computer notes
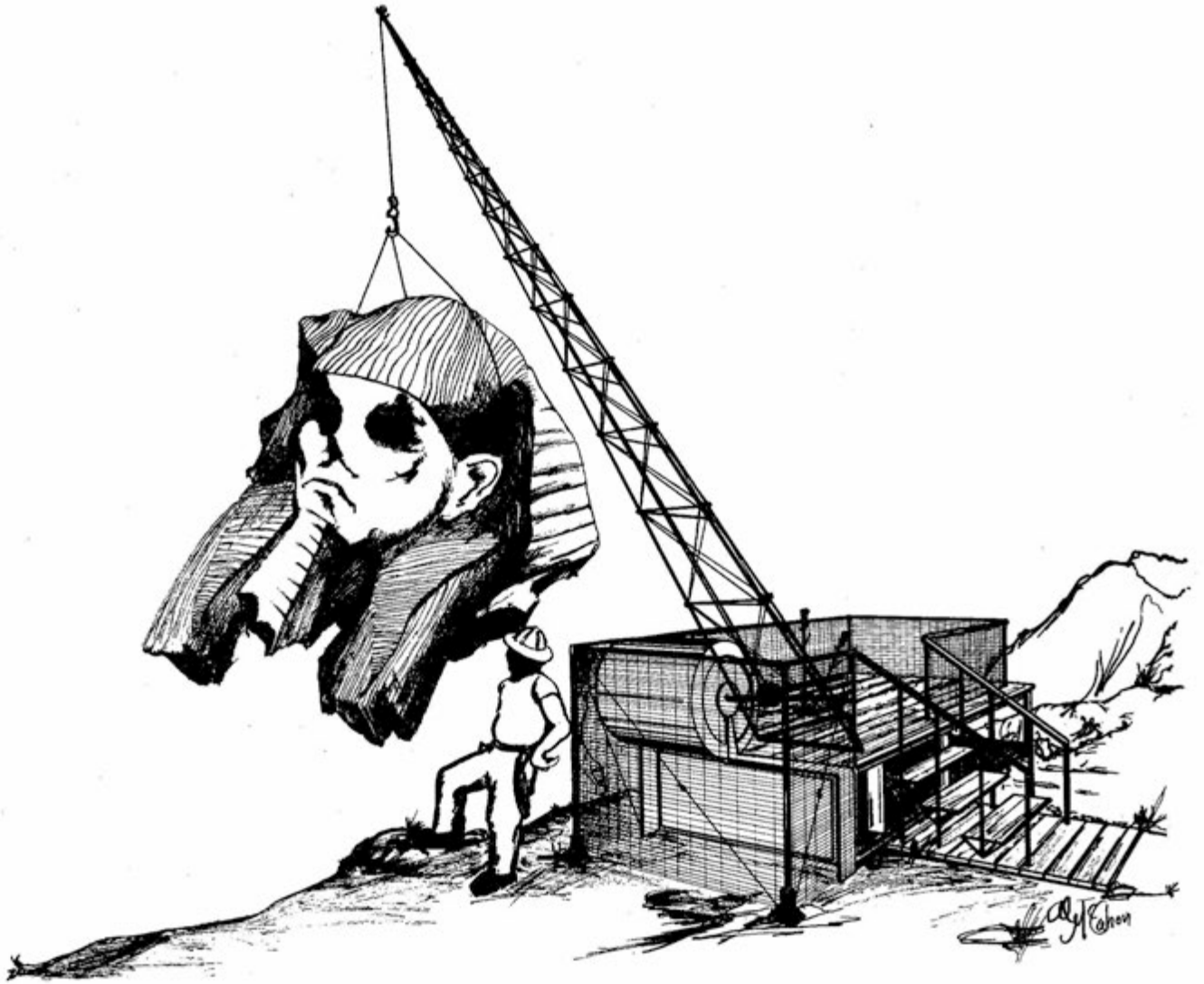
magazine

This month Computer Notes has gotten a . . . well, facelift. Our new, smaller magazine format makes C/N easier to read and more convenient to file. Each issue will still contain the usual amount of top-quality technical information, and to provide even more space for articles, we've cut back the number of advertising pages.

Many readers have requested the change, and MITS is always anxious to provide continuing support to all Altair computer owners whenever possible.

BASIC File Handler Developed for Hams

By Dr. George L. Haller

This article describes a simple program in BASIC which will keep track of your QSLs and also give you information on the repeaters that you may wish to use. The program may also be extended to provide other functions in logkeeping, but it was kept simple to give the hams who are new computer freaks a chance to get acquainted with their new toy. It is written in Altair™ BASIC version 3.2 but could be easily modified to any other BASIC.

A Sample Run of the Ham File

RUN

REPEATER AND QSL FILE

FOR REPEATERS USE LAST THREE LETTERS OR FREQUENCY AS 1372 OR CITY NAME, FOR AMATEUR STATIONS USE COMPLETE CALL

STATION? AAA
WR4AAA, 13/73, 190, SALISBURY, NC.

END OF SEARCH

STATION ? 28/88
WR4ABF, 28/88, 195, SHELBY, NC

END OF SEARCH

STATION ? HICKORY
WR4ACM, 25/85, 195, HICKORY, NC

END OF SEARCH

STATION ? ABQ   Continued

IN THIS ISSUE

BASIC File Handler
Developed for Hams
Continued

NO RECORD IN FILES

STATION ? CR6CA
5/26/71, 0015Z, 14MHZ, RTTY, JOE

STATION ? K4GV
11/6/75, 0710P, 2M-FM, TED
12/5/75, 0655P, 2M-FM, TED

END OF SEARCH

STATION ? W2ABC

NO RECORD IN FILES

STATION ? W4BQ
THAT'S YOU, STUPID!

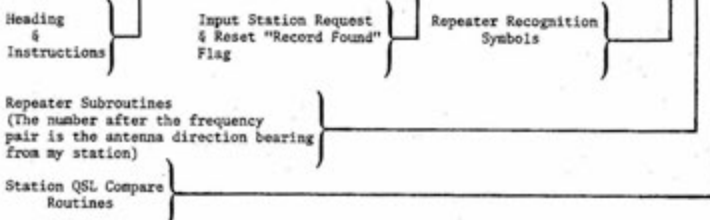END OF SEARCH

STATION ?

OK

Application Program

A sample readout shows that
information about a repeater can be
obtained by replying to the initial
question of the computer ("STATION?")
with the last three letters of the
repeater call, such as "AAA," by
typing in the frequency pair as "37/
97" or even by typing in the loca-
tion city as "SHELBY." If your input
is a city name, all the repeaters in
that city will be printed, and the
same goes for the repeater frequency
pairs. But only one repeater will be
printed for each call. If you should
be working in an area which crosses
district boundaries, and the same
last three letters have been assigned
to repeaters that you work, then the
identifier should be extended to in-
clude the district number (e.g.,
"3AAA" and 4AAA.")

For information from the log
on amateur QSLs, the "STATION?"
request is followed by typing in
the full call of the amateur station.

```
LIST
5 REM HAM FILE IN ALTAIR BASIC BY GEORGE L. HALLER
10 PRINT:PRINT
20 PRINT "REPEATER AND QSL FILE"
30 PRINT
40 PRINT "FOR REPEATERS USE LAST THREE LETTERS OR FREQUENCY AS 13/73"
50 PRINT "OR CITY NAME, FOR AMATEUR STATIONS USE COMPLETE CALL."
60 PRINT:PRINT
70 INPUT "STATION";A$
80 F=0:REM RESET "RECORD FOUND" FLAG
100 IF A$="AAA" OR A$="13/73" OR A$="SALISBURY" THEN GOSUB 300
110 IF A$="ABF" OR A$="28/88" OR A$="SHELBY" THEN GOSUB 310
130 IF A$="ACM" OR A$="25/85" OR A$="HICKORY" THEN GOSUB 330
280 IF F>0 THEN 1000:REM IF REPEATER IS FOUND
290 GOTO 400 :REM JUMPS REPEATER SUBROUTINES
300 PRINT "WR4AAA, 13/73, 190, SALISBURY, NC.":F=1:RETURN
310 PRINT "WS4ABF, 28/88, 195, SHELBY, NC":F=1:RETURN
330 PRINT "WR4ACM, 25/85, 195, HICKORY, NC":F=1:RETURN
400 IF A$="W4BQ" THEN PRINT "THAT'S YOU, STUPID!":F=1
410 IF A$="HB9HK" THEN PRINT "7/20/73, 2030Z, 14MHZ, RTTY, WILLY":F=1
420 IF A$="CR6CA" THEN PRINT "5/26/71, 0015Z, 14MHZ, RTTY, JOE":F=1
430 IF A$="OA4BR" THEN PRINT "7/3/74, 0915Z, 14MHZ, RTTY, ZIP":F=1
440 IF A$="JH1TFF" THEN PRINT "5/27/74, 1150Z, 14MHZ, RTTY, DOC":F=1
450 IF A$="KJ6BZ" THEN PRINT "7/6/73, 0200Z, 14MHZ, RTTY, GEORGE":F=1
460 IF A$="K4GV" THEN PRINT "11/6/75, 0710P, 2M-FM, TED":F=1
470 IF A$="W4MYG" THEN PRINT "11/3/75, 0725P, 2M-FM, RAY":F=1
480 IF A$="K4GV" THEN PRINT "12/5/75, 0655P, 2M-FM, TED":F=1
490 IF A$="K4FZ" THEN PRINT "2/17/76, 0430P, 14MHZ, SSTV, BOB":F=1
1000 PRINT
1010 IF F>0 THEN PRINT"END OF SEARCH":GOTO 60
1020 PRINT "NO RECORD IN FILES":GOTO 60
OK
```

Heading
&
Instructions

Input Station Request
& Reset "Record Found"
Flag

Repeater Recognition
Symbols

Repeater Subroutines
(The number after the frequency
pair is the antenna direction bearing
from my station)

Station QSL Compare
Routines

GLITCHES
Q & A from the Repair Department

By Bruce Fowler

We get many calls on how to
interface terminals to the 2SIO.
The problem is that the ACIA's hand-
shaking signals make interfacing
with the 2SIO a somewhat complicated
matter. An explanation of the sig-
nals and their function should make
the job easier. The three hand-
shaking signals--Data Carrier Detect
(DCD), Request to Send (RTS) and
Clear to Send (CTS)--permit limited
control of a modem or peripheral.
RTS is an output signal, and DCD and
CTS are input signals.

CTS and RTS were designed to
act as handshaking signals with a
modem. When the ACIA contains data
to be transmitted, RTS will go
active. This output signal tells
the I/O device that the ACIA is
ready to transmit. If the I/O
device is ready to receive infor-
mation from the ACIA, it will res-
pond by making CTS active. When
the ACIA CTS input goes active,
this enables the ACIA transmit
buffer, and data is transmitted.
However, if the terminal is not
ready, the terminal will keep CTS
inactive. Data will only leave
the ACIA when CTS is active.

The ACIA will receive data
only when DCD is active. DCD is
normally used with modems. As long
as DCD is inactive, the ACIA's
receiver section is inhibited and
no data can be received by the
ACIA. The purpose of DCD is to
prevent noise from being received
as data by the ACIA. When the mod-
em detects a valid carrier signal,
it informs the ACIA by making DCD
active. The ACIA then will accept
data.

ALTAIR 680B
AIDS IN BRAIN RESEARCH

By: Steve Pollini

Mr. Pollini, a pre-med student at UNM, has been doing neuropsychology research for the past year at the university, under the direction of Dr. D. M. Feeney, Associate Professor of Psychology and Physiology.

The majority of non-congenital epilepsy (epilepsy not present at birth) is caused by brain injury, which may result from a severe blow on the head; a penetrating missile wound or laceration, such as a bullet hole; infection or damage caused by a brain tumor. Until recently, very few researchers have investigated the effects of such injuries on single neuron cell firing patterns (the electrical activity of a cell).

The purpose of my research is to compare the firing patterns of normal cells to that of injured cells. Cats were used in the study because of the similarity of their brain cell structure to that of humans. The cats' brain cells were injured by passing a small electric current through an electrode inserted near the brain cells being studied.

The Altair 680b was chosen for the study for a variety of reasons: its powerful instruction set, ability to use machine language subroutines and ease of interfacing with laboratory and peripheral display devices. It also made my research much more efficient and less time-consuming by effectively analyzing a huge amount of data in a matter of seconds. My lab assistant and I now have more time to design and perform other experiments rather than spending tedious hours in front of an oscilloscope, counting cellular time intervals.

The Altair 680b computer not only helped increase the reliability of the data but also the flexibility of the lab. We can now change experimental paradigms and analysis methods by simply modifying the basic program.

The system consists of an Altair 680b computer, a 680b-BSM 16K Static Memory card and a 680b Universal I/O card with two parallel ports and one serial port. Peripherals attached to the 680b include an ASR-33 Teletype, a Hazeltine 2000 CRT, a window discriminator and a 1 KHz clock. Other equipment includes an oscilloscope, a Pico-Metric Microelectrode System Amplifier and a Grass, Inc. Model 78B Electroencephalographic Amplifier.

The first step in the study consists of surgically implanting a ring mount on the top of a cat's skull. This ring mount is implanted so that when a microelectrode apparatus is mounted to it, the electrode can be lowered into the exact region of the brain that is to be studied. (See Figure 1.)

To permit the microelectrode to enter the brain, a tiny bone flap (approximately 1 cm^2) is removed from the skull where the center of the ring mount is located. The surgery is then completed by screwing a cap on the ring mount. This cap prevents cerebral-spinal fluid from escaping and infection from entering the brain area. The cat spends the next two weeks recovering from the effects of the anesthesia and surgical trauma.

To record from the brain cells, a hydraulic microdrive with a microelectrode is attached to the ring mount. The microdrive is electronically driven with a stepping motor and can advance or retract the electrode as little as one micron at a time. (One micron is the same as one thousandth of a millimeter.) The electrode itself has a tip that is only two or three microns in diameter. Such a small electrode is needed because the neurons from which we record are usually only about 10 microns in diameter.

