SPHERE CORPORATION
PO BOX 129,
BOUNTIFUL UTAH 84010

GLOBAL NEWS

Vol. 1, No. 3
June 1976

THE SWAP NEWSLETTER FOR THE WHOLE EARTH

From the Editors Desk

Even though we are all settled into our new building, hammers and saws have been interrupting our phone calls and work for the past month. Even the SPHERE mobile home is being used as office space occasionally. SPHERE is growing faster than we can have new offices constructed.

Documentation is improving! Currently the SPHERE Operators manual is being re-written with specific "push this button, now push that one" instructions. Complete instructions for SDOS and Basic will be done soon after the Op manual is finished. Schematics are being redone at the rate of one-a-month. Currently we have a very clear layout and schematics for KBD/2 and CPU/2, and printed (not copied) layout and parts list for SIM/1. Blue-line schematics are available for $10.00 per page plus $5.00 handling. The blue prints are full size copies of the originals and are available only on a limited basis to users. For you to make copies of these prints would be a clear infringement of copyright. New assembly instructions for each and every board won't be far behind. We will send upgraded standard documentation to any user who sends a SASE and $1.00 copy fee with his request. When the Op manual is complete we will send it to users who request it for a $5.00 copy fee.

We have a double edged sword that we would like to make into a plowshare. On one hand we have users with programming skill, while on the other we keep hearing from users and potential users who need programming help. To both parties we direct your attention to the Bits and Bytes column.

Many people have asked us for the parts for ECOs. This month we have begun to prepare ECO Kits. The price may be a little high, but it does include the handling fee. Our policy is since the ECOs are mostly reliability improvements and general inhancements, not system failure problems, that we can not retrofit every system ever made.

There has been a slacking off of program submittals lately so the management feels we should hold off on our prizes for this newsletter. If possible the prizes will be awarded next time, we need to have your submittals! Prize winning entries will be published in the newsletter. All others will be mentioned and will be available upon request. Please note however that before we publish it or make it available we will try (to break) it. This may delay the printing of your submittal a while but hang in there, it will be available.

Commencing with the first newsletter after the new Op manual is printed, the newsletter will be aimed at manual updates also. The newsletter is meant to be filed into your Op manual's appendix section for reference. The newsletter's page numbers will be modified so they will be uniform with the appendix page numbers. Just hang on, we are really going to do this thing RIGHT!!

Ernest Dixon
Newsletter Editor

PLEASE NOTE

SPHERE mailing address:
P.O. Box 129
Bountiful, Utah 84010

shipping address and plant location:
940 North 400 East
North Salt Lake, Utah 84054

Dear Sphere Users:

The lead time for system 310 thru 330 Kits has been reduced to under 30 days for prepaid orders. The Assembly and testing department has been tripled in size to reduce lead times on assembled units. We have added a customer relations department in our continuing effort to produce "Satisfied Customers". The comment was made by one publisher recently that Sphere Corporation was the sweetheart of the industry in trying to help their customers. However, I must sound a small note of warning. It is possible to wear out your otherwise welcomed phone calls when it becomes apparent that we are doing your assembly or troubleshooting for you. To date, there are only a few customers out of nearly a thousand who have done so. One is obviously just harrassing us with threats, sarcasm, etc. even after may personal attempts to solve his needs. The others just call for the socializing, I think. Should it take 2 calls per day four days per week for four weeks? No wonder the rest of you have to wait so long. Mr. Waltner a user who almost never called, said of his kit, "I finally received the rest of my computer parts and have it up and running. It works great!" That was on April 20th and might have been April 10th except for the time used to help the few "cranks" amoung our users. Don't let this message discourage your call when you need help.

We appreciate our distributors and their help. Feel free to order new parts through them and consult with them on your troubles.

The new ROM board is just one example of our continued effort to add to our system. We are now announcing new power supplies made by Standard Power. One version is open frame and light duty. The other is enclosed with cord, fuse, over voltage protection, and heavy duty. The use of these "vendor provided" supplies will allow voltage adjustments to offset line drops. But watch out for the cost. That cost factor is why the initial Sphere Systems featured our own version. These new supplies may cause a considerable increase in retail system prices in the future.

Monroe Tyler
President

OOPS

As most SPHERE Basic users have found out, the Sys 2N ROM (E3 on SIM board) has a bug in it. It so happens that when any block's check sum equals 16_X, which is the case of block B6, the next block is missed (block B7 is not found). By rewinding some and re-reading the missed block the problem can be detoured. This problem has been corrected in the Sys 2 NF ROM (also known as P/N 000105 NR) and we will be glad to exchange proms if you will send in your Sys 2N ROM or send a $25.00 deposit and we will send you a new ROM. Since this is our goof (heaven forbid), no service/handling charge will be necessary.

Management

The management of SPHERE Corporation is pleased to announce the creation of the Customer Relations Department with Ernest Dixon as manager. Ernie was one of our first customers from Southern California and has been serving as a field representative there. Welcome to UTAH Ernie.

Doug Hancey
Executive Vice President

Missing parts in your order!!

To help keep our costs (and your price) down and to improve our quality control, we request the following information when you call to inform us of shortages in your shipment:

1) Your name, address and invoice number
2) What parts you are missing
3) What parts you got extra

Please be specific - what board, location, part description, and quantity; SIM/1, E1 and E17,7404, 2 of them.

We will be happy to send your missing parts free of handling charge ONCE, however, if you call again, we will have to charge you for the parts and add the standard $5.00 handling fee and shipping charges. We don't say that we are perfect, quite often parts end up in the wrong bin and are shipped incorrectly. We feel we can ask you to check your entire order at one time and only call us once for this problem.

Review of My Computer Likes Me when i speak in basic!

"My Computer Likes Me..." provides a good introduction to Basic programming. Starting with simple print commands, it takes the reader through arithmetic operations, strings, variables and scientific notation. Using a sample problem on population growth it continues with manual data input, formatting, data statements, go to, and if/then statements. It concludes with descriptions and examples of for/next loops, subscriped variables (single and double) and subroutines.

The book is written with a humorus overtone and has sufficient detail and examples to lead even the novice to the point where he should be able to write very powerful programs.

Although it is aimed at the beginner, the book should provide a good review of the fundamental Basic operations for anyone who may be somewhat rusty using Basic and at $2.00 it is certainly a bargain.

R. S. Mason

Q. What is the status of the Sys 2 ROM set mentioned in the last newsletter?

A. Due to the fact that the Sys 2 PDS software does not contain a mini-assembler in ROM, SPHERE does not feel that this piece of software can be sent out as standard PDS software for our systems. When this software is completed, probably a month or so, it will be available to our system owners for a copying fee of $40.00 if you send in your PROMs (or a $70.00 deposit) or may be purchased ROMs and all for $120.00 This purchase will include the mini-assembler on cassette, complete instructions and for an added $6.00 a source listing of both the proms and tape. If you are ready to send your money now, specify which keyboard you have (KBD/1 or KBD/2).

Q. A friend of mine wants a copy of Basic. Is this program copyrighted?

A. All documentation, software, and hardware manufactured by SPHERE is either copyrighted or marked PROPRIETARY MATERIAL. SPHERE does not feel that software should be hidden under a bushel, however, to drop it from a plane as confetti is expensive. We want our software, as well as yours, to be available to others, but we feel we have the right to know who has it, just as you have the right to know who has the software you developed. We don't charge qualified SPHERE users for development of new software, however we will charge a non-SPHERE user for that software, we charge everyone the copy fee. What we are really trying to say is use our software, but let us copy it for you at a minimal cost so that we have it in our records and can verify that you get what you really need. See note on software prices.

Q. I have an application I might like to market. Would SPHERE be able to help me announce my development to the SPHEREiverse?

Q. I would like to get in touch with other users in my area. Can SPHERE send me a list of those close by?

A. Sorry, we can not send out our users list to everyone, some users are developing things that the people they work with might not like or just don't want to be bothered. Beginning with this issue, we will devote some space to users and their systems. If you have an application, hardware modification, announcement of some kind, or just plain want others to know you are out there, tell us. See the Bits and Bytes column.

Q. Why doesn't the REPEAT key on my new KBD/2 do anything?

A. The repeat key on your keyboard does its job quite well, when it is pushed the 2^7 data bit turns on. The job of reading the PIA for that bit isn't difficult either - the problem is that there just isn't any room in the ROM to implement it. We expect to have it in the Sys 2 ROMs when they are finished.

