

# Central Data 2650 Newsletter

I bet you thought you were never going to see this issue of the newsletter! Well, I apologize for the (large) delay, but I wanted to make sure that all of our software packages were finished up before starting it.

I am glad to say that we accomplished that goal--all of the software which we have announced has been shipped. Also, many of you will be glad to know that we will probably be offering a Startrek program within the next two months which includes a special character generator with all of the space ships, etc. as well as a full set of chess characters!

The response from the release of our ALP and 12K BASIC packages have been great! There is no doubt that these are the easiest to use and best software products we have ever made. The 12K BASIC runs much faster than the 8K BASIC, and has a vast number of improved and additional statements and functions. The ALP (assembly language package) has a copy of our super-editor/assembler and debugger packages. Since all three of these programs are "under one roof", alot of space is saved by having common sub-routines.

To the right you can see a picture of the staff at Central Data. We are not going to divulge the identity of the people in the picture until next issue so you can make wild guesses as to who belongs to those bodies.

I hope that you enjoy this issue of the newsletter, and appreciate any input that you send to me for the next issue.



## LETTERS

Dear Mr. Roloff:

Enclosed is a copy of a disassembler chart for the 2650. We find this a very handy reference device and would like to pass it on to you and perhaps readers of your newsletter. You may feel free to reproduce it directly.

We hope to receive the tape version of the 12K BASIC soon. (Maybe this month?)

Best Regards,

Bruce Block  
Henry Gut  
Space Research Building  
University of Michigan  
Ann Arbor, MI 48109

Dear Bruce and Henry:

Thanks for the chart! I'm sure it will help people out when they need to quickly de-cipher some op-codes. 12K BASIC was sent recently, which completes all of the software items that we have previously announced.  
ED.

Dear Jeff:

Enclosed is a program written by C. Mundt called 'SMART' which allows the Central Data 2650 system to be used as a terminal for another system.

Before we begin I would like to thank the many people who have helped me, particularly Clarence Mundt. Without his help I would still be back in the dark ages. I would also like to thank Mike Herbach and you, Jeff. My many 'dumb' questions undoubtably drove you both up the wall--thanks everybody!!!

Hardware--my system uses the flag and sense lines from the 2650 to transmit and receive data. Those lines are connected to optical isolators to isolate the two microprocessors. This provides not only electrical isolation but immunity from noise (one of my major problems). The lines, through two twisted pairs--one ground and one data--are connected to a serial (RS232) port on my 8080 system. What this means is that the Central Data unit can be used as a terminal for any system which has a serial port (ie an 8080 Z-80, 6800, TRS-80, PET, etc.)

Software--the software must be run in high memory. The version included at the end of this does not echo keyboard inputs as most existing software has internal echo routines. The comments should be self explanatory.

If you should have any questions concerning the software or hardware please send me a cassette and I will be glad to help you. My motives are not altruistic in this regard. I'm hoping that all of you will make additions and improvements and that you will let me know about the alternative uses for this software.

Clarence is working on a subroutine that will allow a buffer area to be created in the 2650 system memory. At that point we will be able to dump programs and data from the 2650 terminal to the second system and vice versa. In my case this will allow me to store 2650 software on

my floppy disk in my 8080 system. The buffer routine will be sent in for the next newsletter

Mike Kelley  
9951 Delco Avenue  
Chatsworth, CA 91311  
213-349-8796

Mike:

Thanks for the new version of the terminal program. I know that I have talked to alot of people who have used their system for this purpose. Unfortunately, we don't have room in this issue for the program, but we will run it for sure in April. ED.

---

Dear Jeff,

Thank you for sending the newsletter. I am looking forward to receiving the next issue. My computer is now running OK after a few minor problems! (The problems seemed anything but minor at the time)

My system consists of:

CDC 2650 and S-100  
Jameco keyboard with seperate hex pad  
23" all transistor TV  
16K static RAM  
RS232C interface for teletype 43  
Upper and lower case char generator in  
2758 PROM  
Modified supervisor in 2758 EPROM

With considerable help from a friend, Mike Durham, I now have software that allows me to print "anything and everything" with the teletype 43. I can print text, the screen, and assembled programs.

Mike, who also has a CDC 2650 unit, has developed a number of interesting programs which I am enclosing for possible inclusion in your newsletter.

The programs are:

1) MORSE CODE. This program will generate audible morse code at a user selectable rate. It will convert to morse code any recognisable character that is typed in the text area of memory. Some sort of "noise maker" is required. I used a BELL

AUDIOALARM but I find the frequency is a bit high. A similiar device with a frequency of 700 to 1200 hertz would be OK. A speed code of 28/33 gives a morse rate of about 10 words per minute. Connect audio device to OUT6 (plug 84, pin 13) via an NPN transistor with a 2.2K resistor in series with the base.

2) TEXT PRINTER program for the Teletype 43. (all at 300 baud)

3) GOLF GAME (8K BASIC). This program generates: obstacle height, hole position, and wind direction and velocity.

