

# TRS8BIT



WELCOME TO  
THIS, THE FIRST  
EDITION OF 2008.  
THANKS FOR TAKING  
THE TIME TO  
DOWNLOAD AND READ  
IT! DON'T FORGET  
THAT THE  
NEWSLETTER FORMAT

IS DESIGNED TO PRODUCE AN  
'A4' BOOKLET WHEN PRINTED.  
I FIND THAT AN 'A5' SIZED  
BOOKLET IS MUCH EASIER TO  
HOLD AND READ.

I HOPE THAT, ONCE AGAIN,  
I'VE MANAGED TO FIND  
INTERESTING ITEMS TO CATER  
FOR A SELECTION OF TASTES  
AND INTERESTS.

THE MODEL 1 DOES SEEM TO  
DOMINATE BUT NEVER HAVING  
OWNED ANY OTHER TRS-80 I'M  
A BIT IN THE DARK WITH THE  
LATER MODEL 3'S AND 4'S.  
IF THERE IS ANY SUBJECT YOU  
WISH TO BE FEATURED, PLEASE  
EMAIL ME WITH SUGGESTIONS  
AND I'LL DO MY BEST TO  
OBLIGE.

I'VE NOTICED, ON EBAY, OVER  
THE LAST 6 TO 8 WEEKS, 2  
VIDEO GENIES, 3 MODEL  
100/2'S AND VARIOUS MODEL  
4 COME UP FOR SALE. I WAS  
SURPRISED AT HOW LITTLE  
THEY FETCHED. ONE OF THE  
GENIES (A MODEL 3003) WENT  
FOR JUST UNDER £40. ONE OF  
THE MODEL 100 ONLY FETCHED  
99P AND A MODEL 4P ONLY  
MANAGED JUST OVER £15.

WHILST RUMMAGING THROUGH MY  
BIT AND PIECES, I CAME  
ACROSS ITEMS FOR BOTH THE  
MODEL 100 AND THE VIDEO  
GENIE. THIS LEAD ME TO  
DISCOVER ANOTHER  
INTERESTING WEB SITE IN NEW  
ZEALAND. I'VE PUT A LINK TO  
THIS ON TRS-80.ORG.UK

THIS EDITION INCLUDES AN  
ARTICLE BY KEN ROBINSON

WHICH LISTS DETAILS OF THE  
GENIE'S 50 WAY BUS AS  
MATCHED TO THE MODEL 1'S 40  
WAY BUS. (KNUTT HAS VARIOUS  
PIECES OF GENIE INFORMATION  
ON HIS WEB SITE. IT'S WELL  
WORTH A VISIT IF YOU'RE  
INTERESTED).

THERE'S A CLEAR AND  
COMPREHENSIVE WALK-THROUGH  
INSTRUCTION SET FOR  
DOWNLOADING MODELS 1 AND  
100 USING ST-80 III AND  
DETAILS OF HOW TO BUILD A  
NULL MODEM. BOTH OF THESE  
ARTICLES WERE ORIGINALLY  
WRITTEN BY LEON HELLER AND  
PUBLISHED IN NATGUG NEWS.

TALKING OF LEON, HE WAS THE  
FIRST PERSON I CAME ACROSS  
WHO SPARKED MY INTEREST IN  
THE LANGUAGE 'C'. IF I  
REMEMBER CORRECTLY, THERE  
WAS A VERSION OF 'SMALL C'  
AVAILABLE FOR THE MODEL 1,  
BUT IT WAS ONLY PRODUCED  
FOR DISK SYSTEMS.

ON EARLY PC'S I BECAME  
INVOLVED WITH THE AMERICAN  
'MIX' 'C' LANGUAGE. IT HAD  
A BIG USER BASE IN THE  
STATES WITH MASSIVE AMOUNTS  
OF FREEWARE FUNCTIONS. BUT  
AS THE YEARS PROGRESSED, I  
FOUND THE CURLY, CURVED AND  
STRAIGHT BRACKETS MORE AND  
MORE DIFFICULT TO SEE (NO  
PUN INTENDED!!). SO I MOVED  
BACK TO COBOL, PICK AND AS  
A CONSEQUENCE BASIC.

I STILL THINK THAT 'C' IS A  
NEAT, POWERFUL AND COMPACT  
LANGUAGE, HOWEVER, AT  
TIMES, IT CAN BE A BIT ON  
THE OBTUSE SIDE!

