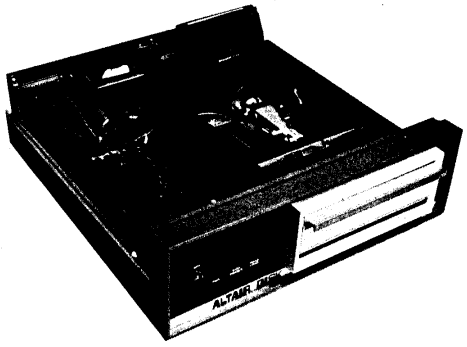
Diskette: Hard sectored, 32 sectors + index hole (Dysan #101, ITC #FD 32-100)

Disk Drive Unit Weight: 40 pounds

C. Operating Principle

The Disk Controller cards provide the interface between the Disk Drive Unit and the Altair bus. Serial read data from the disk is converted into 8-bit parallel form by the controller for transfer to memory via the CPU. Data is written on the disk by converting the 8-bit bytes outputted from the Altair CPU to serial form. All read and write data is transferred one byte at a time through the CPU.

Disk Controller Board #1 controls I/O address selection, sector counting, read data, and disk status. Disk Controller Board #2 controls disk drive addressing, write data, and disk drive functions.



Ordering information:

1. 88-DCDD

   Includes:
   - Set of controller cards
   - 1 Disk Drive Unit
   - 1 interconnect cable—6 ft. long
   - 1 Assembly and Operators Manual
   - 1 Disk Extended BASIC Manual
   - 1 Blank Diskette

2. 88-DISC

   Includes:
   - 1 Disk Drive Unit (117 VAC unless otherwise requested)
   - 1 Interconnect cable—6 ft. long
   - 1 Blank Diskette

3. Altair Disk Extended BASIC
   Requires a minimum 20K of memory for operation.

   Includes:
   - Altair Disk Extended BASIC on diskette
   - Altair Disk Extended BASIC Manual
   - Paper tape or cassette magnetic tape bootstrap loader (specify when ordering)

4. Disk Bootstrap Loader on PROM:
   Order 88-PMC (PROM Memory Card) and DBL PROM (PROM programmed with disk bootstrap loader routine)

5. Manuals only:
   - Disk Hardware Manual
   - Altair Disk Extended BASIC Manual

mits
2450 Alamo S.E./Albuquerque, New Mexico 87106

altair ambassadors: MITS, Money, and You

By Mike Hunter

MITS, the originator and leader of the personal computing revolution, has developed a program to further extend support to those Altair users living in regions where Altair retail centers presently do not exist. The Altair Ambassador program will offer qualified individuals the opportunity to be local MITS Altair representatives in cities where computing interest is large, yet the likelihood of having an Altair dealership is small due to the relative size of the community. Thus, the Altair Ambassador will be able to give MITS support through the selling and servicing of Altair computing equipment, for which he will receive a commission on each sale, and be able to do so using his home as his place of business! Personal computing will become even more personal, for the local MITS representative will be a member of your community--perhaps even yourself!

If you live in a community where an Altair retail center is not available and you are an Altair System owner, you have met the first requirement on the way to becoming an Altair Ambassador. Another criterion is that the Ambassador have a working knowledge of MITS hardware and software so that he may offer technical assistance and repair capabilities to other Altair users.

MITS will conduct weekend training sessions for potential ambassadors which will include discussions of software capabilities, repair techniques, product scope, and sales techniques. Each applicant will be required to attend one weekend training session at his expense. Upon notification of his acceptance into the program, the Ambassador will receive a program package including business cards, forms, catalogues and all current product literature.

Thus, the Altair Ambassador will receive full support from MITS so that he may best sell to and service his community, while making money at the same time. The only investment the Ambassador need make is travel and lodging expenses for the weekend training session in Albuquerque. Thus, if you own a MITS Altair System, and reside in a city without an Altair retail center, you may very well be the local computer expert whose future is to be an Altair Ambassador.

Please write for more information and an application form to:

Altair Ambassador Program
MITS, Inc.
2450 Alamo SE
Albuquerque, NM 87106

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.

BOOK REVIEW

Continued from Page Three

The second portion of the book brings the reader up-to-date with chapters on "The Present and Potential," "Applications" and "Governmental Uses" of computers. In "Justice, the Constitution and Privacy" Sam Ervin, Jr., U-S Senator from North Carolina, raises some interesting questions concerning the computer's role in government surveillance and the individual's right to privacy. On a more humorous side, Art Buchwald's "The Curse" warns of the horrible consequences a computer metes out when a defiant citizen dares to fold, bend and mutilate his phone bill and send it (with payment) back to the company.

