

# Consol 1K Resident Operating System

**By Processor Technology  
Emeryville, CA**

## INTRODUCTION

CONSOL is a resident software operating system for Processor Technology's new single PCB intelligent microcomputer terminal product called SoL Terminal Computer™. The basic SoL Terminal Computer includes the following functional sections:

- 8080A CPU
- 1K bytes of static, low power RAM
- UART controlled RS-232 and 20 ma TTY serial I/O port with multiple baud rates of 75 → 9600 - Switch selectable.
- Video display circuit identical to the VDM-1 sold by Processor Technology. The video display section includes its own 1K bytes of RAM refresh memory. This RAM is in addition to the 1K RW RAM program memory.
- Parallel input-output port for data communications with fully implemented handshaking logic.
- ASCII Keyboard input port.
- PROM/ROM plug-in personality module for up to 2048 bytes of stored program.
- 300 or 1200 baud rate Kansas City Standard Audio cassette tape interface.
- MIB bus compatibility and expansion capability with all Altair/IMSAI/PTC bus plug-in products.

## SOL TERMINAL COMPUTER SOFTWARE CONFIGURATIONS

The SoL Terminal Computer™ can be configured by plug-in resident software modules as a stand alone microcomputer or as an intelligent remote editing terminal.

Basic system operating modes are stored in ROM or PROM on plug-in personality modules with a capacity of up to 2048 words. These modules may be changed in a few seconds to totally reconfigure the system for different applications. Other operating programs, such as BASIC AND FOCAL High level languages, can be loaded automatically into read/write memory (RAM) from cassette tape or floppy disc.

Software control programs for the operation of SoL are designed for three different levels of use. The first level program, CONSOL™, is contained in 1K of PROM and is designed to allow simple terminal operations. In addition, CONSOL allows direct control of the basic computer functions for entering data to, or examining data in, any memory location, or executing a program stored at a known location in memory.

The second level, designed for advanced terminal operations, is the SOLED™ editing terminal system. SOLED uses the full 2048 word capacity of a personality module and contains code to allow screen, file and cassette tape editing/transmission operations.

The stand-alone operating system, SOLOS,™ turns the SoL into a versatile computer that is easy to use, but every bit as powerful as any 8080-based system available today. Using SOLOS and the built-in cassette interface, BASIC can be loaded in less than a minute following power-on. BASIC programs can be both saved and executed from cassette. The SoL operating under SOLOS, brings true 8080 computer power away from hardware tinkering to direct application and problem solving.

## CONSOL SOFTWARE

CONSOL is configured to allow the SoL TERMINAL/COMPUTER to operate as a standard CRT terminal and to provide access to the essential computer capabilities inherent within SoL. The CONSOL software allows self test and small diagnostic programs to be entered to the system memory and executed thus providing verification of correct system operation. In addition, CONSOL contains standardized entry points for all normal I/O operations. These entry point routines are common with each of the SoL System Software allowing each personality module in the SoL product line to interface with external programs in an almost identical manner.

A cassette read routine is included in the CONSOL module software allowing SoL System Software to be loaded and run in a SoL System with additional memory. SoL System Software includes BASIC, FOCAL, a Scientific Calculator and numerous "game" packages including an 8K assembly language version of STARTREK called TREK80.

## CONSOL OPERATION

When power is applied to the SoL unit, CONSOL initializes the system ram area, clears the screen, and enters the terminal mode.

In this mode the SoL System acts as a standard CRT terminal sending keyboard data to an output port and displaying received data on the screen. The COMMAND KEYS of the keyboard are not transmitted to the output port but are interpreted as direct internal operation keys. CURSOR MOVEMENT, HOME and CLEAR SCREEN all operate in this manner, while

## SOFTWARE SECTION

MODE causes an immediate change in the operation of the SoL Terminal Computer.

When the MODE key is depressed CONSOL issues a prompt (>) and waits for a command line to be entered via the keyboard. The SoL is now operating as a computer and is ready to accept one of the following commands:

DUMP	Dump memory locations to screen
ENTER	Enter data to memory
EXECUTE	Execute a program in external memory
BASIC	Execute a program located at address zero
TERMINAL	Return to terminal mode
TLOAD	Load program or data from cassette tape
MODE	Press key to start new command line

## CONSOL COMMANDS

The seven SoL CONSOL commands are defined in the following:

**DUMP <addr> <addr>**

The DUMP command displays memory data on the screen in ASCII Hexidecimal representation. As with all SoL commands the command is recognized by the first two characters and up to ten additional characters can be input without an error being forced. Thus, DU; DUST; DUMP; DUMPTHESE would all be recognized as being a DUMP command.

At least one address must follow the command or an error will result. Entering the command DU followed by addr will result in the data at 'addr' being displayed

## MICROCOMPUTER DEVELOPMENT SOFTWARE

on the screen. If two addresses are defined then all values from the first address to the last address will be displayed. The following example shows the DUMP command with the start and terminating addresses specified;

**DUMP 0 EF**

Up to ten blanks may be inserted between each parameter without forcing an error condition. Errors are flagged by a question mark (?) replacing the character where the error occurred. For example if the DU command were given without an address, the question mark would appear ten spaces to the right of the 'U' character of the dump command.

**ENTER addr**

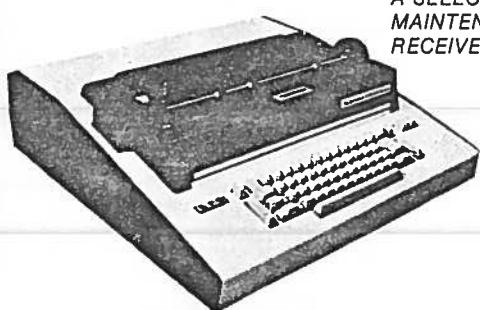
The enter command places sequential bytes into memory beginning at the specified address. Data, represented as hexidecimal values, are entered from the keyboard for storage in memory. Values are entered one line at a time with each line terminated by a carriage return or linefeed. The ENTER command function itself is terminated with a slash (/) and the CONSOL operating system returns to the command mode when the slash is encountered.

Data input lines are terminated with a carriage return or line feed. If the terminator is a C/R, CONSOL will erase all characters from the current cursor location to the end of the screen line. In this case, all valid input should be to the left of the cursor. If an error occurred during input the cursor may be moved back to

# RONDURE COMPANY

Where We Ship from Inventory the Same Day Your Order Arrives\*

A SELECTRIC TERMINAL COMPLETE WITH RS-232/C INTERFACE AND CERTIFIED FOR MAINTENANCE BY A NATIONAL SERVICE COMPANY. SHIPPED THE SAME DAY WE RECEIVE YOUR CHECK\*



**\$895.00**

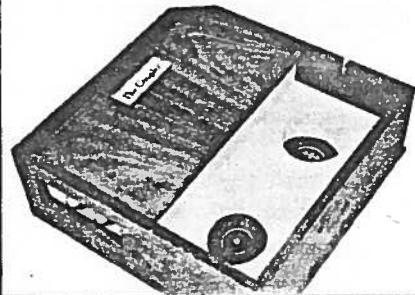
**TERMS:** Check or Money Order. For Modems, Base Keyboard, Switch Blk., add \$2.00 shipping and handling. All others shipping packaging and shipping collect.

\*Maintenance limited to cities in which service now offered. Shipped the same day as certified check or money order arrives. When regular checks accompany order, equipment is shipped when regular check clears.

