Crowds flock to the club's booth at the Pacific Coast Computer Fair held September 18th and 19th.

Jim Butterfield dispenses his wisdom to CCC members at Killarney auditorium.

Don checks program for possible copyright infringement.

Hu solicits new membership applications; Kim prays no one will lean on his plexiglass display case.

Jim Butterfield faces barrage of inquiries after his lecture.

Vince demonstrates program for throngs surrounding the club booth.

PHOTOS BY MIKE QUIGLEY

WE'RE ON THE MOVE!

By HU REIJNE

Those who have been attending the club meetings for a while have seen us outgrow the facilities at Sunset. This place has served us well over the years, but with the growth in members after the VICs and 64s came out, we started looking for a larger place. The new place for the general meetings will be the King Edward Campus of Vancouver Community College, located at 1155 East Broadway, near Clark Drive.

The auditorium holds 250 people. It is located on floor 2 of the main building, near the southwest corner. To get there, enter from the north on the second floor. The Broadway entrance is not open in the evening. Parking is available in the STUDENT parking lot. This is north of 7th Avenue and requires a block of walking. DO NOT park in the faculty parking lot as you will be towed away. Street parking is also available.

We are able to get this facility on a Tuesday night. Because this is our traditional meeting night, we have worked to retain it. If this facility works out we will make a long term booking.

ALL THE NEWS THAT FITS...

We have a lot of brief news items for you this issue, so rather than make up a bunch of separate stories, we decided to lump them together in one batch.

‡‡‡

At the club's annual general meeting to be held in January, elections for all the club's major positions -- President, Vice-President, Secretary, Treasurer, and so forth -- will be held. Now is the time to start thinking of people to nominate. Nominations will be heard at the December general meeting, and voting will take place the following month. It should be stressed that assuming any of these executive positions does not require any knowledge of computing!

‡‡‡

While the club's general meetings, held on the 3rd Tuesday of the month, will hopefully be held regularly at King Edward Campus, the workshop meeting will be moving back and forth during the next few months. The November meeting will be at David Thompson Secondary School, while the December meeting will hopefully be back

at Killarney. In any case, now is the time to remind you to keep in touch with the club through its answer phone: PET-3311 (738-3311). It will provide you with a recorded message 24 hours a day, and will also take any messages or questions you may have about the club.

‡‡‡

There is some doubt as to whether Jim Butterfield will be at the November general meeting as previously announced. Please call the club's phone to confirm details.

The Club's Executive reminds members that solicitation of goods or services at club meetings is a no-no without prior approval. Terms and conditions of any such offers must be presented in writing to the group through the club executive or a designated representative at least one week prior to the announcement of the offer at a club meeting.

Also forbidden is the copying of commercial or copyrighted software at

meetings. Members caught doing so are subject to dismissal from the club.

‡‡‡

It has been requested that the newsletter offer a service to members who wish to sell, swap, or buy software and equipment in the form of unclassified ads. OK! So we're doing that starting with this issue. If you have anything to contribute, please submit them, either by mail to the club or to the editor or members of the executive at any of the club's meetings. We reserve the right to edit these ads for taste, length, and so forth, and to reject any that are deemed too commercial. (You can always take out a paid ad.)

‡‡‡

Speaking of ads, you probably notice the lack of same in this issue of the newsletter. That's why it's only four pages long, unlike the Computer Fair special, which was 16. Ads help offset the cost of the paper, and are available in sizes ranging from 1/12 of a page to a full page, at reasonable rates. Call PET-3311 for details.

SIMONS' BASIC — A Review

By DAVE WHITE

My love-hate relationship with Simons' BASIC began in the summer of 1983. I had strolled into the premises of one of our local PC vendors and inquired as to the availability of new higher level languages for my 64. "You might take a look at Simons' BASIC." So I did.

I examined the manual for the better part of two hours before deciding to part with $100, plus tax, in exchange for 114 new BASIC commands. I rationalized, "Just think of it this way -- you're paying less than a dollar for each command!!!"

Simons' BASIC comes as a ROM cartridge and there is a sizeable manual, which is beautifully formatted. No comic strip pictures -- just the meat of what Simon does, with examples. It is easy to read, with clear, large print.