Q. My assembled system didn't work on arrival. Do you guys really check these systems ???

A. YES!!! All assembled boards and systems are tested thoroughly, however in shipment things tend to come loose. Cables come out of sockets, chips fall out, and

A. (continued)
yes, we even have had a couple of broken CRTs reported to us. Sometimes while putting chips back in their sockets you notice a bent pin and claim we don't test things, but realize that while a bent pin won't be holding the chip down, it is still making contact. Our recently enlarged Quality Control Department is on the lookout for improperly seated components. When you get your assembled unit, DO NOT sign the shipper's release until you unpack, open, plug in, and use the unit. Mark on the shipping papers the condition of the machine, not the carton. ALL systems were working when they were shipped !!!!!!!!!

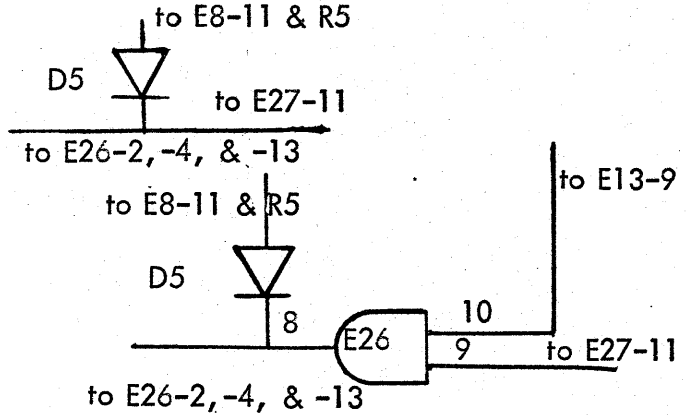
Engineering Change Orders

Beginning immediately parts for ECOs are available in kit form. Reference Global News Vol. 1 No. 2 for details on ECOs 1 through 10.

ECO 1	no kit
ECO 2	$5.50
ECO 3	$7.00
ECO 4	$5.50
ECO 5	$5.25
ECO 6	$5.30
ECO 7	$8.00
ECO 8	$5.75
ECO 9	no kit
ECO 10	$5.50
ECO 11	no kit

Submitted by Richard S. Mason
1037 Park Hill Lane
Escondido, Calif. 92025

CRT/1 to decrease white hash (This mod provides a compromise between that of ECO 1 and ECO 2. Ed.)



Bits and Bytes from our friends

In this new column of Global News, we will feature free announcements from our users. If you want to place an announcement write to Bits and Bytes, Global News, P.O. Box 129, Bountiful, Utah 84010.

We reserve the right to edit and comment on submittals to this column. Address inquires directly to the submitter, not to Global News, or SPHERE Corporation.

Graphics

Black and white or Color graphics with 128 by 128 or 256 by 256 resolution is available from Vocor Corp. 96 East 500 South, Bountiful, Utah 84010. Color requires 3 graphics boards. Each board $169.50, deliveries said to begin in June 1976. Plotting software included (We have yet to see it work, but the developer has done good work before. Ed.)

Users Group

Sphere User's Group of the Southern Calif. Computer Society:
Warren Weimer, President, 23025 Kinnard Ave., Carson, Ca. 90745 (213)835-9417.

Utah Chapter of Southern California Computer Society: contact
Wayne Bates, Organizing committee chairman, (801) 376-2525.
Mike Wise (801)292-8159 or Ernie Dixon (801) 467-9100. Next meeting scheduled June 19 in or around Salt Lake City.

Lonely Users

Vincent Johnston, 402 Wildwood Avenue, Pitman, N.J. 08071 (609) 589-3461

Gary Shell, 3731 Hyde Park Avenue, Cincinnati, Oh. 45209 (513)531-1343 have working DAA Modem

Computer Widows

Jerrie Raehl, 943 Begonia, Escondido, Ca. 92027 (was married May 29, Jim was talking computers less than an hour after "I Do").

THE PART NUMBER GAME

In an effort to keep track of our Stock and help you get the parts you need promptly, SPHERE is in the process of assigning Part Numbers to just about everything you can order. If you need something, please check the serialization of "The Part Number Game", to see if it is listed.

Part number	Old designation	Use		Remarks
000100NR	PDSV3A	CPU/1 E6	CPU/2 E6	Use with KBD/1
000100A	PDSV3D	CPU/1 E6	CPU/2 E6	Use with KBD/1
000101NR	PDSV3A	CPU/1 E12	CPU/2 E12	Use with KBD/1
000101A	PDSV3N	CPU/1 E12	CPU/2 E12	
000102NR	PDSV3A	CPU/1 E20	CPU/2 E20	Use with KBD/1
000102A	PDSV3N	CPU/1 E20	CPU/2 E20	
000103NR	PDSV3A	CPU/1 E33	CPU/2 E35	Use with KBD/1
000103A	PDSV3N	CPU/1 E33	CPU/2 E35	
000104NR	PDSV3N	CPU/1 E6	CPU/2 E6	Use with KBD/2
000105NR	SYS2NF	SIM/1 E3		
000106NR	SDOS FC-9	CPU/1 E6	CPU/2 E6	
000107NR	SDOS FD-4	CPU/1 E12	CPU/2 E12	
000108NR	SDOS FE-10	CPU/1 E20	CPU/2 E20	Use with KBD/1
000109NR	SDOS FF-11	CPU/1 E33	CPU/2 E35	Use with KBD/1
000110NR	SDOS FE-12	CPU/1 E20	CPU/2 E20	Use with KBD/2
000111NR	SDOS FF-17	CPU/1 E33	CPU/2 E35	Use with KBD/2
000115NR	1 card computer	CPU/2 E20		
000116NR	1 card computer	CPU/2 E35		

Suffix designations:
NR = No Revision
alpha character = revision level

SOFTWARE AVAILABLE FOR SPHERE SYSTEMS

All require signed non-disclosure agreement and pre-payment before shipment.

NAME	MEDIUM	MIN SYS*	USER COST	NON-USER COST	REMARKS
BASIC version 1	CT	330	$10.	$300.	1 copy free to Min Sys users
	PT	330	25.	325.	
	D	340	25.	325.	
BASIC extended 16K	CT	330	10.	1000.	1 copy free to Min Sys users upon return of Basic Ver 1.
	PT	330	25.	1000.	
	D	340	25.	1000.	
8K	CT	330	50.	800.	
	PT	330	75.	800.	
	D	340	75.	800.	
4K	CT	320	50.	600.	
	PT	320	75.	600.	
	D	340	75.	600.	
	ROM	310	1000.	1500.**	
MEMORY TEST PROGRAM	CT	330	400.	600.	
	PT	330	450.	550.	
	D	340	410.	610.	
	ROM	310	600.	800.	
PERRY ASSEMBLER	CT	320	10.	500.	
	PT	320	35.	525.	
	D	340	25.	510.	
	ROM	310	360.	860.	
I/O HANDLER ROUTINES	CT	320	400.	600.	
	PT	320	425.	625.	
	D	340	410.	610.	
	ROM	310	500.	800.	

CT=Cassette Tape PT=Paper Tape D=Floppy Disk ROM=1702 Read Only Memories

* = User Minimum system, the sum of all modules purchased from SPHERE, ie. a 310 user does not qualify for a $50. copy of 4K Basic on cassette, but will have to pay $600. If he buys a SIM board he now has a system 320 and can get Basic for $50. System 330 users who buy a printer and floppy disk from SPHERE qualify for both 330 and 340 software (they must pay the copy fee on SDOS). The reverse also applies to 340 users who buy a SIM board. To buy a 4K MEM board and fill it up with outside chips does not qualify one to the things available to those buying a SPHERE 16K MEM board. Having the hardware does not qualify, it must be bought from SPHERE.

** = Includes ROM/1 board

Effective on all System sales after April 1, 1976, and on all System upgrades after January 1,1976.

Note that the assembler is in two parts. The second part is designed to overlap the last instruction of the first part, so that label 'Y' refers to the same location in both parts. The first part can be used alone to assemble a program in mnemonic format, but this will change the source text to Sphere mini-assembler format. The second part is required if you want the source text to end up in mnemonic format. The second part, when used alone, will translate mini-assembler text into mnemonic format. Note that two listings of each part have been included. One listing is in mini-assembler format and the other is in mnemonic format. Also included is a hexadecimal memory dump of the entire program (parts 1 and 2 plus mnemonic table).

Unfortunately, this program requires a 16K memory board, but it could be moved into low memory if care is taken to limit the text buffer size so as not to interfer with the assembler. The program occupies 1792_{10} bytes of memory, of which 1024_{10} bytes are used for the mnemonic table. A description of the program and its use is also enclosed.

For convenience in entering the mnemonic table I have included a mnemonic table loader. This program need be used only once, since the table can be stored on tape along with the assembler, once it has been entered.