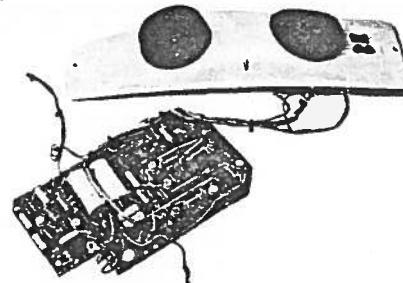
**ALSO NOTE:** NO EQUIPMENT INCLUDES PRINTS OR DOCUMENTATION (unless stated). NO CONNECTING CORDS OR CONNECTORS. EQUIPMENT IS SHIPPED ON AN AS IS WHERE IS BASIS. EXCEPT WHERE EXPRESSLY STATED IN WRITING, NO REPRESENTATION OR WARRANTY IS MADE AS TO THE QUALITY, CONDITION OR WORKING ORDER OF ANY EQUIPMENT OR PART.

**NEW ADDRESS**  
2522 BUTLER, DALLAS, TEXAS 75235  
Phone: (214) 630-4621

**LEFT: ACOUSTICAL MODEMS**  
\* ORIGINATE ONLY \* USED \*  
UNTESTED \* IN WOOD ENCLOSURE  
\$20.00ea.  
2 for \$35



**RIGHT: ACOUSTICAL MODEMS**  
\* ORIGINATE ONLY \* USED \*  
UNTESTED \* Physically fit into Model 33 Teletype. Manufactured by Paragon Partial Documentation 2 for \$25



CIRCLE INQUIRY NO. 53

the left to correct the error.

### TLoad <speed>

CONSOL includes routines to read standardized software from cassette tape. This standardized software is recorded on tape with a sixteen byte header that includes NAME, LOAD INFORMATION, FILE TYPE and execute address. CONSOL, because of space limitations, is unable to search for a program or file by name. After receiving the TLOAD command, CONSOL locates the next available data, uses the header information and loads the file to memory.

After loading the data, CONSOL returns to the command mode where the EXEC command can be used to execute the just loaded program. In addition, a return can normally be made to the command mode by pressing the MODE key. Program Space limitations again limited the mechanization of the escape function during the header search, so if the operation software system locks up in this routine the standard SoL restart must be used to escape.

The Audio Cassette Interface electronics within the SoL will record or receive data at either of two standard speeds. TLOAD will accept a parameter to select this speed, 0 being high speed and 1 being low (300 and 1200 bits per second). If no parameter is given, CONSOL will default to high speed operation as all standard SoL-System Software is recorded at this speed.

### Execute addr

The execute command is used to run programs located in external memory. CONSOL branches to the external routine in a manner similar to a CALL so that the program can return to the command mode using a standard RET instruction if normal stack operations are used.

### BASIC

The BASIC command is provided for executing programs whose starting address is 0. (Such as SoL-BASIC5).

### STANDARD I/O ROUTINES

All SoL System personality modules contain similar I/O code for input/output operations. CONSOL, using 1K of memory, has routines for KEYBOARD and SERIAL PORT input as well as SERIAL and VIDEO DISPLAY OUTPUT. Although the same code for SOLOS and SOLED contains expanded functions, the I/O operations appear almost identical when used with external software.

SoL BASIC5 for example performs all I/O using the jump table of the personality modules. Thus, without altering BASIC the user may output to either the serial port or to the display screen. Provision is also made within BASIC to programmatically change to any of the four available Input or Output options. CONSOL is of course limited to the two provided.