The book's final three chapters --"The Impact of Computers," "Controls or Maybe Lack of Controls" and "Your Future"--explore the many significant effects the computer has upon our everyday lives and the potential role it plays in our country's future. Among the various articles in this section discussing both sides to the computer questionnaire, "Computerized Dating or Matchmaking," "Computer Crime" and "Machines Hold Powers for Good and Evil."

Interspersed among the many informative articles are imaginative poems, computer-generated illustrations and cartoons. Throughout the book, the famous comic strip character Doonesbury and his friend, Mark, marvel at the many wonders of the computer. A newspaper ad for computer operators convinces them that they have found their true vocation in life. "Earn $7,000, impress your friends. MEET GIRLS!"

In addition to all that humor, intrigue and important information to both the computer and noncomputer specialist, The Compleat Computer offers fictional romance about a computer named Max who almost breaks up a marriage. For $5.95 a copy, who could ask for more?

Copies of The Compleat Computer can be obtained directly from the publishers: Science of Research Associates, 1540 Pagemill Road, Palo Alto, California.

Van Tassel has also published Program Style, Design, Efficiency Debugging and Testing (Prentice Hall, Inc.).

I am writing this letter for two reasons. Before I commence with the diatribe, however, I would like to say that I enjoy very much the 17K Altair 8800 system that is up and running at our house. We use the 8K version of BASIC.

On the very first page of the BASIC Reference Manual you state that BASIC was originally developed at Dartmouth University. This is incorrect. I am a student at Dartmouth College; to myself and all other Dartmouth students and alumni this reference to the "university" is a slur.

The distinction, seemingly minor, is historically and legally quite important. In 1818 the state of New Hampshire tried to take control of the College and turn it into a state school. They were partially successful, in that Dartmouth University was created. The University used the same classrooms, dormitories, and chapel as the College. Needless to say, there was considerable unrest from both students and faculty.

In 1819 the problem reached the Supreme Court. Daniel Webster (class of 1801) argued eloquently for the College and triumphed. The Dartmouth College case was a landmark decision that guaranteed the inviolability of legal contracts.

So you see when you attribute BASIC to Dartmouth University, you credit an unenduring and illegal "splinter school" of the early 1800's. Please try to correct this error in subsequent printings of the BASIC manual.

When working on the Dartmouth computer, I have found that the VAL and STR$ functions are useful. VAL takes a single string argument and converts it to a constant:

```
X$="-17.69"
X=VAL(X$)
```

X equals -17.69. STR$ does the opposite:

```
X=25*25
X$=STR$(X)
```

Therefore, X$="625". Also useful is the POS function, which searches a string for the presence of another string and returns the substring's location:

```
Q=POS(A$,B$,C)
POS(A$,"Z",1)=0
POS(A$,"LTA",3)=0
```

Thank you for your time.

Sincerely,
John Sotos

Ed. Note: STR$ and VAL are present in 8K and Extended BASIC, and the equivalent of the POS function in Extended BASIC Ver. 3.4 is INSCR.

CUSTOMER SERVICE NEWS
- continued from page 2

Shipping and handling charges - Rates for continental United States shipments - Prepaid or UPS COD:

$8.00 each for mainframes and terminals

$3.00 for up to three (3) peripheral boards, add $1.00 per board thereafter.

We are unable to ship COD to the following:

(1) Post Office Boxes
(2) Companies or Educational Institutions
(3) APO or FPO Addresses
(4) Foreign Countries (including Canada)

Foreign Countries (including Canada) - Repair shipments will be made via Emery Air Freight Collect. Also, charges incurred for units coming in from foreign countries to MITS, e.g. customs charges, will be billed as part of the customer's repair charge. (These charges average between $30.00 and $40.00.)

Please remember to send in payment for these charges, otherwise a delay will occur while we contact you for payment or COD authorization. We will accept for payment: Master Charge, BankAmericard, money order, personal check (three week delay for processing) or authorization for COD for charges under $15.00. Companies and educational institutions should remember to send in their Purchase Orders authorizing repair and return shipment charges.

I hope this will help those of you asking what our procedure is for returning items for repair. If you have further questions, please contact the Repair Department.