If, for convenience, you would like paper tapes of some of these programs, I can furnish them, provided you can give me details on the format you want. I use my own unconventional format.

Submitted by

Anthony Pierry
90 Ruxton Street
Uniondale, N.Y. 11553

(This implementation is for PDS V3A, to use with PDS V3N, all references to routines in PDS should be checked. Ed.)

Program Description

The program consists of two major parts, the mnemonic format to Sphere format translator and the Sphere format to mnemonic format reverse translator. The mnemonic to Sphere translator scans the buffer starting at location 200, and looks first for an equate statement ('=') or an 'END' (column 1 does not contain either '=' or a blank). If an equate statement is found or a line which contains only blanks or comments is encountered, then nothing is changed on that line and the next line is scanned. If the 'END' is encountered then the mnemonic to Sphere translation is finished and the mini-assembler is called in. If the line contains a mnemonic, the mnemonic is stored in locations C0-C3. Next, column six is examined to begin deciphering the addressing mode. If column six contains '%' then the direct mode has been selected. A subroutine is entered which replaces the '%' with a 'D', and then a search subroutine is called to search the mnemonic table for the mnemonic previously stored at locations C0-C3. It should be noted here that only regions of the table that contain direct addressing mode instructions are searched. This avoids ambiguities (i.e. the same mnemonic is repeated in the table for each addressing mode that applies to that instruction) and also speeds up program execution.

To understand how this search is accomplished it is necessary to understand how the table is organized. Only four bytes are reserved for each instruction. This leaves room for up to four characters for each mnemonic but it does not leave any room to store the op-codes. However, if we store each mnemonic in order of its op-code, leaving blanks for unimplemented op-codes we can then compute the op-code, since it will be a function of the location where the mnemonic is found. The table starts at location 4A00, so if the mnemonic is found at location L then the op-code (O) is given by O=(L-4A00)/4. This computation takes place in the search subroutine.

However, since a given mnemonic may occur in more than one place in the table, the search routine must search only the parts of the table which correspond to the desired addressing mode op-codes. The search subroutine searches the table from the location pointed to by the index register to the location pointed to by the contents of location D0. The op-code is returned in accumulator A in binary. If the mnemonic was not found in the table a zero is returned in accumulator A. Since instructions with the same addressing mode occur together in groups in the table, only a few regions of the table need be searched once the addressing mode is known. In the case of the direct addressing mode, for example, the table is searched from 4C40 to 4C80 and from 4D40 to 4D80. In the case of branch instructions the BSR instruction (op-code 8D) is treated as a special case since it is in the middle of a group of immediate mode instructions. Other special cases are LDX, CPX, and LDS immediate mode, which are three byte instructions whereas all other immediate mode instructions require two bytes. These are the only cases requiring special treatment.

Determining the addressing mode is not always that simple. If column six contains a blank then we must determine whether the mode is extended, relative, indexed, or inherent. If an operand is absent (column 7 is blank) then then inherent mode is assumed, otherwise columns 8 and 9 are examined to detect the indexed mode. If neither of these modes are detected then the first character of the mnemonic is tested. If it is the letter 'B' then the relative mode is assumed. If the addressing mode has not yet been detected, then it must be the extended mode by default. Once the proper mode is known, the table can be searched, the op-code can be put into columns 3 and 4, columns 2 and 5 can be replaced with blanks and the proper character (E, D, or R) can be put into column 6. If the instruction was an indexed mode, then columns 8 and 9, or 9 and 10 are given blanks to erase the ',X' characters.

The Sphere format to mnemonic format reverse translator begins in the same way as the mnemonic to Sphere translator, by skipping over equate statements or lines without instructions, and looking for the 'END' statement. The op-code is then placed in location D0 and the location of the mnemonic is computed using L=(4*0) +4A00. Next, the mnemonic found at that location is written into columns 2 through 5. Then column 6 is examined. If an 'R' is present we simply remove it and we are done. Otherwise, we look for an 'E'. If an 'E' is present we look at the op-code previously stored at D0. If it is CE, 8C, or 8E we replace the 'E' in column 6 with a '#'. Otherwise, we put a blank in column 6 to remove the 'E'. We are then finished with this line. If instead a 'D' was present in column 6 we must determine whether we have a direct, immediate, or indexed instruction. This is determined by the range in which the op-code happens to fall. For example, if the op-code lies between 70 and 90 then the instruction mode is immediate and we replace the 'E' with '#' and go to the next line. If the indexed mode is detected then we must remove the 'E' and place the characters ',X' after the operand. This is all there is to the reverse translation.

Simple Mnemonic Assembler
by Anthony Pierry

This program is simple because it utilizes the Sphere P.D.S. mini-assembler to do the actual assembly. Therefore, this program isn't really an assembler at all. It is in fact a translator whose primary function is to enable the Sphere mini-assembler user to assemble programs without having to look up the op-codes or concern himself with the number of bytes used by an instruction (E, D, or R). This is accomplished in the program by a very simple process. The user simply loads his program, using the editor, in a format similar to the Sphere mini-assembler format but with mnemonics replacing op-codes. The Simple Mnemonic Assembler program is then run. The user's program is automatically translated into Sphere mini-assembler format, the mini-assembler is called in to do the actual assembly, and then the program is automatically translated back into the original mnemonic format.

The entire assembler with mnemonic table occupies 1792_{10} bytes of which 1024_{10} bytes are used for the mnemonic table. The mnemonic program format is illustrated below in comparison with the Sphere mini-assembler format. Note that the '@' character is still used to refer to a label as an operand but the characters E,D, and R are not used. The '#' symbol is used to indicate the immediate addressing mode. The '%' symbol is used to indicate the direct addressing mode. The operand followed by ',X' indicates the indexed mode. If none of these characters appears the extended, implied, or relative modes will be assumed (depending upon the particular instruction, of course). Note also that the characters must appear in the proper places, just as when using the mini-assembler and that you must space out past column seven or instructions that have no operand.

```
0123456789
A LDAA#1F
  STAA 2000
  CLRA
  CMPB%F0
  BNE @A
  LDAA 0,X
  LDX #10A1
END
```

```
0123456789
A  86 D1F
   B7 E2000
   4F
   D1 DF0
   26 R@A
   A6 D0
   CE E10A1
END
```

Be careful! No errors are flagged, so if you make a syntactical error you may not get back exactly the program you put in. Use Re-edit to check your program after it has been assembled.

Simple Mnemonic Assembler

Instructions For Use:

1) Enter user program in mnemonic format using editor.
2) Scroll up and hit 'ESC' as usual
3) Call debug (Ctrl D) and open location 4700 HEX
4) Hit 'G' key
5) When cursor returns, program will be assembled
6) Call Re-edit to examine the buffer. Program should appear unmodified.
7) Return to debug to run user program

NOTES:

1) Re-edit should be called using the following program:

```
BD EFC68
7E EFC18
END
```

   This prevents a run-away condition when 'ESC' is pressed, which could cause problems.
2) In this version of Simple Mnemonic Assembler the comma character may not be used as a label, since it interferes with detection of the indexed addressing mode.

Mnemonic Table Loader

Instructions For Use:

1) Open location 2000 and hit 'G'
2) A '0' should appear followed by a space and a blinking cursor
3) Since 0 is an unimplemented op-code, hit 'ESC' key.
4) A '1' should appear on the next line
5) Type NOP and then 'ESC' key
6) A '2' should appear on the next line
7) 2 is unimplemented so hit 'ESC' key
8) Continue this process, typing the correct mnemonics must be entered more than once. For accumulator instructions do not leave an intervening blank.
   ex: type LDAA not LDA A

```
                MNEMONIC TABLE
                   LOADER
                   1/13/76
=       2000  BY ANTHONY PIERRY
    C6 D10  /SET BASE TO .16
    D7 D5   /STORE IN ARB
    5F
    D7 D4
    D7 DE0  /INITIALIZE OP CODE
    CE E4A00 /BEGINNING OF TABLE
A   DF DD0  /TABLE POINTER
    BD EFD16 /CRLF
    DE D1C  /CURSOR POINTER
    96 DE0  /CURRENT OP CODE
    5F
    BD EFF64 /PRINT OP CODE
    DF D1C  /ADVANCE CURSOR
    36 D20
    BD EFCAD /PRINT 1 BLANK
    7C EE0
    BD EFC6E /GET MNEMONIC
    DE D24  /SCNPTR
    A6 D3   /4TH LETTER
    36      /SAVE ON STACK
    A6 D2   /SAME FOR REST OF...
    36      /MNEMONIC
    A6 D1
    36
    A6 D0
    DE DD0  /TABLE POINTER
    A7 D0   /PUT 1ST CHAR IN
    32      /GET NEXT CHAR
    8       /UPDATE POINTER
    A7 D0   /SAME FOR REST OF...
    32      /MNEMONIC
    8
    A7 D0
    32
    8
    A7 D0
    8
    20 B0A  /NEXT OP CODE
END
```

```
                MNEMONIC TABLE
                   LOADER
                   1/18/76
=       1000  BY ANTHONY PIERRY
 LDAB#10  /SET BASE TO .16
 STAB%5   /STORE IN ARB
 CLRB
 STAB%4
 STAB%E0  /INITIALIZE OP CODE
 LDX #4A00 /BEGINNING OF TABLE
A STX %D0  /TABLE POINTER
 JSR  FD16 /CRLF
 LDX %1C  /CURSOR POINTER
 LDAA%E0  /CURRENT OP CODE
 CLRB
 JSR  FF64 /PRINT OP CODE
 STX %1C  /ADVANCE CURSOR
 LDAA#20
 JSR  FCAD /PRINT 1 BLANK
 INC  E0
 JSR  FC6E /GET MNEMONIC
 LDX %24  /SCNPTR
 LDAA 3,X /4TH LETTER
 PSHA     /SAVE ON STACK
 LDAA 2,X /SAME FOR REST OF...
 PSHA     /MNEMONIC
 LDAA 1,X
 PSHA
 LDAA 0,X
 LDX %D0  /TABLE POINTER
 STAA 0,X /PUT 1ST CHAR IN
 PULA     /GET NEXT CHAR
 INX      /UPDATE POINTER
 STAA 0,X /SAME FOR REST OF...
 PULA     /MNEMONIC
 INX
 STAA 0,X
 PULA
 INX
 STAA 0,X
 INX
 BRA  @A  /NEXT OP CODE
END
```

```
                SIMPLE
          MNEMONIC ASSEMBLER
               -PART 1-
          (MNEMONIC TO SPHERE)
               VERSION 1.1
                 1/25/76
=      4700  BY ANTHONY PIERRY
   CE E1FF
L  8      /LOC BEGINNING OF LINE
   DF DF0 /SAVE IT
   8       /COLUMN 1
   A6 D0
   81 D3D  /'='?
   26 R@M
A  8      /LOOK FOR END OF LINE
   A6 D0
   81 D60
   26 R@A
   20 R@L /FOUND END OF LINE
M  81 D20 /COLUMN 1 IS BLANK?
   27 R@Z
   BD EFDA1 /CALL PDS ASSEMBLER
   7E E@Y /REVERSE TRANSLATE
Z  8
   A6 D0
   81 D20 /IS LINE BLANK?
   27 R@A /YES->GET NEXT LINE
   97 DC0 /STORE MNEMONIC IN...
   8       /LOCATIONS C0-C3
   A6 D0
   97 DC1
   8
   A6 D0
   97 DC2
   8
   A6 D0
   97 DC3
   8       /COLUMN 6
   A6 D0
   81 D25 /'%'?
   26 R@@
   7E E@D /YES,GO TO DIRECT
@  81 D23 /'#'?
   26 R@*
   7E E@I /YES,GO TO IMMEDIATE
*  8       /COLUMN 7
   A6 D0
   81 D20 /NO OPERAND?
   26 R@%
$  7E E@0 /NO,GO TO INHERENT
%  81 D60 /NO OPERAND?
   27 R@$
   8       /COLUMN 8
   A6 D0
   81 D60 /NOT INDEXED?
   27 R@W /YES GO TO DECISION
   81 D2C /COMMA?
   27 R@X /YES->GO TO INDEXED
   8       /COLUMN 9
   A6 D0
   81 D2C /COMMA?
```

```
W  86 D42 /EXTENDED OR RELATIVE
   91 DC0 /FIRST LETTER=B?
   27 R@R /YES->RELATIVE
   7E E@E /NO, EXTENDED
R  8D R@+ /CALL RELATIVE
\  7E E@F /GO TO FINISH
X  8D R@. /CALL INDEXED
   20 R@\
I  BD E@; /CALL IMMEDIATE
   20 R@\
D  BD E@: /CALL DIRECT
   20 R@\
E  BD E@' /CALL EXTENDED
   20 R@\
0  BD E@= /CALL INHERENT
   20 R@\
+  86 D53 /RELATIVE SUBROUTINE
   91 DC1 /SECOND LETTER=S
   26 R@1 /NO->NOT BSR
   86 D8D /YES, OP CODE IS 8D
   20 R@2
1  CE E4AC0 /STOP SEARCH HERE
   DF DD0
   CE E4A80 /START SEARCH HERE
   BD E@S  /CALL SEARCH ROUTINE
2  DE DF0  /COLUMN 0
   C6 D52  /'R'
   E7 D6   /PUT INTO COLUMN 6
   39
.  86 D20  /INDEXED SUBROUTINE
   A7 D0   /ERASE COMMA
   A7 D1   /ERASE 'X'
   DE DF0  /COLUMN 0
   86 D44  /'D'
   A7 D6   /PUT 'D' IN COL 6
   CE E4BC0 /STOP SEARCH HERE
   DF DD0
   CE E4B80 /START SEARCH HERE
   BD E@S  /CALL SEARCH
   26 R@3
   CE E4CC0 /MNEMONIC NOT FOUND
   DF DD0   /CONTINUE SEARCHING
   CE E4C80
   BD E@S
   26 R@3   /MNEMONIC FOUND
   CE E4DC0 /MNEMONIC NOT FOUND
   DF DD0   /CONTINUE SEARCHING
   CE E4D80
   BD E@S
3  39   /MNEMONIC FOUND, RETURN
;  96 DC0 /IMMEDIATE SUBROUTINE
   81 D4C /FIRST LETTER=L?
   26 R@4
   96 DC2 /YES
   81 D58 /THIRD LETTER=X?
   26 R@5
   DE DF0 /COLUMN 0
   86 D45 /'E'
   A7 D6   /PUT 'E' IN COL 6
   86 DCE /OP CODE=CE
   39
5  81 D53 /THIRD LETTER=S
   26 R@6
   DE DF0 /COLUMN 0
   86 D45
```

```
    A7 D6   /PUT 'E' IN COL 6
    86 D8E  /OP CODE=8E
    39
 4  81 D43 /FIRST LETTER=C
    26 R@6
    96 DC2 /YES
    81 D58 /THIRD LETTER=X
    26 R@6
    DE DF0 /COLUMN 0
    86 D45 /'E'
    A7 D6   /PUT 'E' IN COL 6
    86 D8C  /OP CODE=8C
    39
 6  DE DF0  /COLUMN 0
    86 D44  /'D'
    A7 D6
    CE E4C40 /SET UP FOR SEARCH
    DF DD0
    CE E4C00
    BD E@S  /CALL SEARCH
    26 R@7
    CE E4D40 /MNEMONIC NOT FOUND
    DF DD0   /CONTINUE SEARCHING
    CE E4D00
    BD E@S
 7  39
 :  DE DF0  /DIRECT SUBROUTINE
    86 D44  /'D'
    A7 D6   /PUT 'D' IN COL 6
    CE E4C80 /SET UP FOR SEARCH
    DF DD0
    CE E4C40
    BD E@S  /CALL SEARCH
    26 R@7
    CE E4D80 /MNEMONIC NOT FOUND
    DF DD0   /CONTINUE SEARCHING
    CE E4D40
    BD E@S
    39
 '  DE DF0  /EXTENDED SUBROUTINE
    86 D45  /'E'
    A7 D6   /PUT 'E' IN COL 6
    CE E4C00 /SET UP FOR SEARCH
    DF DD0
    CE E4BC0
    BD E@S  /CALL SEARCH
    26 R@8
    CE E4D00 /NOT FOUND...
    DF DD0   /CONTINUE SEARCH
    CE E4CC0
    BD E@S
    26 R@8
    CE E4E00 /NOT FOUND...
    DF DD0   /CONTINUE SEARCH
    CE E4DC0
    BD E@S
 8  39
 =  CE E4A80 /INHERENT SUBR.
    DF DD0   /SET UP FOR SEARCH
    CE E4A00
    BD E@S   /CALL SEARCH
    26 R@8
    CE E4B80 /NOT FOUND...
```

```
    DF DD0   /CONTINUE SEARCH
    CE E4AC0
    BD E@S
    39
F   DE DF0   /FINISH
    C6 D20
    E7 D2    /ERASE COLUMN 2
    8        /NEXT WE WRITE...
    8        /THE OP CODE INTO...
    8        /COLUMNS 3 & 4
    C6 D10   /BASE .16
    D7 D5    /PUT INTO ARB
    5F
    D7 D4
    BD EFF64 /BINASC
    C6 D20
    E7 D0    /ERASE COLUMN 5
    7E E@A   /FIND END OF LINE
S   96 DC0   /SEARCH SUBROUTINE
    A1 D0    /COMPARE 1ST LETTER
    26 R@N   /NO MATCH?
    96 DC1   /A MATCH!
    A1 D1    /COMPARE 2ND LETTER
    26 R@N   /NO MATCH?
    96 DC2   /ANOTHER MATCH!
    A1 D2    /COMPARE 3RD LETTER
    26 R@N   /NO MATCH?
    96 DC3   /A THIRD MATCH!
Q   A1 D3    /COMPARE 4TH LETTER
    26 R@P   /STILL A CHANCE.
    DF DF6   /MNEMONIC FOUND!