\*\* AL8-8 PROGRAM DEVELOPMENT SYSTEM \*\*  
 PROCESSOR TECHNOLOGY CORP.  
 6289 BOULLE STREET  
 EUREVILLE, CALIF. 94668

```

  C608    .0001
  C609    .0002
  C60A    .0003
  C60B    .0004
  C60C    .0005
  C60D    .0006
  C60E    .0007
  C60F    .0008
  C610    .0009
  C611    .0010
  C612    .0011
  C613    .0012
  C614    .0013
  C615    .0014
  C616    .0015
  C617    .0016
  C618    .0017
  C619    .0018
  C61A    .0019
  C61B    .0020
  C61C    .0021
  C61D    .0022
  C61E    .0023
  C61F    .0024
  C620    .0025
  C621    .0026
  C622    .0027
  C623    .0028
  C624    .0029
  C625    .0030
  C626    .0031
  C627    .0032
  C628    .0033
  C629    .0034
  C62A    .0035
  C62B    .0036
  C62C    .0037
  C62D    .0038
  C62E    .0039
  C62F    .0040
  C630    .0041
  C631    .0042
  C632    .0043
  C633    .0044
  C634    .0045
  C635    .0046
  C636    .0047
  C637    .0048
  C638    .0049
  C639    .0050
  C63A    .0051
  C63B    .0052
  C63C    .0053
  C63D    .0054
  C63E    .0055
  C63F    .0056
  C640    .0057
  C641    .0058
  C642    .0059
  C643    .0060
  C644    .0061
  C645    .0062
  C646    .0063
  C647    .0064
  C648    .0065
  C649    .0066
  C64A    .0067
  C64B    .0068
  C64C    .0069
  C64D    .0070
  C64E    .0071
  C64F    .0072
  C650    .0073
  C651    .0074
  C652    .0075
  C653    .0076
  C654    .0077
  C655    .0078
  C656    .0079
  C657    .0080
  C658    .0081
  C659    .0082
  C65A    .0083
  C65B    .0084
  C65C    .0085
  C65D    .0086
  C65E    .0087
  C65F    .0088
  C660    .0089
  C661    .0090
  C662    .0091
  C663    .0092
  C664    .0093
  C665    .0094
  C666    .0095
  C667    .0096
  C668    .0097
  C669    .0098
  C66A    .0099
  C66B    .0100
  C66C    .0101
  C66D    .0102
  C66E    .0103
  C66F    .0104
  C670    .0105
  C671    .0106
  C672    .0107
  C673    .0108
  C674    .0109
  C675    .0110
  C676    .0111
  C677    .0112
  C678    .0113
  C679    .0114
  C680    .0115
  C681    .0116
  C682    .0117
  C683    .0118
  C684    .0119
  C685    .0120
  C686    .0121
  C687    .0122
  C688    .0123
  C689    .0124
  C68A    .0125
  C68B    .0126
  C68C    .0127
  C68D    .0128
  C68E    .0129
  C68F    .0130
  C690    .0131
  C691    .0132
  C692    .0133
  C693    .0134
  C694    .0135
  C695    .0136
  C696    .0137
  C697    .0138
  C698    .0139
  C699    .0140
  C69A    .0141
  C69B    .0142
  C69C    .0143
  C69D    .0144
  C69E    .0145
  C69F    .0146
  C6A0    .0147
  C6A1    .0148
  C6A2    .0149
  C6A3    .0150
  C6A4    .0151
  C6A5    .0152
  C6A6    .0153
  C6A7    .0154
  C6A8    .0155
  C6A9    .0156
  C6A0    .0157
  C6A1    .0158
  C6A2    .0159
  C6A3    .0160
  C6A4    .0161
  C6A5    .0162
  C6A6    .0163
  C6A7    .0164
  C6A8    .0165
  C6A9    .0166
  C6A0    .0167
  C6A1    .0168
  C6A2    .0169
  C6A3    .0170
  C6A4    .0171
  C6A5    .0172
  C6A6    .0173
  C6A7    .0174
  C6A8    .0175
  C6A9    .0176
  C6A0    .0177
  C6A1    .0178
  C6A2    .0179
  C6A3    .0180
  C6A4    .0181
  C6A5    .0182
  C6A6    .0183
  C6A7    .0184
  C6A8    .0185
  C6A9    .0186
  C6A0    .0187
  C6A1    .0188
  C6A2    .0189
  C6A3    .0190
  C6A4    .0191
  C6A5    .0192
  C6A6    .0193
  C6A7    .0194
  C6A8    .0195
  C6A9    .0196
  C6A0    .0197
  C6A1    .0198
  C6A2    .0199
  C6A3    .0200
  C6A4    .0201
  C6A5    .0202
  C6A6    .0203
  C6A7    .0204
  C6A8    .0205
  C6A9    .0206
  C6A0    .0207
  C6A1    .0208
  C6A2    .0209
  C6A3    .0210
  C6A4    .0211
  C6A5    .0212
  C6A6    .0213
  C6A7    .0214
  C6A8    .0215
  C6A9    .0216
  C6A0    .0217
  C6A1    .0218
  C6A2    .0219
  C6A3    .0220
  C6A4    .0221
  C6A5    .0222
  C6A6    .0223
  C6A7    .0224
  C6A8    .0225
  C6A9    .0226
  C6A0    .0227
  C6A1    .0228
  C6A2    .0229
  C6A3    .0230
  C6A4    .0231
  C6A5    .0232
  C6A6    .0233
  C6A7    .0234
  C6A8    .0235
  C6A9    .0236
  C6A0    .0237
  C6A1    .0238
  C6A2    .0239
  C6A3    .0240
  C6A4    .0241
  C6A5    .0242
  C6A6    .0243
  C6A7    .0244
  C6A8    .0245
  C6A9    .0246
  C6A0    .0247
  C6A1    .0248
  C6A2    .0249
  C6A3    .0250
  C6A4    .0251
  C6A5    .0252
  C6A6    .0253
  C6A7    .0254
  C6A8    .0255
  C6A9    .0256
  C6A0    .0257
  C6A1    .0258
  C6A2    .0259
  C6A3    .0260
  C6A4    .0261
  C6A5    .0262
  C6A6    .0263
  C6A7    .0264
  C6A8    .0265
  C6A9    .0266
  C6A0    .0267
  C6A1    .0268
  C6A2    .0269
  C6A3    .0270
  C6A4    .0271
  C6A5    .0272
  C6A6    .0273
  C6A7    .0274
  C6A8    .0275
  C6A9    .0276
  C6A0    .0277
  C6A1    .0278
  C6A2    .0279
  C6A3    .0280
  C6A4    .0281
  C6A5    .0282
