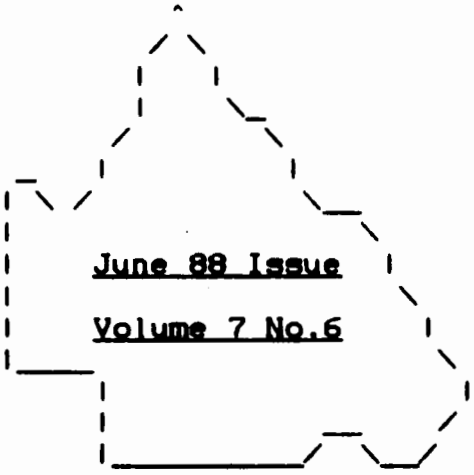
APPLE-BUG



The Newsletter of APPLE-Q Inc.

the Brisbane User's Group

Post Office Box 721

South Brisbane

Queensland 4101

June 88 Issue

Volume 7 No.6

>>> In this issue <<<

[CREDITS]

Once again the credit for getting this issue to press, must go to Dale Rodgie and the tireless efforts of the Executive Committee, along with the help of a few members who have taken the time to put fingers to the keyboard. The Hooper Education Centre has once again performed the impossible task of accepting our pages, making enough sense out of them to make the plates, doing the actual printing, collating, stapling, folding and stapling again, and getting them to Australia Post, who in turn attempted to get them to you in time for you to read them before the sun sets on another Open Day, for yet another month.

>> Dale Rodgie -- He typed it all in <<
>> Graham Black -- He typed some too <<
>> The Members -- For their contributions <<
>> Appleworks -- Word Processing <<
>> ImageWriter II -- Typesetting <<
>> The Hooper Centre -- Printing & Distribution <<
>> The APPLE Computer -- The reason for it all <<

$1.50

Registered by Australia Post - Publication No. QBG 3485.

[Executive Committee]

Bernie Benson	- President	Ph.(07) 345-1545
Eric Conolly	- Vice-President	Ph.(07) 261-1860
Bob Godbehere	- Secretary	Ph.(07) 808-3892
Tony Truscott	- Treasurer	Ph.(07) 266-4795
Graham Black	- Registrar	Ph.(07) 883-1525
Dale Rodgie	- Editor	Ph.(075) 38-6942
Ken Smith	- Software Librarian	Ph.(07) 345-1995
Brett Dutton	- Hardcopy Librarian	Ph.(07) 870-8599

[Bulletin Board System <BBS>]

Apple-Q BBS : online 24 hrs
Telephone : (07) 284-6145 (DATA)
: (07) 883-1525 (VOICE)
Baud Rates : 300, 1200/75, 1200 and 2400 (CCITT and BELL)
Data Specs : 8 Data bits - 1 Stop bit - No Parity (Full Duplex)
Sysops : Graham Black - Vince Crosdale - Brett Dutton
Calls to the system : 56
Registered Users : 29 (as of 10.00 p.m. 2nd June)
SYSOP stands for : SYStem OPerator
BBS stands for : Bulletin Board System

[What's When]

at the Hooper Education Centre - Kuran Street - Wavell Heights

Open Day
Sunday 19th June 1988
Hours: 9.00.am. till 4.30.pm.

Committee Meeting
Monday 20th June 1988
Starts: 7.00.pm.

Open Day
Sunday 17th July 1988
Hours: 9.00.am. till 4.30.pm.

Committee Meeting
Monday 18th July 1988
Starts: 7.00.pm.

[Membership Fees]

Joining Fee: $10 Adults/Family: $20
Full Time Students: $12 Pensioners: $12
Corporate Membership: $50
Associate Membership: $ 5 plus $5 Joining Fee (BBS only)
(Full Time Students under 21 years on production of Student Card)
(Pensioners on production of Pensioner Card)
[.....at the discretion of the Executive Committee]

Copying fees for the Club's Public Domain Software are 50 cents per disk side. with a minimun charge of of $2.00 for 5.25" disks. The copying fee for 3.5" disks is $3.00 per disk. We cater for the][.][+, //e. //c, //GS and MAC. The copying of Commercially produced software cannot be sanctioned by APPLE-Q Inc. and members who do so risk expulsion from the group.

All contributions for the newsletter should be handed to a committee member at the Open Day, or posted to P.O.Box 698, Redcliffe, Queensland 4020. The deadline date is the committee meeting immediately following the Open Day.

[Editorial]

by Dale Rodgie

Welcome to the June edition of Apple-Bug. In the last edition of the newsletter. I said that we would be reviewing TML Basic and Copy II+ version 8. Well, they just arrived a few days ago, so I will include the reviews in the July Apple-Bug.

This brings me to the subject of articles. We are constantly looking for articles for the newsletter. My interest leans toward the Apple IIGS. so I write articles about the GS. However, most of the members have other Apple II models. So we need articles on any subject related to the Apple computer. They could be a software review, a helpful hint. or articles on programming. All members get a lot from the group. so how about giving some of your time back to the group to help others.