Frustrated game player enters stroke power and elevation. Silly computer then proceeds to miss the hole just about every time and has the nerve to abuse the player. Good luck suckers, you will need it!!

4) MATH TEST (8K BASIC). I modified this program from May 78 Personal Computing to suit CDC 8K BASIC. A slight modification to the BASIC is required. The POKE commands at the start of the program delete the LFCR from the INPUT command. I think the rest of the program is self explanatory.

5) 2758 EPROM MOD. Diagram of modifications required to fit 2758 EPROMS in place of 3624's for the supervisor and character generator. A 24 pin wire wrap socket soldered to a PROM holder with a few modifications (as shown in the diagram) between the two does the job nicely.

6) FLASH. This is a modification to both the Editor/Assembler and BASIC that flashes the data on the screen (upper right hand corner) whilst loading or storing source code and BASIC programs.

Editor/ Assembler  
H259C 3F 33 C9  
H33C7 3F 02 4F CB F9 17  
H25FB 3F 33 C0  
H33BE CD E5 52 CB F9 17

BASIC  
24FB 1F 17 C2  
17C0 3F 02 E9 CB F9 1F 24 FB  
2574 3F 17 CD  
17CB 3F 02 4F CB F9 1F 25 77

Mike has a few problems with the Editor/Assembler that I hope you can help with.

- 1) HALT and NOP don't always generate correct machine code.
- 2) "T" error message is never printed.
- 3) Assm sometimes does not put 0000 at end of tape dump.
- 4) Some programs printed in the newsletter use arithmetic on labels. The CDC assembler does not understand them. This causes great confusion to inexperienced programmers (me!).

I have one problem with the BASIC. Is there a fix that allows the SIN of angles from 180 to 360 degrees?

I have noticed that some people are having trouble loading tapes made on different recorders. I had this problem and overcame it in the following way. The tape recorder I use has a DC motor with a speed trimming pot inside the motor. I removed this pot, attached flying leads out to a 10 turn 500 ohm pot with a multi-turn dial. The end result is that with a bit of adjustment I can load any tape (so far anyhow!).

We had some trouble getting the morse code program to run on my computer (it ran without problem on Mikes unit). After a lot of worry, abuse, phone calls, and finally a visit from Mike, the trouble was tracked down.

The problem was caused by the fact that the lower part of the program status word (PSL), in particular the register select bit (bit 4) is not set to select register bank 0 on start up. The PSL is set to the random value that appears in memory location H17F2 at start up. The above caused the morse code program to do some funny things when the program hits the first register bank change instruction. The cure to this problem is to preset the PSL at the start of the program.

Credit for the enclosed programs go to:

Math Test: Jim Charlton  
All Others: Mike Durham  
27 Kingsley Grove  
Mount Waverley  
Victoria 3149  
Australia

Since I started this letter Mike has developed two more rather clever mods for the Editor/Assembler which I will enclose.

- 7) "E" writes end of text address on screen using command E.
- 8) ERRORS. Prints number of errors that have occurred during assembly of a program.

Note: programs 7 and 8 were assembled by me--they work--but may not be what you would call elegant programming.

I am fast running out of memory so I had better finish up.

Yours faithfully,

Jim R. Charlton  
7 Oakton Close  
Mulgrave, Victoria 3170  
Australia

Jim:

Thanks alot for the letter and all of the programs. Unfortunately, we don't have room to run all of the programs this issue, but we will put them in future issues.

Concerning your problems with the assembler, I suggest that on HALT and NOP instructions you extend the line to at least the operand field. You can do this by simply hitting TAB after entering the opcode. This fills the register field with spaces, which solves the problem.

I can't understand why the "T" error message is never printed for you, but it is true that the assembler sometimes doesn't send out a zero block at the end of tape dumps. To solve this, I suggest you increase the size of your BSB area. Finally, the Editor/Assembler does not allow arithmetic expressions in the operand field (unlike our disk assembler). The ALP package, however, does allow such operations.

The 8K BASIC does have problems when it comes to the trig functions at certain angles. Unfortunately, there is no way to correct this problem. Naturally, the Extended BASIC has this problem fixed. ED.

Jeff:

You probably have received many suggestions for modifications to the Executive ROM, such as making the dump portion of TPOT a sub-routine so it can be called from user programs. This is a replacement for the tape delay routines for the 2.5MHz 2650.

(Address H026F)

D6	BSTR,UN	D2	Do it once
	BSTR,UN	D2	Do it again
*			Fall through, do again
D2	LODI,R0	DB	
DL2	RETC,EQ		Dummy--3 cycle delay
	BDRR,R0	DL2	Branch Back
*			Fall Through for 1/2 bit
D1	LODI,R0	DB	
DL1	RETC,EQ		
	BDRR,R0	DL1	
	RETC,UN		
	NOP		Dummy--fill out 16 bytes

You may want to suggest in your newsletter using the A15 address line on the S-100 board for the INTACK line, since that would disable the memory and allow inserting a vector onto the data bus.

Thanks,

Nathan Myers

Nathan:

Thanks for the suggestion. I know several people are planning on updating their system for the new 2650.