ON THE SUBJECT OF 'C' I'VE  
FOUND (ON ONE OF THE MANY  
SHAREWARE CD'S I HAVE) A  
NICE LITTLE 'C' SOURCE  
PROGRAM FOR USE ON  
COMPUTERS WITH ONLY UPPER  
CASE.

AND WHILE I'M MENTIONING THE LACK OF LOWER CASE, I'VE ALSO INCLUDED ONE OF THE MOST USEFUL LOWER CASE DRIVERS I'VE EVER USED. (IT WORKS GREAT WITH MATTHEW REED'S MODEL 1 EMULATOR, WHICH, BY THE WAY, HAS HAD ANOTHER UPDATE. SEE TRS-80.ORG FOR DETAILS). ALL OF THE ONE-LINERS AND PROGRAMS FOR TRS8BIT ARE EITHER WRITTEN, DEvised OR TESTED USING THE EMULATOR. I'M QUITE HOOKED ON THE WINDOWS VIRTUAL CASSETTE!

THAT'S ABOUT IT FOR NOW, KEEP IN TOUCH.

DUSTY



----= 000 =----

AT THE  
READY>  
PROMPT



WITHIN LEVEL II BASIC THERE ISN'T AN XOR INSTRUCTION (ONLY AND, OR, NOT). HOWEVER IT IS POSSIBLE, ACCORDING TO JOHN PHELAN, TO USE THE FOLLOWING :-

A XOR B = (A OR B)-(A AND B)

AND OBTAIN THE CORRECT RESULT.

IN THE LAST EDITION I MENTIONED THE PRINT USING "!" SIGN SO IT SEEMS ONLY FAIR TO 'FEATURE' THE OTHER STRING

OUTPUT CODE WHICH IS THE PERCENT "%" SIGN. THE PERCENT SIGN MARKS THE BEGINNING AND THE END OF A PRINT LOCATION TO BE FILLED WITH STRING DATA. BETWEEN THE 2 PERCENT SIGNS IS A GIVEN NUMBER OF SPACES. THESE SPACES DEFINE THE WIDTH OF THE STRING FIELD. FOR EXAMPLE, THE EXPRESSION "% %" DEFINES A STRING FIELD OF 4 POSITIONS. (THE BEGINNING AND END PERCENT SIGNS COUNT AS ONE SPACE EACH.) WHEN YOU USE THIS PRINT MODE, THE COMPUTER TAKES THE OPERAND STRING SUPPLIED BY YOU AND TRIES TO PUT IT INTO THE 'PRINT USING' FIELD. FOR EXAMPLE -  
10 PRINT USING "% %";  
"DUSTY"

WILL PRINT - DUST.  
ALL CHARACTERS THAT DON'T FIT INTO THE 'PRINT USING' STATEMENT ARE TRUNCATED. ANY PERCENT SIGN THAT IS NOT BEGINNING OR ENDING A STRING FIELD IS REGARDED AS JUST ANOTHER CHARACTER.

YOU CAN, OF COURSE, PRE-DEFINE YOUR PRINT USING STRING TO SAVE MEMORY (AND EFFORT), FOR EXAMPLE -  
10 A\$="%"  
20 B\$="27 LETSBIE AVENUE"  
20 PRINT USING A\$;B\$  
WILL PRINT - 27 LETSBIE.  
THIS IS A NEAT LITTLE DEVICE FOR LIMITING THE WIDTH USED BY FIELDS WITHIN PRINT OUTS OR SCREEN DISPLAYS.

----= 000 =----

HERE'S A GREAT ONE-LINER FOR DISPLAYING CHARACTERS RAPIDLY TO THE SCREEN BY PATRICK BOYLE

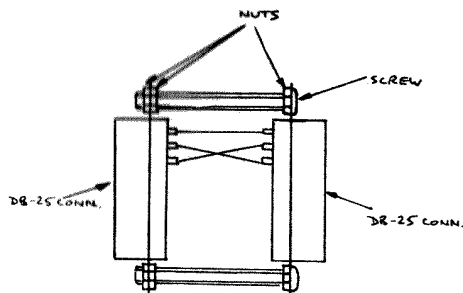
```
0 CLEAR22:A$=STRING$(22,32):J=VARPTR(A$):I=PEEK(J+1)+256*PEEK(J+2):FORK=ITOI+21:READZ:POKEK,Z:NEXT:POKE16526,PEEK(J+1):POKE16527,PEEK(J+2):FORX=1TO2:POKEI+10,RND(255):L=USR(0):X=1:NEXT:DATA33,0,60,17,1,60,1,255,3,54,0,237,176,6,5,33,0,0,43,124,181,201
```