Before examining Simon in detail, it may well serve us to digress for a moment to discuss the question, "What is to be expected of a higher level language?" I have found that I can make my 64 do wondrous things, particularly in graphics and sound. I could not count the hours -- nay, days -- that I have spent happily building routines to set up high resolution graphics, plot points, draw circles, you name it. Of course, the routines were usually painfully slow, being written in BASIC. So I swatted up my machine code, got out the manuals, and did it all over again in machine language. A bit "unpolished" here and there, but they worked, and it was fast! Now I find that these excursions are an enjoyable, even essential, part of programming. But there comes the time when one just wants to get the job done. And here lies the power of a good high level language. It will do the bookwork for you, better and faster than you could do it yourself. You are now free to devote your time to the program proper. Let's look at Simon to see whether he gets the job done.

Simon introduces so many new commands that it will be possible to examine only a few of them. I shall talk mostly of graphics, my favorite.

To get a two-color high resolution graphics screen (that's 320 pixels by 200 pixels) painting white on black:

    10 HIRES 1,0

To get multicolor hi res, with 3 available plotting colors of green,

blue, and yellow:

    15 MULTI 5,6,7

Now, you say you want a blue circle of radius 50, in the center of your screen? That's easy!

The center is at X=80 Y=100. We will need to define radii for both the X and Y directions, since our circle statement can also be used to define an ellipse. Set RX=RY=50. Now, we want it to be blue. That's a 6, and 6 was the second color in our multi statement. So we use 2. Here it all is:

    CIRCLE 80, 100, 50, 50, 2

You want it filled in yellow? OK. I need to pick any coordinates that fall inside the circle. I'll use 80,100. Yellow is the third color in the multi statement.

    30 PAINT 80, 100, 3

We can also plot points, lines, arcs and radii for pie charts or wheels. We can draw rectangles and "shapes". Shapes are neat. They are a bit like sprites and can be displayed at any magnification and also rotated. We can put letters or strings on the hi res screen.

There is an entire set of commands for sprites. We can use as many colors as we wish. To bring three new colors to our palette:

    40 LOW COL 8,9,10

This gives the use of yellow, orange, and brown while still retaining colors already painted. We can also get dramatic effects by switching, in the twinkling of an eye, all colors painted so far. Just give a new multi statement, or even several in quick succession. We have here the makings of a disco light show!

Simon has many commands which just sort of make life easier for the programmer. Many of these are related to structured programming. IF...THEN now has an ELSE; LOOP...END LOOP and REPEAT...UNTIL are two more kinds of loop.

Labels can also be used -- CALL LABEL-1 replaces GOTO 420, and EXEC SUB 3 replaces GOSUB 1800, for example.

There is a fistful of commands for sound and string handling. I really could go on and on, but will have to settle for a list of a few of the many more commands: MERGE (to either merge or append one program to another);

TRACE; FIND (searches for a given string in the program); PRINT USING (edits a numeric field); COLD START; DUMP (displays values of all variables -- except for arrays); RENUMBER (renumbers all program lines).

At this point the reader might well be asking, "Why does he call this a love-hate thing? Seems all love to me." Well, that's because we've only talked about the good times. Simon has its faults.

OPTION 10 is a statement which has brought me a lot of grief. It causes Simon keywords to be highlighted in reverse video. I find this effective and use it all the time.

OPTION 0 turns the highlighting off -- that is, it seems to be off. The listings no longer show highlighting, but (alas) attempts to get a 1525 printer listing will fail! The only way I know of to recover from this program is to save the program, issue the COLD command (equivalent to power off/on), and RELOAD the program. As a result of this bug I had, for weeks, believed that printer listings were impossible.

Graphics commands related to color are poorly documented. It is, in fact, difficult to believe that the person who did the documentation fully understood how the colors work! The LOW COL command (to redefine color) doesn't always work. It is, though, pretty easy to predict when it will fail.

RENUMBER would be more effective if it updated GOTO and GOSUB line numbers.

A couple of less frequently used commands will freeze the system when encountering invalid parameters (as program line is entered).

I am not sure just how the ROM cartridge works, but I imagine that one could "patch" these errors wit a bit of machine language code in the right place. If so, I shall be looking for Simon, Version 6411.

Did Simon get the job done? I think so, and in spades. On a scale of one to ten, I'd give Simon an 8.6, not bad for the creation of a 16-year-old Englishman!

I would very much appreciate your comments on queries, or even just a call (224-3082) to tell me you also have Simons' BASIC.

■ -- COMPUTERS AND HAM RADIO -- ■

BY Hu Reijne, VE7CHW

In this second part of computers and ham radio I would like to go on about using the computer as a terminal to send and receive Morse code, Radio teletype signals and other uses around the ham station.