Concerning your suggestion about the INTACK line, I think it would be best to use pin 96 for this since it is given that use in other systems. ED.

---

#### SUBSCRIPTION INFORMATION

The Central Data 2650 Newsletter is published every 8 weeks by Central Data Corporation. Any comments should be sent to Central Data Corp.; PO Box 2530 Station A; Champaign, IL 61820.

Subscription Rate: \$4.00 per year.

Editor: Jeff Roloff

Gentlemen:

Enclosed is a copy of BACKUP, a program I have written and used for some time. Having altered both the assembler and BASIC many times, I find it invaluable to create new copies of these programs. It is compatible with your supervisor's tape loader. You may publish it in your newsletter if you wish.

I am creating this form letter with a program I call EDITOR ONLY. It is a copy of your editor taken from the assembler (full function version) plus my text writer code. Perhaps this is a practical use for a home computer. I can put a letter completely together on the screen before putting it on paper.

Sincerely,

Ray Krygier  
25353 Crown Point Court  
Farmington Hills, MI 48018

Ray:

Its good that you sent in this program which so many people have asked for! It is especially nice because of all of the messages that it gives, which aid greatly in the use of the program. ED.

---

Dear Jeff:

Just a note to say that I enjoyed the newsletter last month. I have enclosed a program that some of the hams out there may enjoy. It sure makes antenna design easy.

I am now conducting a weekly net dedicated to the 2650. It is held every thursday at 2330Z on 3.993 MHz SSB. It is called the 2650 computer users net and is only one week old at this time (9/78). So far there are only three of us, but I have been in contact with several others who are going to join us. I would appreciate it if you would mention us in the newsletter as soon as possible so that we might have maximum participation. So far we have discussed saturation recording, Signetics application notes, RTTY application:

RAM supervisor, RAM character generator, static/dynamic memory, and power supplies. I am looking forward to a great exchange of ideas about this system.

Thanks,

Jerry J. Johnson  
1707A Forrest Avenue  
Fort Meade, MD 20755  
301-672-2033

Jerry:

Thanks for the program! (Readers can find it later in the newsletter.) I'm sure that it will save alot of time for our ham radio readers. Hopefully this letter will get more people tuning into your net. It sure sounds like you are covering alot of bases with it! Let us know how it goes. ED.

---

Jeff,

As careful as I was in writing up my modifications for the Radio Shack keyboard, two errors crept in. The first was from my additions to the typed copy not being fully legible and the second was pure oversight. At any rate, here they are, with my apologies: first, the two AND gate sections on the right hand side are sections of Z19, the 7408 added to the circuit for the mod. They can easily be read incorrectly as Z14. The second error is the exclusion of two necessary connections. To fully correct the instructions, here is a new instruction "C":

Jumper lines to be added:

- 1) From trace above cut at break key to trace below cut at break key.
- 2) From Z12-16 to line between "." and up arrow keys.
- 3) From broken trace at repeat key to line between "U" and "5" keys.

Again, I apologize for these errors and assure everyone that with them fixed the keyboard will produce all codes from 0 to 7F.

Mike Herbach

Below is a continuation of the list of people who want to talk to others about their system. The name, address, and city are all separated by a semicolon, and if they gave a telephone number it is preceded by a '#'. Any special interests are preceded by a colon.

Remember that you can send this information in on a 3x5 card in order to be published in the newsletter. Chances are, someone out there has the same interests as you do!

Alan Shigemura; 2030 Fern Street; Honolulu, HI 96826 #808-949-0330  
: Multiplexing multiterminal interface  
Frits Schoute; Orionlaan 94; 1223 AJ Hilversum, Netherlands  
#31 35 831881: Word processing, Telephone switching  
Mark J. Schlaffer; 2401 W. Broadway, Apt 627; Col., MO 65201  
#314-445-3128  
Robert A. Robards; Independence Hill; Van Voorhis Rd.; Morgantown,  
WV 26508 #304-599-6507: Business hardware and software  
Jack Harris; 1240 Fair Oaks Drive; Irving, TX 75060 #214-253-4059  
: Residential energy and security management  
Douglas Doane; 2227 Amlevst; Kalamazoo, MI 49008: 16K byte on-  
board RAM expansion  
Steven Gibbs; Merkas Klita; Ben Yehuda, Natanya, Israel: games  
Jim Charlton; 7 Oakton Close; Mulgrave, Victoria 3107, Australia  
#(03) 560-3206: Development of weather monitoring station,  
Educational programs (for children)  
Tony E. Pajnic; 3859 Cloverdale Road; Medway, OH 45341  
#513-849-0347: Learning assembly programming.  
N. F. Simmons; 8192 Santa Margarita Lane; La Palma, CA 90623  
Nathan Myers; 48 Kahoa Street; Hilo, HI 96720 #808-935-7807  
: 2650 applied to everything--multiprocessing, word processing,  
assembly programming, APL

### EPROM Mod by Mike Durham

Enables 2758 to be plugged into 3624 socket.