We spend the first five work sessions making two-minute recordings of normal cells in a track and storing that information on magnetic tape. A track is the space formed each time the electrode is placed in a new position in the cat's brain and moved downward to record the cells within its path. The electrode covers a distance of 5,000 microns or 5 millimeters and usually records 15-20 cells per track. Four tracks are recorded during each session, which makes a total of about 80 cells per session.
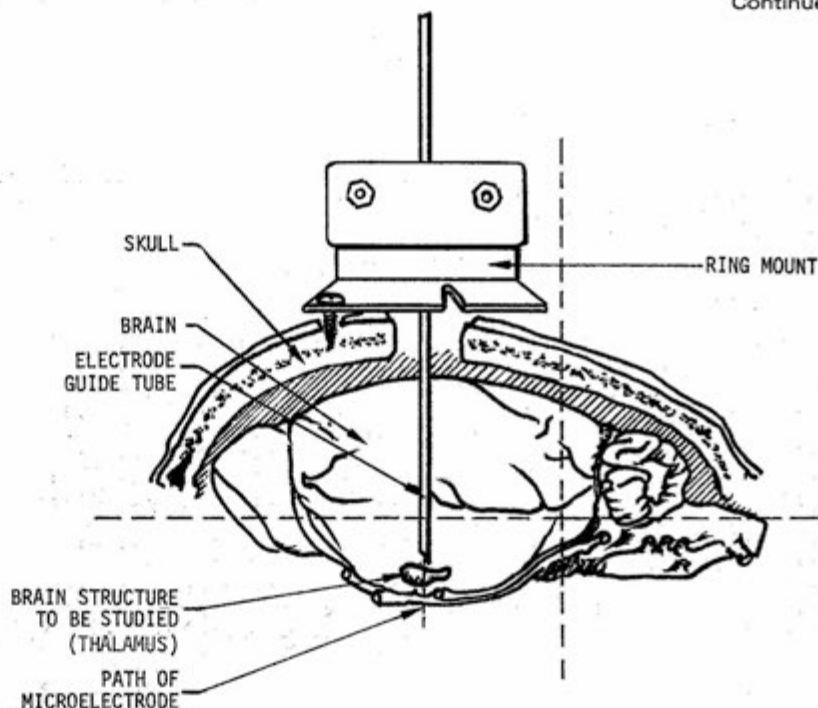
Continued

SKULL

BRAIN
ELECTRODE
GUIDE TUBE

RING MOUNT

BRAIN STRUCTURE
TO BE STUDIED
(THALAMUS)

PATH OF
MICROELECTRODE

Figure 1.

When this search for cells in
a track is done, small pulses can
sometimes be seen on the oscillo-
scope. But the pulses aren't large
enough to record, so the electrode
must be moved down into the brain
until sizeable "spike" potentials
or pulses show up on the oscillo-
scope. (See Figure 2.) Then the
neuron activity in this area is
recorded for two minutes before
moving on to the next cell.

After five sessions or 400
cells have been recorded, a lesion
is made next to the track sites by
placing a special electrode at the
site and passing a small current
through it. Then a series of
recordings are made on following
days to study the effects of the
lesion on the surrounding brain
cells.

When this process is complete,
two-minute recordings of over 1,000
neurons have been accumulated. Each
of these recordings must then be
analyzed to determine the type of
firing pattern the cell displayed
as well as the average and fastest
rates at which it fired. Without
a computer, this would be a tedious
and time-consuming process.

To study these different pat-
terns of neuron firing, the interval
between spikes (the amount of time
between firings) is measured. These
intervals can be short, which means
that the cell is firing very rapidly,
or they can be long, which indicates
slow firing. Normal cells fire any-
where from once a second up to
about 700 times per second. They
often appear to be firing spon-
taneously with no particular pat-
tern. (See Figure 2.) Other times
they fire in bursts. (See Figure
3.)

Constant amplitude or the
ability to produce spikes of the
same height is an important pro-
perty of these spike potentials.
This may not appear to be true in
Figure 4, which shows spikes of
two different heights. But that's
because Figure 4 shows two differ-
ent cells being recorded at the
same time. One cell has a greater
amplitude because it is closer to
the electrode tip than the other
one. This presents an obvious pro-
blem. How does the computer dis-
tinguish between one cell and the
other? A window discriminator
solves this problem. It allows
the spikes of only one cell to get
through to the computer to be ana-
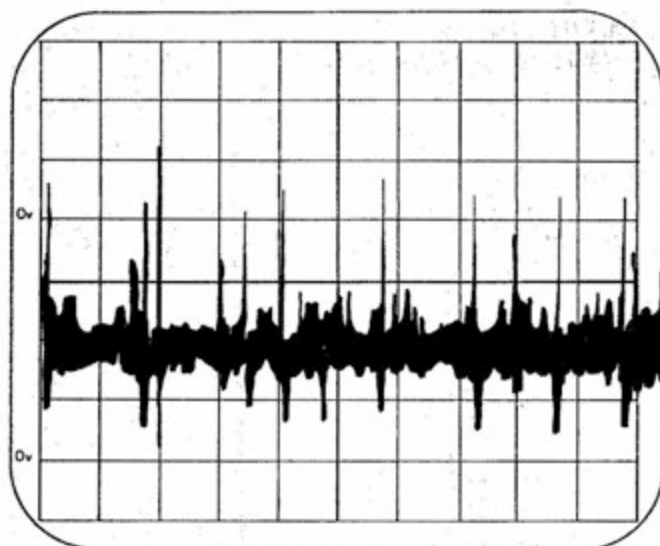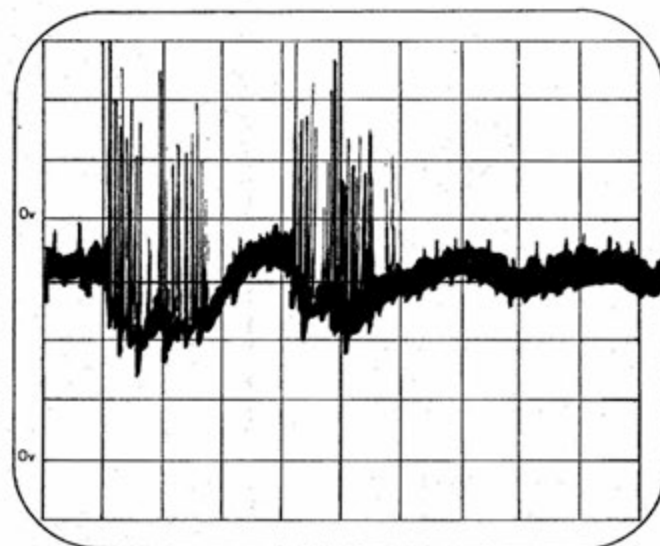lyzed. (See Figure 5 on page 6.)
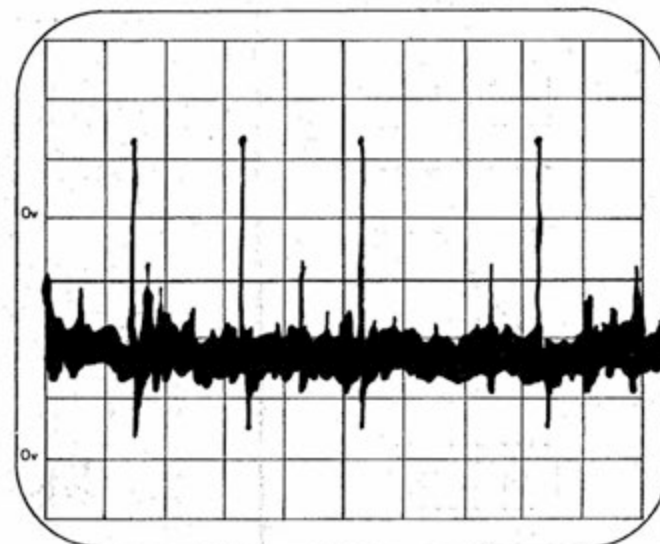

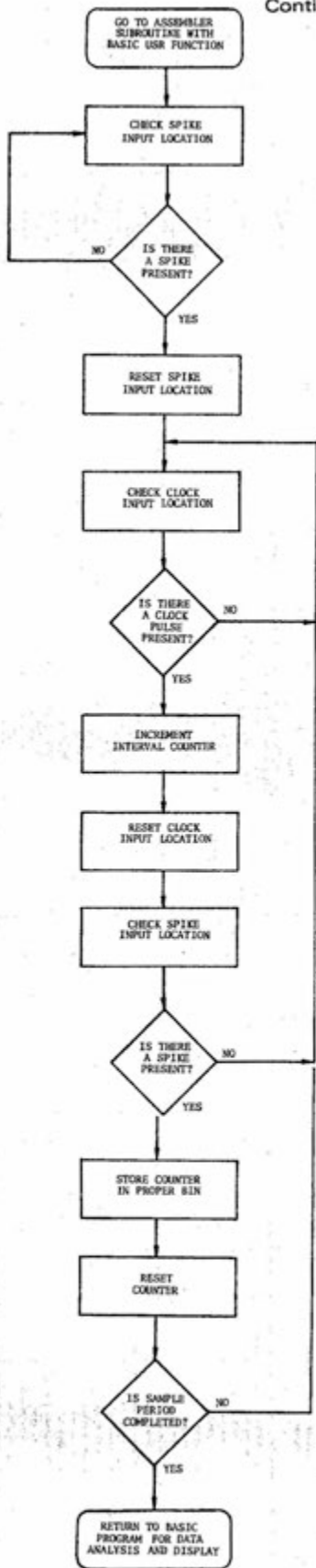Figure 2.

Figure 3.

Figure 4.

FOUR

Continued

Figure 6.

Continued

Case Study:

ALTAIR COMPUTER SCORES HIGH AT GOLF TOURNAMENT

By O. E. (Gene) Dial

Dr. Dial is a professor in the Graduate School of Public Affairs at the University of Colorado. This is the second in a series of two articles by Dr. Dial, explaining how he used an Altair 8800 computer to score a golf tournament.

The portability of the Altair 8800 computer makes it an excellent candidate for a potential new market in scoring golf tournaments, which is becoming an increasingly complex task.

The fun of amateur golf invitationals is enhanced greatly by the handicap system, which makes it possible for players of different abilities to compete not only against each other but against their own track record. But golfers are a suspicious lot when it comes to viewing the handicaps of other players, particularly if the players are from out of town.

Many strategies have been devised to improve the handicap system so that all players are equally satisfied. One method is the points system, which seems to be an increasingly popular method used at invitationals. The major disadvantage of the system is that it requires many tedious, error-prone calculations.

Point System

Each local player is matched with an invited player. The handicaps of each are summed and then subtracted from 72. The result is called the Initial Quota. On the first day each player earns gross points on each of the 18 holes by comparing his hole score with the par value of the hole. Points range from $\emptyset$ (a triple-bogie/or worse) to 6 (an eagle--two under par). These points are added up for every two-man team, then the total is reduced by the amount of the team's Initial Quota. The result is plus or minus Net Points earned for the day's play.

As a further precaution against sandbagging, the team's Initial Quota is adjusted by one-half the Net Points earned. For example, if a team accumulates 10 points, its quota would be raised 5 points. The impact of this Revised Quota is felt on the second day of play.

This is just a sample of the tens of thousands of calculations that must be done, and each must be verified independently by the tournament committee. The Altair microcomputer and an enterprising operator who wants to go into business would be an ideal pair for this task. The following programs, which include some minor improvements, were used in the 1976 Boulder Country Club Frolics--a two-day invitational golf tournament for the club's members and guests.

Based on practical considerations (discussed in my first article), the package is composed of two programs: 1) a Scores Processing Program (SPP), and 2) a Video Display Program (VDP). As score cards come in, the SPP digests the data (Revised Quota, Points and Scores) from four or five of them at a time, so it can later be assembled as DATA statements for the VDP. The SPP is usually run for about an hour at noon and again in the evening each day of the tournament. The VDP is RUN at all other times.

Extended BASIC is used for both programs. The SPP requires about $111\emptyset$ words for an average load of data, and the VDP, fully loaded, requires about $3\emptyset\emptyset\emptyset$ words in addition to about $5\emptyset\emptyset$ words of string space. Obviously, the VDP should run safely with 16K of memory. But this will vary, of course, with the number of teams in the tournament. The safest approach is to fully load the program with dummy data for the maximum number of teams anticipated in the tournament. Then, with crossed fingers, and fully expecting the need for another 4K board, make a test run.

RAW DATA INPUT TO WINDOW DISCRIMINATOR

OUTPUT WHEN WINDOW SET BETWEEN 2 AND 3

OUTPUT WHEN WINDOW SET BETWEEN 1 AND 2

Figure 5.

The machine language program once initialized, continually checks the data bit location where the spike data appears. A set condition of the bit means that a spike (neural firing) has occurred. If the bit is still clear upon checking, the program loops until the bit is set. Once a spike has occurred and has been registered, the input bit is cleared so that the next spike can be seen. The program then checks a clock input location for the presence of a clock pulse. A 1KHz crystal controlled clock, which provides a clock pulse once every millisecond, is used. When a clock pulse has been registered, a counter is incremented. So the counter keeps track of the time interval between spike recordings.