IF YOU ARE CONNECTING A COMPUTER TO ANOTHER VIA THEIR RS-232 SERIAL PORTS, YOU NEED A DEVICE CALL A 'NULL MODEM'. THIS IS JUST A CABLE WITH APPROPRIATE CONNECTORS, WIRED IN SUCH A WAY THAT EACH COMPUTER APPEARS AS A MODEM TO THE OTHER. WITH A COUPLE OF DB-25 CONNECTORS AND A FEW PIECES OF WIRE, TWO SCREWS AND A FEW NUTS, YOU CAN EASILY BUILD A NULL MODEM FOR A FEW POUNDS IN ABOUT 20 MINUTES.

ALL YOU DO IS PLACE THE CONNECTORS BACK TO BACK, AND LINK THE PINS BETWEEN THE CONNECTORS ACCORDING TO THE TABLE BELOW.

SIGNAL	PIN	PIN
GND	1	1
TD	2	2
RD	3	3
RTS	4	4
CTS	5	5
DSR	6	6
GND	7	7
CD	8	8
DTR	20	20

THEN YOU FIX THE TWO CONNECTORS TOGETHER USING THE SCREWS AND NUTS AS SHOWN BELOW.



IF YOU WANT TO BE REALLY POSH, YOU COULD USE STUDDING, SPACERS AND A SMALL BOX OF COURSE.

I USED A MALE AND FEMALE CONNECTOR, AS THIS ENABLES ME TO CONNECT MY MODEL 1 OR MODEL II TO MOST OTHER SYSTEMS.

BY WIRING THE PINS ON A ONE TO ONE BASIS, WITH CONNECTORS OF THE APPROPRIATE GENDER, YOU CAN EASILY MAKE WHAT IS TERMED AS A GENDER-CHANGER.

---= 000 =---

TRS-80 MODEL 1/GENIE EDGE CONNECTOR CROSS-REFERENCE FROM AN ORIGINAL ARTICLE BY KEN ROBINSON

THE FOLLOWING TABLE SHOWS THE PIN INTER-CONNECTIONS TO OBTAIN THE TRS-80 BUS FROM THE VIDEO GENIE. IN THE TABLE THE SYMBOLS USED EITHER SIDE OF THE GENIE PIN NUMBER HAVE THE FOLLOWING MEANING. \*NN\* INDICATES AN ADDITION TO THE GENIE BUS #NN# INDICATES A SIMILARITY BETWEEN THE GENIE AND THE TRS-80 BUT NOT AN IDENTITY.

AN ACULAB FLOPPY TAPE AND AN LNW EXPANSION BOARD CONTROLLING DISKS CAN BE RUN OFF AN EXPANSION BUS USING THE FOLLOWING SUBSTITUTIONS-  
 PIN 12 OUT\* CAN BE OBTAINED BY OR'ING GENIE PINS 40 AND 38  
 PIN 13 VR\* CAN BE OBTAINED BY OR'ING GENIE PINS 40 AND 43  
 PIN 14 INTAK\* CAN BE OBTAINED BY OR'ING GENIE PINS 38 AND 45  
 PIN 15 RD\* CAN BE OBTAINED BY OR'ING GENIE PINS 41 AND 43  
 PIN 19 IN\* CAN BE OBTAINED BY OR'ING GENIE PINS 38 AND 41

SUITABLE OR GATES CAN BE FOUND ON THE 74LS32, WHICH IS A QUAD ARRAY. UNLESS INTAK\* IS DEFINITELY REQUIRED IR CAN BE LEFT CUT FOR MOST APPLICATIONS. IT IS WORTH NOTING THAT RD\* AND WR\* ON THE TRS-80 ARE IN FACT [NENRD] AND [MEMWR] AND