A7	1	24	+5 VOLT
A6	2	23	A8
A5	3	22	A9---> PIN 19 ON 3624 PLUG (CS3)
A4	4	21	VPP-->+5V (PIN 24)
A3	5	20	CS---> PIN 21 ON 3624 PLUG (CS1)
A2	6	19	AR---> 0V OR TO ADDRESS LINE 10 FOR 2K MEMORY
A1	7	18	PD/PGM--> 0V (PIN 12)
A0	8	17	07
O0	9	16	06
O1	10	15	05
O2	11	14	04
GND	12	13	03

LEADS ARE CONNECTED DIRECTLY THROUGH (PIN FOR PIN) FROM 2758 SOCKET TO 3624 PLUG, EXCEPT WHERE INDICATED

1/ FOR 2K MEMORY, 2758 PIN 19 IS CONNECTED TO A10 VIA FLYING LEAD TO IC2 PIN 6

2/ IF 2758 SUPERVISOR IS TO BE USED, PLUG CONVERTER INTO IC 10 SOCKET, THIS CONNECTS 2758 A9 INPUT TO CORRECT LINE

3/ IF 2758 CHARACTER GENERATOR IS USED, FIT TO IC 40 SOCKET

Dear Sir:

The newsletter is, I find, very interesting and informative and I am looking forward to it being published on a regular basis every 8 weeks.

I have written two programs that I would like to sell to any interested readers. I would like to purchase a second disk drive and a printer but I need a little extra cash. The first program is a game, SPACEWAR, which is very similar to SCELBI's GALAXY GAME FOR THE 8008/8080. This program is supplied in machine code on cassette tape for \$10.00 and occupies less than 5K starting at H2000. The second program is a disassembler which converts machine code to assembler using absolute addresses instead of labels. Its display format is similar to the Editor/Asm format during Pass 2 except that there are no line numbers or labels.

The disassembler was developed to debug a fault on the DOS software which isn't supplied with any program listings. It is suitable for my present requirements but requires a few finishing touches. I don't intend to dress it up unless others are interested in the program. The disassembler is available for \$5.00 and will be available approximately two weeks after I receive the first order. This program occupies less than 2K and starts at H4000 and H7000. i.e. the same program dumped in two separate memory locations, you load just one from cassette tape.

The editor/assembler tape control by Mike Herbach was found to be very useful and led me to see if the 8K BASIC interpreter could also be easily modified. It can, the modification I used is:

2538 3F 17 67  
2056 3F 17 7A  
1767 04 08 F0 04 24 05 FE 06 02 F8 7E F9 7C  
FA 7A 3F 23 D6 17 20 F0 3F 21 4E 17

With both the editor/assembler and BASIC I find it inconvenient shifting for UPPER CASE so I incorporated the following mods:

BASIC

25BC 1F 17 50  
1750 08 89 F4 60 98 02 44 5F C8 81 17 25 B1

Editor/Assembler

21EA change 3F 03 0F to where you want the mod. I put mine at 3580. It is better to keep it out of the display memory block i.e. dont put it between H1500 and H1800.  
3580 73 F7 80 18 7B F7 60 98 02 47 5F CB 10  
3F 03 96 73 F7 80 98 7B 0B 06 E7 1B 1C  
00 83 17 00

The above modifications have been proof read and hopefully no errors exist. If any do exist please accept my sincere apologies.

Yours sincerely,

Roger Miskowicz

Dear Roger:

It is good to see that you have come up with a couple programs that you can offer our customers...and the price is very reasonable. Thanks for the patches to the programs. I am sure that they will help out many people. ED.

\*\*\*\*\*

Below is listed a computer club that we recently learned about as well as the only name we received who wanted to talk to others about his system. If anyone else is interested in having their name and interests published, just send it in on a post card.

Amateur Computer Club  
2650 Library  
51 Beechwood Drive  
Feniscowles  
Blackburn, Lancs BB2 5AT  
ENGLAND

Dave Peterson WA4EPW  
4670 Pennsylvania Avenue  
Roanoke, VA 24019  
(703) 563-9611