  C6A6    .0283
  C6A7    .0284
  C6A8    .0285
  C6A9    .0286
  C6A0    .0287
  C6A1    .0288
  C6A2    .0289
  C6A3    .0290
  C6A4    .0291
  C6A5    .0292
  C6A6    .0293
  C6A7    .0294
  C6A8    .0295
  C6A9    .0296
  C6A0    .0297
  C6A1    .0298
  C6A2    .0299
  C6A3    .0300
  C6A4    .0301
  C6A5    .0302
  C6A6    .0303
  C6A7    .0304
  C6A8    .0305
  C6A9    .0306
  C6A0    .0307
  C6A1    .0308
  C6A2    .0309
  C6A3    .0310
  C6A4    .0311
  C6A5    .0312
  C6A6    .0313
  C6A7    .0314
  C6A8    .0315
  C6A9    .0316
  C6A0    .0317
  C6A1    .0318
  C6A2    .0319
  C6A3    .0320
  C6A4    .0321
  C6A5    .0322
  C6A6    .0323
  C6A7    .0324
  C6A8    .0325
  C6A9    .0326
  C6A0    .0327
  C6A1    .0328
  C6A2    .0329
  C6A3    .0330
  C6A4    .0331
  C6A5    .0332
  C6A6    .0333
  C6A7    .0334
  C6A8    .0335
  C6A9    .0336
  C6A0    .0337
  C6A1    .0338
  C6A2    .0339
  C6A3    .0340
  C6A4    .0341
  C6A5    .0342
  C6A6    .0343
  C6A7    .0344
  C6A8    .0345
  C6A9    .0346
  C6A0    .0347
  C6A1    .0348
  C6A2    .0349
  C6A3    .0350
  C6A4    .0351
  C6A5    .0352
  C6A6    .0353
  C6A7    .0354
  C6A8    .0355
  C6A9    .0356
  C6A0    .0357
  C6A1    .0358
  C6A2    .0359
  C6A3    .0360
  C6A4    .0361
  C6A5    .0362
  C6A6    .0363
  C6A7    .0364
  C6A8    .0365
  C6A9    .0366
  C6A0    .0367
  C6A1    .0368
  C6A2    .0369
  C6A3    .0370
  C6A4    .0371
  C6A5    .0372
  C6A6    .0373
  C6A7    .0374
  C6A8    .0375
  C6A9    .0376
  C6A0    .0377
  C6A1    .0378
  C6A2    .0379
  C6A3    .0380
  C6A4    .0381
  C6A5    .0382
  C6A6    .0383
  C6A7    .0384
  C6A8    .0385
  C6A9    .0386
  C6A0    .0387
  C6A1    .0388
  C6A2    .0389
  C6A3    .0390
  C6A4    .0391
  C6A5    .0392
  C6A6    .0393
  C6A7    .0394
  C6A8    .0395
  C6A9    .0396
  C6A0    .0397
  C6A1    .0398
  C6A2    .0399
  C6A3    .0400
  C6A4    .0401
  C6A5    .0402
  C6A6    .0403
  C6A7    .0404
  C6A8    .0405
  C6A9    .0406
  C6A0    .0407
  C6A1    .0408
  C6A2    .0409
  C6A3    .0410
  C6A4    .0411
  C6A5    .0412
  C6A6    .0413
  C6A7    .0414
  C6A8    .0415
  C6A9    .0416
  C6A0    .0417
  C6A1    .0418
  C6A2    .0419
  C6A3    .0420
  C6A4    .0421
  C6A5    .0422
  C6A6    .0423
  C6A7    .0424
  C6A8    .0425
  C6A9    .0426
  C6A0    .0427
  C6A1    .0428
  C6A2    .0429
  C6A3    .0430
  C6A4    .0431
  C6A5    .0432
  C6A6    .0433
  C6A7    .0434
  C6A8    .0435
  C6A9    .0436
  C6A0    .0437
  C6A1    .0438
  C6A2    .0439
  C6A3    .0440
  C6A4    .0441
  C6A5    .0442
  C6A6    .0443
  C6A7    .0444
  C6A8    .0445
  C6A9    .0446
  C6A0    .0447
  C6A1    .0448
  C6A2    .0449
  C6A3    .0450
  C6A4    .0451
  C6A5    .0452
  C6A6    .0453
  C6A7    .0454
  C6A8    .0455
  C6A9    .0456
  C6A0    .0457
  C6A1    .0458
  C6A2    .0459
  C6A3    .0460
  C6A4    .0461
  C6A5    .0462
  C6A6    .0463
  C6A7    .0464
  C6A8    .0465
  C6A9    .0466
  C6A0    .0467
  C6A1    .0468
  C6A2    .0469
  C6A3    .0470
  C6A4    .0471
  C6A5    .0472
  C6A6    .0473
  C6A7    .0474
  C6A8    .0475
  C6A9    .0476
  C6A0    .0477
  C6A1    .0478
  C6A2    .0479
  C6A3    .0480
  C6A4    .0481
  C6A5    .0482
  C6A6    .0483
  C6A7    .0484
  C6A8    .0485
  C6A9    .0486
  C6A0    .0487
  C6A1    .0488
  C6A2    .0489
  C6A3    .0490
  C6A4    .0491
  C6A5    .0492
  C6A6    .0493
  C6A7    .0494
  C6A8    .0495
  C6A9    .0496
  C6A0    .0497
  C6A1    .0498
  C6A2    .0499
  C6A3    .0500
  C6A4    .0501
  C6A5    .0502
  C6A6    .0503
  C6A7    .0504
  C6A8    .0505
  C6A9    .0506
  C6A0    .0507
  C6A1    .0508
  C6A2    .0509
  C6A3    .0510
  C6A4    .0511
  C6A5    .0512
  C6A6    .0513
  C6A7    .0514
  C6A8    .0515
  C6A9    .0516
  C6A0    .0517
  C6A1    .0518
  C6A2    .0519
  C6A3    .0520
  C6A4    .0521
  C6A5    .0522
  C6A6    .0523
  C6A7    .0524
  C6A8    .0525
  C6A9    .0526
  C6A0    .0527
  C6A1    .0528
  C6A2    .0529
  C6A3    .0530
  C6A4    .0531
  C6A5    .0532
  C6A6    .0533
  C6A7    .0534
  C6A8    .0535
  C6A9    .0536
  C6A0    .0537
  C6A1    .0538
  C6A2    .0539
  C6A3    .0540
  C6A4    .0541
  C6A5    .0542
  C6A6    .0543
  C6A7    .0544
  C6A8    .0545
  C6A9    .0546
  C6A0    .0547
  C6A1    .0548
  C6A2    .0549
  C6A3    .0550
  C6A4    .0551
  C6A5    .0552
  C6A6    .0553
  C6A7    .0554
  C6A8    .0555
  C6A9    .0556
  C6A0    .0557
  C6A1    .0558
  C6A2    .0559
  C6A3    .0560
  C6A4    .0561
  C6A5    .0562
  C6A6    .0563
  C6A7    .0564
  C6A8    .0565
  C6A9    .0566
  C6A0    .0567
  C6A1    .0568
  C6A2    .0569
  C6A3    .0570
  C6A4    .0571