The other day, the BBS user file was lost. So, when you connect with the BBS, you will need to enter your name. You will then be given a new user number. While the user file was lost, the messages on the BBS are ok. It also means that the number of cals to the system and the number of users have been lost. This is why these figures are quite low on the last page. This creates problems if you "Auto-Reply" to a message, since the user number is now different. Please do not use Auto-Reply for a while.

Last month we had nine new members join the group, they are: Richard Tomlins. James Grainger, the Piat-niski family, Peter Mc Call, Carolyn Allen, Andrew Hamilton. Norman Carey, Jenny Loader and Ross Hall. The committee would like to welcome you all to the group. If you have any questions, I am sure one of the other members would be able to answer them for you. We are now getting members from all over Australia. One of the members that joined last month comes from the Indian Ocean. or at least on an island in the Indian Ocean.

Remember to display your membership card on your Apple this Open Day. There is only limited space available, so only Apple-Q members can set up their machines. People not displaying their membership card will be asked to pay the membership fee or remove their computer.

Both the Trading Table and the Software Library will close at 1.30 p.m. this Open Day. So please orginize your business around these new business hours. Also, please don't go behind the counter at the Open Day.

A small number of power boards are available, at Open Days, for a deposit of $20.00. Due to the number of missing power boards, we had to increase the deposit. I recommend that you bring your own as the few available will not last long.

This Open Day is the last chance you have to submit entries for the Logo Competition. We are looking for a logo to print on the front page of the newsletter each month. The prize. for the winner, is a years free membership. The winner will be announced in the July newsletter along with the winning entry. You can submit your entry to any member of the committee. We need both the logo on paper and on disk (in a high-res file).

We have two special interest groups operating on Open Days. If you are interested in Machine Language programming on the Apple II's then join Brett Dutton in the Software Library Room at 1.30 pm. For more information, read the Hardcopy Report. We also a special interset group for serious users. So if you are interested in Business programs, Programming, etc. for both the Macintosh and Apple II's, the meeting will be held in Room 22 in the main Hooper Centre Building between 1 and 4 p.m. on Open Day's.

[Hardcopy Library]

by Brett Dutton

As you will have noticed for anyone who has seen our library that there is a large amount of information available (and quite a deal of reading). Since I have taken on this position on the committee, I have found a wealth of knowledge on one of the subjects that has interested me for quite a while. I have since taught myself assembly language with a lot of help from our library and a bit of help from QIT.

I know that there are a few people who would like to understand and program in assembly language, so, I will be commencing classes in assembly language for the few who want to know.

I do not claim to be an expert, but I will endeavour to empart all the knowledge necessary to start programming in assembler, and hopefully you can donate some good programs to the software library.

So the classes will begin at the next meeting at 1.30 p.m. in the Software Library Room (to your right just before you enter the front door). People who come to the classes must be fully fluent in programming in either basic or any other higher level language. But for more information come to the first class.

[Software Library]

by Ken Smith

If you are new to the club you are probably wondering what is in the club's library and is it good value? and if I am interested in a particular disk, how do I go about obtaining a copy?

Firstly, if you haven't a copy of the club's library, see me at the next Open Day. There is a catalog of all the public domain/shareware programs we own. You can then scroll through the listing at home at your leisure or send it to your printer for a hard copy.

Are the programs any good? We don't sell programs but we do have a copy charge. Most of the programs have some value and for a couple of dollars represent good value. The standard charge is fifty cents per side with a minimum charge of two dollars for five and a quater inch disks, or three dollars for a three and a half inch disk. Our group of programs represent the best value for your money. For example we have quite a number of data disks of graphics for use with Print Shop. At a total cost of ten dollars for twenty sides, or part there of. you will have access to hundreds of graphics, the set also contains Print Shop utilities. I challenge you to find a better value on the commercial market.

Are you into adventure games? We have the Eamon adventure series in the library starting with #36, Beginners Cave. You should start with this disk if you wish to follow this series as it sets up the scenario and the characters. You can then branch out to #37 Wonderful World of Eamon, #44 Cave of the Mind, #45 Death Star, #57 Eamon Dungeon Design, #58 Lair of the Minotaur. And all of this for only fifty cents per side. What a bargain!!!

Below is a list of the contents of four Public Domain game disks available in the Software Library. The letter in the square brackets represents the file type. ie. [A] = Applesoft Basic, [B] = Binary, [I] = Integer Basic and [T] = Text. If you have an an Enhanced Apple IIe, Apple IIc or Apple IIGS, you may need to get Integer Basic before you can run programs marked [I].