There are several programs around that have been published in magazines etc. that will generate morse code on a computer. One of the things holding many people back from getting their ham radio license is learning the code. Morse code is a requirement by international convention to obtain your ham licence. The speed that you have to copy is 10 or 15 words per minute depending on the class of

license you are writing. To get proficient at this you require a lot of practice. This is where the computer comes in handy. A fairly simple basic program can pound out random code at up to 20 words per minute with no sweat. Not only that the computer is a faithfull servant and will keep this up until you are tired of it.

The way it works is you generate a random number on your PET than use an algorithm to convert this to an ASCII letter or number which is printed on the screen. In another part of the program is a table that is looked up and the correct combination of dits and dahs are sent out.

After being on the air for a

LET'S SHARE, NOT SWAP

By JIM BUTTERFIELD

Occasionally, I run across someone who has an attractive program. And when I ask, "Is that public domain? May I have a copy?", I get the reply, "What can you swap me for it?" My answer: "Nothing. All my programs are in the TPUG library." So I don't get a copy of the program.

This amazes me. The other person may have dozens -- or hundreds -- of my programs. But I'm not going to get the new program, because I have nothing to swap.

The whole swapping thing makes no sense to me. The name of the game is sharing, not swapping.

Let's look back at the origins of the club. Suppose I -- and several other programmers -- had said to TPUG, "You don't get programs from us unless you can swap us something equally good". Suppose that TPUG said to its members, "You don't get a program until you submit a program of equal quality". We'd have a pretty weak operation. User groups don't work that way. Thank heavens.

I fear that the swap syndrome encourages program theft.

Some poor beginner who isn't skilled in program writing is coerced by swappers into giving a program as a swap. What is he or she going to give? The pressure is to buy a program and give away a copy. And that's wrong, wrong, wrong.

Sometimes I send people programs. I usually refer them to the club, but occasionally I need to send a program or two directly. I don't expect anything in return; in fact, sometimes my return address isn't on the package. Some people reply and say, "Thank you", which is OK. On a couple of occasions, people have replied by sending me bootleg copies of commercial programs. They shouldn't do that. I have a feeling that these people have been brainwashed into the "swapping" thing. They think that they must give something in return -- even if it's illegal . They shouldn't.

Let's get off this swapping bandwagon.

Any programs I have, provided they are not copyrighted or commercial, are freely available to anyone who wants them. They are in the club library, for that matter.

How about your programs? Surely you don't think that they are too good for the club? Throw them into the pot -- make them available.

The whole business of having a club is to share ideas, experiences. Let's share -- not swap.

-- from *TORPET*, September

(CONTINUED FROM PAGE ONE)

The club executive is pursuing the idea of regional meetings in areas like Richmond, Langley, etc. Such meetings will need people to organize and run them, so volunteers along these lines will be welcome.

$$$

The newsletter needs your contributions -- articles, reviews, or what have you. No newspaper experience is necessary!

---

64 LIBRARY UPDATE

More confusion from the 64 library. This time I mean to mess the disk ID's up. Remember when there was only ONE disk in the 64 library? This disk was called DISK #1. Then came DISK #2. That started the confusion. DISK #1 was ID 01 on side one and ID 02 on the back side. DISK #2 had ID 03 on the front side and ID 04 on the back. Have I confused you? Well, if not you then I did it to some people. They'd ask for ID 02 when DISK #2 was the one they wanted.

With hope of curing this problem I've changed the ID's on the library discs. The new ID for DISK #1 is A1 (front side) and A2 (back side). DISK #2 is B1 (front side) and B2 (back side). Does this look like a cure? I hope so.

What is new, you may ask? Good question. Well, I did some sorting of the material received from OTTAWA and came up with two double sided disks full of new material we didn't have before. So the total 64 library has 6 disks recorded on both sides. Yes, look on the back side for goodies.

Disks available (when returned) are A, B, C, D, E and F. New ID's are A1, A2, B1, B2, C1, C2, D1, D2, E1, E2, F1, F2. See me for the 64 library (on a meeting night).

