The Greatest Show on Earth...

Last month Tom, Grady, and I showed our stuff at the Long Beach Rainbowfest. We had a booth at the end of an aisle, which we expanded to include the whole end of the aisle. As people came in, we snagged 'em! Brought them over to the table, forced them to look at Chromasette and SilverWare, showed them the prices (wow, good deal!) and threatened to have Tom follow them all over the convention unless they bought something. When Tom turned and gave them a little smile, they bought...

```
****************************************************************************
*                                                                          *
*     Filename       English Translation        PMODE PCLEAR   Locations   *
*                                                             CTR-80  CCR-81*
*     SHUTCOV        Shuttle Cover                 4     5     7/149   5/126 *
*     MATCH4         Match4                        0     1    25/162  17/141 *
*     COUNT          Count 'de Money               3     4    49/180  34/162 *
*     PILOT          CCPILOT                       0     1    69/195  50/182 *
*     SAMPLE         Sample (for PILOT)           (2)   (4)   84/206  62/198 *
*     TAIPAN         Taipan                       (0)   (1)   92/212  70/206 *
*     COLRDUMP       *Colordump                   (2)   (4)  120/235  95/240 *
*     EZSKI          EZ Ski (CLOADM:EXEC)         (2)   (4)  129/244 105/254 *
*                                                                          *
*                                                                          *
*   Locations are for the R/S recorders. If the first copy of a program won't load, try the second. If *
*   neither copy loads, return the tape for disciplining and a prompt replacement. If you get an OM, FC, *
*   or SN error while loading or running a program, you probably have to enter the PMODE and PCLEAR values *
*   for the program directly from the keyboard (values in parenthesis are not set in the program). *     *
*   These programs may use high speed. Be sure that the computer is slowed down again before doing I/O to *
*   tape or disk (POKE 65494,0). Subscribers - The month on the mail label is the last month of your *
*   subscription. If you have a cassette subscription, the number next to the month is the amount it *
*   would cost to convert the rest of your subscription to the disk version. *
*                                                                          *
****************************************************************************
```

This one gets used over and over also... **Shuttle Cover** (by Eric Faden) 'simulates' the liftoff and gives an artist's impression of the shuttle orbiting the earth. Note: If you wish to be prompted for the correct colors at the beginning of the cover (so the U.S. flag always has red stripes), remove the apostrophe (') from the beginnings of lines 103, 105, 106, and 107, change the **'DRAW** in line 102 to **DRAW**, and change line 108 to **GOTO 105**.

You might wonder why the author had to do the above hoop-jumping to create the right colors for his cover. First, he chose to display his cover in the highest resolution graphic mode (PMODE 4). This limited him to just black and buff for his screen colors. However, because a color television is not precise (don't ask me to get too technical, 'cause I can't), you often get other colors on the screen (red and blue, for example). You get these colors whenever you put alternating pixels of black and buff next to each other. Unfortunately, there is no way upon power up, to tell if you are going to get red or blue. Sometimes you get blue when the pixels are black-buff and sometimes you get blue when the pixels are buff-black. The author solves this problem by making you tell him if a square drawn with the black-buff arrangement is blue. If it is blue, then he knows that black-buff will give a blue tint and buff-black will give a red tint. If it is red, then from then on he draws his blue lines in buff-black and his red ones in black-buff.

A perfect match - **Match4** (by Paul Cheramy). Using the arrow keys or the joystick to move back and forth, you stack your piece on one of the seven columns by pushing <down

arrow> or by pulling the joystick down. The object is to get 4 in a row (across, up-down, or diagonal) and to block your opponent from getting 4 in a row. Note: If you have 16k you may want to turn the computer off and then on again before loading this program.

One of my favorite subjects (unfortunately, I know very little about it) is money. So **Count 'de Money** (by Phill Beistel) interests me! This educational program is designed to help one learn to count money by showing dollars and change graphically and asking you to type the amount shown on the screen. If you play level 2, you will be asked to type the amount of money left if some of the money shown on the screen is spent. Use <left arrow> or <clear> to clear a value if you err in typing.