GAMES DISK #31 (Dos 3.3) ==========

[I] AIR FORCE BOMBER
[I] DEATH STAR
[I] LIFE (LOMEM-3072 HIMEM-8192)
[I] TOWERS OF HANOI SOLVED
[A] SHIP CAPTAIN CREW
[I] LIFE
[I] BINGO
[I] BINGO CARD
[I] SLOTTS
[I] STARWARS RESCUE
[I] SKUNK
[I] LAS VEGAS SLOT MACHINE
[A] SNAKE ARCADE GAME
[I] WENATCHEE APPLESTAND (GAME)
[A] MOONLANDING
[A] POKER DICE MK V
[I] SLOT MACHINE
[A] RUBIK-RANDOM
[A] RUBIK-ROTATE
[B] CHAIN
[I] OTHELLO
[I] TWENTY-THREE BRICKS
[I] SINK THE SHIP
[I] SLOT MACHINE ASB 1
[I] STAR WARS
[I] SHOOTOUT
[I] MISSION- U-BOAT
[I] PENTOMINOES A2048 L1792
[B] LIFE.OBJ(AS)
[I] POKER STUD
[I] BINGO CARD
[I] SLOT MACHINE LOTTERY
[I] NEW FLY KILLER
[I] AWARI
[I] OTHELLO
[A] SIMON
[B] H-DICE
[I] NJ LOTTERY
[I] AUTOMATIC BINGO
[I] PHILA. PINBALL
[A] HANGMAN
[I] HANGMAN 2
[I] MOON LANDER
[I] NEVADA CRAPS
[B] SNAKE LOGO
[A] CANYON RUN - COLOUR
[A] COLOR BARS
[A] EVASION
[A] RUBIK
[A] RUBIK-PLOT
[A] RUBIK-END
[I] PINBALL
[I] CATCH
[I] TOWERS OF HANOI
[I] MASTERMIND
[I] BLACKJACK
[I] INTERCEPT
[I] CHASER
[B] PENTOMINOES
[A] LIFE(AS)
[A] HANGPERSON
[B] POKER STUD.X
[I] BINGO
[I] CASINO

GAMES DISK #32 (Dos 3.3) ==========

[I] GAME PAK 4
[I] PINBALL
[I] POKER DRAW 2
[I] SEA HUNT
[I] SLOT MACHINE COLOR
[B] ALIVADER
[A] LIFE
[B] L.TUMBLER
[B] L.GUN
[A] EVOLUTION INTRO
[A] HI-RES EVOLUTION
[B] ANANA.ANM
[A] F
[A] FOUR IN A ROW
[B] FLIP IT
[B] SUPER BLAST
[A] NUCLEAR REACTOR
[I] PINBALL MARBLE
[I] CHASE REAL TIME
[B] POKER DRAW.X
[I] SHOOTING STARS 4
[I] TOWERS
[A] HORSE.RACE
[B] L.MICE
[B] L.TRAIN
[B] L.TRASH
[A] COLOR EVOLUTION
[A] ANIMATED ANAGRAMS
[B] LETTERS.ANM
[B] CARD.SHPS
[A] MAD CUBE
[B] MAXX'S MADNESS
[B] THE MAXX MACHINE

GAMES DISK #33 (Dos 3.3) ==========

[I] LIFESAVER
[A] FIREWORKS2
[A] CRASH
[A] FIREWORKS1
[A] CALENDAR
[A] SPACE ANIMATION

[B] LIFE
[A] GUIDED MISSILE
[B] HI-RES
[A] COLOR GUN
[B] HIRES.SKETCH.BIN
[A] BLAST AWAY
[A] MAZE
[I] STARS
[I] APPLAYER MENU
[B] APPLAYER
[I] GALACTI-CUBE
[A] THERMOMETER
[A] LUNA (ROB NEARY)
[A] MATCHES
[I] CODES AND HANGMAN
[I] CONCENTRATION

GAMES DISK #34 (Dos 3.3) ==========

[A] ^CLC GAME 038
[I] CHESS 2
[I] CONCENTRATION
[I] FIRE
[I] GOLF SCORE & HANDICAP
[I] POKER STUD
[I] SIMON
[I] STORY TELLER
[I] TIME BOMB
[A] ^CLC GAME 048
[I] APPLE WARS 1
[I] BAGELS 1
[I] BULLS AND BEARS
[I] GUIDED AIRPLANE
[I] INTERSTELLAR
[I] SLOT MACHINE $2 + SOUND
[I] SPACE WAR FOR TWO
[I] STAR TREK MINI
[I] STAR WARS FOUR
[I] WAR LORDS
[I] BATTLESTAR
[B] CHESS 2.X
[I] DRAGON MAZE 3
[I] FISHING->NEEDS WORK
[I] PINBALL SUPER
[B] POKER STUD.X
[I] SOLITAIRE CHECKERS BW
[I] TAX MAN
[I] WUMPUS 2
[I] ANTI AIRCRAFT
[I] APPLE WARS 1.1
[I] BINGO AUTO->NEEDS WORK
[I] GAME PAK NIGHTMARE
[I] HIDDEN MOUSE 2
[I] KENO 1
[I] SPACE PILOT
[I] STAR SHIP ATTACK
[I] STAR WARS
[I] TREK

==========

[Letter to Editor]

Dear Sir.

I would like to commend your committee on thier move to establish the information base of people, who would be willing to assist mugs like myself, to understand something about my computer's make-up, and how I can improve its performance.