    D6 DF6   /COMPUTE OP CODE...
    C0 D4A   /SUBTRACT 4A00
    96 DF7   /DIVIDE BY 4
    56
    46
    56
    46
    39
N   8        /NEXT MNEMONIC
    8
    8
    8
    9C DD0   /END OF SEARCH?
    26 R@S   /IF NO,TRY AGAIN
    4F       /YES,NO OP CODE
    39
P   81 D20   /3 LETTER MNEMONIC?
    26 R@N   /IF NO,TRY AGAIN
    86 D60   /60 INSTEAD OF 20
    7E E@Q   /RECHECK
Y   3F       /REVERSE TRANSLATOR...
END          /STARTS HERE
```

```
                SIMPLE
          MNEMONIC ASSEMBLER
              -PART 1-
          (MNEMONIC TO SPHERE)
              VERSION 1.1
                1/25/76
=     4700  BY ANTHONY PIERRY
  LDX #1FF
L INX    /LOC BEGINNING OF LINE
  STX %F0 /SAVE IT
  INX     /COLUMN 1
  LDAA 0,X
  CMPA#3D  /'='?
  BNE  @M
A INX    /LOOK FOR END OF LINE
  LDAA 0,X
  CMPA#60
  BNE  @A
  BRA  @L /FOUND END OF LINE
M CMPA#20 /COLUMN 1 IS BLANK?
  BEQ  @Z
  JSR  FDA1 /CALL PDS ASSEMBLER
  JMP  @Y /REVERSE TRANSLATE
Z INX
  LDAA 0,X
  CMPA#20 /IS LINE BLANK?
  BEQ  @A /YES->GET NEXT LINE
  STAA%C0 /STORE MNEMONIC IN...
  INX     /LOCATIONS C0-C3
  LDAA 0,X
  STAA%C1
  INX
  LDAA 0,X
  STAA%C2
  INX
  LDAA 0,X
  STAA%C3
  INX     /COLUMN 6
  LDAA 0,X
  CMPA#25 /'%'?
  BNE  @@
  JMP  @D /YES,GO TO DIRECT
@ CMPA#23 /'#'?
  BNE  @*
  JMP  @I /YES,GO TO IMMEDIATE
* INX     /COLUMN 7
  LDAA 0,X
  CMPA#20 /NO OPERAND?
  BNE  @%
$ JMP  @0 /NO,GO TO INHERENT
% CMPA#60 /NO OPERAND?
  BEQ  @$
  INX     /COLUMN 8
  LDAA 0,X
  CMPA#60 /NOT INDEXED?
  BEQ  @U /YES->GO TO DECISION
  CMPA#2C /COMMA?
  BEQ  @X /YES->GO TO INDEXED
  INX     /COLUMN 9
  LDAA 0,X
  CMPA#2C /COMMA?
```

```
  BEQ  @X /YES->GO TO INDEXED
W LDAA#42 /EXTENDED OR RELATIVE?
  CMPA%C0 /FIRST LETTER=B?
  BEQ  @R /YES->RELATIVE
  JMP  @E /NO, EXTENDED
R BSR  @+ /CALL RELATIVE
\ JMP  @F /GO TO FINISH
X BSR  @. /CALL INDEXED
  BRA  @\
I JSR  @; /CALL IMMEDIATE
  BRA  @\
D JSR  @: /CALL DIRECT
  BRA  @\
E JSR  @' /CALL EXTENDED
  BRA  @\
0 JSR  @= /CALL INHERENT
  BRA  @\
+ LDAA#53 /RELATIVE SUBROUTINE
  CMPA%C1 /SECOND LETTER=S?
  BNE  @1 /NO->NOT BSR
  LDAA#8D /YES, OP CODE IS 8D
  BRA  @2
1 LDX #4AC0 /STOP SEARCH HERE
  STX %D0
  LDX #4A80 /START SEARCH HERE
  JSR  @S  /CALL SEARCH ROUTINE
2 LDX %F0  /COLUMN 0
  LDAB#52  /'R'
  STAB 6,X /PUT INTO COLUMN 6
  RTS
. LDAA#20  /INDEXED SUBROUTINE
  STAA 0,X /ERASE COMMA
  STAA 1,X /ERASE 'X'
  LDX %F0  /COLUMN 0
  LDAA#44  /'D'
  STAA 6,X /PUT 'D' IN COL 6
  LDX #4BC0 /STOP SEARCH HERE
  STX %D0
  LDX #4B80 /START SEARCH HERE
  JSR  @S  /CALL SEARCH
  BNE  @3
  LDX #4CC0 /MNEMONIC NOT FOUND
  STX %D0   /CONTINUE SEARCHING
  LDX #4C80
  JSR  @S
  BNE  @3   /MNEMONIC FOUND
  LDX #4DC0 /MNEMONIC NOT FOUND
  STX %D0   /CONTINUE SEARCHING
  LDX #4D80
  JSR  @S
3 RTS  - /MNEMONIC FOUND, RETURN
; LDAA%C0 /IMMEDIATE SUBROUTINE
  CMPA#4C /FIRST LETTER=L?
  BNE  @4
  LDAA%C2 /YES
  CMPA#58 /THIRD LETTER=X?
  BNE  @5
  LDX %F0 /COLUMN 0
  LDAA#45 /'E'
  STAA 6,X /PUT 'E' IN COL 6
  LDAA#CE /OP CODE=CE
  RTS
```

```
5 CMPA#53 /THIRD LETTER=S?
  BNE  @6
  LDX %F0 /COLUMN 0
  LDAA#45
  STAA 6,X /PUT 'E' IN COL 6
  LDAA#3E  /OP CODE=3E
  RTS
4 CMPA#43 /FIRST LETTER=C?
  BNE  @6
  LDAA%C2 /YES
  CMPA#58 /THIRD LETTER=X?
  BNE  @6
  LDX %F0 /COLUMN 0
  LDAA#45 /'E'
  STAA 6,X /PUT 'E' IN COL 6
  LDAA#3C  /OP CODE=3C
  RTS
6 LDX %F0  /COLUMN 0
  LDAA#44  /'D'
  STAA 6,X
  LDX #4C40 /SET UP FOR SEARCH
  STX %D0
  LDX #4C00
  JSR  @S  /CALL SEARCH
  BNE  @7
  LDX #4D40 /MNEMONIC NOT FOUND
  STX %D0   /CONTINUE SEARCHING
  LDX #4D00
  JSR  @S
7 RTS
: LDX %F0  /DIRECT SUBROUTINE
  LDAA#44  /'D'
  STAA 6,X /PUT 'D' IN COL 6
  LDX #4C80 /SET UP FOR SEARCH
  STX %D0
  LDX #4C40
  JSR  @S  /CALL SEARCH
  BNE  @7
  LDX #4D80 /MNEMONIC NOT FOUND
  STX %D0   /CONTINUE SEARCHING
  LDX #4D40
  JSR  @S
  RTS
' LDX %F0  /EXTENDED SUBROUTINE
  LDAA#45  /'E'
  STAA 6,X /PUT 'E' IN COL 6
  LDX #4C00 /SET UP FOR SEARCH
  STX %D0
  LDX #4BC0
  JSR  @S  /CALL SEARCH
  BNE  @8
  LDX #4D00 /NOT FOUND...
  STX %D0   /CONTINUE SEARCH
  LDX #4CC0
  JSR  @S
  BNE  @8
  LDX #4E00 /NOT FOUND...
  STX %D0   /CONTINUE SEARCH
  LDX #4DC0
  JSR  @S
```

```
8 RTS
= LDX #4A80 /INHERENT SUBR.
  STX %D0   /SET UP FOR SEARCH
  LDX #4A00
  JSR  @S   /CALL SEARCH
  BNE  @8
  LDX #4B80 /NOT FOUND...
  STX %D0   /CONTINUE SEARCH
  LDX #4AC0
  JSR  @S
  RTS
F LDX %F0  /FINISH
  LDAB#20
  STAB 2,X /ERASE COLUMN 2
  INX      /NEXT WE WRITE...
  INX      /THE OP CODE INTO...
  INX      /COLUMNS 3 & 4
  LDAB#10  /BASE .16
  STAB%5   /PUT INTO ARB
  CLRB
  STAB%4
  JSR  FF64 /BINASC
  LDAB#20
  STAB 0,X /ERASE COLUMN 5
  JMP  @A  /FIND END OF LINE
S LDAA%C0  /SEARCH SUBROUTINE
  CMPA 0,X /COMPARE 1ST LETTER
  BNE  @N  /NO MATCH?
  LDAA%C1  /A MATCH!
  CMPA 1,X /COMPARE 2ND LETTER
  BNE  @N  /NO MATCH?
  LDAA%C2  /ANOTHER MATCH!
  CMPA 2,X /COMPARE 3RD LETTER
  BNE  @N  /NO MATCH?
  LDAA%C3  /A THIRD MATCH!
Q CMPA 3,X /COMPARE 4TH LETTER
  BNE  @P  /STILL A CHANCE.
  STX %F6  /MNEMONIC FOUND!
  LDAB%F6  /COMPUTE OP CODE...
  SUBB#4A  /SUBTRACT 4A00
  LDAA%F7  /DIVIDE BY 4
  RORB
  RORA
  RORB
  RORA
  RTS
N INX      /NEXT MNEMONIC
  INX
  INX
  INX
  CPX %D0  /END OF SEARCH?
  BNE  @S  /IF NO,TRY AGAIN
  CLRA     /YES,NO OP CODE
  RTS
P CMPA#20 /3 LETTER MNEMONIC?
  BNE  @N /IF NO,TRY AGAIN
  LDAA#60 /60 INSTEAD OF 20
  JMP  @Q /RECHECK
Y SWI    /REVERSE TRANSLATOR...
END      /STARTS HERE
```

```
                    SIMPLE
                MNEMONIC ASSEMBLER
                    -PART 2-
                (SPHERE TO MNEMONIC)
                    VERSION 1.1
                      1/25/76
 =      48E7  BY ANTHONY PIERRY
Y  CE E1FF
>  8      /LOC BEGINNING OF LINE
   DF DF0 /SAVE IT
   8      /COLUMN 1
   A6 D0
   81 D3D /'-'? .
   26 R@?
<  8      /LOOK FOR END OF LINE
   A6 D0
   81 D60
   26 R@<
   20 R@> /FOUND END OF LINE
?  81 D20 /COLUMN 1 IS BLANK?
   27 R@!
   3F     /STOP,JOB IS DONE
!  8
   8      /COLUMN 3
   A6 D0
   81 D20 /IS LINE BLANK?
   27 R@< /YES->GET NEXT LINE
   BD EFF22 /ASCII TO BINARY
   97 DD0  /STORE OP CODE
   C      /COMPUTE ADDDRESS...
   49     /OF MNEMONIC
   59
   49     /MULTIPLY BY 4
   59
   CB D4A /ADD 4A00
   D7 DD2
   97 DD3
   DE DD2 /ADDRESS OF MNEMONIC
   A6 D3   /PUSH MNEMONIC...
   36      /ONTO STACK
   A6 D2
   36
   A6 D1
   36
   A6 D0
   DE DF0  /COLUMN 0
   A7 D2   /WRITE MNEMONIC...
   32      /INTO COLUMNS 2-5
   A7 D3
   32
   A7 D4
   32
   A7 D5
   81 D60  /LAST CHAR=60
   26 R@&
   86 D20  /YES,CHANGE TO 20
   A7 D5
&  A6 D6   /EXAMINE COL 6
   81 D52  /'R'?
   26 R@↑
```

```
    86 D20   /YES
    A7 D6    /ERASE THE 'D'
    7E E@<   /GO TO END OF LINE
†   81 D45   /COL 6='E'?
    26 R@1
    96 DD0   /YES,GET OP CODE
    81 DCE   /OP CODE=CE?
    27 R@2
    81 D8C   /OP CODE=8C?
    27 R@2
    81 D8E   /OP CODE=8E?
    27 R@2
    86 D20   /NO TO ALL 3
    A7 D6    /ERASE THE 'E'