Interests: amateur radio and terminal use

\*\*\*\*\*

12K BASIC Updates by Central Data

FILE 'BASICMOD' AS ASSEMBLED BY SYSTEM ON 01-24-79

LINE	ADDR	B1	B2	B3	B4	LABEL	OPCODE	OPERAND	COMMENTS
0001	4025					*			
0002	4025					*			
0003	4025					*	BASIC12 VERSION 1.3 MODIFICATIONS		
0004	4025					*			
0005	4025					*	PROGRAM PATCHES		
0006	4025					*	PRINT USING FIX		
0007	4025					*			
0008	179C						ORG	179C	#
0009	179C 22 A2					IPUSP	ACON		#
0010	179E 08 FC					NEWC7	LODR.R0	*IPUSP	#
0011	17A0 E4 FF						COML.R0	FF	#
0012	17A2 18 0B						BCTR.E0	NEWC10	#
0013	17A4 1E 07						BCTR.UN	NEWC9	#
0014	17A6 04 20					NEWC8	LODI.R0	20	#
0015	17A8 3F 24 C5						BSTA.UN	SENDIT	#
0016	17AB A5 01						SUBI.R1	01	#
0017	17AD 59 77					NEWC9	BRNR.R1	NEWC8	#
0018	17AF 03					NEWC10	LOI2.R3		#
0019	17B0 1F 30 B5						BCTA.UN	INCSP2	#
0020	17B3					*			
0021	17B3					*	TAPE CONTROL--		
0022	17B3					*	URNS ON AND OFF TAPE FROM ET		
0023	17B3					*			
0024	17B3 04 00					ZTOFF	LODI.R0	0	*
0025	17B5 F0						WRTD.R0		*
0026	17B6 17						RETC.UN		*
0027	17B7 3B 7A					ZOUT1	RSTR.UN	ZTOFF	*
0028	17B9 07 49						LODI.R3	49	*
0029	17BB 1F 2F 4C						BCTA.UN	ERR2	*
0030	17BE 3F 73					ZOUT2	BSTR.UN	ZTOFF	*
0031	17C0 1F 4A 7F						RCTA.UN	DISP	*
0032	17C3 04 24					ZWAIT	LODI.R0	24	*
0033	17C5 0F FE						LODI.R1	FE	*
0034	17C7 06 02						LODI.R2	02	*
0035	17C9 F8 7E					ZLOOP	BDRR.R0	ZLOOP	*
0036	17CB F9 7C						BDRR.R1	ZLOOP	*
0037	17CD FA 7A						BDRP.R2	ZLOOP	*
0038	17CF 17						RETC.UN		*
0039	17D0 04 08					ZTON	LODI.R0	08	*
0040	17E2 F0						WRTD.R0		*
0041	17E3 17						RETC.UN		*
0042	17D4 3B 7A					ZREAD	BSTR.UN	ZTON	*
0043	17D6 1F 51 FC						BCTA.UN	AOK	*
0044	17D9 3B 7F					ZWRITE	BSTR.UN	ZTON	*
0045	17DB 3B 66						BSTR.UN	ZWAIT	*
0046	17DD 1F 51 FC						BCTA.UN	AOK	*
0047	1710					*			
0048	1710					*	BASIC12 CHANGES		
0049	1710					*	PRINT USING FIXES		
0050	1710					*			
0051	2325						ORG	PRNTS3	#
0052	2325 1F 17 9E						BCTA.UN	NEWC7	#
0053	2328					*			
0054	2328					*	TAPE CONTROL		
0055	2328					*			

FILE 'BASICMOD' AS ASSEMBLED BY SYSTEM ON 01-24-79

LINE	ADDR	B1	B2	B3	B4	LABEL	OPCODE	OPERAND	COMMENTS
0056	5130						ORG	5130	*
0057	5130 3F 17 D4						BSTA.UN	ZREAD	*
0058	513C						ORG	513C	*
0059	513C 1F 17 E7						BCTA.UN	ZOUT1	*
0060	4989						ORG	4989	*
0061	4989 3F 17 BE						BSTA.UN	ZOUT2	*
0062	5141						ORG	5141	*
0063	5141 3F 17 D9						BSTA.UN	ZWRITE	*
0064	44C7						ORG	44C7	*
0065	44C7 1C 24 9B						RCTA.E0	PRNTD6	*
0066	44CA					*			
0067	44CA					*	BASIC12 VERSION 1.4 MODIFICATIONS		
0068	44CA					*			
0069	44CA					*	BASIC12 CHANGES		
0070	44CA					*	VARPTR FUNCTION FIX		
0071	44CA					*			
0072	3E06						ORG	BTOD2	#
0073	3E06 04 66						LODI.R0	66	#
0074	3E08 08 8F						ADDR.R0	BTODS+2	#
0075	3E0A C8 2A						STRR.R0	BDS+2	#
0076	3E0C 04 66						LODI.R0	66	#
0077	3E0E 08 66						ADDR.R0	BTODS+1	#
0078	3E10 C8 73						STRR.R0	BDS+1	#
0079	3E12 04 66						LODI.R0	66	#
0080	3E14 08 61						ADDR.R0	BTODS	#
0081	3E16 C8 1C						STRR.R0	BDS	#
0082	3E18 08 08						FPSL	WC	#
0083	3E1A D7						RRL.R3		#
0084	3E1C D2						RRL.R2		#
0085	3E1E 02 19						LODR.R0	BDS-2	#
0086	3E20 08 59						ADDR.R0	BTODS+2	#
0087	3E22 94						DAR.R0		#
0088	3E24 C8 56						STRR.R0	BTODS+2	#
0089	3E26 08 10						LODR.R0	BDS+1	#
0090	3E28 08 51						ADDR.R0	BTODS+1	#
0091	3E2A 94						DAR.R0		#
0092	3E2C C8 4E						STRR.R0	BTODS+1	#
0093	3E2E 08 08						LODR.R0	BDS	#
0094	3E30 08 49						ADDR.R0	BTODS	#
0095	3E32 94						DAR.R0		#
0096	3E34 C8 1D F7						STRR.R0	BTODS	#
0097	3E36 1E 17						BCTR.UN	BTODS	#
0098	3E38 08 08 08					BDS	RES	3	#
0099	3E37					*			
0100	3E37					*	CLEAR STATEMENT FIX		
0101	3E37					*			
0102	3E37 CC 13 4F					BASIC2	STRA.R0	CTEMP+1	*
0103	3E3A 3F 49 91						BSTA.UN	ERASE	*
0104	3E3D 1F 20 4C						BCTA.UN	BASIC3	*
0105	3E40					*			
0106	3E40 CC 08 6A					PURGET	STRA.R0	*TABEND	*
0107	3E43 1F 20 31						RCTA.UN	BASIC4	*
0108	3E46					*			
0109	2019						ORG	2019	*
0110	2019 3F 22 52						BSTA.UN	PURGES	*