  C6A5    .0572
  C6A6    .0573
  C6A7    .0574
  C6A8    .0575
  C6A9    .0576
  C6A0    .0577
  C6A1    .0578
  C6A2    .0579
  C6A3    .0580
  C6A4    .0581
  C6A5    .0582
  C6A6    .0583
  C6A7    .0584
  C6A8    .0585
  C6A9    .0586
  C6A0    .0587
  C6A1    .0588
  C6A2    .0589
  C6A3    .0590
  C6A4    .0591
  C6A5    .0592
  C6A6    .0593
  C6A7    .0594
  C6A8    .0595
  C6A9    .0596
  C6A0    .0597
  C6A1    .0598
  C6A2    .0599
  C6A3    .0600
  C6A4    .0601
  C6A5    .0602
  C6A6    .0603
  C6A7    .0604
  C6A8    .0605
  C6A9    .0606
  C6A0    .0607
  C6A1    .0608
  C6A2    .0609
  C6A3    .0610
  C6A4    .0611
  C6A5    .0612
  C6A6    .0613
  C6A7    .0614
  C6A8    .0615
  C6A9    .0616
  C6A0    .0617
  C6A1    .0618
  C6A2    .0619
  C6A3    .0620
  C6A4    .0621
  C6A5    .0622
  C6A6    .0623
  C6A7    .0624
  C6A8    .0625
  C6A9    .0626
  C6A0    .0627
  C6A1    .0628
  C6A2    .0629
  C6A3    .0630
  C6A4    .0631
  C6A5    .0632
  C6A6    .0633
  C6A7    .0634
  C6A8    .0635
  C6A9    .0636
  C6A0    .0637
  C6A1    .0638
  C6A2    .0639
  C6A3    .0640
  C6A4    .0641
  C6A5    .0642
  C6A6    .0643
  C6A7    .0644
  C6A8    .0645
  C6A9    .0646
  C6A0    .0647
  C6A1    .0648
  C6A2    .0649
  C6A3    .0650
  C6A4    .0651
  C6A5    .0652
  C6A6    .0653
  C6A7    .0654
  C6A8    .0655
  C6A9    .0656
  C6A0    .0657
  C6A1    .0658
  C6A2    .0659
  C6A3    .0660
  C6A4    .0661
  C6A5    .0662
  C6A6    .0663
  C6A7    .0664
  C6A8    .0665
  C6A9    .0666
  C6A0    .0667
  C6A1    .0668
  C6A2    .0669
  C6A3    .0670
  C6A4    .0671
  C6A5    .0672
  C6A6    .0673
  C6A7    .0674
  C6A8    .0675
  C6A9    .0676
  C6A0    .0677
  C6A1    .0678
  C6A2    .0679
  C6A3    .0680
  C6A4    .0681
  C6A5    .0682
  C6A6    .0683
  C6A7    .0684
  C6A8    .0685
  C6A9    .0686
  C6A0    .0687
  C6A1    .0688
  C6A2    .0689
  C6A3    .0690
  C6A4    .0691
  C6A5    .0692
  C6A6    .0693
  C6A7    .0694
  C6A8    .0695
  C6A9    .0696
  C6A0    .0697
  C6A1    .0698
  C6A2    .0699
  C6A3    .0700
  C6A4    .0701
  C6A5    .0702
  C6A6    .0703
  C6A7    .0704
  C6A8    .0705
  C6A9    .0706
  C6A0    .0707
  C6A1    .0708
  C6A2    .0709
  C6A3    .0710
  C6A4    .0711
  C6A5    .0712
  C6A6    .0713
  C6A7    .0714
  C6A8    .0715
  C6A9    .0716
  C6A0    .0717
  C6A1    .0718
  C6A2    .0719
  C6A3    .0720
  C6A4    .0721
  C6A5    .0722
  C6A6    .0723
  C6A7    .0724
  C6A8    .0725
  C6A9    .0726
  C6A0    .0727
  C6A1    .0728
  C6A2    .0729
  C6A3    .0730
  C6A4    .0731
  C6A5    .0732
  C6A6    .0733
  C6A7    .0734
  C6A8    .0735
  C6A9    .0736
  C6A0    .0737
  C6A1    .0738
  C6A2    .0739
  C6A3    .0740
  C6A4    .0741
  C6A5    .0742
  C6A6    .0743
  C6A7    .0744
  C6A8    .0745
  C6A9    .0746
  C6A0    .0747
  C6A1    .0748
  C6A2    .0749
  C6A3    .0750
  C6A4    .0751
  C6A5    .0752
  C6A6    .0753
  C6A7    .0754
  C6A8    .0755
  C6A9    .0756
  C6A0    .0757
  C6A1    .0758
  C6A2    .0759
  C6A3    .0760
  C6A4    .0761
  C6A5    .0762
  C6A6    .0763
  C6A7    .0764
  C6A8    .0765
  C6A9    .0766
  C6A0    .0767
  C6A1    .0768
  C6A2    .0769
  C6A3    .0770
  C6A4    .0771
  C6A5    .0772
  C6A6    .0773
  C6A7    .0774
  C6A8    .0775
  C6A9    .0776
  C6A0    .0777
  C6A1    .0778
  C6A2    .0779
  C6A3    .0780
  C6A4    .0781
  C6A5    .0782
  C6A6    .0783
  C6A7    .0784
  C6A8    .0785
  C6A9    .0786
  C6A0    .0787
  C6A1    .0788
  C6A2    .0789
  C6A3    .0790
  C6A4    .0791
  C6A5    .0792
  C6A6    .0793
  C6A7    .0794
  C6A8    .0795
  C6A9    .0796
  C6A0    .0797
  C6A1    .0798
  C6A2    .0799
  C6A3    .0800
  C6A4    .0801
  C6A5    .0802
  C6A6    .0803
  C6A7    .0804
  C6A8    .0805
  C6A9    .0806
  C6A0    .0807
  C6A1    .0808
  C6A2    .0809
  C6A3    .0810
  C6A4    .0811
  C6A5    .0812
  C6A6    .0813
  C6A7    .0814
  C6A8    .0815
  C6A9    .0816
  C6A0    .0817
  C6A1    .0818
  C6A2    .0819
  C6A3    .0820
  C6A4    .0821
  C6A5    .0822
  C6A6    .0823
  C6A7    .0824
  C6A8    .0825
  C6A9    .0826
  C6A0    .0827
  C6A1    .0828
  C6A2    .0829
  C6A3    .0830
  C6A4    .0831
  C6A5    .0832
  C6A6    .0833
  C6A7    .0834
  C6A8    .0835
  C6A9    .0836
  C6A0    .0837
  C6A1    .0838
  C6A2    .0839
  C6A3    .0840
  C6A4    .0841
  C6A5    .0842
  C6A6    .0843
  C6A7    .0844
  C6A8    .0845
  C6A9    .0846
  C6A0    .0847
  C6A1    .0848
  C6A2    .0849
  C6A3    .0850
  C6A4    .0851
  C6A5    .0852
  C6A6    .0853
  C6A7    .0854
  C6A8    .0855
  C6A9    .0856
  C6A0    .0857
  C6A1    .0858
  C6A2    .0859
  C6A3    .0860
  C6A4   
```