The clock input location is then reset. After each clock pulse is recorded, the 680b must check the spike input location to see if the interspike interval has ended. If no spike has occurred since the last clock pulse, another clock pulse is recorded. When the program senses that a spike has occurred, the amount of time in the interval counter is stored and once again the interval timing sequence is begun until the next spike occurs.
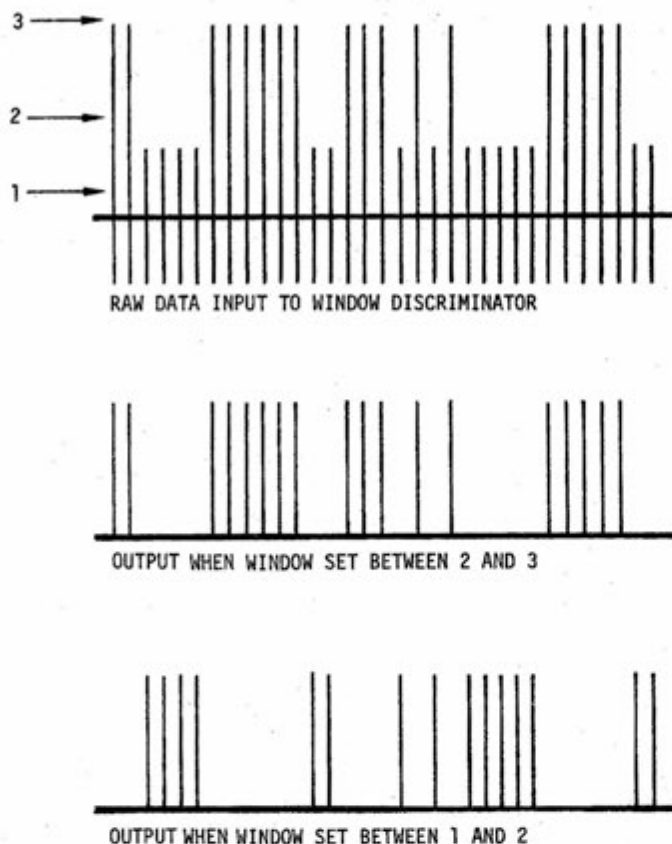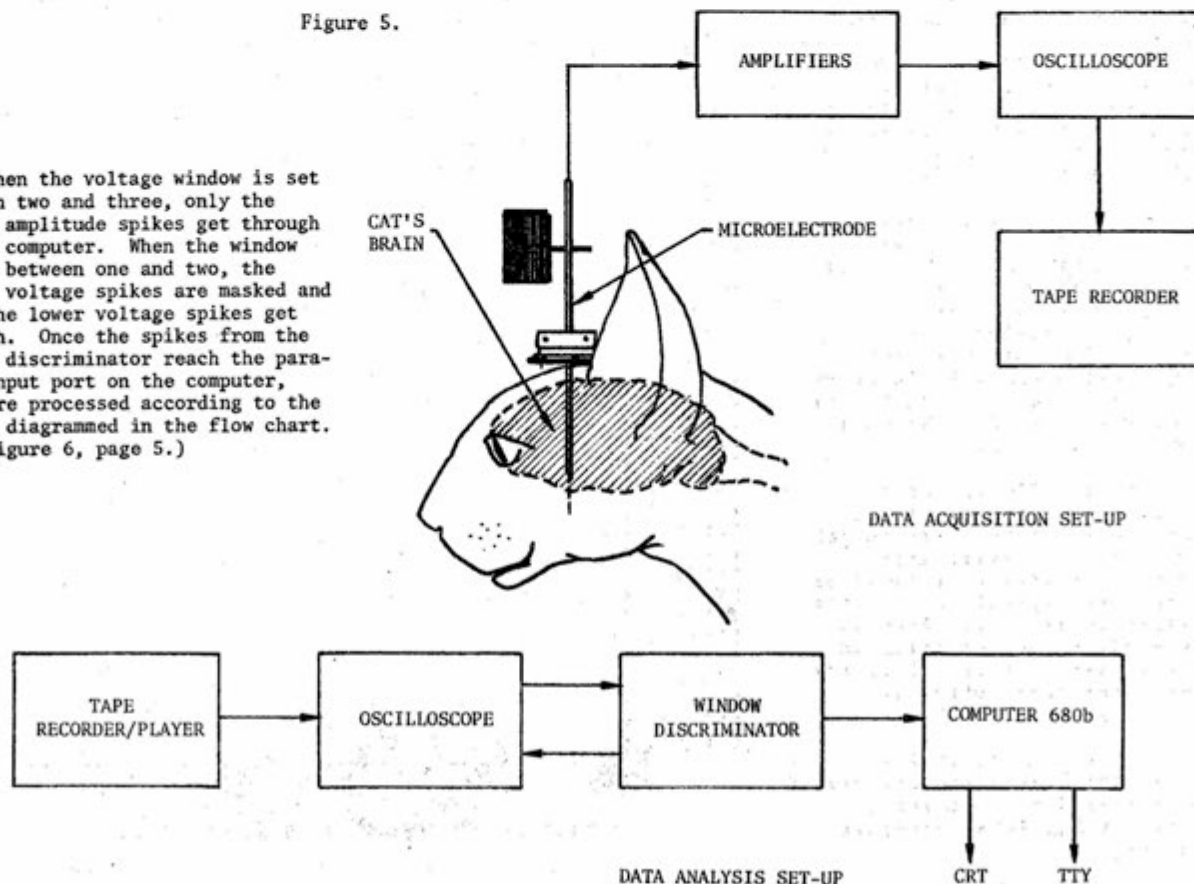
When the voltage window is set between two and three, only the higher amplitude spikes get through to the computer. When the window is set between one and two, the higher voltage spikes are masked and only the lower voltage spikes get through. Once the spikes from the window discriminator reach the parallel input port on the computer, they are processed according to the scheme diagrammed in the flow chart. (See Figure 6, page 5.)

DATA ACQUISITION SET-UP

DATA ANALYSIS SET-UP

STUDY: CHRONIC SINGLE UNIT RECORDING IN THE CAT THALAMUS
CAT NUMBER: CSU-05
RECORDING SESSION: PRE-LESION #3
DATE: 10 NOVEMBER 76
TRACK NUMBER: 17
ELECTRODE HEIGHT: 14768

Each time an interval is completed, the program checks another preset location to see if the sample period has ended. When the sample period is complete, all of the specific intervals are stored in one of 65 bins. (A bin is the memory location that stores the number of times a particular interspike interval has been counted.) After the information has been stored, the machine language or assembler subroutine then jumps back to a BASIC program.

Next, the main BASIC program plots the spike interval data as a histogram, such as the one in Figure 7. The position of the asterisk in each time bin designates how many interspike intervals equalled that amount of time. This histogram represents a cell that fired in bursts. (See Figure 3.) Notice that when the cell fires, there are many short intervals between the firings. On the histogram this is represented by the peak between five and 12 milliseconds. Between each burst is a large interval--somewhere between 34 and 47 milliseconds, as shown by the second peak on the histogram. A nonbursty cell would show a single but generally wider distribution peak.
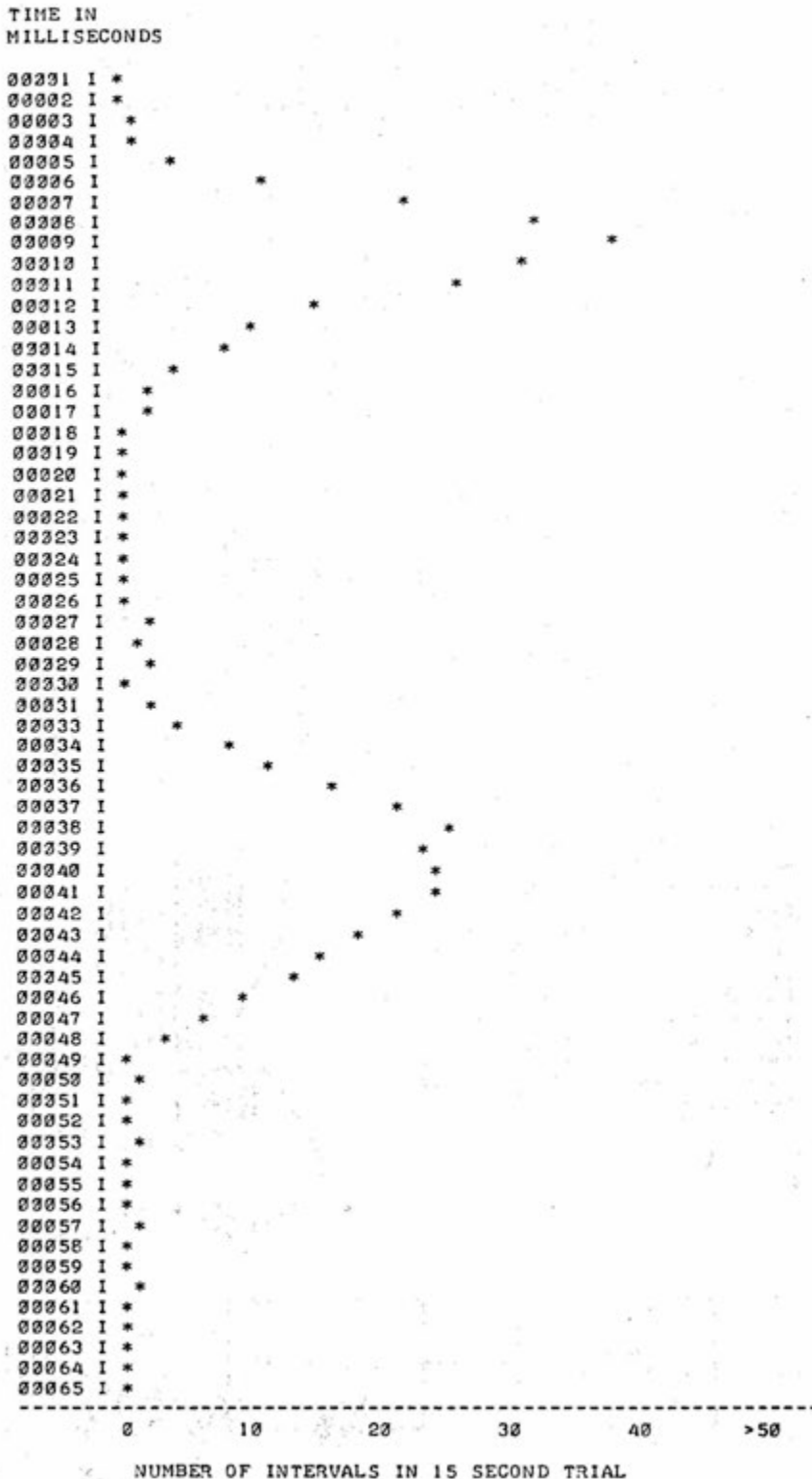
To see the difference between normal and abnormal or epileptic neurons or cells, we look for shifts in the peaks on the histograms. If the peak is very close to the 1 to 2 millisecond end of the histogram, the cell would be firing upwards to 1000 times per second, which normal cells generally do not do.

Two or three such histograms are usually done for each cell recording to make sure that the cell was stable in its firing pattern during the two-minute recording session. The prelesion and postlesion cell histograms are then compared to determine if the number of abnormal cells in the postlesion are greater than the number of prelesion abnormal cells.

Another BASIC subroutine is currently being developed to perform statistical tests upon the cell data. This will provide even greater information about the effects of the lesion upon the neuronal populations of the brain. Thanks to the Altair 680b microcomputer and MITS software, our research has progressed much more rapidly and efficiently.

I'd also like to thank Dr. Feeney, whose unique insights into brain research have continually inspired and guided me throughout this project.

INTERSPIKE INTERVAL HISTOGRAM
Figure 7.

```
TIME IN
MILLISECONDS

00001 I *
00002 I *
00003 I  *
00004 I  *
00005 I     *
00006 I        *
00007 I              *
00008 I                 *
00009 I                    *
00010 I                *
00011 I               *
00012 I             *
00013 I          *
00014 I        *
00015 I     *
00016 I   *
00017 I   *
00018 I *
00019 I *
00020 I *
00021 I *
00022 I *
00023 I *
00024 I *
00025 I *
00026 I *
00027 I   *
00028 I   *
00029 I   *
00030 I *
00031 I   *
00032 I      *
00033 I        *
00034 I          *
00035 I            *
00036 I               *
00037 I                 *
00038 I                   *
00039 I                 *
00040 I                *
00041 I                *
00042 I               *
00043 I             *
00044 I           *
00045 I          *
00046 I        *
00047 I      *
00048 I    *
00049 I *
00050 I  *
00051 I *
00052 I *
00053 I  *
00054 I *
00055 I *
00056 I *
00057 I  *
00058 I *
00059 I *
00060 I  *
00061 I *
00062 I *
00063 I *
00064 I *
00065 I *
------------------------------------------------------
        0       10      20      30      40      >50

        NUMBER OF INTERVALS IN 15 SECOND TRIAL
```

WACC-II DATE CHANGED

By Charles Olsen
Director of Public Relations

WACC-II has been rescheduled for May 4-7--two weeks earlier than originally announced. These dates have been confirmed, so please plan your vacation and travel time accordingly.

Everyone at MITS is enthusiastic about this year's WACC, especially since it's going to be even bigger and better than last year's convention. We've made some important changes that will make WACC-II more informative and enjoyable for everyone.

Facilities:

We have expanded the facilities by selecting the spacious Albuquerque Convention Center for this year's WACC. Equipped with large exhibit halls, 20 meeting rooms, lounges and an auditorium that seats 2500 people, the Convention Center is an ideal place for WACC-II.

Contests:

$15,000 in hardware, software and door prizes will be awarded at the banquet. Up to 20 contestants will compete for a new CT96-CRT and other prizes in the demo contest.

Remember, the deadline for renting a booth is April 8, 1977, OR when space runs out. Look for details about booth rental on your WACC-II brochure or contact me, and I will send you an exhibitor's packet.

Seminars:

Last year we had four seminars. But this year we will have 12, with many guest speakers discussing a variety of topics, ranging from games to advanced business applications, each day of the convention.

Booths:

Altair dealers, computer magazines and clubs from all over the country as well as companies that manufacture Altair compatible peripherals, are just a few of the organizations that will have booths set up in the exhibit halls at WACC-II.

Remember, the deadline for renting a booth is April 8, 1977, OR when space runs out, so reserve your booth soon! Look for details about booth rental on your WACC-II brochure or contact me, and I will send you an exhibitor's packet.

May is one of the most beautiful months in "The Land of Enchantment." It's a perfect time to attend this year's bigger and better WACC and to enjoy the unique beauty and tri-cultural heritage of Albuquerque. I am looking forward to seeing all of you at the 1977 WACC--May 4-7.

ALTAIR COMPUTER SCORES AT GOLF TOURNAMENT

Scores Processing Program (SPP)

Large invitational tournaments usually use a shotgun start with players positioned on all Tee Boxes to begin simultaneously at the sound of a shotgun. With this type of start, players complete the course within about a half-hour. This means that the operator is confronted with a large number of score cards within a brief period of time. Players usually linger around the large, exterior Display Board until their scores are posted and verified. Since players sometimes challenge their scores, the operator must be ready with instant feedback.

This can be accomplished if the operator processes only four or five score cards in a batch. The operator notes the results, in duplicate, for each batch on a pad of special forms. The original of each entry is quickly dispatched to the exterior Display Board Monitor, and the copy is retained for later entry into the VDP. When all scores are in for the group of players just completing the course, the VDP is loaded into the Altair 8800 computer, and the new data is entered as DATA statements.

SPP Preparation

The SPP requires certain preparation at the beginning of the tournament and at the beginning of each day of play. The par value of each hole (PA($\emptyset$)-PA(17)) is entered consecutively in DATA statements 300-310, as may be required. Then, the number of teams (T) is entered, either as INPUT statements (no. 1$\emptyset$) or as a DATA statement. These variables remain constant throughout the tournament.