FILE 'BASICMOD' AS ASSEMBLED BY SYSTEM ON 01-24-79

LINE	ADDR	B1	B2	B3	B4	LABEL	OPCODE	OPERAND	COMMENTS
0111	201C CC 01 2E						STRA.R0	PDATA	*
0112	201F CC 06 34						STRA.R0	ERWAIT	*
0113	2022 CC 0F 4B						STRA.R0	INON	*
0114	2025 CC 10 DB						STRA.R0	SP-1	*
0115	2028 CC 01 2E						STRA.R0	DATPNT+1	*
0116	202B CC 02 5F						STRA.R0	DOLFCR	*
0117	202E 1F 3F 37						BCTA.UN	BASIC2	*
0118	2031 0A 51					BASIC4	LODR.R2	IBST	*
0119	2033 CE 13 4E						STRA.R2	CTEMP	*
0120	2036					*			
0121	2049						ORG	BASIC3-3	*
0122	2049 17						RETC.UN		*
0123	204A					*			
0124	204A					*	BASIC12 VERSION 1.5 MODIFICATIONS		
0125	204A					*			
0126	204A					*	PROGRAM PATCHES		
0127	204A					*	PRINT @ FIX		
0128	204A					*			
0129	1796						ORG	1796	#
0130	1796 77 08					WHYAT	FPSL	WC	#
0131	1798 75 01						CPSI	CRY	#
0132	179A E1						RRL.R1		#
0133	179B 17						RETC.UN		#
0134	179C					*			
0135	179C					*	BASIC12 CHANGES		
0136	179C					*	PRINTL FIX		
0137	179C					*			
0138	22E3						ORG	22E3	*
0139	22E3 1B 03						BCTR.UN	PRNTLL	*
0140	22B5					*			
0141	22B5					*	PRINT @ FIX		
0142	22B5					*			
0143	2431						ORG	2431	#
0144	2431 3F 17 96						BSTA.UN	WHYAT	#

Jeff:

As usual with programs written against a deadline, a few errors crept into the Startrek game. Actually only the first item is an error, the other is an improvement in one of the functions. Here they are:

1517 68  
24C6 3F 17 61  
1761 3F 26 C4 3F 28 57 1F 2B B6

For tape dump purposes, memory used is now from A1510 to A1769 and A2000 to A3FPD. Program execution is still at A2000.

Mike Herbach



Antenna Computer program by Jerry Johnson

(AS) THIS PROGRAM WAS WRITTEN FOR THE CENTRAL-DATA 2650  
(AS) COMPUTER BY JERRY J JOHNSON ON 15 JUNE 78. IT IS  
(AS) WRITTEN IN 8K BASIC.

(AS)  
(AS) ANTENNA COMPUTER  
(AS)  
(AS) (AS) MEANS ASTERIX  
(AS) (PL) MEANS PLUS  
(AS) (AT) MEANS AT  
(AS) (PC) MEANS PERCENT  
(AS) (PD) MEANS POUNDS  
(AS) (LT) MEANS LESS THAN  
(AS) (GT) MEANS GREATER THAN  
(AS) (EQ) MEANS EQUAL