-- Glenn Hazlewood

```
"CCC LIBRARY DISK" A1 2A
1  "C-64 WEDGE"        PRG   23 "HANGMAN"            PRG
24 "BLACKJACK"         PRG   28 "SOLITAIRE"          PRG
14 "TARGET PONG"       PRG   27 "BOMBAIN"            PRG
23 "ANDROID NIM 64"    PRG   28 "YAHTSEE"            PRG
38 "ATARI II.C2"       PRG   42 "PERSONALITY TEST"   PRG
28 "ENTERTAINER"       PRG   43 "ARTILLERY.C2"       PRG
37 "LABYRINTH"         PRG   2  "AQUA 2"             PRG
13 "KEYBOARD"          PRG   7  "TIME GUESSER"       PRG
24 "DISK DOCTOR"       PRG   12 "1 SIN+"             PRG
17 "PETALS/ROSE"       PRG   91 "AFRICAN ADVENTUR"   PRG
78 "CASTLE ADVENTURE"  PRG   4  "DOS 5.1"            PRG
16 "MAXIT"             PRG
87 BLOCKS FREE.
```

```
"CCC LIBRARY DISK" A2 2A
1  "C-64 WEDGE"        PRG   11 "DOS IN BASIC"       PRG
4  "DOS 5.1"           PRG   11 "RES MONT"           PRG
4  "HIRES VIEW 2"      PRG   89 "MUSIC MAKER"        PRG
2  "HIRES VIEW 2.MAC"  PRG   26 "GRUNBY TOWERS"      PRG
32 "1-DOLLAR"          PRG   88 "SUPERTREK 64+"      PRG
32 "2-MAP"             PRG   9  "SUPERDOS207000"     PRG
32 "3-DBL BESSEL"      PRG   3  "SPRITE BOOT"        PRG
32 "4-KAREN"           PRG   43 " SPRITE EDITOR"     PRG
33 "5-DIP"             PRG   2  " SCROLL.DATA"       PRG
28 "FRENCH 64"         PRG   3  "SAMPLE SPRITES"     PRG
9  " STANDARD.SET"     PRG   33 "SPRITE INSTR."      PRG
28 "SINGLE BACKUP"     PRG   4  "CHAR BOOT"          PRG
8  "MOUSE"             PRG   32 " CHAR EDITOR"       PRG
9  "SPRMON.SYS51201"   PRG   1  " ROTATE.DATA"       PRG
72 BLOCKS FREE.
```

```
"CCC LIBRARY DISK" B1 2A
28 "AFRICA & ASIA"     PRG   12 "US PRESIDENT QUI"   PRG
12 "EASY ADD & SUBTR"  PRG   12 "WORLD CAPITAL QU"   PRG
15 "20 QUESTIONS"      PRG   6  "TYPING DRILL"       PRG
6  "ADDITION GAME"     PRG   18 "VOCAB 1"            PRG
2  "ASK"               PRG   13 "ANIMAL"             PRG
11 "BIG MATH 1.1"      PRG   27 "READER"             PRG
9  "CASH REGISTER"     PRG   13 "MATH.QUIZ"          PRG
8  "CRYPTOGRAM"        PRG   10 "ANIMAL.DATA"        SEQ
8  "DONUTS"            PRG   11 "NOT.SO.EASY"        PRG
24 "ELIZA"             PRG   13 "SPELLING.TEST"      PRG
18 "EUROPE"            PRG   14 "G'S & I'S"          PRG
6  "FRACTIONS"         PRG   23 "DISPLAY"            PRG
11 "FRENCH VERBS"      PRG   4  "HISTOGRAM"          PRG
8  "GRAMMAR"           PRG   27 "EDU-TILITIES"       PRG
13 "HAIKU 5"           PRG   16 "EUROPEAN CAPITOL"   PRG
11 "HANGMAN"           PRG   26 "FISHERY"            PRG
11 "HANGMAN(HJS)"      PRG   8  "FLASH CARDS"        PRG
7  "MATH DICE"         PRG   10 "FRENCH"             PRG
13 "MATH TUTOR"        PRG   6  "GLOBAL"             PRG
7  "MISSING NUMBER"    PRG   23 "ICE CREAM PAR"      PRG
9  "NO"                PRG   26 "MATH IQ"            PRG
15 "PETS"              PRG   27 "HANGMAN 2"          PRG
19 "SPELLING BEE+FIL"  PRG   24 "JUMBO JET"          PRG
19 "STATES&CAPITALS"   PRG   28 "SINGLE BACKUP"      PRG
11 BLOCKS FREE.
```

```
"CCC LIBRARY DISK" B2 2A
42 "SPACE PILOT"       PRG   12 "XMON64L(BASIC)"     PRG
42 "SNAKES"            PRG   12 "XMON64C(50138)"     PRG
14 "P&L PLOTTING DEM"  PRG   12 "XMON64M(29700)"     PRG
10 "SPRITE DEMO"       PRG   12 "XMON64T(30003)"     PRG
9  "3-D DEMO"          PRG   10 "SHOOT"              PRG
2  "M/L DRAW DEMO"     PRG   5  "SEQUENTIAL FILE"    PRG
68 "STAR TREK IV-A"    PRG   13 "RANDOM FILE"        PRG
38 "SPRITE MAKER"      PRG   19 "DISASSEMBLER DIS"   PRG
38 "ATARI"             PRG   16 "DISK CHECKER"       PRG
44 "MASTER MIND"       PRG   7  "DISK VIEWER"        PRG
2  "PETSIM"            PRG   9  "DISK LOGGER"        PRG
13 "TRACK & SECTOR"    PRG   6  "SECTOR PRINT"       PRG
8  "MENU"              PRG   7  "DSK VIEW/CHANGER"   PRG
2  "   DIRECTORY    "  SEQ
199BLOCKS FREE.
```

```