See you next month - Gale

LETTERS TO THE EDITOR

Continued from Page Two

Dear Ed.,

I am writing this letter for two reasons. Before I commence with the diatribe, however, I would like to say that I enjoy very much the 17K Altair 8800 system that is up and running at our house. We use the 8K version of BASIC.

On the very first page of the BASIC Reference Manual you state that BASIC was originally developed at Dartmouth University. This is incorrect. I am a student at Dartmouth College; to myself and all other Dartmouth students and alumni this reference to the "university" is a slur.

The distinction, seemingly minor, is historically and legally quite important. In 1818 the state of New Hampshire tried to take control of the College and turn it into a state school. They were partially successful, in that Dartmouth University was created. The University used the same classrooms, dormitories, and chapel as the College. Needless to say, there was considerable unrest from both students and faculty.

In 1819 the problem reached the Supreme Court. Daniel Webster (class of 1801) argued eloquently for the College and triumphed. The Dartmouth College case was a landmark decision that guaranteed the inviolability of legal contracts.

So you see when you attribute BASIC to Dartmouth University, you credit an unenduring and illegal "splinter school" of the early 1800's. Please try to correct this error in subsequent printings of the BASIC manual.

When working on the Dartmouth computer, I have found that the VAL and STR$ functions are useful. VAL takes a single string argument and converts it to a constant:

```
X$="-17.69"
X=VAL(X$)
```

X equals -17.69. STR$ does the opposite:

```
X=25*25
X$=STR$(X)
```

Therefore, X$="625". Also useful is the POS function, which searches a string for the presence of another string and returns the substring's location:

```
Q=POS(A$,B$,C)
```

POS looks for the location of B$ in A$ starting at location C in A$:

```
A$="ALTAIR"
POS(A$,"T",1)=3
POS(A$,"LTA",1)=2
POS(A$,"Z",1)=0
POS(A$,"LTA",3)=0
```

Thank you for your time.

Sincerely,
John Sotos

Ed. Note: STR$ and VAL are present in 8K and Extended BASIC, and the equivalent of the POS function in Extended BASIC Ver. 3.4 is INSCR.

CUSTOMER SERVICE NEWS

- continued from page 2

Shipping and handling charges - Rates for continental United States shipments - Prepaid or UPS COD:

$8.00 each for mainframes and terminals

$3.00 for up to three (3) peripheral boards, add $1.00 per board thereafter.

We are unable to ship COD to the following:

(1) Post Office Boxes
(2) Companies or Educational Institutions
(3) APO or FPO Addresses
(4) Foreign Countries (including Canada)

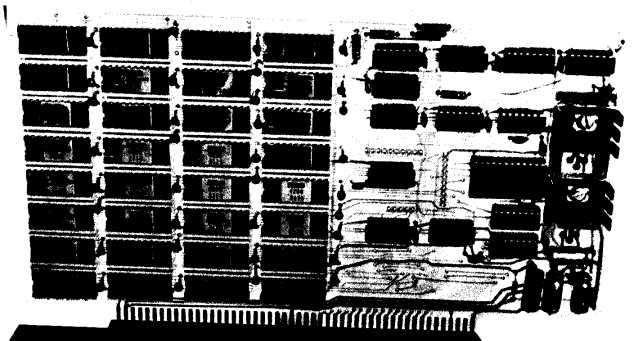
Foreign Countries (including Canada) - Repair shipments will be made via Emery Air Freight Collect. Also, charges incurred for units coming in from foreign countries to MITS, e.g. customs charges, will be billed as part of the customer's repair charge. (These charges average between $30.00 and $40.00.)

Please remember to send in payment for these charges, otherwise a delay will occur while we contact you for payment or COD authorization. We will accept for payment: Master Charge, BankAmericard, money order, personal check (three week delay for processing) or authorization for COD for charges under $15.00. Companies and educational institutions should remember to send in their Purchase Orders authorizing repair and return shipment charges.

I hope this will help those of you asking what our procedure is for returning items for repair. If you have further questions, please contact the Repair Department.

See you next month - Gale

One Slot!



Altair™ 16K Static

Almost too good to be true, the Altair 16K Static RAM board is easily the most advanced memory module yet developed for the Altair 8800, 8800a and 8800b computers.

Four Altair 16K Static boards add up to the entire 64K of memory directly accessible by the Altair.

The Altair 16K Static board offers two surprise features—minimal power requirements and fast access time. One Altair 16K Static board draws less current than any 8800 compatible 4K boards, thus four Altair 16K Static boards can be plugged into the Altair 8800 without beefing up the power supply.