(AS)  
1 ERASE  
PRINT ' ANTENNA LENGTH FORMULAS, WHAT TYPE DO YOU WANT? '  
PRINT 'DIPOLE (ENTER DIPO)'  
PRINT 'VERTICAL (ENTER VERT)'  
PRINT 'FULL WAVE LOOP (ENTER LOOP)'  
PRINT 'QUAD/ DELTA LOOP (ENTER QUAD)'  
PRINT 'BEAM (ENTER BEAM)'  
INPUT B\$  
IF B\$(EQ)'DIPO' GOTO 5  
IF B\$(EQ)'VERT' GOTO 10  
IF B\$(EQ)'LOOP' GOTO 15  
IF B\$(EQ)'QUAD' GOTO 20  
IF B\$(EQ)'BEAM' GOTO 25  
ERASE  
PRINT 'I DON'T KNOW WHAT TYPE ANTENNA THAT IS, TRY AGAIN.'  
FOR Z(EQ)1 TO 20  
NEXT Z  
GOTO 1  
5 ERASE  
PRINT 'DIPOLE'  
INPUT 'FREQUENCY IN MEGAHERTZ(EQ)'  
L(EQ)468/F  
PRINT 'LENGTH OF DIPOLE IS 'L' FEET'  
INPUT 'WANT TO DO IT AGAIN ? (Y/N)'  
IF A\$(EQ)'Y' GOTO 5  
GOTO 100  
10 ERASE  
PRINT 'VERTICAL'  
INPUT 'FREQUENCY IN MEGAHERTZ(EQ)'  
L(EQ)234/F  
I(EQ)240/F  
PRINT 'VERTICAL(EQ) 'L' FT'  
PRINT 'RADIALS(EQ) 'I' FT'  
INPUT 'DO YOU WANT TO DO IT AGAIN ? (Y/N)'  
IF A\$(EQ)'Y' GOTO 0  
GOTO 100  
15 ERASE  
PRINT 'LOOP'  
INPUT 'FREQUENCY IN MEGAHERTZ(EQ)'  
L(EQ)1005/F  
PRINT ' LENGTH(EQ) 'L' FEET CIRCUMFERENCE OR '  
PRINT 'L/4' FEET PER SIDE FOR FOUR SIDES OR '  
PRINT 'L/3' FEET PER SIDE FOR THREE SIDES'  
INPUT 'DO YOU WANT TO DO IT AGAIN ? (Y/N)'  
IF A\$(EQ)'Y' GOTO 15

GOTO 100  
20 ERASE  
PRINT 'QUAD/LOOP'  
INPUT 'FREQ(EQ)'  
L(EQ)1005/F  
D(EQ)963/F  
R(EQ)1030/F  
PRINT 'DRIVEN ELEMENT (QUAD)'L/4' FEET PER SIDE'  
PRINT 'REFLECTOR (QUAD)'R/4' FEET PER SIDE'  
PRINT 'DIRECTOR (QUAD)'D/4' FEET PER SIDE'  
PRINT  
PRINT 'DRIVEN ELEMENT (DELTA)'L/3' FEET PER SIDE'  
PRINT 'REFLECTOR (DELTA)'R/3' FEET PER SIDE'  
PRINT 'DIRECTOR (DELTA)'D/3' FEET PER SIDE'  
PRINT  
PRINT 'DRIVEN ELEMENT (TOTAL CIRCUMFERENCE)'L' FEET'  
PRINT 'REFLECTOR (TOTAL CIRCUMFERENCE)'R' FEET'  
PRINT 'DIRECTOR (TOTAL CIRCUMFERENCE)'D' FEET'  
INPUT 'DO YOU WANT TO DO IT AGAIN ? (Y/N)'  
IF A\$(EQ)'Y' GOTO 20  
GOTO 100  
25 ERASE  
PRINT 'BEAM'  
INPUT 'FREQUENCY IN MEGAHERTZ(EQ)'  
L(EQ)476/F  
D(EQ)450/F  
R(EQ)501/F  
S(EQ)120/F  
PRINT  
PRINT 'DRIVEN ELEMENT (TOTAL TIP-TO-TIP) 'L' FEET'  
PRINT 'REFLECTOR (TOATAL TIP-TO-TIP) 'R' FEET'  
PRINT 'DIRECTOR (TOTAL TIP-TO-TIP) 'D' FEET'  
PRINT 'OPTIMUM SPACING (.2 WAVELENGTH) 'S' FEET'  
INPUT 'DO YOU WANT TO DO IT AGAIN ? (Y/N)'  
IF A\$(EQ)'Y' GOTO 25  
100 INPUT 'DO YOU WANT TO TRY ANOTHER ANTENNA TYPE ? (Y/N)'  
IF A(EQ)'Y' GOTO 1  
ERASE  
PRINT 'THANK YOU'  
STOP