## **SOFTWARE SECTION**

## **MICROCOMPUTER DEVELOPMENT SOFTWARE**

JANUARY 1977

INTERFAZ AGE 91

# SOFTWARE SECTION

# MICROCOMPUTER DEVELOPMENT SOFTWARE

C111 CD 30 C1 8513 \*  
C111 CD 30 C1 8513 COPRC CALL CLR  
C114 ED #1 8514 CALL C1,1 SET FOR CHARACTER POSITION  
C114 ED 25 C1 8515 CALL VDAD2 GET SCREEN ADDRESS  
C116 ED 8516 XCHG  
C12A CD 32 C2 8517 CALL SCR REMOVE THE CURSOR  
C12A CD 32 C2 8518 JS ERRI NO COMMAND?  
C12A CA 88 C3 8519 XCHG H SET BACK TO C1A  
L117 ED 8520 LIT D,CONTAN POINT TO COMMAND TABLE  
C12A CD 11 94 C1 8521 \* THIS ROUTINE SLACKS THROUGH A TABLE POINTED TO  
C12A CD 11 94 C1 8522 \* AT DCN FOR THE ADDRESS OF THE FIRST OF THE COMMANDS  
C12A CD 11 94 C1 8523 \* AT DCN FOR THE ADDRESS OF THE LAST OF THE COMMANDS  
C12A CD 11 94 C1 8524 \* NOT COMPLETE IF NO MATCH IS FOUND THE SCAN ENDS  
C12A CD 11 94 C1 8525 \* BY PLACING A QUESTION MARK WITHIN THE SEARCH STRING.  
C12A CD 11 94 C1 8526 \*  
C12A CD 11 94 C1 8527 FOCUS D TEST FOR TABLE END  
C12A CD 11 94 C1 8528 ORA A NOT FOUND..COMMAND ERROR  
C12A CD 11 94 C1 8529 JS ERRI NO START OF SCRN ADDRESS  
C12A CD 11 94 C1 8530 PUSH H  
C12A CD 11 94 C1 8531 CRF H TEST FIRST CHR  
C12A CD 11 94 C1 8532 LDAX B  
C12A CD 11 94 C1 8533 JNS MCOR  
C12A CD 11 94 C1 8534 \*  
C12A CD 11 94 C1 8535 LDAX B  
C12A CD 11 94 C1 8536 LDAX D  
C12A CD 11 94 C1 8537 LDAX E  
C12A CD 11 94 C1 8538 JNS MCOR NOW SECOND CHARACTER  
C12A CD 11 94 C1 8539 \*  
C12A CD 11 94 C1 8540 POP B CLEAR THE SCRN  
C12A CD 11 94 C1 8541 XCHG D DE HAS SCAN ADDRESS  
C12A CD 11 94 C1 8542 LDAX E HL HAS COMMAND ADDRESS  
C12A CD 11 94 C1 8543 JMS DISPT DISPATCH TO IT  
C12A CD 11 94 C1 8544 \*  
C12A CD 11 94 C1 8545 LDAX E GO TO NEXT ENTRY  
C12A CD 11 94 C1 8546 MCOR INX D  
C12A CD 11 94 C1 8547 LDAX D  
C12A CD 11 94 C1 8548 LDAX E  
C12A CD 11 94 C1 8549 POP B GET BACK ORIGINAL ADDRESS  
C12A CD 11 94 C1 8550 JRP PDCOM CONTINUE SEARCH  
C12A CD 11 94 C1 8551 \*  
C12A CD 11 94 C1 8552 \* OUTPUT A CHAR FOLLOWED BY A PROMPT  
C12A CD 11 94 C1 8553 \* (WITH CONSO ALL OPERATIONS ARE ON THE SCREEN)  
C12A CD 11 94 C1 8554 \*  
C12A CD 11 94 C1 8555 PHROUT CALL CALF  
C12A CD 11 94 C1 8556 AVI B,1 THE PROMPT  
C12A CD 11 94 C1 8557 JRP VDOS PUT IT ON THE SCREEN  
C12A CD 11 94 C1 8558 \*  
C12A CD 11 94 C1 8559 \*  
C12A CD 11 94 C1 8560 CMFL AVI B,L LINE FEED  
C12A CD 11 94 C1 8561 LDAX B  
C12A CD 11 94 C1 8562 AVI B,L CARRIAGE RETURN  
C12A CD 11 94 C1 8563 LDAX B  
C12A CD 11 94 C1 8564 JRP VDOS PUT IT OUT AND RETURN  
C12A CD 11 94 C1 8565 RET  
C12A CD 11 94 C1 8566 \*  
C12A CD 11 94 C1 8567 \* SCAN OVER UP TO 12 CHARACTERS LOOKING FOR A BLANK  
C12A CD 11 94 C1 8568 \*  
C12A CD 11 94 C1 8569 \*  
C12A CD 11 94 C1 8570 \*  
C12A CD 11 94 C1 8571 JS SCR GET A BLANK NOW SCAN PAST IT  
C12A CD 11 94 C1 8572 INK D  
C12A CD 11 94 C1 8573 DCR C NO MORE THAN TWELVE  
C12A CD 11 94 C1 8574 JMS BALE1  
C12A CD 11 94 C1 8575 RET GO BACK WITH ZERO FLAG SET  
C12A CD 11 94 C1 8576 \*  
C12A CD 11 94 C1 8577 \*  
C12A CD 11 94 C1 8578 \* SCAN PAST UP TO 18 BLANK POSITIONS LOOKING FOR  
C12A CD 11 94 C1 8579 \* A NON BLANK CHARACTER.  
C12A CD 11 94 C1 8580 \*  
C12A CD 11 94 C1 8581 SCHM AVI C,12 MAXIMUM COMMAND STRING  
C12A CD 11 94 C1 8582 SCHM1 LDAX D  
C12A CD 11 94 C1 8583 CPI BLANK  
C12A CD 11 94 C1 8584 LDAX D  
C12A CD 11 94 C1 8585 INK D  
C12A CD 11 94 C1 8586 DCX C  
C12A CD 11 94 C1 8587 RS . COMMAND ERROR  
C12A CD 11 94 C1 8588 JRP SCHM1 KEEP LOOPING  
C12A CD 11 94 C1 8589 \* THIS ROUTINE SCANS OVER CHARACTERS, PAST BLANKS AND  
C12A CD 11 94 C1 8590 \* CONVERTS THE FOLLOWING ADDRESS TO HEX. ERRORS RETURN TO  
C12A CD 11 94 C1 8591 \* THE ERROR BANNER.  
C12A CD 11 94 C1 8592 \*  
C12A CD 11 94 C1 8593 \*  
C12A CD 24 C2 8594 SCHM CALL SBLK  
C12A CA BE C3 8595 JS ERRI  
C12A CD 24 C2 8596 \*  
C12A CD 24 C2 8597 \* THIS ROUTINE CONVERTS ASCII DIGITS INTO BINARY FOLLOWING  
C12A CD 24 C2 8598 \* A STANDARD HEX CONVERSION. THE SCAN STOPS WHEN AN ASCII  
C12A CD 24 C2 8599 \* SPACE IS ENCOUNTERED. PARAMETER ERRORS REPLACE THE ERROR  
C12A CD 24 C2 8600 \* CHARACTER WITH A QUESTION MARK.  
C12A CD 24 C2 8601 \*  
C12A CD 24 C2 8602 SHXW LXI H,8 CLEAR H & L  
C12A CD 24 C2 8603 SHX1 LDAX D GET CHARACTER  
C12A CD 24 C2 8604 CPI ZDH IS IT A SPACE?  
C12A CD 24 C2 8605 RS . IT IS  
C12A CD 24 C2 8606 JMS . MAKE ROOM FOR THE NEW ONE  
C12A CD 24 C2 8607 MCWVW DAD H  
C12A CD 24 C2 8608 LDAX B  
C12A CD 24 C2 8609 CPI H  
C12A CD 24 C2 8610 LDAX B  
C12A CD 24 C2 8611 CALL MCWV1 DO THE CONVERSION  
C12A CD 24 C2 8612 JMS ERRI NOT VALID HEXDECIMAL VALUE  
C12A CD 24 C2 8613 ADD L  
C12A CD 24 C2 8614 LDAX H  
C12A CD 24 C2 8615 INK D  
C12A CD 24 C2 8616 JRP SHX1 BUMP THE POINTER  
C12A CD 24 C2 8617 \*  
C12A CD 24 C2 8618 MCWVW SBLK H  
C12A CD 24 C2 8619 CPI 48 REMOVE ASCII BIAS  
C12A CD 24 C2 8620 LDAX B  
C12A CD 24 C2 8621 CPI 1F  
C12A CD 24 C2 8622 RS . IF LESS THAN 9  
C12A CD 24 C2 8623 SHX1 7 IT'S A LETTER?  
C12A CD 24 C2 8624 CPI 1FH  
C12A CD 24 C2 8625 LDAX B  
C12A CD 24 C2 8626 CPI C  
C12A CD 24 C2 8627 LDAX B  
C12A CD 24 C2 8628 LDAX C  
C12A CD 24 C2 8629 LDAX D  
C12A CD 24 C2 8630 LDAX E  
C12A CD 24 C2 8631 LDAX F  
C12A CD 24 C2 8632 LDAX G  
C12A CD 24 C2 8633 LDAX H  
C12A CD 24 C2 8634 LDAX I  
C12A CD 24 C2 8635 LDAX J  
C12A CD 24 C2 8636 LDAX K  
C12A CD 24 C2 8637 LDAX L  
C12A CD 24 C2 8638 LDAX M  
C12A CD 24 C2 8639 LDAX N  
C12A CD 24 C2 8640 LDAX O  
C12A CD 24 C2 8641 LDAX P  
C12A CD 24 C2 8642 THIS ROUTINE GETS CHARACTERS FROM THE SYSTEM KEYBOARD  
C12A CD 24 C2 8643 AND OUTPUT THEM TO THE SERIAL PORT. IT IS  
C12A CD 24 C2 8644 IMPERFECT TO CONFIGURE THE S11 AS A STANDARD VIDEO  
C12A CD 24 C2 8645 TERMINAL. COMMAND KEYS ARE NOT OUTPUT TO THE OUTPUT  
C12A CD 24 C2 8646 PORT BUT ARE IMPERFECT AS DIRECT S11 COMMANDS.  
C12A CD 24 C2 8647 THE MODE COMMAND, RECEIVED BY THE KEYBOARD, PUTS THE  
C12A CD 24 C2 8648 KEY IN THE COMMAND MODE.  
C12A CD 24 C2 8649 \*  
C12A CD 24 C2 8650 \*  
C12A CD 24 C2 8651 TMRH LXI SP,BYSTP JST STAGE POINTER  
C12A CD 24 C2 8652 LDAX T0FT SLOW DOWN THE TAPES  
C12A CD 24 C2 8653 CALL PEASL CLEAR THE SCREEN  
C12A CD 24 C2 8654 \*  
C12A CD 24 C2 8655 KIN CALL RSTAT IS THERE ONE WAITING?  
C12A CD 24 C2 8656 JMS T0FT  
C12A CD 24 C2 8657 JS ERRI GET THE CHARACTER  
C12A CD 24 C2 8658 MOV RS,1  
C12A CD 24 C2 8659 AMI 884 COMMAND KEY? *CP*  
C12A CD 24 C2 8660 JMS T0FT  
C12A CD 24 C2 8661 CALL VDOS PUT IT ON THE SCREEN  
C12A CD 24 C2 8662 JRP TIN LOOP OVER AND OVER  
C12A CD 41 C4 8663 \*  
C12A CD 41 C4 8664 T0FT CALL SERBOT OUTPUT IT TO THE SERIAL PORT  
C12A CD 41 C4 8665 TIN ESTAT GET SERIAL STATUS  
C12A CD 41 C4 8666 JMS T0FT  
C12A CD 41 C4 8667 KIN LOOP IF NOT  
C12A CD 41 C4 8668 LDAX SDATA GET DATA  
C12A CD 41 C4 8669 LDAX H NO DATA FROM HERE  
C12A CD 41 C4 8670 MOV B,A IT'S OUTPUT FROM B'  
C12A CD 41 C4 8671 CALL VDOS PUT IT ON THE SCREEN  
C12A CD 41 C4 8672 JRP TIN LOOP OVER AND OVER  
C12A CD 41 C4 8673 \*  
C12A CD 41 C4 8674 \*  
C12A CD 41 C4 8675 \* DUMP COMMAND  
C12A CD 41 C4 8676 \*  
C12A CD 41 C4 8677 \* THIS ROUTINE DUMPS CHARACTERS FROM READY TO THE  
C12A CD 41 C4 8678 \* CURRENT OUTPUT DEVICE. (WITH CONSO ALL OUTPUT GOES TO  
C12A CD 41 C4 8679 \* THE SCREEN). ALL VALUES ARE DISPLAYED AS ASCII HEX.  
C12A CD 41 C4 8680 \*  
C12A CD 41 C4 8681 \* THE COMMAND FORM IS AS FOLLOWS:  
C12A CD 41 C4 8682 \*