May I suggest also that an additional service such as a "Wanted" and "For Sale" column might be appreciated by some members. I for one would like to upgrade my 64K unit without having to purchase additional drives and monitors etc., which are usually included in second hand packages. Just such a column I think would assist members greatly.

I do realise, however, that such a column will add more work to the editing and publication of the Apple-Bug and perhaps I am being selfish in suggesting it.

Ron Lingard.

Editor: Thank you Ron for the letter. In each edition of Apple-Bug we have an Advertisements section. Financial members can advertise for a particular item or list of items for sale for free. As you point out, most of the computers sold are complete packages. This could be because that most new Apple computers are bought as a package. Anyone who would like to advertise in the Apple-Bug, please contact any committee member at the Open Day, write to the groups post box or leave a message on the Apple-Q BBS under the name of Dale Rodgie.

[Apple News]

A Talking Vending Machine?

A vending machine like the one shown on Beyond 2000 a few months ago, is now in use in the United States. Instead of dispensing soft drinks or mars bars. this one dispenses computer software. How to use it is quite simple. The machine has a built in keyboard and screen - you simply select the software you want and insert your credit card. It then copies the software on to the disk on the spot. The disk then pops out. The manual is printed on a dot matrix printer built into the machine. When the printer is finished. you simply tear off the manual. The American version of the vending machine also talks to you.

The Go-Anywhere Clock

Apricorn, in the United States, have developed a clock/calender chip that can be installed in an empty RAM chip socket or even under an existing RAM chip. It also comes with a menu driven utility to patch ProDOS for the chip. The clock. called Clockpro. costs $59.95 (U.S.) and is available from Apricorn, 10670 Treena Street. Suite 10. San Diego, CA 92131 - Phone 1100-1-619-271-4880.

Floppy Funnies

The software publisher Infocom and Tom Snyder Productions have joined together to release comic books on diskette. The comics, called Infocomics, are now available in the States for $12.00 (U.S.). They are not a game or puzzle, but contain the graphic images that are displayed on your screen. The disks contain 10 times the images than their paper counterparts. Some of the titles available are: Lane Mastodon vs. The Blubberman, Gamma Force in Pit of a Thousands Screems, and Zorkquest: Assault of Egreth Castle.

Apple goes MIDI

Apple in the States have released a MIDI interface for the Apple IIGS and Macintosh. With the MIDI interface, you can connect you Apple to MIDI-equipped musical instruments. With the right software, the computer can be made to handle music sequencing, composing, and editing faster and more easily, says Apple. It is now available in the States for $99.00 (U.S.).

Double the Modem at half the Price!

Micro-Educational are now selling the latest Maestro modem. The Maestro 2400 ZXR has everything you ever wanted in a modem - V21, V22, V23(Viatel) and V22bis CCITT and BELL. Add to that auto-answer, auto-dial, auto-disconnect, auto baud rate sense. pulse and tone dialing. It is also fully Hayes compatible. All of that for only $399.00. Call Micro-Educational for more information - (049) 26 4122.

Desktop Programming

Byte Works, in the States. have just released ORCA/Desktop. Orca/Desktop is a shell that can be used with all the Apple Programmer Workshop languages. It is WIMP interface that can be used to work on source files and compile them. It is quite an achievement since APW uses the text screen. Orca/Desktop also contains a source level and a machine level debugger, step, trace and the ability to view and change the stack and memory. The Desktop costs $60.00(US) and requires an Apple IIGS and one 3.5 inch drive.

Apple's new CDROM drive

Apple Australia have now released the AppleCD SC. It is a compact disk disk player designed to be connected to either the Macintosh or Apple II's. The player reads the compact disk like your standard hi-fi CD player. Like the hi-fi CD player, the information is stored on the disk digitially, and can't be erased to re-recorded. The CD-ROM or Compact Disk Read Only Memory can store up to 270,000 pages of typewritten text - that is about 550 megabytes. However, not just text can be stored on the disk - sound, graphics and text can be mixed to provide dazzling presentations. The AppleCD SC sells for $1,199.00(US).

[Tips & Techniques]

Date Format : "The Aussie Way"

by Bernie Benson

The old problem of dates and which is the better way of expressing them - the Aussie or U.S. format. If you use Beagles Bros. Urtramacros you will be aware that there are two macros to print the current date which has been entered at startup of Appleworks or if you have a clockcard the present date and still further what ever date you want to set using <ba-d>.

Unfortunately both macros return the U.S. format as follows:

<date> May 14, 1988
<date2> 05/14/88

The following macro has been written to replace the <date> macro. It has to be defined as some other name and used instead of <date>. I have defined mine as <sa-f>.