    7E E@<   /GO TO END OF LINE
2   86 D23   /'#'
    A7 D6    /PLACE '#' IN COL 6
3   7E E@<   /GO TO END OF LINE
1   81 D44   /COL 6='D'?
    26 R@3
    96 DD0   /YES,GET OP CODE
    81 D70   /OP CODE<70?
    2B R@X   /YES->INDEXED
    81 D90   /70<OP CODE<90?
    2B R@I   /YES->IMMEDIATE
    81 DA0   /90<OP CODE<A0?
    2B R@%   /YES->DIRECT
    81 DB0   /A0<OP CODE<B0?
    2B R@X   /YES->INDEXED
    81 DD0   /B0<OP CODE<D0?
    2B R@I   /YES->IMMEDIATE
    81 DE0   /D0<OP CODE<E0?
    2B R@%   /YES->DIRECT
X   86 D20   /INDEXED
    A7 D6    /ERASE THE 'D'
    A6 D8    /EXAMINE COL 8
    81 D20
    27 R@4   /COL 8 IS BLANK?
    81 D60   /NO
    27 R@5   /COL 8=60?
    86 D2C   /NO,COL 8 OCCUPIED
    A7 D9    /PUT COMMA IN COL 9
    86 D58
    A7 DA    /PUT 'X' IN COL 10
    7E E@<   /GO TO END OF LINE
4   86 D20   /COL 8 IS BLANK
    A7 D8    /PUT COMMA IN COL 8
    86 D58
    A7 D9    /PUT 'X' IN COL 9
    7E E@<   /GO TO END OF LINE
5   3F       /SYNTAX ERROR
I   86 D23   /IMMEDIATE
    A7 D6    /PUT '#' IN COL 6
    7E E@<   /GO TO END OF LINE
%   86 D25   /DIRECT
    A7 D6    /PUT '%' IN COL 6
    7E E@<   /GO TO END OF LINE
END
2
```

```
I                    SIMPLE
                  MNEMONIC ASSEMBLER
                       -PART 2-
                  (SPHERE TO MNEMONIC)
                       VERSION 1.1
                         1/25/76
 =      48E7  BY ANTHONY PIERRY
Y LDX #1FF
> INX     /LOC BEGINNING OF LINE
  STX %F0 /SAVE IT
  INX     /COLUMN 1
  LDAA 0,X
  CMPA#3D /'-'?
  BNE  @?
< INX     /LOOK FOR END OF LINE
  LDAA 0,X
  CMPA#60
  BNE  @<
  BRA  @> /FOUND END OF LINE
? CMPA#20 /COLUMN 1 IS BLANK?
  BEQ  @!
  SWI     /STOP,JOB IS DONE
! INX
  INX     /COLUMN 3
  LDAA 0,X
  CMPA#20 /IS LINE BLANK?
  BEQ  @< /YES->GET NEXT LINE
  JSR  FF22 /ASCII TO BINARY
  STAA%D0  /STORE OP CODE
  CLC     /COMPUTE ADDDRESS...
  ROLA    /OF MNEMONIC
  ROLB
  ROLA    /MULTIPLY BY 4
  ROLB
  ADDB#4A /ADD 4A00
  STAB%D2
  STAA%D3
  LDX %D2 /ADDRESS OF MNEMONIC
  LDAA 3,X /PUSH MNEMONIC...
  PSHA    /ONTO STACK
  LDAA 2,X
  PSHA
  LDAA 1,X
  PSHA
  LDAA 0,X
  LDX %F0  /COLUMN 0
  STAA 2,X /WRITE MNEMONIC...
  PULA    /INTO COLUMNS 2-5
  STAA 3,X
  PULA
  STAA 4,X
  PULA
  STAA 5,X
  CMPA#60  /LAST CHAR=60
  BNE  @&
  LDAA#20  /YES,CHANGE TO 20
  STAA 5,X
```

```
& LDAA 6,X /EXAMINE COL 6
  CMPA#52  /'R'?
  BNE  @↑
  LDAA#20  /YES
  STAA 6,X /ERASE THE 'R'
  JMP  @<  /GO TO END OF LINE
↑ CMPA#45  /COL 6='E'?
  BNE  @1
  LDAA%D0  /YES,GET OP CODE
  CMPA#CE  /OP CODE=CE?
  BEQ  @2
  CMPA#8C  /OP CODE=8C?
  BEQ  @2
  CMPA#8E  /OP CODE=8E?
  BEQ  @2
  LDAA#20  /NO TO ALL 3
  STAA 6,X /ERASE THE 'E'
  JMP  @<  /GO TO END OF LINE
2 LDAA#23  /'#'
  STAA 6,X /PLACE '#' IN COL 6
3 JMP  @<  /GO TO END OF LINE
1 CMPA#44  /COL 6='D'?
  BNE  @3
  LDAA%D0  /YES,GET OP CODE
  CMPA#70  /OP CODE<70?
  BMI  @X  /YES->INDEXED
  CMPA#90  /70<OP CODE<90?
  BMI  @I  /YES->IMMEDIATE
  CMPA#A0  /90<OP CODE<A0?
  BMI  @%  /YES->DIRECT
  CMPA#B0  /A0<OP CODE<B0?
  BMI  @X  /YES->INDEXED
  CMPA#D0  /B0<OP CODE<D0?
  BMI  @I  /YES->IMMEDIATE
  CMPA#E0  /D0<OP CODE<E0?
  BMI  @%  /YES->DIRECT
X LDAA#20  /INDEXED
  STAA 6,X /ERASE THE 'D'
  LDAA 8,X /EXAMINE COL 8
  CMPA#20
  BEQ  @4  /COL 8 IS BLANK?
  CMPA#60  /NO
  BEQ  @5  /COL 8=60?
  LDAA#2C  /NO,COL 8 OCCUPIED
  STAA 9,X /PUT COMMA IN COL 9
  LDAA#58
  STAA A,X /PUT 'X' IN COL 10
  JMP  @<  /GO TO END OF LINE
4 LDAA#2C  /COL 8 IS BLANK
  STAA 8,X /PUT COMMA IN COL 8
  LDAA#58
  STAA 9,X /PUT 'X' IN COL 9
  JMP  @<  /GO TO END OF LINE
5 SWI      /SYNTAX ERROR
I LDAA#23  /IMMEDIATE
  STAA 6,X /PUT '#' IN COL 6
  JMP  @<  /GO TO END OF LINE
% LDAA#25  /DIRECT
  STAA 6,X /PUT '%' IN COL 6
  JMP  @<  /GO TO END OF LINE
END
=
```

```
4700  CE 01 FF 08 DF F0 08 A6 00 81 3D 26 09 08 A6 00
4710  81 60 26 F9 20 ED 81 20 27 06 BD FD A1 7E 48 E7
4720  08 A6 00 81 20 27 E6 97 C0 08 A6 00 97 C1 08 A6
4730  00 97 C2 08 A6 00 97 C3 08 A6 00 81 25 26 03 7E
4740  47 80 81 23 26 03 7E 47 7B 08 A6 00 81 20 26 03
4750  7E 47 8A 81 60 27 F9 08 A6 00 81 60 27 08 81 2C
4760  27 15 08 A6 00 81 2C 27 0E 86 42 91 C0 27 03 7E
4770  47 85 8D 1B 7E 48 95 8D 32 20 F9 BD 47 DD 20 F4
4780  BD 48 31 20 EF BD 48 50 20 EA BD 48 7C 20 E5 86
4790  53 91 C1 26 04 86 8D 20 0B CE 4A C0 DF D0 CE 4A
47A0  80 BD 48 AF DE F0 C6 52 E7 06 39 86 20 A7 00 A7
47B0  01 DE F0 86 44 A7 06 CE 4B C0 DF D0 CE 4B 80 BD
47C0  48 AF 26 18 CE 4C C0 DF D0 CE 4C 80 BD 48 AF 26
47D0  0B CE 4D C0 DF D0 CE 4D 80 BD 48 AF 39 96 C0 81
47E0  4C 26 1C 96 C2 81 53 26 09 DE F0 86 45 A7 06 86
47F0  CE 39 81 53 26 1C DE F0 86 45 A7 06 86 8E 39 81
4800  43 26 0F 96 C2 81 53 26 09 DE F0 86 45 A7 06 86
4810  8C 39 DE F0 86 44 A7 06 CE 4C 40 DF D0 CE 4C 00
4820  BD 48 AF 26 0B CE 4D 40 DF D0 CE 4D 00 BD 48 AF
4830  39 DE F0 86 44 A7 06 CE 4C 80 DF D0 CE 4C 40 BD
4840  48 AF 26 EC CE 4D 80 DF D0 CE 4D 40 BD 48 AF 39
4850  DE F0 86 45 A7 06 CE 4C 00 DF D0 CE 4B C0 BD 48
4860  AF 26 18 CE 4D 00 DF D0 CE 4C C0 BD 48 AF 26 0B
4870  CE 4E 00 DF D0 CE 4D C0 BD 48 AF 39 CE 4A 80 DF
4880  D0 CE 4A 00 BD 48 AF 26 F2 CE 4B 80 DF D0 CE 4A
4890  C0 BD 48 AF 39 DE F0 C6 20 E7 02 08 08 08 C6 10
48A0  D7 05 5F D7 04 BD FF 64 C6 20 E7 00 7E 47 0D 96
48B0  C0 A1 00 26 1F 96 C1 A1 01 26 19 96 C2 A1 02 26
48C0  13 96 C3 A1 03 26 17 DF F6 D6 F6 C0 4A 96 F7 56
48D0  46 56 46 39 08 08 08 08 9C D0 26 D3 4F 39 81 20
48E0  26 F2 86 60 7E 48 C3 CE 01 FF 08 DF F0 08 A6 00
48F0  81 3D 26 09 08 A6 00 81 60 26 F9 20 ED 81 20 27
4900  01 3F 08 08 A6 00 81 20 27 EA BD FF 22 97 D0 0C
4910  49 59 49 59 CB 4A D7 D2 97 D3 DE D2 A6 03 36 A6
4920  02 36 A6 01 36 A6 00 DE F0 A7 02 32 A7 03 32 A7
4930  04 32 A7 05 81 60 26 04 86 20 A7 05 A6 06 81 52
4940  26 07 86 20 A7 06 7E 48 F4 81 45 26 1C 96 D0 81
4950  CE 27 0F 81 8C 27 0B 81 8E 27 07 86 20 A7 06 7E
4960  48 F4 86 23 A7 06 7E 48 F4 81 44 26 F9 96 D0 81
4970  70 2B 14 81 90 2B 35 81 A0 2B 33 81 B0 2B 08 81
4980  D0 2B 29 81 E0 2B 2C 86 20 A7 06 A6 03 81 20 27
4990  0F 81 60 27 16 86 2C A7 09 86 58 A7 0A 7E 48 F4
49A0  86 2C A7 08 86 58 A7 09 7E 48 F4 3F 86 23 A7 06
49B0  7E 48 F4 86 25 A7 06 7E 48 F4 18 18 18 18 18 18
49C0  00 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18
49D0  18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18
49E0  18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18
49F0  18 18 18 18 18 18 18 18 18 18 18 18 18 18 18 18
4A00  60 60 60 60 4E 4F 50 60 60 60 60 60 60 60 60 60
4A10  60 60 60 60 60 60 60 60 54 41 50 60 54 50 41 60
4A20  49 4E 58 60 44 45 58 60 43 4C 56 60 53 45 56 60
4A30  43 4C 43 60 53 45 43 60 43 4C 49 60 53 45 49 60
4A40  53 42 41 60 43 42 41 60 60 60 60 60 60 60 60 60
4A50  60 60 60 60 60 60 60 60 54 41 42 60 54 42 41 60
4A60  60 60 60 60 44 41 41 60 60 60 60 60 41 42 41 60
4A70  60 60 60 60 60 60 60 60 60 60 60 60 60 60 60 60
```