"CCC LIBRARY DISK" C1 2A
79 "PONIO TUTOR-1.64"  PRG   4  "SAVE FILE(S) PRG"   PRG
67 "PONIO TUTOR-2.64"  PRG   4  "SAVE FILE(S) SEQ"   PRG
74 "PONIO TUTOR-3.64"  PRG   28 "SINGLE BACKUP"      PRG
64 "PONIO TUTOR-4.64"  PRG   33 "SPRITE INSTR."      PRG
67 "PONIO TUTOR-5.64"  PRG   10 "64 MEM CHART.SR"    PRG
61 "PONIO TUTOR-6.64"  PRG   3  "INDEX HARDCOPY"     PRG
68 "PONIO TUTOR-7.64"  PRG   3  "INDEX HARDCOPY 2"   PRG
13 "LISTER.V64"        PRG   8  "INDEX HARDCOPY 3"   PRG
4  "1525 CHAR.EDIT"    PRG   19 "WORLD CLOCK"        PRG
7  "TIME VEN INST"     PRG   11 "SUPER CHASE"        PRG
7  "TIM VEN SET UP"    PRG   25 "STARTREK"           PRG
18 "TIME ADVENTURE"    PRG
8  BLOCKS FREE.
```

```
"CCC LIBRARY DISK" C2 2A
3  " LIBRARY NOTES "   PRG   4  "ADVF 3"             SEQ
7  "SCRLL.LDR.TAPE"    PRG   5  "ADVF 4"             SEQ
46 "SPRT.EDIT.TAPE"    PRG   3  "ADVF 5"             SEQ
25 "SIDMON"            PRG   8  "ADVITM"             SEQ
36 "SOUNDER"           PRG   5  "ADVKEYS"            SEQ
11 "DOS IN BASIC"      PRG   18 "ADVMAP"             SEQ
88 "SUPERTREK 64+"     PRG   15 "ADVSNOR"            SEQ
48 "ADVENTURE-64"      PRG   2  "HIRES VIEW"         PRG
9  "ADVF 0"            SEQ   32 "1-DOLLAR"           PRG
5  "ADVF 1"            SEQ   32 "2-MAP"              PRG
4  "ADVF 2"            SEQ   32 "3-DBL BESSEL"       PRG
3  "ADVF 20"           SEQ   32 "4-KAREN"            PRG
7  "ADVF 21"           SEQ   33 "5-DIP"              PRG
6  "ADVF 22"           SEQ   22 "DISK UTILITIES"     PRG
5  "ADVF 23"           SEQ   2  "COMM64"             PRG
6  "ADVF 24"           SEQ   56 "BACKGAMMON"         PRG
7  "ADVF 25"           SEQ   4  "PARTITION LOADER"   PRG
11 "ADVF 26"           SEQ   5  "PARTITION /4"       PRG
6  "ADVF 27"           SEQ   13 "SPEED TYPE 64"      PRG
5  "ADVF 28"           SEQ   12 "1 SIN+"             PRG
1  "ADVF 29"           SEQ
0  BLOCKS FREE.
```

```
"CCC LIBRARY DISK" D1 2A
33 "ROM 1"             PRG   7  "64 SEARCHER"        PRG
33 "ROM 2"             PRG   4  "ULTRASORT TEST"     PRG
1  "KERNAL BOOT2"      PRG   33 "ULTRASORT"          PRG
1  "TRANS.A000-A"      PRG   1  "C-64 WEDGE"         PRG
1  "KERNAL BOOT1"      PRG   4  "DOS 5.1"            PRG
5  "DISK MENU"         PRG   2  "DOS INSTR."         PRG
15 "DISK BACKUP 1541"  PRG   45 "MONTANA"            PRG
20 "BIORHYTHMS"        PRG   94 "KANON"              PRG
28 "MUNCHMAID"         PRG   26 "STAR PILOT"         PRG
21 "OIL TYCOON"        PRG   14 "SKI-64"             PRG
42 "WUMPUS HUNT"       PRG   23 "DARK STAR"          PRG
21 "JAPANESE AFO"      PRG   27 "MARTIAN INVADERS"   PRG
4  "UNNEW"             PRG   56 "FOOTBALL64"         PRG
12 "FILECOPIER INSTR"  PRG   13 "1 SIN+"             PRG
16 "FILECOPIER"        PRG   3  "STRING THING 64"    PRG
4  "ALARM CLOCK"       PRG   5  "ENTERPRISE C64"     PRG
3  "AUTO PROOFREADER"  PRG   24 "CONSTRICTOR.4"      PRG
14 "DISASSEMBLER 64"   PRG   8  "EXPANDER.4"         PRG
3  "RAMTEST/MACHINE"   PRG   3  "SLOW LISTER"        PRG
5  "FOUR-SPEED BRAKE"  PRG
6  BLOCKS FREE.
```

```
"CCC LIBRARY DISK" D2 2A
13 "C-64 FILE CASE"    PRG   27 "TOKEN64"            PRG
9  "LOCKDISK64"        PRG   22 "AFO"                PRG
6  "CROSS-REF64"       PRG   6  "CROSS-REF"          PRG
10 "COPY-ALL.64"       PRG   18 "SUPERMON64.V2"      PRG
109"BACH FUGUE.C"      PRG   12 "LISTER.V64"         PRG
14 "SPRITE MANIP.64"   PRG   10 "COPY-ALL"           PRG
24 "MUSIC MASTER"      PRG   30 "TELCOM333.4.17.4"   PRG
25 "YESTERDAY.C"       PRG   30 "TELCOMM818.4.8.4"   PRG
17 "HIRES GRAPHICS"    PRG   11 "TELCOM.INST.4"      PRG
1  "PET EM"            PRG   4  "MEMBER"             PRG
6  "SCREEN PLOTTER"    PRG   8  "RS323.INSTR.4"      PRG
98 "MONOPOLE.C"        PRG   6  "FACTORS.VIC64"      PRG
14 "SUBMARINES.C"      PRG   18 "BLACKJACK!"         PRG