The macro as listed will print the date as follows:

14th May, 1988

The listing is rather long however the processing time is comparable with the <date> macro. I wrote another one that did the same job only using about 50% of the code however it used a file to do the job and that slowed down the process so I scrapped the idea infavour of this one.

```
F:<awp><$1 = date><$2 = right $1,4><$3 = date2><$4=left $3,5><$5 = right $4,2><$6 = left $4,2><m = val $6><if m = 1 then l = 7 else if m = 2 then l = 8 else if m = 3 then l = 5 else if m = 4 then l = 5 else if m = 5 then l = 3 else if m = 6 then l = 4 else if m = 7 then l = 4 else if m = 8 then l = 6 else if m = 9 then l = 9 else if m = 10 then l = 7 else if m = 11 then l = 8 else if m = 12 then l = 8 elseoff><$8 = left $1.l> <c = val $5><$9 = "th"><if c = 1 then $9 ="st" else if c =21 then $9 ="st" else if c =31 then $9 ="st" else if c = 2 then $9 ="nd" else if c = 22 then $9 ="nd" else if c = 3 then $9 = "rd" else if c = 23 then $9 = "rd" elseoff><$1 = str$ C + $9 + " " + $8 + ", " + $2><print $1>!
```

Explanation of logic:

```
F:<awp><$1 = date>                    read date into $1
<$2 = right $1,4>                     read year into $2
<$3 = date2>                          read date into $3
<$4=left $3,5><$5 = right $4,2>       read day into $5
<$6 = left $4.2>                      read month into $6
<m = val $6>                          convert month to value
<if m = 1 then l = 7 else             set l to the number
if m = 2 then l = 8 else              of letters in month
if m = 3 then l = 5 else
if m = 4 then l = 5 else
if m = 5 then l = 3 else
if m = 6 then l = 4 else
if m = 7 then l = 4 else
if m = 8 then l = 6 else
if m = 9 then l = 9 else
if m = 10 then l = 7 else
if m = 11 then l = 8 else
```

```
if m = 12 then l = 8 elseoff>
<$8 = left $1,1>                          use l to read month
<c = val $5>                              convert day to value
<$9 = "th">                               set suffix to "th"
<if c = 1 then $9 ="st" else              reset suffix if not "th"
if c =21 then $9 ="st" else
if c =31 then $9 ="st" else
if c = 2 then $9 ="nd" else
if c = 22 then $9 ="nd" else
if c = 3 then $9 = "rd" else
if c = 23 then $9 = "rd" elseoff>
<$1 = str$ C + $9 + " " + $8 + ", " + $2>  build date
<print $1>!                               print date
```

Receipt Number Repeater Macro

by Bernie Benson

Here's another tip from Bernie Benson for users of Appleworks and Ultramacros.

The macro, below, automatically enters receipt numbers in the data base after the first number is entered by the user. The numbers will either be generated across or down depending on whether you have set Return to go across or down. This macro will enter the next 50 numbers.

```
F:<adb><$1 = cell><1 = 50><A =val $1><rtn>

<begin>
< A = A + 1><print A><rtn><1 = 1 - 1><if 1 >0 then rpt>!
   text, clear previous word
```

Lower Case Modification for the Apple //+

by Brett Dutton

When I first joined Apple-Q (about December '86) I knew nothing at all about the Apple. I had been weaned on the old Radio Shack TRS-80 (yes the trash-80). Fortunately my father came across an Apple Europlus at a very good price. So then I had an Apple but I still didn't know anything about the machine. Well I did what most people do and called the dealer. He put me on to the club and from there, well, the rest is history.

So, to the point. I quickly became dissatisfied with the Europlus. The main problem at that time was the fact that I couldn't type anything in lowercase (memory requierments came later). After consulting a few people I had the solution.

This modification doesn't work in 40 column mode unless the software supports it, but most of the things I do are using the 80 columns anyway.

So, to the mod:

First you have to take the plastic case off the Apple being careful when disconnecting the keyboard from the main circuit board, not to bend the pins. Attached to the bottom of the keyboard is a small circuit board connected by two plastic clips and a 26 pin connector. Remove this carefully by squeezing together the plastic clips and gently lifting the small circuit board. Once the plastic clips are loose, just ease the circuit board off the 26 pin connector.

Once this is done, have the circuit board with the component side facing you and the pin connector facing away from you and look in the top-left quater of the board. Find six soldered holes with four of the holes connected by two solder pads. (see fig 1.) These pads look like 2 triangles point to point. The tracks have to be cut at these two points. Once you have done this you will need a DPST switch (two position switch with six pins underneath) and also 6 pieces of wire (I used a piece of ribbon cable). Strip and tin the ends of the six wires and solder one into each of the six holes on the circuit board. Solder the other ends to the six pins on the bottom of the switch. MAKE SURE THEY CORRESPOND (see fig 2.) Then replace the circuit board the way it was removed and you're almost finished.

Drill a hole in the Apple case of an appropriate size to accomodate the switch and in a good position so that it can be reached easily. Then install the switch. MAKE SURE THAT ANY DRILL SHAVINGS DO NOT GET ON THE MAIN CIRCUIT BOARD.

Now just put your Apple back together and you have your lowercase. Unfortunately the 6502 does not recognise lowercase commands but you can use the mod for programs (within quotes), word processing (Appleworks), and talking over the modem to Bullitin Boards. Actually you can try lowercase wherever it works.