TRS-80

VIDEO GENIE

1	RAS*	43
2	SYSRES*	#48#
3	CAS*	*13*
4	A10	26
5	A12	29
6	A13	27
7	A15	21
8	GND	1,2,49,50
9	A11	28
10	A14	23
11	A8	22
12	OUT*	*32*
13	WR*	#40#
14	INTAK*	--
15	RD*	#41#
16	MUX	*25*
17	A9	24
18	D4	20
19	IN*	*33*
20	D7	17
21	INT*	31
22	D1	14
23	TEST*	#39#
24	D6	16
25	A0	10
26	D3	16
27	A1	7
28	D5	11
29	GND	1,2,49,50
30	D0	15
31	A4	6
32	D2	12
33	WAIT*	37
34	A3	6
35	A5	5
36	A7	3
37	GND	1,2,49,50
38	A6	4
39	GND (LEV 2 ONLY)	#39#
40	A2	9

1	GND	8,37,29
2	GND	8,37,29
3	A7	36
4	A6	38
5	A5	35
6	A4	31
7	A1	27
8	A3	34
9	A2	40
10	A0	25
11	D5	28
12	D2	32
*13*	□CAS□	3
14	D1	22
15	D0	30
16	D3	26
17	D7	20
18	D6	24
19	VCC	--
20	D4	18
21	A15	7
22	A8	11
23	A14	10
24	A9	17
*25*	MUX	16
26	A10	4
27	A13	6
28	A11	9
29	A12	5
30	PHI	--
31	PINT	21
*32*	□OUT□	12
*33*	□IN□	15
34	□PHLDA□	--
35	□PHAN□	--
36	□HALT□	--
37	□PWAIT□	--
38	□ILRQ□	--
#39#	□PHOLD□	23
#40#	□WR□	13
#41#	□RD□	15
42	□CCDBS□	--
43	□MREQ□	1
44	□D0DBS□	--
45	□MI□	--
46	□RESET□	--
47	RFSH	--
#48#	□NNI□	2
49	GND	8,37,29
50	GND	8,37,29

NOT □RD□ AND □WR□, HENCE THE \* AS OPPOSED TO THE BAR □CAS□ CAN BE OBTAINED FROM THE CPU BOARD INSIDE THE GENIE ON PIN 6 OF Z39. MUX CAN BE OBTAINED ON PIN 5 OF Z37. ALSO IF YOU WISH □OUT□ IS AVAILABLE ON PIN 8 OF Z15 WHILST □IN□ IS AVAILABLE ON PIN 6 OF Z15.

(THE ORIGINAL ARTICLE WAS IN VERY POOR CONDITION AND CONSEQUENTIALLY VERY DIFFICULT TO READ. I HOPE I'VE MANAGED TO COPY ALL THE DETAILS CORRECTLY).

HERE'S A NICE LITTLE MOD TO THE CASSETTE VERSION OF ELECTRIC PENCIL FROM AN ORIGINAL ARTICLE WRITTEN BY RODNEY SCHREINER.

FOR SEVERAL MONTHS I'VE BEEN USING THE CASSETTE VERSION OF PENCIL TO PREPARE MANUSCRIPTS AND LETTERS USING SINGLE SHEETS OF 8.5 X 11 INCH PAPER. SINCE PENCIL DOES NOT AUTOMATICALLY PAUSE AT THE END OF A PAGE, I HAVE TO WATCH THE PRINTER AND STOP THE PROGRAM MANUALLY SO I CAN INSERT A NEW SHEET OF PAPER.

THE TEDIOUSNESS OF THIS TASK FINALLY GOT TO ME, AND I DECIDED TO INVESTIGATE THE PROGRAM TO SEE IF I COULD MODIFY IT SO THAT IT WOULD STOP AT THE END OF EACH PAGE. AFTER A LENGTHY SEARCH THROUGH THE CODE, I LOCATED THE SPOT AT WHICH THE PROGRAM TESTS FOR THE FINAL LINE OF A PAGE. AT THIS POINT THE PROGRAM JUMPS TO THE PAGE SPACING ROUTINE. BY CHANGING ONLY 2 BYTES I ALTERED THIS JUMP TO A RETURN TO THE MAIN SYSTEM. NOW, WHEN THE PRINTER REACHES THE END OF PAGE, IT STOPS AND I CAN LOAD A FRESH SHEET OF PAPER AT MY LEISURE. ALL THAT IS NECESSARY TO CONTINUE PRINTING IS TO MOVE THE CURSOR TO THE START OF THE NEXT PAGE I REQUIRE PRINTING AND OFF I GO.