Backup program by Ray Krygier

0003	0000	PRNT				
0004	0000	R0	EQU	0		
0005	0000	R1	EQU	1		
0006	0000	R2	EQU	2		
0007	0000	R3	EQU	3		
0008	0000	CUR1	EQU	17FE		
0009	0000	CUR2	EQU	17FF		
0010	0000	LFCR	EQU	0024		
0011	0000	WCHR	EQU	0396		
0012	0000	UN	EQU	3		
0013	0000	EQ	EQU	0		
0014	0000	LT	EQU	2		
0015	0000	GT	EQU	1		
0016	0000	KBIN	EQU	030F		
0017	0000	INHX	EQU	01B6		
0018	0000	HXDT	EQU	006A		
0019	0000	SUMC	EQU	17E9		
0020	0000	ADD1	EQU	17FA		
0021	0000	ADD2	EQU	17FB		
0022	0000	LENT	EQU	17EA		
0023	0000	COCD	EQU	3A		
0024	0000		PRNT			
0025	0000	*				
0026	0000	*****	BACKUP	*****		
0027	0000	*				
0028	1510		ORG	1510		
0029	1510	*				
0030	1510 04 02	BEGIN	LODI,R0	02	INIT PSW	
0031	1512 93		LPSL			
0032	1513 04 40		LODI,R0	40		
0033	1515 92		LPSU			
0034	1516	*				
0035	1516 3F 16 02		BSTA,UN	CLSCR	CLEAR SCREEN	
0036	1519	*				
0037	1519 05 20		LODI,R1	20	POSITION CURSOR	
0038	1518 CD 97 FE		STRA,R1	*CUR1		
0039	151E 05 10		LODI,R1	10		
0040	1520 CD 17 FE		STRA,R1	CUR1		
0041	1523 05 41		LODI,R1	41		
0042	1525 CD 17 FF		STRA,R1	CUR2		
0043	1528 05 1C		LODI,R1	1C		
0044	152A CD 97 FE		STRA,R1	*CUR1		
0045	152D	*				
0046	152D 05 FF	MS01	LODI,R1	FF	WRITE MSG #1	
0047	152F 0D 36 10	MS01A	LODA,R1	MSG1,+		
0048	1532 18 06		BCTR,EQ	INPUT		
0049	1534 C3		STRZ,R3			
0050	1535 3F 03 96		BSTA,UN	WCHR		
0051	1538 1B 75		BCTR,UN	MS01A		
0052	153A	*				
0053	153A 3F 01 B6	INPUT	BSTA,UN	INHX	WAIT FOR REPLY	
0054	153D CF 16 0F		STRA,R3	ENDADR		
0055	1540	*				
0056	1540 3F 00 24	MS02	BSTA,UN	LFCR	WRITE MSG #2	
0057	1543 05 FF		LODI,R1	FF		
0058	1545 0D 36 33	MS02A	LODA,R1	MSG2,+		
0059	1548 18 06		BCTR,EQ	GO		

0060	154A C3				STRZ,R3	
0061	154B 3F 03 96				BSTA,UN	WCHR
0062	154E 1B 75				BCTR,UN	MS02A
0063	1550 3F 03 0F	GO			BSTA,UN	KBIN
0064	1553	*				WAIT FOR OPERATOR RESPONSE
0065	1553 04 08				LODI,R0	08
0066	1555 F0				WRTO,R0	START TAPE
0067	1556 05 10				LODI,R1	10
0068	1558 07 60	DL2			LODI,R3	60
0069	155A 3F 02 78	DL3			BSTA,UN	Q278
0070	155D FB 78				BDRR,R3	DL3
0071	155F F9 77				BDRR,R1	DL2
0072	1561	*				
0073	1561 04 20				LODI,R0	20
0074	1563 CC 16 0E				STRA,R0	SAVE
0075	1566	*				
0076	1566 3F 16 02	START			BSTA,UN	CLSCR
0077	1569 05 FF				LODI,R1	FF
0078	156B 0D 36 55	MS03A			LODA,R1	MSG3,+
0079	156E 18 06				BCTR,EQ	STR1
0080	1570 C3				STRZ,R3	
0081	1571 3F 03 96				BSTA,UN	WCHR
0082	1574 1B 75				BCTR,UN	MS03A
0083	1576	*				
0084	1576 0C 16 0E	STR1			LODA,R0	SAVE
0085	1579 CC 17 FA				STRA,R0	ADD1
0086	157C C2				STRZ,R2	
0087	157D 3F 00 6A				BSTA,UN	HXOT
0088	1580 20				EORZ,R0	
0089	1581 CC 17 FB				STRA,R0	ADD2
0090	1584 C2				STRZ,R2	
0091	1585 3F 00 6A				BSTA,UN	HXOT
0092	1588 04 FF				LODI,R0	FF
0093	158A CC 17 EA				STRA,R0	LENT
0094	158D 3F 15 C7				BSTA,UN	TAPOUT
0095	1590	*				WRITE IT TO TAPE
0096	1590 0C 16 0E				LODA,R0	SAVE
0097	1593 84 01				ADDI,R0	01
0098	1595 CC 16 0E				STRA,R0	SAVE
0099	1598 EC 16 0F				COMA,R0	ENDADR
0100	159B 9D 15 66				BCFA,GT	START
0101	159F	*				
0102	159E 20				EORZ,R0	
0103	159F CC 17 FA				STRA,R0	ADD1
0104	15A2 CC 17 FB				STRA,R0	ADD2
0105	15A5 CC 17 EA				STRA,R0	LENT
0106	15A8 3F 15 C7				BSTA,UN	TAPOUT
0107	15AB	*				WRITE IT TO TAPE
0108	15AB 04 00				LODI,R0	00
0109	15AD F0				WRTO,R0	STOP TAPE
0110	15AE	*				
0111	15AE 3F 00 24				BSTA,UN	LFCR
0112	15B1 3F 00 24				BSTA,UN	LFCR
0113	15B4 05 FF				LODI,R1	FF
0114	15B6 0D 36 6D	MS04			LODA,R1	MSG4,+
0115	15B9 18 06				BCTR,EQ	STOP
0116	15BB C3				STRZ,R3	
0117	15BC 3F 03 96				BSTA,UN	WCHR
0118	15BF 1B 75				BCTR,UN	MS04
0119	15C1 3F 03 0F	STOP			BSTA,UN	KBIN
						ANY KEY TO RETURN TO 2000