The variable "Q(J)" may represent either the Initial Quota or the Revised Quota assigned to each team, depending upon the day of play. On the first day it represents the Initial Quota, and these must be entered into DATA statements 4$\emptyset\emptyset$-41$\emptyset$, as may be required. These same DATA statements are then deleted on the beginning of the second day and substituted with the Revised Quotas substituted. Quotas must be entered in order of team number.

Scores are entered as DATA statements, so numbers 5$\emptyset\emptyset$-51$\emptyset$ have been reserved for this purpose. Hole scores for each team require two DATA statements. The first contains, in sequence, the team number followed by the 18 hole scores of player #1 of the team. The second contains, in sequence, the 18 hole scores of player #2 of the same team.

As the scores for each team are processed, a line of the table is printed out. When the batch has been completed, the operator will get an OUT OF DATA message, and the data can then be entered onto the forms provided for that purpose.

The SPP Listing

```
1Ø INPUT "NUMBER OF TEAMS"; T:  INPUT "DAY OF PLAY"; D:  T=T-1:

   ?:?:?.
```

The DAY OF PLAY is either "1" or "2". The T-1" is to reconcile the number of teams to a matrix which is dimensioned from $\emptyset$.

```
2Ø DIM SC(1), Q(T), PA(17)

30 FOR J=Ø TO 17: READ PA(J): NEXT J: FOR J=Ø TO T: READ Q(J):

   NEXT J

40 GOSUB 11Ø
```

This subroutine prints the heading for the table.

```
50 READ J:J=J-1: SC(Ø)=Ø: SC(1)=Ø
```

The "J=J-1" is to compensate for the matrix dimensioned from $\emptyset$. The remainder of the statement is for initialization purposes.

```
60 FOR A=Ø TO 1: FOR B=Ø to 17: READ HS: SC(A)+HS
```

Here the hole scores are accumulated to player scores for each member of the team.

```
70 GOSUB 17Ø: NEXT B,A: PS=N-Q(J): N=Ø: ON D GOTO 8Ø, 1ØØ
```

The subroutine at 17Ø calculates the points earned on each hole score. These are accumulated for each player and the team, then reduced by the amount of the quota for the team.

```
8Ø QR=Q(J) + PS/2
```

"D" will have a value of "1" when first day's scores are being calculated, and so a Revised Quota must be determined for use on the second day.

```
9Ø GOSUB 26Ø: GOTO 5Ø
```

This subroutine prints out the first day scoring data for each team as soon as it is processed, then returns to look for more data.

```
1ØØ GOSUB 27Ø: GOTO 5Ø
```

This subroutine prints out the second day scoring data for each team as soon as it is processed, then returns to look for more data.

```
11Ø ON D GOTO 12Ø, 14Ø

12Ø ? "TEAM" TAB(7)"SCORES" TAB(15)"POINTS" TAB(23)"REVISED"

13Ø ? "NO," TAB(7)"1" TAB(12)"2" TAB(24)"QUOTA": FOR F=1 TO 32:

    ? "-";: NEXT F: GOTO 16Ø
```

This subroutine (11Ø - 16Ø) provides headings for the first or second day table, as indicated by "D".

```
140 ? "TEAM" TAB(12)"SCORES" TAB(24)"POINTS"

15Ø ?"NO," TAB(12)"1" TAB(17)"2": FOR F=1 TO 32: ?"-";: NEXT F

16Ø RETURN
```

The Points Calculator Subroutine consists of statements 17Ø - 25Ø. The hole score, not a part of the listing, is set forth to the right of each statement.

```
17Ø E=HS-PA(B)+3: IF E>5 THEN 24Ø

18Ø ON E GOTO 19Ø, 2ØØ, 21Ø, 22Ø, 23Ø

19Ø N=N+6: GOTO 25Ø                    (Eagle)

2ØØ N=N+4: GOTO 25Ø                    (Birdie)

21Ø N=N+2: GOTO 25Ø                    (Par)

22Ø N=N+1: GOTO 25Ø                    (Bogie)

23Ø N=N+.5: GOTO 25Ø                   (Double
                                        Bogie)

240 N=N+Ø

25Ø RETURN
```

Statements 26Ø-27Ø are the print out subroutines, one for each day.

```
26Ø ? J+1 TAB5)SC(Ø) TAB(1Ø)SC(1) TAB(14)PS TAB(22)QR: RETURN

27Ø ? J+1 TAB(1Ø)SC(Ø) TAB(15)SC(1) TAB(23)PS:RETURN

3ØØ DATA
```

Enter consecutively the par value for each hole.

```
4ØØ DATA
```

Enter consecutively the Initial or Revised Quotas, according to day of play.

altair computer center opens in Lincoln

One of the latest Altair Computer Centers to join the steadily growing list of dealerships across the country is located in Lincoln, Nebraska.

The Altair Computer Center in Lincoln is dedicated to all facets of the micro-minicomputer market with a complete line of MITS Altair computer products in kit or assembled form, including: Altair software language packages, books, magazines, electronics tools, Vector experimenter boards and accessories and microcomputer-related integrated circuits for experimenters as well as hobbyists.

The Lincoln Computer Center will continue to provide customers with parts, service, advice and information about new products. However, in the future, about 80% of our business will center on developing assembled computer systems for specific applications or small to medium-sized businesses.

Store manager, Steve Conover, has many years of experience in the electronics field in addition to three years work in marketing and promotion. Conover will also be in charge of sales and advertising coordination.

Gene Friesen is the service director for the center. He holds a BS-EE with an extensive background in digital design and logic systems. As a former employee of an electronics firm in Lincoln, Friesen was head of the research and development team for the past two and a half years and was responsible for new product development and production design.

Gary Green is the administrator for the Altair Computer Center and will keep track of all business records.

Friesen and Conover are currently organizing educational seminars for local businessmen, computer clubs and educational institutions. These seminars are aimed at educating the general public to the ever increasing role the microcomputer is assuming in today's society. Seminars will feature specialists in computer programming, software structuring, I/O devices, applications, hardware design and interfacing.

BASIC File Handler Developed for Hams

After the computer prints the information on repeaters at amateur stations, it will print "END OF SEARCH" and then return to "STATION?" for the next request. If the requested information is not in the file, the computer will print "NO RECORD IN FILES" and go back to "STATION?"

The Program

Additions or deletions to the file are accomplished by using the Altair BASIC command "CLOAD," which brings the program into the computer from cassette. A new entry is made by simply adding a new line for each new station, repeater or even additional QSLs with a previously logged station. After the updating has been performed, the program is then put back on cassette using the "CSAVE" command.

The memory required for the listing shown takes about 1250 bytes and a typical entry takes about 45 bytes. The program could be expanded to provide readouts for summaries of QSLs by districts or foreign countries. This could be done by adding an "OR" phrase as was done in the repeater section of the program, but as mentioned at the start, this program has been kept simple. However, it is a practical operating program and is in use at my station. The listing as shown on page 2 shows only a small portion of my log files.

New Product Prices Announced

Two new products have recently been added to the Altair price list-- the Altair Universal I/O Board and the Altair 88-Process Control Interface Board. Domestic prices for these products are now in effect.

Altair 680b Universal I/O Board:

	Kit	Assembled
With no options:	$110	$160
With one serial port:	$165	$260
With two parallel ports:	$160	$235
Full board (one serial port and two parallel ports:	$215	$335

Altair 88-Process Control Interface Board

Assembled only:	$235

TEN

ALTAIR COMPUTER SCORES AT GOLF TOURNAMENT

5ØØ DATA

Use two statements for each team. The first should consecutively test the team number and then the scores for each of 18 holes for player #1. The second, the scores for each of 18 holes for player #2.

Video Display Program

The Scores Processing Program is sufficient in itself to perform the task of scoring golf tournaments where a point system is used. But many golf and country clubs use a common TV antenna network which services a number of sets located in card rooms, bars, dining rooms, etc. Channel 3, often not used by commercial or educational networks, can also be used as a closed circuit channel for displaying tournament results as they occur. Instead of using a video camera for this purpose, a faster, more simple way is to connect the antenna system to the computer system using the VDP. (See first article, p. 16, December issue of C/N.) With the VDP, displays rotate from page to page on the screen during the entire tournament, except during brief periods when the system is required for preliminary scores processing.

Displays added sufficient interest to our 1976 Boulder Country Club Frolics to encourage plans for improved displays and programs in this year's tournament. In the upcoming tournament a video camera will operate continuously from its position on the first Tee Box. The picture will be merged with that of the displays, and both will be adjusted for optimal viewing.

Another planned improvement includes the substitution of PRINT USING statements in the Displays so as to permit correct justification of tabular material.

VDP Preparation

Variables that remain constant throughout the tournament can be input as DATA statements before the start of the tournament. Such variables include the number of teams, names of players and the handicaps of each as well as the par value of each hole on the course.

The program is divided into two parts. The first reads the data and sets the stage for displays to begin. The second provides the displays. There is a break point provided between the two (statement No. 7Ø), which gives the operator time to switch on the output to the antenna network before beginning displays.

A pause is provided for each page of the displays. This pause may be adjusted to local desires by changing the value of TI in statement No. 8Ø for scoring displays and statement No. 6ØØ for bulletins.

Remaining preparation occurs while the tournament is in progress-- at noon and in the evening of each day. At these times the scores of each player and the points earned by each team are produced by the SPP. These results are entered in the VDP as they become available in each of four groupings. First day scores and points are entered in statements 8ØØ-899, and second day scores and points are entered in statements 7ØØ-799. Formats of these entries are discussed in the program listing. As soon as all DATA statements have been loaded into the program, the program should be CSAVE'd as a precautionary measure. The operator is then ready to run the VDP.

65 NEXT K, J

7Ø INPUT "SWITCH ON THE TV DEVICE, READY FOR DISPLAYS? ('Y')";B$

The TV Device is connected via antenna splitters to the TV antenna network in the clubhouse. The arrangement is switched so as to keep background preparation "off the air."

8Ø GOSUB 1Ø1Ø: TI=3ØØØ: ON D GOTO 9Ø, 34Ø, 47Ø

Statements 9Ø through 21Ø provide a paged display of Final Scores. Many of the techniques employed are repeated in subsequent displays, but they will be discussed only as they appear in this one.

90 GOSUB 1030

> Each Display is initiated by a "GOSUB 1030." This provides a "front page" which is fully titled. By way of contrast, the sub-displays which follow include only the titling which is essential to an understanding of the table of information presented.

100 GOSUB 110: GOTO 150

> Statements Nos. 110-140 provide titling for the sub-displays. These are cycled, in this case, for each six lines of data (see statement No. 180). This represents the maximum lines which can be presented at one time. Note that readability is enhanced by spacing a line between each set of three lines. (See statement No. 170).

110 ?TAB(2)"*" TAB(10)"FINAL SCORES" TAB(29)"*"

120 GOSUB 1020: ?"TEAM" TAB(14)"P O I N T S"

130 ?TAB(7)"1ST DAY" TAB(16)"2ND DAY" TAB(25)"TOTAL"

140 GOSUB 1020: RETURN

150 FOR J=0 TO T: IF S2(J,0)=0 THEN 200

> This flag permits skipping over the scores of players who have not yet finished their round, e. g., only about half the scores come in at noon each day, and these are the only scores reported in the Displays.

160 ?J+1 TAB(7)P1(J) TAB(16)P2(J) TAB(24)P1(J)+P2(J)

> The "J+L" reconciles the matrix index to Team Number.

170 IF INT((J+1)/3)=(J+L)/3 THEN ?

175 IF J=T THEN 200 ELSE IF INT((J+1)/6)<>(J+1)/6 THEN 200

> The "IF J=T" avoids a needless pause and page at the end of the Display.

The VDP Listing

10 INPUT "NUMBER OF TEAMS;" T: INPUT "PERIOD OF PLAY;" D:

> T=T-1: D=3-D
>
> It is critical that the correct period of play be entered since this entry synchronizes the sequence of READ statements with the kinds of DATA statements entered. The following flags are used:
>
> "0" - No first day scores have been entered as DATA statements.
>
> "1" - At least some first day scores have been entered, and no second day scores have been entered.
>
> "2" - At least some second day scores have been entered.
>
> The "T - 1" is used to dimension the matrices from 0.
>
> The "D=3-D" is used to sequence READ statements (see No. 30), and the displays (see No. 80).

20 DIM G$(T,1), H(T,1), S1(T,1), S2(T,1), P1(T), P2(T)

30 TAB(8) "* NOW WORKING *": ON D GOTO 40, 50, 60

> The "Now Working" statement can be deleted, but without it, the time required to read all of the files can cause alarm to the operator.

40 READ J: IF J=0 THEN 50 ELSE J=J-1: READ S2(J,0),S2(J,1), P2(J):
   GOTO 40

New Club Planned

Employees from the Thatcher Glass Manufacturing Company in Elmira, New York, are interested in forming a computer club in their area. Individuals interested in becoming a member should contact:

Samual Leonard
Thatcher Glass Manufacturing Co.
1901 Grand Central Ave.
Elmira, NY 14902

Article Corrected

Error Noted in S4K Memory Board

The transistor (Q1) silkscreen is incorrect on the S4K Memory Board (Rev. 1). The "E" should not be on the left transistor pad. Instead, the right transistor should be labelled "E," as shown in Figure 1.
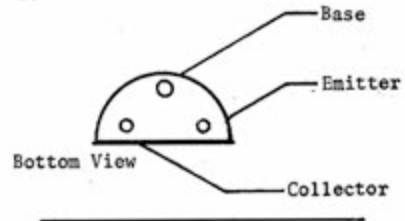
Correction to Dial Article

The following change should be noted in "Data Recording With the Altair," by Dr. O. E. Dial (November, pp. 13-15.) In section 2 (Recording Data) statement 30 should read, Wait 6, 128, 128.

2450 Alamo S.E.
Albuquerque, New Mexico
87106

classified ads

BASIC Goes Bach
With the Altair Polka

By Doug Jones

The Paper Tape Reader Control (PTRC) signal from the Altair 680b may not be terribly exciting in itself, but it does offer a simple way for even a minimal computer system to control events in the real world. For instance, under software control the PTRC can be turned on and off at rates of anywhere from once a week to tens of thousands of times per second. This facility enables the Altair 680b to make music, and the procedure for turning your 680b into a "music box" is a relatively simple one.

Hardware

The Input/Output section of the Altair 680b uses a Motorola MC6850 Asynchronous Communications Interface Adaptor(ACIA) integrated circuit. One of the output signals of the ACIA is Request-to-Send, which is used on the Altair 680b as a PTRC. This signal is under software control and can be turned on by sending D1 hex to location F000 or turned off by sending B1.