How it Works

Well the keyboard processor can handle lowercase but the Europlus cannot. So the manufactures leave the option to convert it the way I have just described. With the original connection is output (before the tracks were cut) but by connecting the other option lowercase is output (with shift converting to uppercase).

Good luck with that, but if you have the slightest doubt don't attempt this. If you need help, I would be happy to give you a hand at the next open day.

fig 1

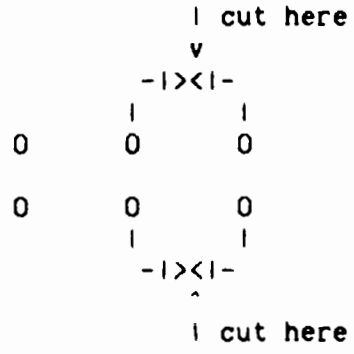
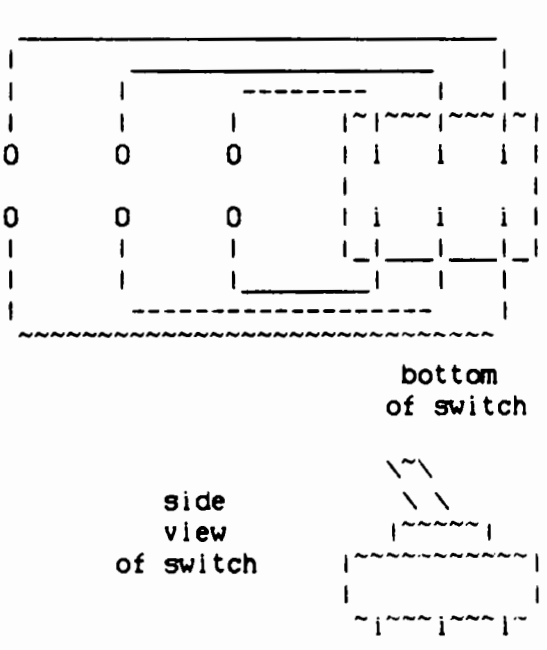


fig 2.



[Brisbane Bulletin Boards]

Apple-Q Inc.....(07) 284-6145	Educational RCP/M.....(07) 266-3369
Amiga Express.....(07) 393-5352	BMUG RBBS.....(07) 366-4833
Ampak PBBS-RCP/M.....(07) 263-7070	Missing Link BBS.....(07) 808-3094
Hi-Tech C Software BBS.(07) 38-6872	Rock Cave BBS.....(07) 359-1809
Kangaroo Point TAFE....(07) 393-1763	Electric Dreams.....(07) 399-1322
Tomorrowland RMSD.....(07) 283-0315	Fix.....(07) 284-5814

[Assembly Language Programming]

by Craig Johnson

What is assembler and machine language?

Most of us have some familiarity with BASIC, with it's english like commands such as "PRINT", "INPUT", and "FOR...NEXT". There are other high level languages, like Fortran and Pascal, which similarly formed of english type commands. The Apple's central processing unit (CPU) cannot understand such English commands, but instead the CPU must "interpret" the commands or "translate" them into a form which it can understand.

When running a program in BASIC, the Apple takes each line of the program and interprets it into a form which is acceptable to the CPU. Once interpreted, the CPU then executes the line. Applesoft is therefore called a "BASIC interpreter" or is an "interpreted BASIC".

The opposite of high level, structured language are assembler and machine language. The terms "assembly language" and "machine language" are often used interchangeably, although there is a difference. Programming in machine is accomplished when hexadecimal or binary data is loaded into the computer byte-by-byte (eg entering the Monitor by CALL -151 and inputing "300: A9 0A...") When computers were in their infancy, this was the only option available. This form of programming can only be intelligible to people who think like a computer. To us lesser mortals, writing a program in machine language is the ultimate in frustration.

(As aside: when Steve Wozniak and Steve Jobs first built and Apple computer, they took it to parties and entered hundreds of hex codes by hand just to load in a program to demonstrate this first Apple's ability. Allegedly, DOS was first written this way.)

Enter Assemblers. An assembler is simply a tool which allows a person to write a program using somewhat more intelligible commands, and then the assembler program converts ("assembles") the English G-like commands into the corresponding machine language code.

While there is a definite and significant difference between "assembly" language and "machine" language, the two terms are often used interchangeably. Throughout this article, such interchangeably shall be used.

Why program in machine language?

If assembly language is tougher than BASIC, why bother being a glutton for punishment? Is there something to be said for such an arcane language? Consider these three reasons: speed, flexibility and memory management.

The most prominent benifit of a machine language program can be seen when you play a commercially produced, arcade style game. Such games are invariably written on an assembler (the people at Southwestern Data Systems say the games from Sirius are written on Merlin). The execution speed of a machine language programis much faster than a corresponding BASIC program - arcade quality games simply cannot be written in BASIC. Even BASIC which has been compiled into machine language (by Applesoft compilers such as TASC or Speedstar) is still slower and more memory hungry than an equivalant program written in assembler.