Easier than French... - **CCPILOT** (by Larry Landwehr) translates programs written in PILOT to BASIC. PILOT is a very simple programming language used mainly to teach people how to program. There are just a few rules and only 13 commands:

1) Each complete command must be followed by a colon.
2) A command may be preceded by a number, which acts as a label for that line or set of lines.
3) Use the BASIC convention for variable names (all variable names are 1 or 2 characters with the first character alphabetic and string names followed by a $).
4) These variables are reserved and used by the program - **AN$ MA$ FL**.
5) All statement and/or procedure numbers must be unique, but they do not have to be in any order.

R Remark. Ignored by program.
> R: THIS LINE WILL NOT AFFECT THE PROGRAM
T Type the following line on the screen. Variables surrounded by slashes (/) are displayed.
> T: HELLO   (HELLO is displayed)
A Accept input into a variable. If a variable is not specified, then **AN$** will be used.
> T: TYPE YOUR NAME AND AGE.   (Asks for and gets
> A: the name in AN$ and
> A: AG   the age in AG, then
> T: /AN$/ IS /AG/ YEARS OLD.   prints them)
M Matches following string to any portion of AN$. Variables can be used if they are surrounded by slashes (/). If there is a match, FL is set to 1. Otherwise, FL is set to 0.
> R: CONTINUING THE ROUTINE ABOVE   (Sees if your name
> M: DAV   is the same as mine)
Y Always followed by another command. Executes that command if the PILOT flag is true (FL=1).
> R: CONTINUING THE 2 ROUTINES ABOVE   (If there was a match,
> YT: MY NAME IS DAVE (OR DAVID) TOO!   FL=1 and message displayed)
N Always followed by another command. Executes that command if the PILOT flag is false (FL=0).
> R: CONTINUING THE 3 ROUTINES ABOVE   (If there was not a match,
> NT: YOUR NAME IS DIFFERENT THAN MINE. FL=0 and message displayed)
C Compute the following arithmetic expression and assign the answer to the specified variable. Variables can be used in the computation and all BASIC operators are acceptable.
> C: M=256*256   (Computes the maximum
> C: K=M/64   addressable memory and
> T: ONE K IS ACTUALLY /K/ BYTES.   then figures 1k worth)
X If the following expression is true, FL is set to 1. Otherwise FL is set to 0. Variables can be compared and all BASIC operators are acceptable.
> R: CONTINUING THE ABOVE ROUTINE   (Checks a Kbyte against
> X: K>1000   1000 bytes, prints message,
> YT: MORE THAN 1000 BYTES IN A KBYTE. then checks max memory with
> X: 32*K>M   my 32k, finds that max mem

```
          YN: 1 HAVE LESS RAM THAN POSSIBLE.     is higher and prints message)
```

U   Use a procedure. Must be followed by a number (not a variable), which is associated with a procedure. After the procedure has been completed, the next statement after the U is done.

```
          U: 10                              (Uses procedure 10 and
          T: THE NUMBER IS /NU/                prints the number it got)
```

Procedure delimeter (marks the beginning and end of a procedure). The first # marking the procedure must be proceded by a procedure number. Procedures must appear at the end of a program. The indentation of the commands is purely for ease in reading the program later.

```
          R: CALLED FROM ROUTINE ABOVE
      10#:                                   (This procedure
          T: WHAT NUMBER DO YOU LIKE?         gets a number, and
          A: NU                               returns it to the
      #:                                      main program)
```

J   Jump to a statement. Must be followed by a number (not a variable), which is associated with a statement.

```
      50T: GUESS MY NUMBER.                  (If you guess 37,
        A: NB                                 it prints WRONG!
        X: NB=37                              and asks you to
       YJ: 30                                 guess again until
        T: WRONG!                             you get it right.
        J: 50                                 Then you get the
      30T: RIGHT!                             RIGHT! message)
```

E   Ends a program (uses the BASIC STOP command).

```
        E:                                   (Stops the program)
```

L   Literal. Inserts the following BASIC line into the translated program.

```
        T: LET'S PRINT YOUR NAME. NAME?      (Gets your name and
        A:                                    prints it double-spaced
        L: PRINT#-2, AN$:PRINT#-2," "         on a printer)
```

That was fun! Now, to create your own **CCPILOT** program, use the BASIC editor. Just start each line with **line number '** (ie: **25 'T: WHAT IS YOUR NAME?**). After you have finished your program, save it to tape or disk in **ASCII** format using **CSAVE"filename",A** (use **SAVE** to save to disk). Once the program is saved, run **CCPILOT**. You will be asked the filename of the program you created. Then the program will be read in and translated into BASIC.