Under software control then, the PTRC can produce 5 volt square waves. The frequency of the waves is determined by the rate at which the signal is turned on and off. By connecting an amplifier and speaker between the PTRC output and ground, the waves can be made audible, and the Altair 680b becomes a musical instrument(of sorts).

Software

The software used to generate music consists of two parts: an assembly language package that generates the notes themselves and a BASIC program to compile the notes into musical compositions.

Altair BASIC is ideally suited for this application since it includes not only powerful string manipulation features but also the USR function, which allows a BASIC program to call and control assembly language subroutines. In addition, the PEEK and POKE features of Altair BASIC permit a BASIC program to access and modify any location in memory.

The assembly language tone and duration routines are simple ones. The length of time the note is to be sustained is POKEd into the address labelled SUST. A loop decrements the value of SUST until it equals zero, and then the routine returns to the BASIC program.

The frequency of the musical tone is stored at label NOTE. This value controls the rate at which the PTRC is turned on and off by means of a WAIT loop.

The relationship between the frequencies of the notes in a scale is a complicated one, governed by the psychology of hearing more than the regularities of mathematics. A few minutes of research in a college physics text revealed the fact that the ratios between the adjacent notes in a C-major diatonic scale are either 8/9, 9/10 or 15/16. These relationships would be sufficient for our purposes if we only wanted to play tunes in C-major and never wanted to play anything with sharps and flats.

The ratios of sharps and flats to their base notes become even more complicated if we want to maintain only diatonic scales. One way around this problem is to adopt the well-tempered scale used by J.S. Bach. In a well-tempered scale, the ratios of the frequencies of any note and its sharp or flat is the twelfth root of 2(there are twelve half-tones in an octave, and the ratio of the frequencies of two notes an octave apart is one-half.) So, the frequency of A# is the frequency of A times the twelfth root of two. The frequency of Ab is A times one, divided by the twelfth root of two. These two constants are known as K4 and K5 in the program.

The BASIC program has three parts, which correspond to the three modes of operation: Random, Play and Compose. When the program starts, it asks which mode is desired. Typing R, P or C in response, selects the desired part of the program.

The Random mode causes the computer to jump to a routine using Altair BASIC's random number generator to produce values for NOTE and SUST. If the numbers are within the required range, they are converted to musical notes and sent out as a strange series of beeps and boops that will enthrall even the most avid music connoiseur.

Now that the intervals have been defined, it is necessary to set up a frequency standard. Once the frequency of a single note is defined, it is a relatively simple matter to calibrate all of the notes so that they correspond with the same notes on all other musical instruments.

Unfortunately, my short wave receiver was sold from the ham shack a month before I began this project, so I couldn't listen to the frequency standard on WWV. The precision audio oscillator on my work bench wasn't any help either, because the "hair line" on the frequency dial looked about 2½ inches wide. Without even a tuning fork or pitch pipe in the neighborhood, I was stumped for finding any kind of a frequency standard.

But in a sudden flash of inspiration, I came up with the following standard:

Middle C shall be that musical note struck by a bunny with a hammer in his right paw on a xylophone pull-toy with four wooden wheels and a short string.

It may not be too precise a definition, but it's entirely adequate for my purposes.

Play Mode

In Play Mode the program reads the notes and durations from data entered into the computer and then "plays" them. The musical score is entered as DATA statements, starting at line 100. For example, the following data plays the C major scale.

    100 DATA C,D,E,F,G,A,B,C',$$

The program is designed to play a six octave range. The octave of a note is specified by following the note with one of the following designators:

!!! Three octaves below middle C
!! Two octaves below middle C
! The octave below middle C
no punctuation The octave of middle C
' One octave above middle C
'' Two octaves above middle C
''' Three octaves above middle C

Thus, A in the octave below middle C appears in the program as A! (Note that '' is two apostrophes and not one quotation mark.) Sharps and flats are denoted by writing a > or a <, respectively, after each note. The sharp and flat signs are placed between the note and the octave multiplier.

Continued

Bach

The duration of a note is specified by an integer placed before the note. The integer must be between one and nine. The default value is 1. The duration values are such that 5B! is held five times longer than B! This constant allows whole notes, half notes, quarter notes, eighth notes and dotted notes to be played.

Here is an example of how a tune looks in the program's notation.

```
100 DATA 2C,2C,2G,2G,A,A,A,A,4G
101 DATA 2F,2F,2E,2E,2D,2D,8C,$$
```

This melody is, no doubt, a commentary on the social, political and ethical background of the anonymous author's time. It has come down to us, however, as something about sheep.

The program uses other operators for special purposes. $$ or END indicates the end of the DATA lists. It tells the program when to stop READing and thereby avoids OD (Out of Data) errors. $ designates a "repeat" just as the double dots do in a musical score. To have "Yes, sir, yes, sir three bags full" play twice, type:

```
101 DATA $,2F,2F,2E,2E,2D,2D,8C,$$
```

R is a rest. It can be weighted in the same way as notes to generate rests of any duration. Be careful, 9R is more of a coffee break than a rest! ↑ is the "cascade." This is used to good effect in such songs as the lovely "Skaters' Waltz." % is the spratz. The spratz operator must have a number following it. The number that follows the spratz is the ASCII code of some character. When the spratz is played, the character is typed on the terminal. The following "spratz" example punctuates the playing of "Jingle Bells" with bells from the terminal.

```
100 DATA A,A,2A,%7,A,A,2A,%7,A,C',F,
G,4A,%7,2R,%7,$$
```

It's a sort of ensemble. On a model 35 Teletype™, %11 produces a crescendo of paper feeds. A vertical tab character is a great down-beat.

Spratzing is really serious musical business. The percussion effects of Sproings, clicks, thunks and braaaaps are unparalleled in any other musical instrument. Perhaps the most famous example of spratzing is Leroy Anderson's, "The Typewriter Song." The Boston Pops spratzes so well in their rendition, that it's easy to imagine the entire orchestra is using real typewriters.

ALTAIR COMPUTER SCORES AT GOLF TOURNAMENT

A flag, "∅," must be entered as the last "Team Number" for scores on both day-1 and day-2 (when scores have been entered for those days), to permit an exit from the READ loops.

Note that the READ statements loop back to themselves until a "∅" is read for team number.

The "J=J-1" is merely to reconcile "Team Number" to the number represented by it in the matrix.

```
5∅ READ J: IF J=∅ THEN 5∅ ELSE J=J-1: READ S1(J,∅),S1(J,1),P1(J): GOTO 5∅

6∅ FOR J=∅ TO T: FOR K=∅ TO 1: READ G$(J,K), H(J,K): IF H(J,K) >24 THEN
   H(J,K)=24
```

A convention observed in the tournament is that no handicap can be credited in excess of 24 toward the calculation of the Initial Quota.

```
18∅ GOSUB 1∅∅∅
```

This GOSUB provides a sufficient pause for the page to be read, then clears the page before the next following is presented.

```
19∅ GOSUB 11∅

2∅∅ NEXT J

21∅ GOSUB 1∅∅∅
```

This assures that the last page of the Display is provided with a pause before continuing to the next.

Statements 22∅-33∅ provide the "Second Day Scores" Display.

```
22∅ GOSUB 1∅3∅

23∅ GOSUB 24∅: GOTO 27∅

24∅ ?TAB(2)"*" TAB(7)"SECOND DAY SCORES" TAB(29)"*"

25∅ GOSUB 1∅2∅:?"TEAM" TAB(6)"PLAYER" TAB(15)"SCORES" TAB((23) "POINTS"

26∅ GOSUB 1∅2∅:RETURN

27∅ FOR J=∅ TO T: IF S2(J,∅)=∅ THEN 32∅ ELSE ? J+1 TAB(7)"1"
    TAB(15)S2(J,∅)

28∅ ? TAB(7)"2" TAB(15)S2(J,1): ? TAB(23)P2(J): ?

29∅ IF J=T THEN 32∅ ELSE IF INT((J+1)/2)<>(J+1)/2 THEN 32∅

3∅∅ GOSUB 1∅∅∅

31∅ GOSUB 24∅

32∅ NEXT J

33∅ GOSUB 1∅∅∅
```

Statements 34∅-46∅ provide the "First Day Scores" Display.

```
34∅ GOSUB 1∅3∅

35∅ GOSUB 36∅: GOTO 4∅∅

36∅ ? TAB(2)"*" TAB(8)"FIRST DAY SCORES" TAB(29)"*"

37∅ GOSUB 1∅2∅: ? "TEAM PLAY - SCORE POINTS REVISED"

38∅ ? TAB(6)"ER" TAB(25)"QUOTA"

39∅ GOSUB 1∅2∅: RETURN

4∅∅ FOR J=∅ TO T: IF S1(J,∅)=∅ THEN 45∅ ELSE ? J+1 TAB(6)"1" TAB(11)S1(J,∅)

41∅ ? TAB(6)"2" TAB(11)S1(J,1)
```

Albuquerque Becoming Personal Computing Center

By Nels Winkless

This story first appeared in the January 7 issue of THE NEW MEXICO INDEPENDENT (p. 13) as "Albuquerque Becoming a Center of Personal Computing Concepts." Nels Winkless is the editor of PERSONAL COMPUTING magazine.

INDUSTRY gossips are buzzing all over the country about the extraordinary surge of activity in Albuquerque that has made this smallish city the virtual capital of personal computing. Four significant institutions dealing with personal computing have been established here and that does cause talk. Personal what?

The Independent has commented before on an amazing new phenomenon loose in the land--the ability of private citizens to go into retail computer stores and buy, over-the-counter, small computer systems of remarkable power, complete with tape and disc memories, printers, video displays and other peripherals that make them as powerful as most major scientific and business systems were just a few years ago. The private citizen who doesn't ever want to know what's inside the magic box can now use computers as working tools to accomplish a great many personal data handling tasks.

Struggling Company

Well, it all started here when a struggling little Albuquerque company, MITS, Inc., got national publicity for its new computer kit that almost anybody could afford.

That was two years ago--January of 1975--and MITS sold 10 times as many computers in that year as hoped--over 5000 units. The wild growth was an agony as well as a blessing to the company, but the firm hung on to become established as the leader in its field--not only first chronologically but biggest in sales and tops in quality and the breadth of its product line. This caused more than a little confusion among eastern business folk who still occasionally inquire about customs regulations affecting their dealings with New Mexico.

Recently, MITS announced its plans to merge with Pertec Computer Corporation of California, becoming a division of that much larger firm. The merger will bring financial muscle and manufacturing expertise to the local operations, building up the enterprise here in a big way. If everything goes through,

Albuquerque will see a wholesome boost to its home-grown economy.

Not only that, when Pete Connor opened his Computer Shack retail outlet here, he was running an experiment of enormous interest to many people all over the country. Could a retail computer store prosper in an isolated community of modest size like ours? There's Pete, plugging away happily, selling systems (even to Europe, we are told) and building a real business. That's not bad in a few months, though he has yet to report the sale of his 40th computer, a landmark he has promised to share with Independent readers. The Computer Shack is used as an example to show that the age of personal computing has really arrived.

Third, Albuquerque is now the home of an organization called The United States Robotics Society, formed by a handful of people who feel that robots are inevitable, for good or for ill. The common view has been that robots can be developed only in huge business or government laboratories, but in fact, a hobbyist now has enough technology at his command to experiment with enormously complex robotics systems. All he needs is a good idea, and that is not often derived from either big business or big government.

USRS literature says: " . . . Society is established 'for good'-- for the good of mankind--not in opposition, for opposition is idle, and not in advocacy, for advocacy is unnecessary . . ." Society president Glenn Norris comments: "We had no idea whether or not we'd develop membership large enough to support a real learned society and pay for the cost of publishing and mailing newsletters, establishing a useful library, and so on. This is a somewhat specialized interest, and there is serious opposition to robotics by people who find the subject frightening and would like to control it legally. We feel that practical control is impossible, of course, and that there is an urgent need to learn how to deal with robots sensibly--those developed in our own society or by someone else. We didn't know whether we were just a small group of people with an odd idea or others felt as strongly as we did."

The group organized formally in the summer of 1976 and tested interest with an advertisement in a national magazine late in November. Somewhat to their surprise, memberships at $12 per year began to

trickle into their Albuquerque Post Office Box at a low but steady rate. Stranger still, the flow is continuing and USRS now has members in all parts of the United States, even a couple in Europe. The group doesn't ever expect to become very large, but its influence may be great and the appearance of this third personal-computing-related institution in Albuquerque has caused more than a trifle of talk.

New Magazine

Last on his list is a magazine called, appropriately enough, Personal Computing. It headquarters in Albuquerque and reaches an international readership. You've never heard of it? Well, drop in at the Computer Shack, Plaza Books (in Coronado) or Electronic Parts (Rhode Island, NE) to look at copies of the Premier Issue, January/February, 1977.

This magazine, too, is an experiment. The story goes like this: Until recently, a chap by the name of David Bunnell was advertising and marketing manager for MITS and was responsible in part for sustaining the great personal computer sales surge. Salesmen frequently called upon Bunnell, hoping to persuade him to purchase space in various publications, including three that were launched early in the game to deal directly with a readership interested in personal computing. However, none of those new journals was named personal computing--a natural, Bunnell thought.

Later, he discussed the idea with folks at the Benwill Publishing Corporation of Brookline, Massachusetts. They said that they'd be happy to apply their hard-earned cash and reputation as a publisher of respected trade magazines to Bunnell's plans for launching a magazine called Personal Computing.

Bunnell's theory behind the project was that there are still more people who don't own computers than do. But a well-done consumer magazine that appeals to beginners as well as the limited number of technically-minded computer hobbyists might do very well.

So Bunnell took the plunge. He hired me, a former writer for The Independent, as editor/writer for the new magazine. Together, we designed the look and the language of Personal Computing before

Albuquerque Becoming Personal Computing Center

Continued

introducing it to the national circulation. Its unique style, with many illustrations and easy-to-understand copy treatment of computers, was specifically designed for non-computer people to enjoy.

So far, so good. Thirty thousand copies of the first issue are going into distribution (to rave notices, if I may say so myself), and the magazine is already recognized as a force of some importance in the personal computing community. Printed in Minneapolis, mechanically assembled, sold out of Boston, and written and edited in Albuquerque, there's still much cause for gossip about Personal Computing.