4A80	42	52	41	60	60	60	60	60	42	43	49	60	42	4C	53	60
4A90	42	43	43	60	42	43	53	60	42	4E	45	60	42	45	51	60
4AA0	42	56	43	60	42	56	53	60	42	50	4C	60	42	4D	49	60
4AB0	42	47	45	60	42	4C	54	60	42	47	54	20	42	4C	45	60
4AC0	54	53	58	60	49	4E	53	60	50	55	43	41	50	55	4C	42
4AD0	44	45	53	60	54	58	53	60	50	53	43	41	50	53	43	42
4AE0	20	20	20	60	52	54	53	60	60	60	60	60	52	54	49	60
4AF0	60	60	60	60	60	60	60	60	57	41	49	60	53	57	49	60
4B00	4E	45	47	41	60	60	60	60	60	60	60	60	43	4F	4D	41
4B10	4C	53	52	41	60	60	60	60	52	4F	52	41	41	53	52	41
4B20	41	53	4C	41	52	4F	4C	41	44	45	43	41	60	60	60	60
4B30	49	4E	43	41	54	53	54	41	60	60	60	60	43	4C	52	41
4B40	4E	45	47	42	60	60	60	60	60	60	60	60	43	4F	4D	42
4B50	4C	53	52	42	60	60	60	60	52	4F	52	42	41	53	52	42
4B60	41	53	4C	42	52	4F	4C	42	44	45	43	42	60	60	60	60
4B70	49	4E	43	42	54	53	54	42	60	60	60	60	43	4C	52	42
4B80	4E	45	47	60	60	60	60	60	60	60	60	60	43	4F	4D	60
4B90	4C	53	52	60	60	60	60	60	52	4F	52	60	41	53	52	60
4BA0	41	53	4C	60	52	4F	4C	60	44	45	43	60	60	60	60	60
4BB0	49	4E	43	60	54	53	54	60	4A	4D	50	60	43	4C	52	60
4BC0	4E	45	47	60	60	60	60	60	60	60	60	60	43	4F	4D	60
4BD0	4C	53	52	60	60	60	60	60	52	4F	52	60	41	53	52	60
4BE0	41	53	4C	60	52	4F	4C	60	44	45	43	60	60	60	60	60
4BF0	49	4E	43	60	54	53	54	60	4A	4D	50	60	43	4C	52	60
4C00	53	55	42	41	43	4D	50	41	53	42	43	41	60	60	60	60
4C10	41	4E	44	41	42	49	54	41	4C	44	41	41	60	60	60	60
4C20	45	4F	52	41	41	44	43	41	4F	52	41	41	41	44	44	41
4C30	43	50	58	60	42	53	52	60	4C	44	53	60	60	60	60	60
4C40	53	55	42	41	43	4D	50	41	53	42	43	41	60	60	60	60
4C50	41	4E	44	41	42	49	54	41	4C	44	41	41	53	54	41	41
4C60	45	4F	52	41	41	44	43	41	4F	52	41	41	41	44	44	41
4C70	43	50	58	60	60	60	60	60	4C	44	53	60	53	54	53	60
4C80	53	55	42	41	43	4D	50	41	53	42	43	41	60	60	60	60
4C90	41	4E	44	41	42	49	54	41	4C	44	41	41	53	54	41	41
4CA0	45	4F	52	41	41	44	43	41	4F	52	41	41	41	44	44	41
4CB0	43	50	58	60	4A	53	52	60	4C	44	53	60	53	54	53	60
4CC0	53	55	42	41	43	4D	50	41	53	42	43	41	60	60	60	60
4CD0	41	4E	44	41	42	49	54	41	4C	44	41	41	53	54	41	41
4CE0	45	4F	52	41	41	44	43	41	4F	52	41	41	41	44	44	41
4CF0	43	50	58	60	4A	53	52	60	4C	44	53	60	53	54	53	20
4D00	53	55	42	42	43	4D	50	42	53	42	43	44	60	60	60	60
4D10	41	4E	44	42	42	49	54	42	4C	44	41	42	60	60	60	60
4D20	45	4F	52	42	41	44	43	42	4F	52	41	42	41	44	44	42
4D30	60	60	60	60	60	60	60	60	4C	44	58	60	60	60	60	60
4D40	53	55	42	42	43	4D	50	42	53	42	43	42	60	60	60	60
4D50	41	4E	44	42	42	49	54	42	4C	44	41	42	53	54	41	42
4D60	45	4F	52	42	41	44	43	42	4F	52	41	42	41	44	44	42
4D70	60	60	60	60	60	60	60	60	4C	44	58	60	53	54	58	60
4D80	53	55	42	42	43	4D	50	42	53	42	43	42	60	60	60	60
4D90	41	4E	44	42	42	49	54	42	4C	44	41	42	53	54	41	42
4DA0	45	4F	52	42	41	44	43	42	4F	52	41	42	41	44	44	42
4DB0	60	60	60	60	60	60	60	60	4C	44	58	60	53	54	58	60
4DC0	53	55	42	42	43	4D	50	42	53	42	43	42	60	60	60	60
4DD0	41	4E	44	42	42	49	54	42	4C	44	41	42	53	54	41	42
4DE0	45	4F	52	42	41	44	43	42	4F	52	41	42	41	44	44	42
4DF0	60	60	60	60	60	60	60	60	4C	44	58	60	53	54	58	60