If an error is discovered during the translation, an error message is displayed and the program halts. Hit . (period) to terminate the program or any other key to continue translating to look for other errors. Then go back, fix the errors in your PILOT program and try the translation again.

If no errors are found, your new BASIC program can be saved to tape or disk. It will be saved with the same filename preceded by an * (asterisk). Note: If you did not specify an extension and save the new program to disk, it will be saved with the **/DAT** extension.

EXTRA! If you wish to create your PILOT program on a real text editor (like January 1983's **Keep Text**), delete the **:INPUT#DV,CL$** at the end of line 157 in **CCPILOT** and type line **102 RETURN**.

To get you started with **CCPILOT**, we have included a sample BASIC program in ASCII called (what else?) **Sample**. To use it:

1) Load and run **CCPILOT**.
2) When asked, give the filename **SAMPLE**.
3) Answer the Tape/Disk questions that come up for loading and saving.

Now just load in and run your newly translated (and saved) BASIC **Sample**! Note: The new BASIC file will be named ***SAMPLE** (***SAMPLE.DAT** on disk).

Will the next program be Shogun? **Taipan** (by Patrick Walker) gives you the chance to be a pirate or a trader in the far eastern seas of the late 1800's. The object is to 'earn' a million units of money. It's not easy as you are pestered by pirates, loan sharks, storms, extortion, etc. Don't buy more than your ship can hold (as you get richer, you can buy bigger ships and more guns). If you have to fight, hit **F** to fire or **R** to run. Your firepower is based on the number of guns (not arms - they are for trading) on your ship. And I thought times were hard now... Note: If you have 16k, you must type **PMODEO:PCLEAR1**<enter> before loading the game.

Put a picture anywhere - **Colordump** (by Drew Haines) makes it EASY to save and/or load a graphic picture to/from tape and/or disk! It lets you play with the PMODE, SCREEN, etc. and see how the graphics screen looks at each point. And, best of all, you can dump the graphic screen to the Radio Shack CGP-115 Printer-Plotter! I got one of the plotters just because of this program! It took 1 1/2 hours to do the drawing (it does all of one color, then goes back to the beginning and does the next color), but I got some dinner and a great drawing of this month's cover (done in PMODE 3 - try it in PMODE 4 and you'll see why).

Dump notes: 1) You can dump the picture to the printer in high speed if your computer can handle it (the program automatically adjusts the printer speed POKEs). 2) The program assumes a printer is on line and dumps you out of the program if one is not on line. If you want to use the program without a printer (it can be useful), just retype line **0 PCLEAR4:PMODE3:CLS:POKE65494,0** and run it for the save/load features. 3) Large dump just doubles the size of the drawing.

The closest I'll get to snow this year is **EZ Ski** (by Paul Griffiths). This is a fun game! You use the joystick to move your skier down the slope, avoiding trees, skiers, and fallen skiers. If you can go through the flagposts, you get an extra 100 points. When you start the game, hit a key to get past the title page. Then use the joystick to choose between EZ, TUFF, and RUFF speeds and hit a key to start the game.

EZ Ski is in machine language, so to load and execute it type **CLOADM:EXEC**<enter> (use **LOADM** from disk). To make a copy of the program type **CSAVEM"EZSKI",8192,10979,8192**<enter> (use **SAVEM** to save to disk).

Lost in a maze...

The author of August 1983's **Blackard's Castle** kindly sent us a map of over 1000 rooms in his adventure. If you've used up 200 sheets of paper and still don't have it mapped, just send us a stamped, self-addressed envelope and ask for a copy.

Breaking the language barrier...

Michael Seymour of Bronx, New York fixed our March 1983 **Language Drill** so that it would drill English-Foreign or Foreign-English:

Add **855 PRINT:INPUT"eNGLISH fOREIGN rANDOM";EF$:IFEF$="F"THENAA=1ELSEAA=0**
Add **995 IFEF$="R"THENAA=RND(2)-1**
Change the **E$=LEFT$(W$(T),E):S$=MID$(W$(T),E+2)** in line 1040 to
    EE=LEN(W$(T))-E:E$=MID$(W$(T),1+(E+1)*AA,E+AA*EE):
        S$=MID$(W$(T),1-(E+1)*(AA-1),E-(AA-1)*EE)

See you at Rainbowfest (New Brunswick - March 30 thru April 1)?

Later,

Dave

ed.