C12A CD 41 C4 8683 \*  
C12A CD 41 C4 8684 \*  
C12A CD 41 C4 8685 \*  
C12A CD 41 C4 8686 \*  
C12A CD 41 C4 8687 \*  
C12A CD 41 C4 8688 \*  
C12A CD 41 C4 8689 \*  
C12A CD 41 C4 8690 \*  
C12A CD 41 C4 8691 \*  
C12A CD 41 C4 8692 \*  
C12A CD 41 C4 8693 \*  
C12A CD 41 C4 8694 \*  
C12A CD 41 C4 8695 \*  
C12A CD 41 C4 8696 \*  
C12A CD 41 C4 8697 \*  
C12A CD 41 C4 8698 \*  
C12A CD 41 C4 8699 \*  
C12A CD 41 C4 8700 \*  
C12A CD 41 C4 8701 \*  
C12A CD 41 C4 8702 \*  
C12A CD 41 C4 8703 \*  
C12A CD 41 C4 8704 \*  
C12A CD 41 C4 8705 \*  
C12A CD 41 C4 8706 \*  
C12A CD 41 C4 8707 \*  
C12A CD 41 C4 8708 \*  
C12A CD 41 C4 8709 \*  
C12A CD 41 C4 8710 \*  
C12A CD 41 C4 8711 \*  
C12A CD 41 C4 8712 \*  
C12A CD 41 C4 8713 \*  
C12A CD 41 C4 8714 \*  
C12A CD 41 C4 8715 \*  
C12A CD 41 C4 8716 \*  
C12A CD 41 C4 8717 \*  
C12A CD 41 C4 8718 \*  
C12A CD 41 C4 8719 \*  
C12A CD 41 C4 8720 \*  
C12A CD 41 C4 8721 \*  
C12A CD 41 C4 8722 \*  
C12A CD 41 C4 8723 \*  
C12A CD 41 C4 8724 \*  
C12A CD 41 C4 8725 \*  
C12A CD 41 C4 8726 \*  
C12A CD 41 C4 8727 \*  
C12A CD 41 C4 8728 \*  
C12A CD 41 C4 8729 \*  
C12A CD 41 C4 8730 \*  
C12A CD 41 C4 8731 \*  
C12A CD 41 C4 8732 \*  
C12A CD 41 C4 8733 \*  
C12A CD 41 C4 8734 \*  
C12A CD 41 C4 8735 \*  
C12A CD 41 C4 8736 \*  
C12A CD 41 C4 8737 \*  
C12A CD 41 C4 8738 \*  
C12A CD 41 C4 8739 \*  
C12A CD 41 C4 8740 \*  
C12A CD 41 C4 8741 \*  
C12A CD 41 C4 8742 \*  
C12A CD 41 C4 8743 \*  
C12A CD 41 C4 8744 \*  
C12A CD 41 C4 8745 \*  
C12A CD 41 C4 8746 \*  
C12A CD 41 C4 8747 \*  
C12A CD 41 C4 8748 \*  
C12A CD 41 C4 8749 \*  
C12A CD 41 C4 8750 \*  
C12A CD 41 C4 8751 \*  
C12A CD 41 C4 8752 \*  
C12A CD 41 C4 8753 \*  
C12A CD 41 C4 8754 \*  
C12A CD 41 C4 8755 \*  
C12A CD 41 C4 8756 \*  
C12A CD 41 C4 8757 \*  
C12A CD 41 C4 8758 \*  
C12A CD 41 C4 8759 \*  
C12A CD 41 C4 8760 \*  
C12A CD 41 C4 8761 \*  
C12A CD 41 C4 8762 \*  
C12A CD 41 C4 8763 \*  
C12A CD 41 C4 8764 \*  
C12A CD 41 C4 8765 \*  
C12A CD 41 C4 8766 \*  
C12A CD 41 C4 8767 \*  
C12A CD 41 C4 8768 \*  
C12A CD 41 C4 8769 \*  
C12A CD 41 C4 8770 \*  
C12A CD 41 C4 8771 \*  
C12A CD 41 C4 8772 \*  
C12A CD 41 C4 8773 \*  
C12A CD 41 C4 8774 \*  
C12A CD 41 C4 8775 \*  
C12A CD 41 C4 8776 \*  
C12A CD 41 C4 8777 \*  
C12A CD 41 C4 8778 \*  
C12A CD 41 C4 8779 \*  
C12A CD 41 C4 8780 \*  
C12A CD 41 C4 8781 \*  
C12A CD 41 C4 8782 \*  
C12A CD 41 C4 8783 \*  
C12A CD 41 C4 8784 \*  
C12A CD 41 C4 8785 \*  
C12A CD 41 C4 8786 \*  
C12A CD 41 C4 8787 \*  
C12A CD 41 C4 8788 \*  
C12A CD 41 C4 8789 \*  
C12A CD 41 C4 8790 \*  
C12A CD 41 C4 8791 \*  
C12A CD 41 C4 8792 \*  
C12A CD 41 C4 8793 \*  
C12A CD 41 C4 8794 \*  
C12A CD 41 C4 8795 \*  
C12A CD 41 C4 8796 \*  
C12A CD 41 C4 8797 \*  
C12A CD 41 C4 8798 \*  
C12A CD 41 C4 8799 \*  
C12A CD 41 C4 8800 \*  
C12A CD 41 C4 8801 \*  
C12A CD 41 C4 8802 \*  
C12A CD 41 C4 8803 \*  
C12A CD 41 C4 8804 \*  
C12A CD 41 C4 8805 \*  
C12A CD 41 C4 8806 \*  
C12A CD 41 C4 8807 \*  
C12A CD 41 C4 8808 \*  
C12A CD 41 C4 8809 \*  
C12A CD 41 C4 8810 \*  
C12A CD 41 C4 8811 \*  
C12A CD 41 C4 8812 \*  
C12A CD 41 C4 8813 \*  
C12A CD 41 C4 8814 \*  
C12A CD 41 C4 8815 \*  
C12A CD 41 C4 8816 \*  
C12A CD 41 C4 8817 \*  
C12A CD 41 C4 8818 \*  
C12A CD 41 C4 8819 \*  
C12A CD 41 C4 8820 \*  
C12A CD 41 C4 8821 \*  
C12A CD 41 C4 8822 \*  
C12A CD 41 C4 8823 \*  
C12A CD 41 C4 8824 \*  
C12A CD 41 C4 8825 \*  
C12A CD 41 C4 8826 \*  
C12A CD 41 C4 8827 \*  
C12A CD 41 C4 8828 \*  
C12A CD 41 C4 8829 \*  
C12A CD 41 C4 8830 \*  
C12A CD 41 C4 8831 \*  
C12A CD 41 C4 8832 \*  
C12A CD 41 C4 8833 \*  
C12A CD 41 C4 8834 \*  
C12A CD 41 C4 8835 \*  
C12A CD 41 C4 8836 \*  
C12A CD 41 C4 8837 \*  
C12A CD 41 C4 8838 \*  
C12A CD 41 C4 8839 \*  
C12A CD 41 C4 8840 \*  
C12A CD 41 C4 8841 \*  
C12A CD 41 C4 8842 \*  
C12A CD 41 C4 8843 \*  
C12A CD 41 C4 8844 \*  
C12A CD 41 C4 8845 \*  
C12A CD 41 C4 8846 \*  
C12A CD 41 C4 8847 \*  
C12A CD 41 C4 8848 \*  
C12A CD 41 C4 8849 \*  
C12A CD 41 C4 8850 \*  
C12A CD 41 C4 8851 \*  
C12A CD 41 C4 8852 \*  
C12A CD 41 C4 8853 \*  
C12A CD 41 C4 8854 \*  
C12A CD 41 C4 8855 \*  
C12A CD 41 C4 8856 \*  
C12A CD 41 C4 8857 \*  
C12A CD 41 C4 8858 \*  
C12A CD 41 C4 8859 \*  
C12A CD 41 C4 8860 \*  
C12A CD 41 C4 8861 \*  
C12A CD 41 C4 8862 \*  
C12A CD 41 C4 8863 \*  
C12A CD 41 C4 8864 \*  
C12A CD 41 C4 8865 \*  
C12A CD 41 C4 8866 \*  
C12A CD 41 C4 8867 \*  
C12A CD 41 C4 8868 \*  
C12A CD 41 C4 8869 \*  
C12A CD 41 C4 8870 \*  
C12A CD 41 C4 8871 \*  
C12A CD 41 C4 8872 \*  
C12A CD 41 C4 8873 \*  
C12A CD 41 C4 8874 \*  
C12A CD 41 C4 8875 \*  
C12A CD 41 C4 8876 \*  
C12A CD 41 C4 8877 \*  
C12A CD 41 C4 8878 \*  