4D0B = 42

MNEMONIC TABLE LOADER

BY ANTHONY PIERRY 1/18/76

```
1000  C6 10 D7 05 5F D7 04 D7 E0 01 01 CE 4A 00 DF D0
1010  BD FD 16 DE 1C 96 E0 5F BD FF 64 DF 1C 86 20 BD
1020  FC AD 7C 00 E0 BD FC 6E DE 24 A6 03 36 A6 02 36
1030  A6 01 36 A6 00 DE D0 A7 00 32 08 A7 00 32 08 A7
1040  00 32 08 A7 00 08 20 C6
```

SPHERE CORP.

ORDER FORM

SOLD TO: _____ DATE: _____

_____ SHIPPING INSTRUCTIONS:

_____

ZIP _____

CUSTOMER PHONE NO. _____

QTY	CATALOG NUMBER	DESCRIPTION	PRICE	TOTAL
			.	.
			.	.
			.	.
			.	.
			.	.
			.	.
			.	.

PAYMENT

INDIVIDUALS-must include cashiers check, money order, or complete bank card information below. Other checks will clear before shipment.

COMPANIES: Special offers are available on a quantity OEM basis in kits, assembled units, or our one card computer. All orders are subject to credit approval. Invoices older than 10 days will be charged at .05% per calender day on invoice amount.(18¼% per year max.)

TOTAL	.
UTAH RESIDENTS ADD 5% SALES TAX	.
SHIPPING	.
TOTAL AMOUNT ENCLOSED OR TO BE CHARGED TO MY CREDIT CARD.	.

SHIPMENT

All costs of shipment are to be paid by purchasing party. Shipment will be made according to shipment instructions on a "best efforts" basis. No claims for lost or damaged shipments will be accepted by this corporation.

BANK CARD INFORMATION

Sign your name as it is on your credit card _____ BankAmericard____ Master Charge____

Credit Card Number _____ Interbank Number _____ Expiration Date _____

MAIL TO: SPHERE CORP. P. O. Box 129 , Bountiful, Utah, 84010

SPHERE CORP.

WARRANTEES

Warrantee (Assembled Units)

Warrantee units which fail due to defects in material or workmanship within 90 days of shipment will be repaired or replaced at our option when delivered at 940 North 400 East, North Salt Lake, Utah, with return shipment prepaid. Suspect modules may be sent. Send all correspondence to P.O. Box 129, Bountiful, Utah 84010.

Kits Warrantee (All Expensive Repairs)

Any part which fails due to defect within 90 days of shipment will be replaced. Replacement parts will be sent when failing parts are received with a $5.00 handling fee at the above shipping address. Warrantee period begins 10 days after shipment from factory.

EXCLUSIONS

1. No warrantee as to assembled units shall extend to any unit upon which unauthorized repairs or replacements have been attempted prior to return.

2. The warrantee on kit units is expressly limited to replacement of defective component parts; no warrantee, express or implied, attaches or extends to any assembly process or assembled part.

3. The warrantees described above shall be IN LIEU OF any other warrantee, express or implied, including, but not limited to, any implied warrantee of MERCHANTABILITY or fitness for a particular purpose.

Purchaser acknowledges he has read, understood and assents to the preceeding warrantee and delivery terms by his signature to this purchase order.

_____

Signature

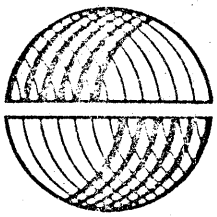
PROGRAMMING SHEET

PROGRAM NAME _____ VERSION ___

DATE _____ PAGE ___ OF ___

ADDRESS DECIMAL	ADDRESS HEX	MACHINE LABEL EQUATE	MACHINE OP CODE	MACHINE OP TYPE	MACHINE OPERAND HEX,OCT,DEC	SYMBOLIC LABEL	SYMBOLIC OP CODE	SYMBOLIC OP TYPE	SYMBOLIC OPERAND	COMMENTS

4 SPACES 4



SPHERE

corporation

PROGRAM SUBMITTAL FORM

Program Title

Date

Author

Author Address

Type of Program: _____; 1 - Games, 2 - Utilities, 3 - Demo,
4 - Business, 5 - I/O handlers, 6 - Other _____

Program Abstract:

Program Length (bytes) _____ Source Language _____

Use other side for flowchart. Use standard Sphere coding forms for source and object listings.

791 South 500 West Bountiful, Utah 84010 (801) 292-8466