THE CHANGES I'VE MADE ARE DETAILED BELOW:

CHANGE

4F3FH FROM 28 TO C8  
4F40H FROM 06 TO 00

ONLY BYTES 4F3F AND 4F40 ARE CHANGED. THIS CAN BE ACCOMPLISHED BY USING T-BUG,

RELOCATED TO HIGH MEMORY. AFTER THE 2 CHANGES ARE COMPLETED, A NEW SYSTEM TAPE CAN BE WRITTEN FROM T-BUG. THE STARTING ADDRESS IS 4350H, THE ENDING ADDRESS IS 5374H AND THE ENTRY POINT IS 4350H.

THIS NEW VERSION OF PENCIL CANNOT PRINT CONTINUOUSLY, SINCE THE ALTERATIONS ABOVE FORCE IT TO STOP AT THE END OF EACH PAGE. HOWEVER, IF I NEED CONTINUOUS PRINTING I CAN USE MY ORIGINAL PENCIL CASSETTE.

---- 000 ----

FILE TRANSFERS ON THE MODELS 1 AND 100 BY LEON HELLER

IF YOU ARE USING THE TANDY RS-232 BOARD, IT IS NOT ADVISABLE TO JUST PLUG THE CABLE FROM THE MODEL 1 INTO THE BACK OF THE MODEL 100. BOTH MACHINES ARE CONFIGURED AS 'DATA TERMINAL EQUIPMENTS' (DTE) AND ALTHOUGH YOU CAN REVERSE PINS 2 AND 3 ON THE RS-232 BOARD IN THE MODEL 1, THE OTHER SIGNALS WILL BE TRYING TO OUTPUT TO ONE ANOTHER. THIS SHOULD NOT RESULT IN ANY DAMAGE, BUT WILL PREVENT PROPER HANDSHAKING BETWEEN THE SYSTEMS. YOU WILL NEED WHAT IS TERMED A NULL MODEM CONNECTOR, WITH ALL THE SIGNALS CROSSED OVER. THESE CAN BE OBTAINED AT MANY LOCAL COMPUTER SUPPLY SHOPS OR YOU CAN EASILY MAKE ONE FOR YOURSELF. (SEE THE ARTICLE ON PAGE 3 ED.)

ASSUMING YOU HAVE A SMART TERMINAL PROGRAM LIKE ST80-III ON THE MODEL 1, YOU SET THE SYSTEM UP FOR 300 BAUD, 7 DATA BITS, EVEN PARITY, 1 STOP BIT, TRANSMIT AUTO LINE FEED ON, AND RECEIVE

AUTO LINE FEED ON.  
ON THE MODEL 100, SELECT  
'TELECOM' FROM THE MAIN  
MENU, THEN SET THE RS-232  
PARAMETERS USING F3 (STAT:  
37E1E). NOW USE F4 TO GET  
INTO TERMINAL MODE. KEYS  
PRESSED ON THE MODEL 1  
SHOULD BE ECHOED ON THE  
MODEL 100, AND VICE-VERSA,  
SHOWING THAT ALL  
CONNECTIONS ON BOTH SYSTEMS  
ARE FUNCTIONING.

IF YOU ARE TRANSFERRING A  
FILE FROM THE MODEL 1 TO  
THE MODEL 100, YOU NOW GET  
THE FILE INTO THE MODEL 1  
BUFFER, USING (SHIFT) G.  
RESPOND TO THE TRANSMISSION  
SPEED QUESTION WITH 1, TO  
BE ON THE SAFE SIDE. ON THE  
MODEL 100, USE F2 TO SELECT  
THE DOWNLOAD MODE. THE  
SYSTEM WILL REQUEST A  
FILENAME. AFTER INPUTTING  
THE FILENAME, F2 WILL BE  
DISPLAYED IN REVERSE VIDEO.  
NOW PRESS (SHIFT) O ON THE  
MODEL 1 KEYBOARD, AND THE  
FILE WILL BE TRANSFERRED  
FROM THE MODEL 1 TO THE  
MODEL 100, BEING  
SIMULTANEOUSLY DISPLAYED ON  
THE MODEL 1 AND MODEL 100  
SCREENS. WHEN THE FILE HAS  
BEEN TRANSFERRED ST80-III

'READY' WILL BE DISPLAYED  
ON THE MODEL 1. IF YOU  
PRESS F2 ON THE MODEL 100,  
THE FILE WILL BE CLOSED,  
AND YOU CAN EXIT FROM  
'TELECOM' BY PRESSING F8  
TWICE.