But MITS, USRS and Personal Computing are just the beginning. It will be interesting to see what computing institutions are formed in Albuquerque in the next few years. Odd, but in the middle of nowhere.

altair computer center re-opens

Customers will find modern decor and a new manager at the recently re-opened Altair Computer Store of New York.

The store was closed for three months for complete remodeling and now offers customers not only a newly painted, carpeted and shelved interior, but a more extensive line of Altair computers and equipment. Three separate demo areas displaying an Altair 8800a computer with a disk, an 8800a with a cassette and an Altair 680b computer have been set up in the store so that customers can easily try out different Altair computer systems.

The new store manager, Dan Goodman, holds a degree in electrical engineering from Bucknell University and also has 15 years experience in the data processing field. Before Goodman assumed his managerial duties at the New York store, he was manager of the computer center at BASF Wyndotte--the American subsidiary of the largest chemical manufacturer in the world, located in Germany.

Bach

Continued

Compose Mode

In the Compose Mode, each note is heard as it is entered. The notes are also held in a buffer for later recall. When the sequence of notes is satisfactory, typing .D (for Dump) prepares a complete DATA statement that will play the sequence in Play Mode. The .D command generates all leaders, nul, line numbers and Control/0's necessary to punch the sequence on paper tape. In this way masterpieces can be saved for posterity or just until the next week.

Example

```
RUN
(P)LAY  (C)OMPOSE OR (R)ANDOM? C         (C)ompose is chosen.
W=WRITE L=LIST P=PLAY .D=DUMP             Here are (C)ompose
WRITE . . .                              options available at
? 2C                                     any time.
? 2C
? 2G
? 2G
? 2A
? -1                                     Delete the last note.
#NOTES NOW 4
? A
? A
? A
? A
? 4F
? -1                                     Oops!
#NOTES NOW 8
? 4G
? 2F
? 2F
? 2F
? -1
#NOTES NOW 11
? 2E
? 2E
? 2D
? 2D
? 8C
? L                                      List what we have so far.
2C, 2C, @2G, 2G, A, A, A, A, 4G, 2F
2F, 2E, 2E, 2D, 2D, 8C,
#NOTES 16
? P                                      (P)lay what we have so far.
? -7                                     Delete 7 notes.
#NOTES NOW 9
? $                                      Insert repeat sign.
? 2F
? 2F
? 2E
? 2E
? 2D
? 2D
? 8C
? P
? .D                                     (.D)ump.  Be sure to
                                         turn on tape-punch.
100 DATA 2C,2C,2G,2G,A,A,A,A,4G,$
101 DATA 2F,2F,2E,2E,2D,2D,8C,$$

RUN                                      Still part of tape.

WRITE . . .                             Now back to top.
?
```

THE SKATERS' WALTZ and OTHER FAVORITES

A W (for Write) command is provided to clear the buffer of all previously composed notes. The following is an example of how to use the program in the Compose Mode.

Conclusion

I hope this program will provide hours of fun. Someday, perhaps, a professional musician, lyricist and a programmer will get together to produce some really beautiful songs for the Altair 680b and Paper Tape Reader Control.

Q & A from the Repair Department

Information from the two input signals, CTS and DCD, is present in the ACIA status register. Bit 2 represents $\overline{DCD}$, and bit 3 represents $\overline{CTS}$. When bit 2 is high, DCD is inactive. When bit 3 is high, CTS is inactive. This permits software to detect the direction of data flow. When status bit 2 goes low, valid data is sent to the ACIA by a modem. When bit 3 goes low, data can be transmitted to the modem.

The 2SIO receive, transmit and ground are interfaced with a terminal in the same way as any other I/O card. The 2SIO transmit connects to the terminal's receive, the 2SIO's receive connects to the terminal's transmit and the 2SIO's ground connects to the terminal's ground. It's a rather complicated process to connect CTS with DCD and RTS. First of all, CTS, DCD and RTS are active high at the terminal's 25-pin connector, but they are active low on the ACIA's pins. From here on, it's necessary to study the terminal instruction booklet in order to use CTS, DCD or RTS.

There is a simpler way to interface a terminal to the 2SIO if the user doesn't care to use CTS, DCD or RTS. Simply carry only three wires to the terminal—receive, transmit and ground. The handshaking input signals are then tied active permanently. For RS-232 levels on the 2SIO, DCD and CTS must be made active high by connecting them to +5v at the 10-pin Molex connector. For TTY or TTL levels, it isn't necessary to tie these active high, because the circuitry will make them active to the ACIA. If DCD and CTS were not wired from the ACIA to the 10 Molex pins, simply connect the appropriate D1, D2 or E1, E2 pads to ground. Similarly, tie the terminal handshaking input signals (DCD and CTS) active at the 25-pin connector.

This also applies to any other control input signals on the terminal such as Data Set Ready. This is probably the easiest way to interface the 2SIO with a terminal if no modem is involved.

"Glitches" offers a perfect opportunity to have your hardware questions quickly and thoroughly answered. But the column depends upon your input. So take a few minutes to jot down a question and send it to:

"Glitches"
Computer Notes
MITS, Inc.
2450 Alamo SE
Albuquerque, NM 87106

ALTAIR COMPUTER SCORES AT GOLF TOURNAMENT

```
415 ? TAB(17)P1(J) TAB(24)72-H(J,Ø)-H(J,1)+P1(J)/2

42Ø IF J=T THEN 45Ø ELSE IF INT((J+1)/2)<>(J+1)/2 THEN 45Ø

43Ø GOSUB1ØØØ

44Ø GOSUB 36Ø

45Ø NEXT J

46Ø GOSUB 1ØØØ
```

Statements 47Ø-59Ø provide the "Team List" Display.

```
47Ø GOSUB 1Ø3Ø

48Ø GOSUB 49Ø: GOTO 53Ø

49Ø ? TAB(2)"*" TAB(11)"TEAM LIST" TAB(29)"*"

5ØØ GOSUB 1Ø2Ø: ? "TEAM" TAB(8)"PLAYERS" TAB(17)"HANDI-" TAB(24)
    "INITIAL"

51Ø ? TAB(1)"NO," TAB(18)"CAPS" TAB(25)"QUOTA"

52Ø GOSUB 1Ø2Ø:RETURN

53Ø FOR J=Ø TO T: ? J+1 TAB(6)G$(J,Ø) TAB(18)H(J,Ø)

54Ø ? TAB(6)G$(J,1) TAB(18)H(J,1): ? TAB(27)72-H(J,Ø)-H(J,1)

55Ø IF J=T THEN 58Ø ELSE IF INT((J+1)/2)<>(J+1)/2 THEN 58Ø

56Ø GOSUB 1ØØØ

57Ø GOSUB 49Ø

58Ø NEXT J

59Ø GOSUB 1ØØØ

6ØØ GOSUB 1Ø3Ø: TI=6ØØØ: ? TAB(2)"*" TAB(11)"BULLETINS" TAB(29)"*"
```

Note that the length of the pause (TI=6ØØØ) has been doubled to permit reading bulletins.

```
61Ø GOSUB 1Ø2Ø

62Ø ?" "
```

Bulletins should be entered as PRINT statements (Nos. 62Ø-629) with no more than 31 characters per line (this latter is to prevent skipping a line in the display). Should it be required to enter a new bulletin during a run, be sure to CSAVE the existing program and DATA statements, e.g., on cassette tape.

```
63Ø GOSUB1ØØØ;GOTO 8Ø
```

Note that with the completion of the Displays selected by the value of "D," the program loops back to Statement 8Ø, where it obtains a new value for the length of pause, then cycles back through the selected Displays.

```
7ØØ DATA
```

About noon on the second day, the scores of half the teams are processed, then entered as DATA statements in the VDP. Statements 7ØØ-799 are reserved for this purpose. The following format must be strictly adhered to:

<u>Team Numbers</u>

```
7ØØ  DATA 1,0,0,0,2,78,75,-4.5,3,0,0,0,[etc.]
```

PRINT USING Applications Increase

By O. E. Dial

As microcomputers are increasingly employed in professional environments, greater demands will be placed on the software to provide esthetically pleasing application outputs. Typically, these applications include tables of data, the completion of forms and check-writing.

The novice computer users focus their interest on the data itself, while the professional DP gives equal attention to the attractiveness of the package in which the data is delivered. The DP is also concerned with proper captions, equal spacing, underscoring, right justification of numerical tabular data, left justification of alpha tabular data as well as the functional or esthetic use of dollar signs, stars and plus or minus signs.

The software support for providing these features exists with the PRINT USING statement in the MITS Extended BASIC interpreter. However, the discussion of the command (pages 47-52 of the MITS BASIC Reference Manual) provides only examples which rely on INPUT statements of a single variable. Although the discussion and examples are generally very thorough, they may nonetheless lead a programmer to conclude that PRINT USING is dependent upon INPUT statements and that it is limited to single-field numeric strings.

The fact is that the potential use of PRINT USING is far more broad and rich than may be apparent at first reading. It may provide for multiple fields of both alphas and numerics but with one limitation--

the alphas must be contained within the quoted string.

The following examples illustrate this potential for multiple fields. READ statements are used in each example.

Multiple Numeric Fields

The first problem is solved when the programmer realizes he can have more than one numeric field in a string, provided each is separated by one or more spaces. For example, to output a table of right justified financial data print:

```
100 A$=" $$###.## $$###.## $$###.##"
```

```
110 PRINT USING A$;A;B;C
```

Using arbitrarily assigned variables, this produces:

```
$12.34 $123.45 $0.60
```

Multiple Alpha/Numeric Fields

The programmer may also combine alphas and other symbols with his numerics, provided they are contained within the quoted string. For example:

```
100 A$="THE TOTALS ARE:  (1) ##.##,
       (2) ##.##, AND (3) ##.##"
```

```
110 PRINT USING A$;A;B;C
```

Note that the interior commas, parentheses, numerics, alphas and the colon will be reproduced in the output. Supplying arbitrary values for the variables produced the following output:

```
     THE TOTALS ARE:  (1) 12.34,
(2) 23.32, AND (3) 1.10
```

PRINT USING in FOR-NEXT Loops

PRINT USING may be an interior statement to a FOR-NEXT loop for outputting the contents of arrays. For example:

```
100 A$="##.## ##.## ###"
```

```
110 FOR J= 1 TO N
```

```
120 PRINT USING A$;A(J); B(J); C(J)
```

```
130 NEXT J
```

Printing the Index in the Loop

The index of the loop or some calculation based upon the index may be contained in the PRINT USING statement. For example:

```
100 A$="#### #### ####"
```

```
110 FOR J=1 TO N
```

```
120 PRINT USING A$; (J+1)*100; A(J);
    B(J)
```

```
130 NEXT J
```

ALTAIR COMPUTER SCORES AT GOLF TOURNAMENT

Scores   Points
(Enter as zero's if unreported)

799 0   The End-of-File Flag.

Important: The flag should not be placed in this file until at least some scores are in.

800 DATA

Exactly as above, except that the data relates to the 1st day of play and statements 800-899 are used.

900 DATA

Statements 900-999 are reserved for team data. The following format should be strictly adhered to:

900 DATA BROWN,20,JONES,30,H Smith,10,R SMITH,12, [etc.]

Player #1, Team #1,   Player #2, Team #1,
and his handicap.   and his handicap.

Note that teams should be entered in order of player number (1-2) and team number (1-T).

1000 FOR L=1 TO TI: NEXT L

1010 FOR L=1 TO 14: ? :NEXT L: RETURN

This is the Pause, tied to a "page eraser".

1020 FOR L=1 TO 32:?"-";: NEXT L: RETURN

This subroutine provides underscoring.

1030 ? "* * *" TAB(10)"1976 FROLICS" TAB(27) "* * *"

1040 ? " * *" TAB(6)"BOULDER COUNTRY CLUB" TAB(28) " * *" : ? : RETURN

This subroutine provides the "Front Page" printed before each Display.

PRINT USING
Applications Increase
Continued

But, remember that the results of any calculations will automatically be rounded to an integer when the results are reported in a PRINT USING statement. The same holds true for interior variables. For example: PRINT USING A$; (J+1)*100; A(J); A(J)+B(J) / A(J)

String Matrices in Conjunction with PRINT USING

PRINT USING will not accomodate a string array, but this limitation can be avoided by providing a normal PRINT statement followed by a semi-colon and colon immediately before the PRINT USING statement. The quoted string used in PRINT USING will begin immediately after the PRINT statement output. Let's suppose a file containing a description of each person identified in the file is to be printed. It might include the following:

A$(J)=Name of Person
B$(J)=Color of Eyes
C$(J)=Color of Hair
B(J)=Height in Inches
C(J)=Weight in Pounds
D(J)=Age

```
100 D$="### ### ##"

110 FOR J = 0 TO N

120 PRINT A$(J) TAB(B0)B$(J) TAB(15)
    C$(J);

130 PRINT USING D$; B(J);C(J);D(J)

140 NEXT J
```

Note that this technique permits left justification of alpha columns and right justification of numeric columns.

In conclusion, once the wider applications of PRINT USING are understood, it can provide all the capability the programmer needs to produce esthetically pleasing documents without sacrificing any of the convenience of the conventional PRINT statement.