27 "MASTER"            SEQ   6  "BIRTHDAY.64"        PRG
14 "COWBOY SHOOTOUT"   PRG   12 "TWIN BAGELS"        PRG
11 "RES MONT"          PRG   14 "BATTLESHIPS"        PRG
1  "PET EMU BOOT"      PRG   9  "LINCOLNSHIRE"       PRG
17 "EMULATOR"          PRG   2  "COPY FILE 64"       PRG
4  "555 TIMER DESIGN"  PRG
11 BLOCKS FREE.
```

```
"CCC LIBRARY DISK" E1 2A
5  "CONT.LDR.NL"       PRG   32 "EYES"               PRG
9  "HI RES LOADER"     PRG   32 "FRIENDS"            PRG
1  "SCREEN"            PRG   32 "SIMCOS2"            PRG
7  "CONT.LDR.PAL"      PRG   32 "SATELLITE"          PRG
32 "SQUEEZIE"          PRG   32 "DONALD.DUCK"        PRG
32 "TEX"               PRG   32 "VM.THINGS"          PRG
32 "HOPALONG"          PRG   32 "SESAME.ST"          PRG
32 "SUY"               PRG   32 "NUDE.REV"           PRG
32 "FIG1"              PRG   32 "VIS.ROSETTE"        PRG
32 "FIG2"              PRG   32 "VIS.S1A"            PRG
32 "FIG3"              PRG   32 "IMAG.CARD.1"        PRG
32 "HOUND2"            PRG   32 "4HILLS"             PRG
2  BLOCKS FREE.
```

```
"CCC LIBRARY DISK" E2 2A
1  "LIST-ME CS1.L"     PRG   32 "DES.1"              PRG
5  "CONT.LDR.NL"       PRG   32 "7-SHILL"            PRG
8  "HI RES LOADER"     PRG   32 "MUSIC"              PRG
1  "SCREEN"            PRG   32 "MAP"                PRG
6  "CONT.LDR.PAL"      PRG   32 "DIANE"              PRG
32 "SPIRAL.1"          PRG   32 "WILLY"              PRG
32 "SUE"               PRG   32 "RACCOON"            PRG
32 "KAREN"             PRG   32 "SIMCOS1"            PRG
32 "SNOOPY"            PRG   32 "WATCH"              PRG
32 "ALBERT"            PRG   32 "WINSTON"            PRG
32 "DOLLAR"            PRG   32 "MICROMETER"         PRG
32 "DIP"               PRG   32 "NUDE"               PRG
32 "SNAIL"             PRG
3  BLOCKS FREE.
```

```
"CCC LIBRARY DISK" F1 2A
47 "BITS AND BYTES"    PRG   52 "DISKVIEW-64"        PRG
5  "BYTESPRITES"       PRG   13 "MINIWORDPRO.64"     PRG
1  "DOS.BOOT.64"       PRG   20 "LIST"               PRG
4  "+DOS 5.1.64"       PRG   18 "ARTILERY"           PRG
2  "+DOS.INST.64"      PRG   26 "BACKGAMMON"         PRG
37 "PILOT INST.64"     PRG   13 "BUGS 64"            PRG
13 "PILOT TRANSL.64"   PRG   33 "CAVES & CAVERNS"    PRG
4  "DISK ADDR CHANGE"  PRG   12 "CHECKERS"           PRG
11 "COPY/ALL"          PRG   24 "CONCENTRATION 64"   PRG
13 "HOW TO USE"        PRG   12 "HAMMURABI"          PRG
5  "HOW PART TWO"      PRG   8  "HUKADE"             PRG
1  "C-64 WEDGE"        PRG   31 "OTHELLO"            PRG
4  "DOS 5.1"           PRG   16 "PI HUNT 64"         PRG
9  "PRINTER TEST"      PRG   16 "PI HUNT 64.2"       PRG
6  "VIEW BAM"          PRG   24 "YAHTZEE"            PRG
4  "CHECK DISK"        PRG   33 "HUSHAMPS"           PRG
14 "DISPLAY T&S"       PRG   56 "DISK CHDS"          PRG
9  "PERFORMANCE TEST"  PRG   67 "NUCLEAR DEMO"       PRG
5  "SEQUENTIAL FILE"   PRG   7  "SABBER"             PRG
13 "RANDOM FILE"       PRG
1  BLOCKS FREE.
```

```
"CCC LIBRARY DISK" F2 2A
5  "BOOT.LRC2.64"      PRG   58 "QUIET AFTERNOON"    PRG
25 "+SPRITES.64"       PRG   43 "B-BACK CHALLENGE"   PRG
3  "+MUSIC.64"         PRG   8  "GUESS A GRAPH"      PRG
7  "+LAND.64"          PRG   8  "SAFE CRACK"         PRG
71 "+DEMO.64"          PRG   6  "TORPEDO"            PRG
18 "1000 MILES/INST"   PRG   52 "ZOMAIE"             PRG
28 "1000 MILES"        PRG   9  "FIFTEEN.64"         PRG
13 "1 SIN+"            PRG   14 "OUTPOST 64"         PRG
79 "STAR TREK-C64"     PRG   23 "IT"                 PRG
23 "DRAW POKER"        PRG   53 "SNOOPY MATH"        PRG
23 "CARD SNAP"         PRG   44 "LONE(64)RANGER"     PRG
23 "CLUE"              PRG   2  "ALPHABET TYPE"      PRG
17 "SPACE NIM"         PRG   3  "TEMPERATURE CONV"   PRG
14 "DUCKSHOOT"         PRG
2  BLOCKS FREE.
```