TO UPLOAD FROM THE MODEL  
100 TO THE MODEL 1, FIRST  
SET THE MODEL1 UP FOR  
RECEIVE AUTO LINE FEED ON  
(SHIFT B), AND OPEN THE  
MEMORY BUFFER (SHIFT) C.  
KEYING F3 ON THE MODEL 100  
WILL RESULT IN THE REQUEST  
FOR THE FILE NAME. INPUT  
THE FILE NAME, AND RESPOND  
TO THE WIDTH QUESTION WITH  
(ENTER). 'UP' WILL BE  
DISPLAYED IN REVERSE VIDEO,  
AND THE FILE WILL BE  
TRANSFERRED TO THE MODEL 1,  
BEING DISPLAYED ON THE  
MODEL 1 SCREEN AS TRANSFER  
OCCURS.

WHEN THE FILE HAS BEEN  
TRANSFERRED, 'UP' ON THE  
MODEL 100 SCREEN WILL  
CHANGE FROM REVERSE VIDEO  
TO NORMAL. KEYING (SHIFT) X  
ON THE MODEL 1 WILL CLOSE  
THE MEMORY BUFFER, AND THE  
FILE CAN BE TRANSFERRED TO  
DISK USING (SHIFT) F.

---= 000 =---

THIS PROGRAM IS ONE OF THE  
MOST USEFUL ONES I'VE EVER  
HAD FOR A 16K MODEL 1! (IT  
WILL ACTUALLY FIT DOWN TO A  
ONE-LINER). IT'S A SUPER  
LITTLE LOWER-CASE DRIVER.  
IT WORKED WELL (SUBJECT TO  
ANY NECESSARY HARDWARE MOD  
BEING COMPLETED) WITH  
ORDINARY LEVEL II BASIC,  
ACULAB XBAS, DISK BASIC AND

MORE RECENTLY, MATTHEW  
REEDS EMULATOR.  
THE REALLY NICE THING ABOUT  
IT WAS THAT IT WORKS IN  
REVERSE! I.E. WITH THE  
SHIFT HELD DOWN, LOWER-CASE  
LETTERS ARE PRODUCED. THIS  
AVOIDED ACCIDENTAL LOWER-  
CASE ENTRIES. A REAL BOON  
FOR PROGRAM DEVELOPMENT AND  
DE-BUGGING!

```
10 REM FOR 16K MACHINES SET MEM SIZE TO 32737
20 POKE 16553,255:FORI=32738TO32767:READJ:POKEI,J:NEXT
30 POKE 16414,226:POKE16415,127
40 DATA 221,110,3,221,102,4,218,154,4,221,126,5,183,40,1
50 DATA 119,121,254,32,218,6,5,254,128,210,166,4,195,125,4
```

## "CONVERT"

WRITTEN BY LEOR ZOLMAN

THIS PROGRAM CONVERTS REGULAR C SOURCE FILES INTO A FORMAT SUITABLE FOR EDITING ON THE TRS-80 (OR ANY UPPER-CASE-ONLY SYSTEM.) SINCE THERE ARE QUITE A FEW ASCII CHARACTERS THAT NEED TO BE REPRESENTED EVEN THOUGH THEY DON'T SHOW UP ON UPPER-CASE-ONLY SYSTEMS, A SPECIAL NOTATION HAS BEEN CREATED FOR REPRESENTING THESE CHARACTERS. THE POUND SIGN IS USED AS A SORT OF 'SHIFT' KEY, WITH THE LETTER FOLLOWING THE POUND SIGN DENOTING THE SPECIAL CHARACTER NEEDED. NOTE THAT THE C COMPILER DOES NOT RECOGNIZE THIS SPECIAL SCHEME, AND BEFORE YOU CAN COMPILE A SOURCE FILE CONTAINING THE SPECIAL CODES YOU MUST PREPROCESS THE FILE USING THE "CCØT" COMMAND.

THE SPECIAL CODES AND THE CHARACTERS THEY REPRESENT ARE:

```
#L LEFT BRACKET (FOR SUBSCRIPTING) (5B HEX)
#R RIGHT BRACKET (5D HEX)
#C CIRCUMFLEX (BITWISE "NOT") (7E HEX)
#H UP-ARROW (EXCLUSIVE "OR" OPERATOR)
    (5E HEX)
#V VERTICAL VAR (LOGICAL AND BITWISE "OR")
    (7C HEX)
#B BACKSLASH (FOR ESCAPE SEQUENCES)(5C HEX)
#U UNDERSCORE (5F HEX)
```

FOR EXAMPLE, THE COMMAND  
A>CONVERT FOO.C BAR.CT

WILL EXPECT FOO.C TO BE A NORMAL C SOURCE FILE ON DISK, AND WILL CONVERT IT INTO A FILE NAMED BAR.CT. THE FILE BAR.CT MAY THEN BE EDITED TO YOUR TASTE, BUT REMEMBER TO PREPROCESS IT WITH "CCØT" BEFORE APPLYING THE C COMPILER.