See "BASIC goes BACH" page 12

THE SKATERS' WALTZ

Soft Solo   **Arranged by DLJ**
Medium-fast Waltz

```
100 DATA 9A!,9C,4D,4D,',9A>!,9D,E,9E,',9G
101 DATA 9F,2C,9C,9A>!,9A!,9G!,F!,9F!,F',G',3F'
102 DATA G',G',3F',G',G',5F',A>',A>',9A>',F',G'
103 DATA 3F',G',G',3F',G',G',3F',B',B',9B',4
104 DATA 9A!!!,9C!!,4D!!,9D!!,',9A>!!!,9D!!,E!!,9E!!,9G!!,9F!!
105 DATA 2C!!,9C!!,9A>!!!,9A!!!,9A!!!,9G!!!,3F!!!,9F!!!,',9E,3B!
106 DATA 9C,9E,9A!,9F,3C>,9D,9F,9B!,9G,3D>,3E
107 DATA 5G,9C,3D,3C>,5D,5G,9C,',9A!!,9C!,4D!
108 DATA 4D!,',9A>!!,9D!,E!,9E!,',9G!,9F!,2C!,9C!
109 DATA 94>!!,9A!!,9G!!,F!!,9E>!!,5$
```

RUN

OTHER FAVORITES

GUSTO   **Unknown**

```
100 DATA D,D>,E,2C',E,2C',2E,6C',D',D>'
101 DATA E',C',2D',E',C',2D',6C',D,D>,E
102 DATA 2C',E,2C',2E,6C',A,G,F>,A,2C'
103 DATA E',D',2C',6D',D,D>,E,2C',E,2C'
104 DATA 2E,6C',D',D>',E',C',2D',E',B,2D'
105 DATA 6C',C',D',E',C',2D',2E',D',C',E'
106 DATA C',2D',2E',D',C',E',C',2D',E',B
107 DATA 2D',2C',2G,4C',8R,8R,2G,2G,4G
108 DATA 2A,2G,2G,2G,2C',2D',8E',4D',4D',4G
109 DATA 2B,2C',4D',2E',2D',9C',2C',2B,4A,2A
110 DATA 2B,4C',2B,2A,4G,4C',4G,2B,2A,2G
111 DATA 2G,4G,2G,2B,2D',9C',4G!,4F!,4E!,4E!
112 DATA 4E!,4E!,4G,2A,2G,4G,2C',2D',8E',4D'
113 DATA 4D',4G,2B,2C',4D',2E',2D',9C',2C',2B
114 DATA 4A,2A,2B,4C',2B,2A,4G,4C',4G,2B
115 DATA 2A,2G,2G,4G,4G,2A,2B,8C',8R,8R
116 DATA 8R,G!,G!,2G!,G!,G!,2G!,G!,G!,2C
117 DATA 2D,2E,G!,G!,2G!,G!,G!,2G!,G!,2E,2D
118 DATA 2B!,2G!,G!,G!,2G!,G!,G!,2G!,G!,G!
119 DATA 2C,2D,2E,C,E,2G,G,F,E,D
120 DATA 2C,2E,2C,G!,G!,2C,2E,2C,E,E
121 DATA E,E,E,2E,E,2A,2E,2A,2E,2A
122 DATA 2E,2D,2B!,2A!,E,E,2E,E,E,2E
123 DATA E,E,2E,2A,2E,2A,2E,2A,2G,2F>
124 DATA 2G,2F>,2G,C,E,2G,G,F,E,D
125 DATA 2C,2E,5C,8R,8R,8R,G,3G,G,A
126 DATA G,F>,F,2E,G,G,3G,G,3G,G
127 DATA A,G,F>,G,2D,G,G,G,F,E
128 DATA D,2C,2E,2G,2C',D',C',B,C',3A
129 DATA C',3B,B,B,F,A,B,8C',8R,8R
130 DATA 8R,4A,6A,2G,2G,2F,8C,2C,2D,2E
131 DATA 2F,4G,4C',9F,8R,8R,8R,2G,2E,4G
132 DATA G!!!,2G,2E,4G,G!!!,2A,2G,2F,2E,2D
133 DATA 2E,2F,E,F,2G,2C,C,C,2C,2C!!!
134 DATA C,D,E,F,4G,2G,2D,2D,D!!!,2F
135 DATA 2E,2D,9D!!!,8C,5$
```

RUN   Continued

"Some set-up, wouldn't you say?"

EIGHTEEN

CN/January-February 1977

88-PCI Offers Unlimited Potential

We introduced the Altair 88-Process Control Interface board in the October issue of C/N (see pp. 6-9), and we're now happy to announce that the new similarly designed 680b-Process Control Interface board is also available.

In the same manner as the 88-PCI interfaces various devices with the Altair 8800b computer, the 680b-PCI enables the Altair 680b computer to communicate with the real world of relays, switches, motors, fans, contacters, alarms, solenoids, lights, heaters and many other electromechanical devices. Both the 680b-PCI and 88-PCI boards have virtually unlimited applications potential and can be used in almost any instance where the computer must control large amounts of power.

The 680b-PCI and the 88-PCI boards each have eight relay outputs featuring single pole single throw (SPST) operation which are capable of switching 1 amp at 120 VAC. Both boards also have eight optically isolated inputs, which can be configured by the user to accept a wide range of input signals. The relays can directly control up to 120 watts, but with external contacts and solenoids added, the amount of power that can be controlled is essentially unlimited. Two pairs of optically-isolated "handshake" lines for control of and communication with external devices are also included. All lines are isolated and balanced for operation in electrically noisy environments.

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

Both the 88-PCI and the 680b-PCI boards are quite simple to use. The respective manuals contain extensive instructions for board set-up and initialization for adapting the board to the user's specific application. Once the board is set up and initialized, all that's necessary to control the relays is to output an eight bit word to the relay control channel. The lines set to a logic "1" by this method will energize their respective relays, and those set to "0" will de-energize. An input from the opto-isolator input channel reads the data from the opto-isolated input lines.

There are many household uses for both of the Process Control Interface boards, such as: lawn sprinkler control, turning lights on or off or even acting as a combination alarm clock and cook that will start fixing breakfast while you catch a few extra minutes of sleep.

Thanks to the PCI boards, the Altair computer can now also be used in your home or business as an alarm system that can act discriminately in dangerous situations. When a break-in occurs, the Altair computer can now "decide" whether a phone call should be made or an alarm should be sounded in the building or home or at the police station.

The possibilities for use of the 680b-PCI and 88-PCI in industrial process control applications are endless. Some of the following applications will be topics for future articles in C/N on detailed uses of the 88-PCI or 680b-PCI.

--Monitoring and correcting temperature, pressure, pH, viscosity, etc., in chemical production processes.

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

--Control of test processes, such as AC power, heater/cooler, electromechanical loading; and collecting data either with 88-PCI boards alone or in conjunction with the 88-A/D Converter and the 88-Multiplexer.

SKATER'S WALTZ and OTHER FAVORITES, continued
See "BASIC goes BACH" page 12

The USER's Program

```
00001                       NAM     MUSIC
00002      F000   READER EQU     $F000
00003      43FD   SUSI   EQU     $43FD
00004      43FE   NOIE   EQU     $43FE
00005              *
00006              ********************
00007              *VALUE OF SUSI MUST BE POKED
00008              *
00009              * SET BASIC MEMORY SIZE = 16383
00010              *
00011              ********************
00012 4000              ORG     $4000
00013 4000 FE 0115       LDX     $0115
00014 4003 AD 00         JSR     X
00015 4005 DE AE         LDX     0174
00016 4007 FF 43FE       SIX     NOIE
00017 400A B6 43FD       LDA A   SUSI
00018 400D 8D 0A     11  BSR     OFF
00019 400F 8D 10         BSR     ON
00020 4011 4A            DEC A
00021 4012 26 F9         BNE     11
00022 4014 FE 0117       LDX     $0117
00023 4017 6E 00         JMP     X
00024              *
00025 4019 C6 B1    OFF  LDA B   #$B1
00026 401B F7 F000       SIA B   READER
00027 401E 8D 09         BSR     WAII
00028 4020 39            RIS
00029              *
00030 4021 C6 D1    ON   LDA B   #$D1
00031 4023 F7 F000       SIA B   READER
00032 4026 8D 01         BSR     WAII
00033 4028 39            RIS
00034              *
00035 4029 FE 43FE  WAII LDX     NOIE
00036 402C 09            DEX
00037 402D 26 FD         BNE     *-1
00038 402F 39            RIS
00039              *
00040              END
```

TOTAL ERRORS 00000

SKATER'S WALTZ and OTHER FAVORITES, continued
See "BASIC goes BACH" page 12

The BASIC Program

```
FORN=110200::?CHR$(0)::NEXT:LIST

1 CLEAR2000
2 POKE 287,64:POKE 288,00
3 S1(0)=2:S1(1)=5:S1(2)=10:S1(3)=20:S1(4)=40:S1(5)=80:S1(6)=160
4 X1(0)=992:X1(1)=496:X1(2)=248:X1(3)=124
5 X1(4)=62:X1(5)=31:X1(6)=16
7 K1=8/9:K2=9/10:K3=15/16
8 K4=(1/(2↑(1/12)))
9 K5=(2↑(1/12))
100 REM ***** DATA STATEMENTS GO HERE *****
1000 PRINT"(P)LAY  (C)OMPOSE  OR  (R)ANDOM";
1002 INPUTRPT$
1004 IFRPT$="P"ORRPT$="Y"THEN1050
1006 IFRPT$="C"THEN2999
1008 IFRPT$="R"THEN1700
1030 GOTO 01000
1050 GOSUB2000
1060 COUNT=COUNT+1
1070 IF D$="S" THEN MRFL=1
1080 IF MRFL=1 THEN COUNT=COUNT+1
1090 IF D$="SS" OR D$="END" THEN 01110
1100 GOTO 01050
1110 COUNT=COUNT-1
1120 IFMRFL=1 THEN COUNT=COUNT-3
1140 DIMNI(COUNT):DIMLN(COUNT)
1160 RESTORE
1170 FORN=1TOCOUNT
1172 GOSUB2000
1174 GOSUB1190
1176 NEXTN
1180 GOTO1595
1190 F=1:K6=1:ERFLG=0
1191 C$="":A$=""
1200 IFD$="S"THENGOSUB2000
1202 IFD$="+"THENLN(N)=-1:NT(N)=-1:RETURN
1210 IFD$="SS"ANDF1=0THENF1=1:RESTORE
1230 IFD$<>"SS"THENGOTO1270
1240 GOSUB2000
1250 IF D$<>"S" THEN 01240
1260 GOSUB2000:GOTO1190
1270 L=LEN(D$)
1280 A$=LEFT$(D$,1)
1281 IFA$="Z"THEN1562
1290 IFA$<"A"THEN1310
1300 GOTO1340
1310 F=VAL(A$):IFF=0THENERFLG=1:RETURN
1320 D$=MID$(D$,2,(L-1)):IFD$=""THENERFLG=1:RETURN
1330 L=L-1
1340 A$=LEFT$(D$,1)
1341 C$=MID$(D$,2,(L-1))
1350 IF MID$(D$,2,1)="<" OR MID$(D$,2,1)=">" THEN 01370
1360 GOTO 01420
1370 IF MID$(D$,2,1)=">"  THEN K6=K4
1380 IF MID$(D$,2,1)="<" THEN K6=K5
1390 D$=MID$(D$,2,(L-1))
1400 L=L-1
1410 C$=MID$(D$,2,(L-1))
1420 IF C$="" THEN C=3
1430 IFA$="R"THEN1560
1440 IF C$="'" THEN C=4
1450 IFC$="''"THENC=5
1455 IFC$="'''"THENC=6
1460 IF C$="!" THEN C=2
1470 IF C$="!!" THEN C=1
1475 IFC$="!!!"THENC=0
1480 X=X1(C):S=S1(C)
1490 S=S+F*K6:IFS>255THENS=255
1495 AVGS=AVGS+S
1500 IF A$="D" THEN X=X*K1
1510 IF A$="E" THEN X=X*K1*K2
1520 IF A$="F" THEN X=X*K1*K2*K3
1530 IF A$="G" THEN X=X*K1*K2*K3*K1
1540 IF A$="A" THEN X=X*K1*K2*K3*K1*K2
1550 IF A$="B" THEN X=X*K1*K2*K3*K1*K2*K1
1560 X=X*K6
```

Continued Page Twenty-Two

88-PCI Offers
Unlimited Potential
Continued

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

The applications listed below are just a few of the many areas in which the 88-PCI and the 680b-PCI may be utilized. Hopefully, these applications will stimulate new ideas. Of course, we welcome any interesting concepts developed by users and will consider these original applications ideas for publication in Computer Notes.

1. Household:

Input	Output
a. thermostats	e. heater/
b. alarm sensors	cooler
c. clock	f. alarms
d. manual switches,	g. coffee
remote control	h. stereo
transducers	

2. Ham Radio:

Input	Output
a. signal strength	c. antenna
comparator	rotator
b. end of tape	d. recording
sensor	equipment
	e. power switch
	for trans-
	mitters

3. Test Control:

Input	Output
a. thermostats	d. heaters/
b. remote logic	coolers
input	e. power
c. comparators	supplies
	f. load switch-
	ing

4. Agriculture:

Input	Output
a. flow meters	f. pumps
b. pressure	g. solenoid
sensors	valves
c. automatic mat-	h. conveyor
erial weighing	motors
d. position	i. hopper
switches	control
e. thermostats	j. distribu-
	tion control

5. Chemical Process Control:

Input	Output
a. ph meters	e. agitator
b. thermostats	motors
c. viscosity	f. pumps
testers	g. valves (sol-
d. flow meters	enoid)
	h. hopper
	control
	i. heaters

Continued

88-PCI Offers
Unlimited Potential
Continued

6. Industrial Process Control:

Input	Output
a. strain gauges	g. tool motor power
b. limit switches	
c. pressure transducer	h. tool position motors
d. thermostats	i. valves
e. position sensor	j. heaters
f. miscellaneous tester outputs	k. rejector solenoid
	l. solenoid operated stampers

One particularly interesting application for either the 88-PCI or the 680b-PCI is solar tracking. Solar tracking mechanisms have recently become widely implemented in conjunction with solar boilers for heating and electrical generation, solar cooking ovens and many scientific uses.

Several questions should be considered when designing a solar tracker control system.

1) How will the tracker know if it is on target?

2) If it is not on target, how will it know which way to go?

3) How will it know how far to go?

4) Will it search endlessly at night, or will it track the moon?

One means of resolving these problems is to construct a solar collector and solar sensor in a common chassis with parallel center focal axes, as shown in Figure 1.

The sensor is aligned so that it can detect an offset in rotation. (See Figure 2.) Thus, when the sun is not on the sensor's focal axis, light will fall on one of the photo transistors. The geometry of the sensor can be adjusted for the degree of tracking accuracy desired. Two sensors are required--one for the elevation axis and one for the azimuth. Infrared type photo transistors should be used since they are the most discriminating (i.e., they will not track the moon, clouds, etc.).

The circuit shown in Figure 3 can be used to connect the sensors to the 88-PCI or the 680b-PCI. Four such circuits should be used to provide the four inputs to the board (above, below, right and left). All are low active at the PIA. The software should scan the inputs periodically and adjust the position until all the signals are extinguished. The adjustment may use four relay outputs--two for the elevation motor (up, down) and two for the azimuth motor (right, left).