The advantage of speed can be used in other situations which will significantly enhance the Apple's capability. Sorting, matrix operations and disk utilities are but a few reasons to use the speed of machine language.

The second reason to program in machine language is the increased in flexibility which it offers. The commands in Applesoft can be enhanced by the use of

shortmachine language routines hidden away in memory. Print formatting functions. sound generation, sorting, string handling and swapping, decimal to hex conversions, garbage collection and memory clearance, and array operations are but some of the more commonly used subroutines. While such machine language enhancements are available commercially, they can also be had at a very low cost, as they are listed in such publications as Nibble and Call-A.P.P.L.E.

The third advantage relates to memory management. Machine language programs are more efficient in their consumption of memory than are their BASIC counterparts. Because of this, it is possible to have several machine language routines residing in memory concurrently, thereby enhancing Applesoft while at the same time avoiding a significant reduction in the memory available to the user.

What is needed to get started

To begin programming in assembly/machine language, it is not necessary to purchase any software tools. A good book, however, can be invaluble. Once your interest is sparked, you may develop a sudden urge to purchase an assembler. Other machine language utilities are helpful.

There are three different levels of sophistication at which you may enter machine language programming:

1) Machine language using the monitor

Although not the most flexible approach to machine language, this is the cheapest route to go, and it forms a good starting place. The way to begin is to exit Applesoft and get into the Monitor by means of the command:

```
CALL -151
```

which yields the "*" prompt.

From here we can enter a program by giving the starting address and the hexadecimal data. Try the following:

```
*300:AD 30 C0 88 D0 FD 4C 00 03 <ret>
```

Where you enter the number 300, a colon, then 9 sets of data. Be sure that you do not leave a space between the colon and the first number. At the end of the line, enter a <carriage return>. Up to 85 hexadecimal numbers can be entered at once this way.

What this program does is toggle the speaker by addressing it's memory location (AD 30 C0), counting downwards from 255 to 0 (88 D0 FD) and then jumping back to the start (4C 00 03). In other words, this short routine keeps a constant tone going on the speaker until you hit <reset> to get out of it. To start this program once you have typed in the above line, simply go to the location $300 and do what's there by the command:

```
*300G<ret>
```

You have just entered your first machine language program. Now go on to greater things. Grab an assembly/machine language listing and enter it into your computer. If you are entering an assembly listing, you will see the hex code in the left hand columns. For example, the above program looks like:

```
0300 - AD 30 C0 1 TONE LDA $C030
0303 - 88       2 LOOP DEY
0304 - D0 FD    3      BNE LOOP
0306 - 4C 00 03 4      JMP TONE
```

Remember to save your work on disk. This is done with the BSAVE command. It is necessary to specify the starting address and length parameters using the "A$" and "L$" commands. If a program went from 0300 to 0320, it would be (hex) 20 or (decimal) 32 bytes long. Therefore it would be saved by the command:

*BSAVE MYPROGRAM,A$300,L$20

where A$ is the starting address and L$ is the length of the program in hex. It could also be saved by the command:

*BSAVE MYPROGRAM,A768,L32

where A is the starting address in decimal (768 = 300 hex), and L is the length of bytes.

To check a program listing enter the start and end address, separated by a period. For the short TONE program above, enter:

*300.308<ret>

If you want to see the dissembled code, enter the starting address and "L" to list 20 commands. To see this program, enter:

*300L<ret>

(To make the entry and correction of machine language programs, one of the disks from the International Apple Core (Disk 25, Machine Language Utilities) contains a utility to facilitate this process. It is called BEU (for Binary Entry Utility and will locate itself on a 16K RAM card or in the Apple's 48K RAM.) It is an excellent utility.

2) Programming with the mini-assembler

In the original Apple II (that had Integer Basic, not Applesoft) machine there was a monitor function entitled "Mini-Assembler". It's job was to facilitate entry of machine code into the computer. When Applesoft was developed for the Apple II plus (and //e). this function (along with the STEP and TRACE in the monitor ROM) was deleted. That is the bad news. Now for the good news. If you have a 16K RAM card (or a //e), loading Integer BASIC upon booting up the DOS 3.3 SYSTEM MASTER disk will load the Mini-Assembler into your machine. This can be accessed by entering Integer with the command:

INT

which yeilds the ">" prompt. Then enter:

>CALL -151

to get to the monitor "*" prompt. Then go to the Mini-Assembler with the command:

*F666G

which yields the exclaimation (!) prompt.

The Mini-Assembler is still available to the owner of a 48K Apple II Plus. as it has been modified and included on the IAC Disk #25. Instructions for it's use are given in an accompanying file, while more complete instructions are given in the Apple II Reference Manual. Apple //e owners can access the Mini-Assembler through Integer BASIC.

3) Assembler

Once you've got your feet wet by gaining access to the monitorand entering a hex listing, you will realize the fact that there has to be a better way. Obtaining an assembler is that way. There are at least ten commercially available editor/assemblers, and the perfect one is probably still to be written.