AS YOU MAY HAVE GATHERED FROM ALL THIS, THE LANGUAGE "C" WAS NEVER INTENDED TO BE IMPLEMENTED ON A SYSTEM HAVING UPPER-CASE ONLY; NEVERTHELESS, HERE IS A WAY FOR IT TO BE DONE.

THIS PROGRAM IS RATHER SIMPLE, AND THUS IT WILL NOT RECOGNIZE THAT SPECIAL CHARACTERS IN QUOTES ARE NOT SUPPOSED TO BE CONVERTED.

```
#DEFINE LEFTCURLY 0X7B
#DEFINE RIGHTCURLY 0X7D
#DEFINE LEFTBRACK 0X5B
#DEFINE RIGHTBRACK 0X5D
#DEFINE CIRCUM 0X7E
#DEFINE UPARROW 0X5E
#DEFINE VERTIBAR 0X7C
#DEFINE BACKSLASH 0X5C
#DEFINE UNDERSCORE 0X5F
```

```
CHAR IBUF[134], OBUF[134];
```

```
MAIN(ARGC, ARGV)
INT ARGC;
CHAR *ARGV[];
BEGIN
    INT FD1, FD2;
    CHAR C;
    IF (ARGC != 3) BEGIN
        PRINTF("USAGE: CONVERT OLD NEW <CR>\N");
        EXIT();
    END
```

```

FD1 = FOPEN(ARGV[1],IBUF);
IF (FD1 == -1) BEGIN
    PRINTF("NO SOURCE FILE.\N");
    EXIT();
END
FD2 = FCREAT(ARGV[2],OBUF);
IF (FD2 == -1) BEGIN
    PRINTF("CAN'T OPEN OUTPUT FILE.\N");
    EXIT();
END

WHILE ((( C = GETC(IBUF)) != 0X1A) && C != 255) BEGIN
    SWITCH (C) BEGIN
        CASE LEFTCURLY: PUTST(" BEGIN ");
            BREAK;
        CASE RIGHTCURLY: PUTST(" END ");
            BREAK;
        CASE LEFTBRACK: PUTSPEC('L');
            BREAK;
        CASE RIGHTBRACK: PUTSPEC('R');
            BREAK;
        CASE CIRCUM: PUTSPEC('C');
            BREAK;
        CASE UPARROW: PUTSPEC('U');
            BREAK;
        CASE VERTIBAR: PUTSPEC('V');
            BREAK;
        CASE BACKSLASH: PUTSPEC('B');
            BREAK;
        CASE UNDERSCORE: PUTSPEC('U');
            BREAK;
        DEFAULT: PUTC(TOUPPER(C),OBUF);
    END
END

IF (C==255) C = 0X1A; /* DIGITAL RESEARCH....WOW. */
PUTC(C,OBUF);
FFLUSH(OBUF);
END

PUTST(STRING)
CHAR *STRING;
BEGIN
    WHILE (*STRING) PUTC(*STRING++,OBUF);
END

PUTSPEC(C)
CHAR C;
BEGIN
    PUTC('#',OBUF);
    PUTC(C,OBUF);
END

```



---= 000 =---

I HOPE YOU'VE ENJOYED THIS ISSUE. IF YOU HAVE, WHY NOT DROP ME A LINE AND TELL EVERYONE WHAT YOU'RE UP TO WITH YOUR TANDY? IF NOT. LET ME KNOW AND I'LL TRY AND IMPROVE THE NEXT ISSUE, WHICH BY THE WAY, SHOULD BE OUT EARLY JUNE 08.

AND FINALLY, HAS ANYONE A PDF COPY OF 'PATHWAYS THROUGH THE ROM' FROM IRA'S SITE? IF YOU HAVE, WOULD YOU EMAIL ME A COPY? IT'S THE SAME ADDRESS FOR ALL CORRESPONDENCE - TIA

DUSTYM@BEEB.NET