The maximum access time of the Altair 16K Static board is 215 nanoseconds, which makes this board the **fastest Altair compatible static board in existence.**

The Altair 16K Static is now in full production. Special introductory price is $765 in kit form and $945 assembled.

MAIL THIS COUPON TODAY

Enclosed is check for $_____

BankAmericard #_____

or Master Charge #_____

☐ Altair 16K Static ☐ Kit ☐ Assembled
(include $3 for postage and handling)

☐ Please send free information package and price sheet.

NAME_____

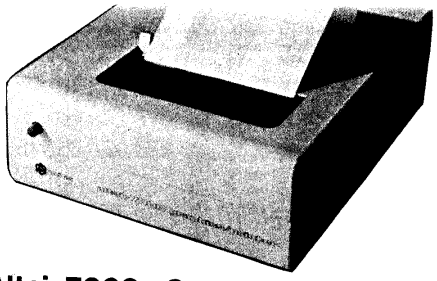
ADDRESS_____

CITY_____ STATE AND ZIP_____

MITS/2450 Alamo SE/Albuquerque, NM 87106/505-243-7821

Prices, delivery and specifications subject to change. Allow up to 60 days for delivery.

2450 Alamo SE/Albuquerque, NM 87106/505-243-7821



Altair 7000 Graphics Printer

Keeping up with its innovative tradition, MITS has recently announced several new products which will greatly expand the capabilities of the Altair Computer system.

The tremendous flexibility of the 7000 Graphics/Printer, which acts as a printer, plotter and graphics device, makes it one of the fastest and most economical methods of electrostatic printing. The new Universal I/O board for the Altair 680b expands the I/O capabilities of the 680b beyond the one serial port on the main board. Thanks to the 88-Mux (24 Channel Multiplexer), the input capacity of the 88-Analog-to-Digital Converter for applications requiring a large number of analogs has been greatly increased. The new 88-S4K Memory Board now makes totally synchronous *memory logic available to Altair 8800 users.*

The 7000 Graphics/Printer

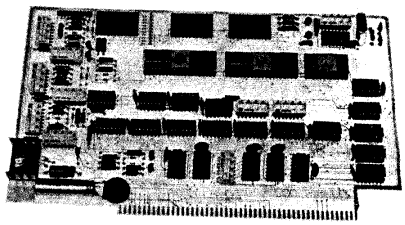
Although there are a number of methods used for printing computer output, electrostatic printing is finally being recognized as the only method which is fast, economical and, now with the MITS 7000, is also the most flexible means. In last month's C.N. we introduced the 7000 Graphics/Printer as a multifunction, hard-copy output device which is plug compatible with the 680 and 8800 mainframes via one PIO port. The enthusiastic response to the 7000 warrants a more detailed explanation of its operation and applications.

The flexibility of the 7000 is due to eight print electrodes, driven directly by software, instead of the usual seven found in 5 x 7 matrix printers. Copies made from the printed output are actually more legible than copies of typed paper and can be made for about 1¢ per foot of electrosensitive paper.

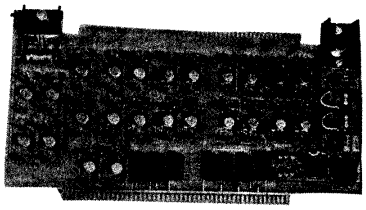
When the 7000 is used as a line printer, characters are generated using a 5 x 7 dot matrix. Altair BASIC supports three different sizes of character sets (each with upper and lower case) to produce line widths of 20, 40 or 80 characters in the four-inch wide printing area. The speed is 160 characters per second (80 characters per line) or 120 lines per minute. Different character sizes are selected with the CHR$ function in BASIC.

The eighth or extra printing electrode in this unit provides symmetry along the horizontal and vertical axes to permit plotting. With the vertical distance between electrodes equal to the distance between lines, there's no gap from line to line. This special feature makes the 7000 ideal for graphics. Pictures can be produced that show either a distinct outline or a sophisticated, detailed picture with shaded areas. When the eight-dot columns are printed close together,

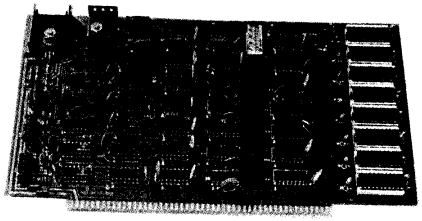
– Continued on Page Four –



680-b Universal I/O Board



88 - Multiplexer



88 -S 4k Memory Board