A general purpose sun sensor which measures the amount of sunlight is another possible input. This sensor enables the computer to sense whether or not there is sufficient light to track. The sensor can also be used to signal the computer during damaging weather conditions, since the sky is usually dark. In this case, the computer will cause the elevation motor to turn the collector face down. The sensor should be placed where it can sense the sun at any time of day.
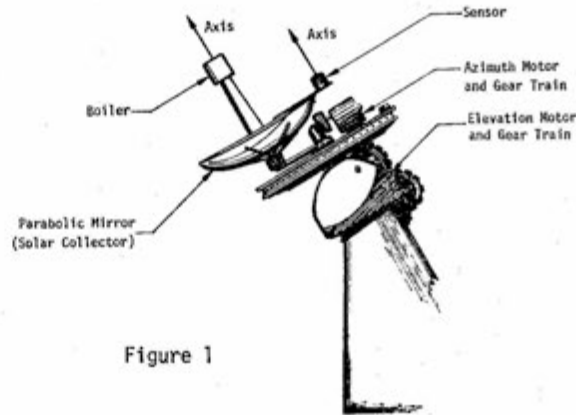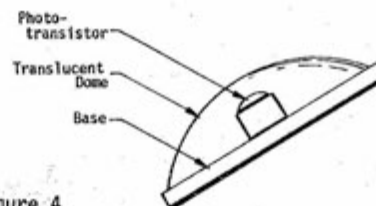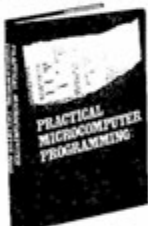
Figure 1

Figure 2

Figure 3

Figure 4

SKATER'S WALTZ and OTHER FAVORITES, continued
See "BASIC goes BACH" page 12

```
1561 IFA$="R"THENX=-F:GOTO1570
1562 IFA$="X"THENX=VAL(MID$(D$,2,(L-1))):S=-1:GOTO1570
1564 IFASC(A$)<65ORASC(A$)>71THENERFLG=1:RETURN
1570 LET NT(N)=X
1580 LET LN(N)=S
1590 RETURN
1595 AVGS=AVGS/COUNT
1600 FORN=1TOCOUNT
1610 S=LN(N):X=NT(N)
1614 IFS<0ANDX<0THENGOSUB12000:GOTO1630
1615 IFX<0THENX=(AVGS)*(-X):GOTO1800
1616 IFS<0THENPRINTCHR$(X)::GOTO1630
1620 GOSUB 01660
1630 NEXT N
1639 IFRP1$="Y"THEN1650
1640 IFRP1$="P"THENPRINT"AGAIN    (Y)ES OR (N)O   "::INPUTRP1$
1641 IFRP1$="Y"THEN1600
1642 IFRP1$="N"THENEND
1643 RETURN
1650 FORN=1TO5000:NEXT:GOTO1600
1660 POKE 17405,S
1670 X=((INT(100*X))/100)
1680 Y=USR(X)
1690 RETURN
1700 S=INT(100*(RND(3)))
1710 X=INT(150*(RND(2)))
1740 IF S=0 OR X=0 THEN GOTO 01700
1750 IF S>255 THEN 01700
1760 GOSUB 01660
1770 GOTO 01700
1800 FORLP=1TOX:NEXT
1810 GOTO1630
2000 IFRP1$="P"ORRP1$="Y"THENREADD$
2010 IFLEFT$(D$,1)=" "THENDS=MID$(D$,2,(LEN(D$)-1))
2020 RETURN
2999 DIMS1$(200):DIMNT(200):DIMLN(200)
3000 PRINT"W=WRITE L=LIST P=PLAY .D=DUMP"
3005 N=0:MRFL=0:COUNT=0:RFL=0
3100 PRINT"WRITE . . ."
4020 N=N+1:INPUTEMP$:D$=TEMP$
4021 IFD$="W"THEN3005
4022 IFD$="P"THEN8000
4024 IFD$=".D"THEN7000
4025 IFD$="S"THEN9000
4026 IFD$="S$"THEN7000
4027 IFD$="L"THEN11000
4028 IFMRFL<>0THENMRFL=MRFL+1
4050 IFLEFT$(D$,1)="-"THEN4052
4051 GOTO4060
4052 N=N-VAL(MID$(D$,2,(LEN(D$)-1)))-1
4053 PRINT"#NOTES NOW"N
4054 IFMRFL>0THENMRFL=MRFL-VAL(MID$(D$,2,(LEN(D$)-1)))-2
4056 IFMRFL=0THENRFL=0
4057 GOTO4020
4060 GOSUB1190
4062 IFERFLG=1THENPRINT"REJECT":N=N-1:IFMRFL<>0THENMRFL=MRFL-1
4064 IFERFLG=1THEN4020
4070 IFS<>0ANDX<>0THENGOSUB10000
4080 S1$(N)=TEMP$:GOTO4020
7000 N=N-1:CNT=99:X=1:X1=1
7010 NULL5:GOSUB7600:PRINTCHR$(15)
7020 CNT=CNT+1:C$=RIGHT$(STR$(CNT),3)
7030 PRINTC$;" DATA ";
7040 PRINTS1$(X);
7050 IFX<=NTHENX=X+1
7055 IFX>NTHEN7200
7060 IFX1=10THENPRINT;X1=1:GOTO7020
7070 X1=X1+1:PRINT",";:GOTO7040
7200 PRINT",S$";:PRINTCHR$(15):PRINT"RUN"
7210 NULL1:GOSUB7600:GOTO3010
7600 FORQ=1TO50:PRINTCHR$(0);:NEXT:PRINT:RETURN
8000 N=N-1:COUNT=N:SAVE=N
8002 IFRFL=0THENGOSUB1595:GOTO8060
8003 IFCOUNT=0THENCOUNT=.1:AVGS=AVGS/COUNT
8004 FORN=1TO(RFL-1)
8006 GOSUB10000
8010 NEXT
8020 FORX1=1TO2
8022 FORN=(RFL+1)TOCOUNT
8040 GOSUB10000
```

88-PCI Offers
Unlimited Potential
Continued

680b-PCI Process Control Board Addresses

A0	(RS0)	A1	(RS1)	C/SRA-2	C/SRB-2	
0	(1)	0	(0)	X	X	Section A Control/Status Register
1	(0)	0	(0)	0	X	Section A Data Direction Register
1	(0)	0	(0)	1	X	Section A Data Register (Opto Inputs)
0	(1)	1	(1)	X	X	Section B Control/Status Register
1	(0)	1	(1)	X	0	Section B Data Direction Register
1	(0)	1	(1)	X	1	Section B Data Register (Relay Control Output)

680b-PCI Process Control Board
Preliminary Specifications

*Opto-Isolators

    Isolation: Resistance Input to Output: 200G ohms
                Voltage: 1500V min.

    Input Active Current: 10-100 mA

    Propagation Delay: Varies with diode current from input to data
                        (D) line, diode and transistor:
                        Turn on: 6 μsec., typical
                        Turn off: 10 μsec., typical

*Relay Outputs

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

    Contact Data: Unloaded mechanical life: 20 x 10^6 operations
                   1 amp at 120 VAC: 5 x 10^5 operations
                   Contact Resistance: 100 milliohms, maximum (silver,
                                        gold overlay)

    Propagation Delay: Pull In: 3.5 nsec.
                        Release: 4 nsec.
                        Bounce: 1.2 nsec.

*Handshake Lines

    Input: Same as Opto-Isolators
    Output: Off leakage current @ 25°C: 10^{-9} amps
             On Current: Approximately 30 mA

SKATER'S WALTZ and OTHER FAVORITES, continued
See "BASIC goes BACH" page 12

```
8050 NEXTN
8055 NEXTX1
8060 N=SAVE:GOTO4020
9000 IFRFL<>0THENPRINT"REJECT":N=N-1:GOTO4020
9002 MRFL=1000+N:RFL=N
9010 STS(N)=DS
9020 GOTO4020
10000 S=LN(N):X=NT(N)
10008 IFS<0ANDX<0THENGOSUB12000:GOTO10030
10010 IFX<0THENX=AVGS*(-X):FORLP=1TOX:NEXT:RETURN
10012 IFS<0THENPRINTCHRS(X);:GOTO10030
10020 GOSUB1660
10030 RETURN
11000 N=N-1
11010 FORX=1TON:PRINTSTS(X);
11020 IF(X/10)=INT(X/10)THENPRINT:GOTO11040
11030 PRINT","";
11040 NEXTX
11050 PRINT:PRINT"#NOTES":N:GOTO4020
12000 S=10
12010 FORX=1TO30:GOSUB1660:NEXT
12020 RETURN
OK
```

88-PCI Offers
Unlimited Potential
Continued

*Supply Requirements

    Unregulated +9 volt bus: Idle: Approximately 100 mA.
                                 All relays energized: add 300 mA

Below are some diagrams of the Opto-Isolator Input and Output along with
a diagram of a Relay Output. Also, included is a block diagram of the
board.

    **Except for the following changes, all of the specs for the 88-PCI
Process Control Interface Board are the same as those for the 680b-PCI.

          88-PCI Process Control Interface Board Addressing

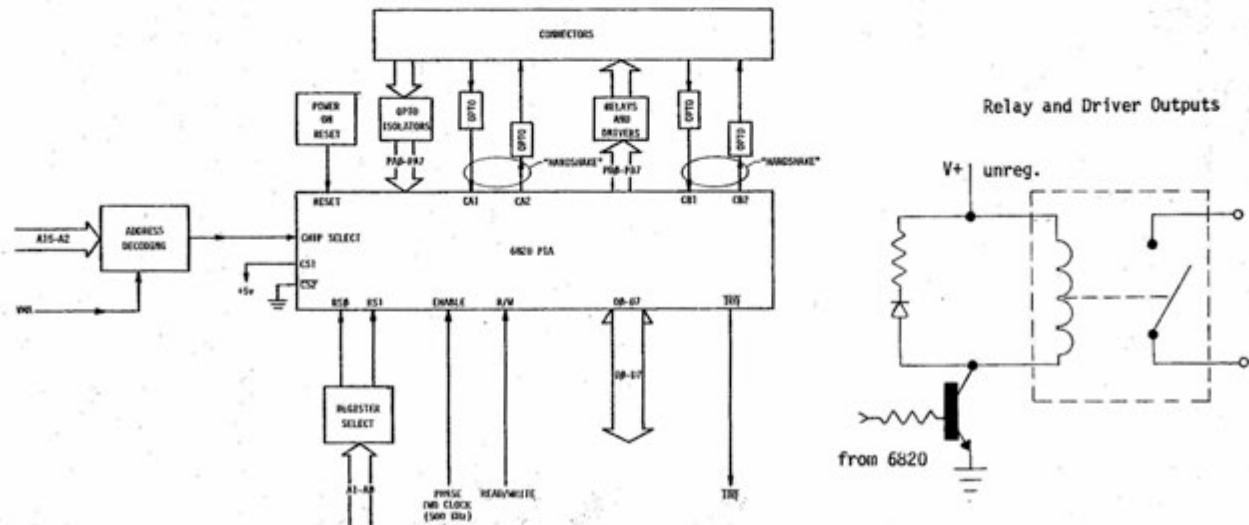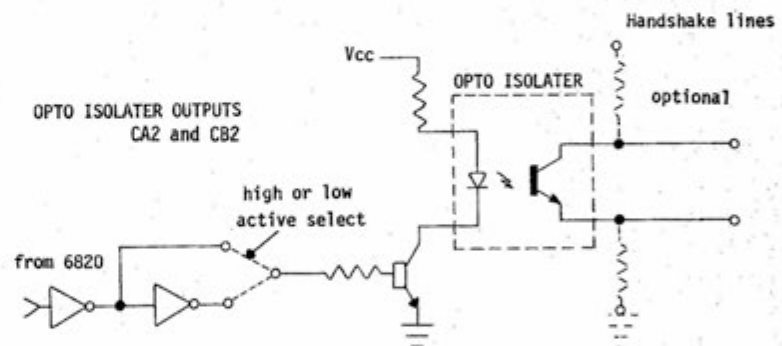| A0	(RS0)	A1	(RS1)	C/SRA-2	C/SRB-2	
0	(1)	0	(0)	X	X	A Section Control/Status Register
0	(1)	1	(1)	0	X	A Section Data Direction Register
0	(1)	1	(1)	1	X	A Section Data Register (Opto Inputs)
1	(0)	0	(0)	X	X	B Section Control/Status Register
1	(0)	1	(1)	X	0	B Section Data Direction Register
1	(0)	1	(1)	X	1	B Section Data Register (Relay Control Output)

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

*Opto Isolators

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

*Supply Requirements

    Unregulated 8v bus: Idle: Approx.
    150 mA. All
    relays ener-
    gized: add
    200 mA. CA2,
    CB2 on: add
    100 mA.

OPTO ISOLATER OUTPUTS
CA2 and CB2

high or low
active select

from 6820

Vcc

OPTO ISOLATER

Handshake lines

optional

Relay and Driver Outputs

V+ unreg.

from 6820

CONNECTORS

POWER ON RESET

OPTO ISOLATORS

OPTO

OPTO

RELAYS AND DRIVERS

OPTO

OPTO

PA0-PA7

"HANDSHAKE"

PB0-PB7

"HANDSHAKE"

RESET

CA1

CA2

CB1

CB2

ADDRESS DECODING

A15-A2

VMA

CHIP SELECT

CS1

CS2

+5v

6820 PIA

RS0

RS1

ENABLE

R/W

D0-D7

IRQ

REGISTER SELECT

A1-A0

PHASE TWO CLOCK (500 kHz)

READ/WRITE

D0-D7

IRQ

Whoa!

Computer Notes is continually seeking quality manuscripts on applications, troubleshooting, interfacing, software, book reviews, fiction and a variety of other computer-related topics. Articles and article ideas are always welcome. Honorariums are based on an article's technical quality and its suitability for *C/N's* readership, and range from $20 up to $50 per typeset magazine page. All articles are subject to editing to fit space requirements and content needs of our readership. Payment for articles which are accepted will be sent within 4 weeks after receipt of a manuscript.

Articles submitted to *C.N.* should be typed, double-space, with the author's name, address and the date typed in the upper left-hand corner of each numbered page. Authors should also include a brief statement about their job title, background and/or experience written separately under the article's title.

Photos, charts, programs and figures should be clearly labeled and referred to by number within the text of the manuscript. Only clear, glossy black photos (no Polaroid pictures) can be accepted. All photos should be taken with uniform lighting and sharp focus.

Program listings should be recorded with the darkest ribbon possible on new blank white paper.

Articles which are not accepted will be returned.

Computer Notes
Attn. Linda Blocki
2450 Alamo SE
Albuquerque, New Mexico 87106