A COMPLEAT GUIDE TO MACHINE LANGUAGE PROGRAMMING ON THE PET

(Continued from last issue)

By HAROLD BROCHMANN

A FEW MORE POINTERS [2-1]

We have seen that bytes 42 and 43 contain the pointer to the index which lists the variables defined in a program. There are a number of other pointers, some of which we will look at now.

PRINT PEEK (41)#256 + PEEK(40) yields 1025. These two locations contain the pointer to the START OF BASIC. In the next chapter we are going to see how this pointer can be altered so as to re-locate BASIC!

PRINT PEEK(53)#256 + PEEK(52) will produce different results depending on what size memory your PET has. This is the pointer to the TOP OF AVAILABLE RAM.

Locations 54 and 55 contain the line number currently being executed by a program. Try running this one:

Locations 60 and 61 contan the line number of the DATA item that is being read. This program will illustrate:

```
10 DATA 5
20 DATA 6
30 DATA 7
40 FOR X=1 TO 3
50 READ Y
60 PRINT PEEK (61)#256 + PEEK(60)
70 NEXT
```

Bytes 62 and 63 contain the byte address of the DATA item currently being read.

Locations 196 and 197 contain the pointer to the start of the line where cursor is located.

This is not an exhaustive list, but serves rather to illustrate the nature of pointers.

ASSIGNMENT 2-1

Write programs that illustrate the function of the DATA and the CURSOR LINE pointers.