Buying an assembler is like obtaining any other software, as the amount of money to be spent and the configuration of your machine will be controlling factors.

Other aids

Other types of aids are also worthy of consideration. A dissasembler is a handy tool which will permit you to take a machine language (hexadecimal) code and reconvert it into an assembler source listing. From there you can edit, alter and re-assemble the code to make it function in a different manner or at a new memory location. A disassembler is available in the Apple's monitor and is able to disassemble to the screen using the

*XXXX L<return>

command. where XXXX is the memory location at which to begin disassembly. A public domain disassembler is included on the machine language disk, and some very good disassemblers are included as part of the packege (eg. Merlin) Others are available as a stand alone product (such as DISASM from RAMWARE).

Debuggers are for those who are very serious - they permit step and trace function and a host of other advanved features. Two examples of commercial products are Bug Byter and Munch-a-Bug. BEU on the machine language disk also contains a debug routine.

There is even a new product which will turn your machine into a "visible" Apple as it shows you on the hi-res screen how your Apple works, displaying the instructions, registers, flages and memory locations as executes a program one step at a time.

And where would we be without books? A variety are available, each on for a slightly different reader. There are some general references on programming the 6502 microprocessors. several good books specifically written for Apple 6502 machine language. and books which serve as a road-map or atlas to the Apple's internals. But the best starting place is the Apple II reference manual which lists the opcodes available on the 6502 and some important internal routines.

When all is said and done. those who know a few BASIC commands can easily find there way around machine language. There is a certain joy in entering, debugging and running a program, be it in BASIC, assembler or whatever. And isn't that joy a big reason for owning an Apple?

[Advertisements]

FOR SALE Contact: Graham Black at the Trading Table or phone (07) 883-1525 after 12.00 noon (he does shift work)

1 x	Auto Ice Printer Card	$ 60.00
1 x	Zardax Word Processor (Dos 3.3 version)	$155.00
1 x	Speed Demon	$450.00
	Makes you Apple run 3.6 times faster than normal - excellent with Data Bases and Spreadsheets.	
1 x	Roland 800 DXY Plotter (8 colour)	$995.00
	Used A3 paper and there are lots of pens to go with it.	
1 x	Epson LX-800 printer	$495.00

1 x Alf Music Synthesizer $450.00
Includes AIWA amp. with over 30 disks of music.
1 x Super Serial Card (new and sealed) $140.00
1 x IBM XT TURBO $2500.00
Includes: Samsung Amber Monitor
30 meg. Miniscribe Hard Disk
two 360K floppy drives
two Serial Ports
Parallel Port
Mouse
Keyboard
Mulitfunction card
plus several manuals

FOR SALE Contact: Vince Crosdale at the Trading Table or phone (07) 355-3090

1 x Enhanced Apple IIe $1900.00
Includes: 512K of RAM
Apple Green screen monitor
two disk drives
Parallel printer card
RGB card
manuals and much software

FOR SALE Contact: Bob Clews, phone (079) 857415

1 x Apple //c (128k RAM and inbuilt drive) $1700.00
Includes: //c Monitor and stand
//c External Drive
Apple Scribe Printer
Maestro Supermodem 300/300, 1200/75 baud
Apple Joystick
Apple Mouse
Complete set of Manuals
Programs: Type-Quick, Terminapple, Viaterm, Appleworks, Apple Pascal
plus approx. 120 double sided disks of software
All in perfect condition.

FOR SALE Contact: Dale Rodgie, phone (075) 386942

1 x Flight Simulator (disk only) $5.00
1 x ProDOS Assembler 3.0 $30.00
1 x Apple Secrets (manual and disk) $20.00
1 x Music Construction Set (Apple //e version) $30.00
1 x Newsroom (includes back-up disk) $40.00
1 x Apple Mechanic $20.00
1 x Alpha Plot $20.00
1 x Machine Language for Beginners (manual) $15.00
1 x ProDOS Technical Reference Manual (inc.disk) $25.00
1 x Microsoft Multiplan (spreadsheet) $50.00

[Next Month]

In the next edition of Apple-Bug, we have the long awaited reviews of TML Basic and Copy II+ version 8. We also will announce the winner of the Logo Competion and more Apple news from around the world. Also we have review on a new communications program called DataWorks. We look at some of its many features. It is not available commercially as yet, but Apple-Bug gets the scoop. Read it all in the July edition of Apple-Bug - comming to a letter box near you.....

REGISTERED PUBLICATION

(Bundle)
For Post-town or Area

Post Code (or Group)

APPLE-BUG

The Newsletter of Apple-Q Inc. the Brisbane Users' Group
If not claimed within 7 days, please return to:
P.O. Box 721, South Brisbane Qld. 4101, Australia.

Please deliver to ->

POSTAGE
Paid
Chermside
Qld.Aust.
4032

Registered by Australia Post - Publication No. QBG 3485

ADVERTISING RATES

FREE: to all Financial Members

$18.00 per Half Page

$30.00 per Full Page