Can you think of ways in which it might be useful for a program to peek or poke these locations?

FLAGS [2-2]

A flag is a byte which indicates whether or not a certain condition is true. So, for example, 152 is the SHIFT KEY STATUS flag. Try the following one-liner:

```
10 PRINT PEEK(152): GOTO 10
```

While this program is running try pressing the SHIFT key down to observe the results.

Location 167 contains the CURSOR ENABLE flag. Try PRINT PEEK(167). The zero tells us that the cursor is enabled when we are in keyboard mode.

```
10 PRINT PEEK (167): GOTO10
```

yields a flow of 1's. This means that the cursor is disabled when a program is running. Try:

```
10 POKE 167,0
20 GOTO 20
```

OTHER UTILITY AREA LOCATIONS [2-3]

Try this program:

```
10 PRINT PEEK(151): GOTO 10
```

Location 151 contains the code for the key which is currently being pressed. When no key is pressed we get the code 255.

BASIC 3 and BASIC 4 PETs use a different code in location 151. In BASIC 4 machines the PETSCII code for the letter pressed is used. In BASIC 3 computers a totally different code is used.

Quite a few PET programs make use of location 151 as part of their input routines. Many problems were created when Commodore changed this code!

THE KEYBOARD BUFFER [2-4]

Whenever a key is pressed the PETSCII code for that key is stored in the KEYBOARD BUFFER which is located in bytes 623 to 633.

When we are in keyboard mode each keystroke is temporarily stored in 623, and immediately moved on to the screen. When a program is running, however, the keystrokes are stored in 623 and up. They become printed on the screen when the program stops. To illustrate this, run the following program:

```
10 FOR X=1 TO 5000:NEXT
```

While the program is running, press several keys. They become printed out on the screen when the program stops.

The number of keystrokes stored in the keyboard buffer at any one time is kept track of in location 158.

```
10 PRINT PEEK(158):GOTO 10
```

Again, while this program is running, press several keys to observe the effect.

When the GET statement is executed in BASIC, the keystrokes are "gotten" from this buffer.

HEXADECIMAL NUMBERS [2-5]

So far we have been referring to byte addresses using the familiar BASE 10 or DECIMAL numbering system. This has yielded some rather awkward numbers like 32768.

The PET's memory is actually organized according to quite a different numbering system -- the HEXADECIMAL system. These are numbers which are BASE 16. HEX numbers are normally differentiated from decimal numbers by placing a # sign in front of them.

It may not be immediately obvious at this point why people who design computers like to work with HEX numbers; but it is nevertheless so, and if we are to explore machine language at all we are going to have to become familar with them.

Decimal numbers make use of ten digits 0-9. Hexadecimal numbers must have sixteen digits available. These are 0,1,2 ... 9,A,B,C,D,E and F.

Here is a partial list to illustrate:

DECIMAL	HEXADECIMAL
1	#01
2	#02
3	#03
.	...
10	#0A
11	#0B
12	#0C
13	#0D
14	#0E
15	#0F
16	#10
etc.	

CONCLUSION [2-6]

In these first two chapters we have explored a variety of topics as a prelude to the remaining material in this book. Hopefully it has given the reader a slightly better insight into how the PET operates and he should now be in a better position to tackle machine language.

(Continued in next issue)

COMPUTERS & HAM RADIO (From page 2)

while you want to try sending and receiving teletype signals with the computer. As these come at you pretty fast a machine language program is required. Some of these are quite sophisticated in that they provide a split screen. On one half of the screen you receive the signal, on the other half you can prepare your next transmission.

The radio is connected to the user port through a terminal unit (modem) which converts the signals to frequency shift keying to go out on the air. Some of these terminal units also allow you to copy Morse code. With a computer Hams send Morse at a speed of 60 or sometimes even 100 words per minute. The best most can do by hand is maybe 30 wpm. so there is quite a speed advantage!

New radio equipment being manufactured now has a computer interface that allows the control of the radio by computer. You can turn it on, change frequency, adjust passband, change modes and do everything that you used to do from the front panel.

As development in electronics continues these two hobbies will continue to advance in the sophistication of their interfaces. Software development and continued reductions in computer costs will see further interfacing between the two pieces of electronic gear. Ham gear continues to make more use of processors all the time as this provides much better control than the older forms of electronics at a lesser cost.

So if you get tired of writing and debugging programs at two in the morning, here is a way to vent your frustrations at someone else who maybe listening in Europe, Australia or even Russia. Give ham radio some thought, it too is a great hobby.

#####################################

CCC UNCLASSIFIED ADS

#####################################

#####################################