C12A CD 41 C4 8879 \*  
C12A CD 41 C4 8880 \*  
C12A CD 41 C4 8881 \*  
C12A CD 41 C4 8882 \*  
C12A CD 41 C4 8883 \*  
C12A CD 41 C4 8884 \*  
C12A CD 41 C4 8885 \*  
C12A CD 41 C4 8886 \*  
C12A CD 41 C4 8887 \*  
C12A CD 41 C4 8888 \*  
C12A CD 41 C4 8889 \*  
C12A CD 41 C4 8890 \*  
C12A CD 41 C4 8891 \*  
C12A CD 41 C4 8892 \*  
C12A CD 41 C4 8893 \*  
C12A CD 41 C4 8894 \*  
C12A CD 41 C4 8895 \*  
C12A CD 41 C4 8896 \*  
C12A CD 41 C4 8897 \*  
C12A CD 41 C4 8898 \*  
C12A CD 41 C4 8899 \*  
C12A CD 41 C4 8900 \*  
C12A CD 41 C4 8901 \*  
C12A CD 41 C4 8902 \*  
C12A CD 41 C4 8903 \*  
C12A CD 41 C4 8904 \*  
C12A CD 41 C4 8905 \*  
C12A CD 41 C4 8906 \*  
C12A CD 41 C4 8907 \*  
C12A CD 41 C4 8908 \*  
C12A CD 41 C4 8909 \*  
C12A CD 41 C4 8910 \*  
C12A CD 41 C4 8911 \*  
C12A CD 41 C4 8912 \*  
C12A CD 41 C4 8913 \*  
C12A CD 41 C4 8914 \*  
C12A CD 41 C4 8915 \*  
C12A CD 41 C4 8916 \*  
C12A CD 41 C4 8917 \*  
C12A CD 41 C4 8918 \*  
C12A CD 41 C4 8919 \*  
C12A CD 41 C4 8920 \*  
C12A CD 41 C4 8921 \*  
C12A CD 41 C4 8922 \*  
C12A CD 41 C4 8923 \*  
C12A CD 41 C4 8924 \*  
C12A CD 41 C4 8925 \*  
C12A CD 41 C4 8926 \*  
C12A CD 41 C4 8927 \*  
C12A CD 41 C4 8928 \*  
C12A CD 41 C4 8929 \*  
C12A CD 41 C4 8930 \*  
C12A CD 41 C4 8931 \*  
C12A CD 41 C4 8932 \*  
C12A CD 41 C4 8933 \*  
C12A CD 41 C4 8934 \*  
C12A CD 41 C4 8935 \*  
C12A CD 41 C4 8936 \*  
C12A CD 41 C4 8937 \*  
C12A CD 41 C4 8938 \*  
C12A CD 41 C4 8939 \*  
C12A CD 41 C4 8940 \*  
C12A CD 41 C4 8941 \*  
C12A CD 41 C4 8942 \*  
C12A CD 41 C4 8943 \*  
C12A CD 41 C4 8944 \*  
C12A CD 41 C4 8945 \*  
C12A CD 41 C4 8946 \*  
C12A CD 41 C4 8947 \*  
C12A CD 41 C4 8948 \*  
C12A CD 41 C4 8949 \*  
C12A CD 41 C4 8950 \*  
C12A CD 41 C4 8951 \*  
C12A CD 41 C4 8952 \*  
C12A CD 41 C4 8953 \*  
C12A CD 41 C4 8954 \*  
C12A CD 41 C4 8955 \*  
C12A CD 41 C4 8956 \*  
C12A CD 41 C4 8957 \*  
C12A CD 41 C4 8958 \*  
C12A CD 41 C4 8959 \*  
C12A CD 41 C4 8960 \*  
C12A CD 41 C4 8961 \*  
C12A CD 41 C4 8962 \*  
C12A CD 41 C4 8963 \*  
C12A CD 41 C4 8964 \*  
C12A CD 41 C4 8965 \*  
C12A CD 41 C4 8966 \*  
C12A CD 41 C4 8967 \*  
C12A CD 41 C4 8968 \*  
C12A CD 41 C4 8969 \*  
C12A CD 41 C4 8970 \*  
C12A CD 41 C4 8971 \*  
C12A CD 41 C4 8972 \*  
C12A CD 41 C4 8973 \*  
C12A CD 41 C4 8974 \*  
C12A CD 41 C4 8975 \*  
C12A CD 41 C4 8976 \*  
C12A CD 41 C4 8977 \*  
C12A CD 41 C4 8978 \*  
C12A CD 41 C4 8979 \*  
C12A CD 41 C4 8980 \*  
C12A CD 41 C4 8981 \*  
C12A CD 41 C4 8982 \*  
C12A CD 41 C4 8983 \*  
C12A CD 41 C4 8984 \*  
C12A CD 41 C4 8985 \*  
C12A CD 41 C4 8986 \*  
C12A CD 41 C4 8987 \*  
C12A CD 41 C4 8988 \*  
C12A CD 41 C4 8989 \*  
C12A CD 41 C4 8990 \*  
C12A CD 41 C4 8991 \*  
C12A CD 41 C4 8992 \*  
C12A CD 41 C4 8993 \*  
C12A CD 41 C4 8994 \*  
C12A CD 41 C4 8995 \*  
C12A CD 41 C4 8996 \*  
C12A CD 41 C4 8997 \*  
C12A CD 41 C4 8998 \*  
C12A CD 41 C4 8999 \*  
C12A CD 41 C4 9000 \*  
C12A CD 41 C4 9001 \*  
C12A CD 41 C4 9002 \*  
C12A CD 41 C4 9003 \*  
C12A CD 41 C4 9004 \*  
C12A CD 41 C4 9005 \*  
C12A CD 41 C4 9006 \*  
C12A CD 41 C4 9007 \*  
C12A CD 41 C4 9008 \*  
C12A CD 41 C4 9009 \*  
C12A CD 41 C4 9010 \*  
C12A CD 41 C4 9011 \*  
C12A CD 41 C4 9012 \*  
C12A CD 41 C4 9013 \*  
C12A CD 41 C4 9014 \*  
C12A CD 41 C4 9015 \*  
C12A CD 41 C4 9016 \*  
C12A CD 41 C4 9017 \*  
C12A CD 41 C4 9018 \*  
C12A CD 41 C4 9019 \*  
C12A CD 41 C4 9020 \*  
C12A CD 41 C4 9021 \*  
C12A CD 41 C4 9022 \*  
C12A CD 41 C4 9023 \*  
C12A CD 41 C4 9024 \*  
C12A CD 41 C4 9025 \*  
C12A CD 41 C4 9026 \*  
C12A CD 41 C4 9027 \*  
C12A CD 41 C4 9028 \*  
C12A CD 41 C4 9029 \*  
C12A CD 41 C4 9030 \*  
C12A CD 41 C4 9031 \*  
C12A CD 41 C4 9032 \*  
C12A CD 41 C4 9033 \*  
C12A CD 41 C4 9034 \*  
C12A CD 41 C4 9035 \*  
C12A CD 41 C4 9036 \*  
C12A CD 41 C4 9037 \*  
C12A CD 41 C4 9038 \*  
C12A CD 41 C4 9039 \*  
C12A CD 41 C4 9040 \*  
C12A CD 41 C4 9041 \*  
C12A CD 41 C4 9042 \*  
C12A CD 41 C4 9043 \*  
C12A CD 41 C4 9044 \*  
C12A CD 41 C4 9045 \*  
C12A CD 41 C4 9046 \*  
C12A CD 41 C4 9047 \*  
C12A CD 41 C4 9048 \*  
C12A CD 41 C4 9049 \*  
C12A CD 41 C4 9050 \*  
C12A CD 41 C4 9051 \*  
C12A CD 41 C4 9052 \*  
C12A CD 41 C4 9053 \*  
C12A CD 41 C4 9054 \*  
C12A CD 41 C4 9055 \*<br

## SOFTWARE SECTION

```

C300 C2 T3 C3    d554    JNE    M001    STILL WORK IF NOT READ
C300 d555    CALL    CLKCR    CHECK CRC AND FALL THROUGH TO ERROR IF NO GOOD
C300 C0 C1 C3    d557    JE    LOADUP    TEST FOR
C300 C4 C1 C3    d558    LD    0000
C300 d559    BRA    A6_G-400    BELL CHARACTER
C300 C0 C2 C4    d560    CALL    VPRINT    PUT IT ON THE SCREEN
C300 C1 C4 C1    d561    JMP    COMM0
C300 d562    *
C300 d563    *
C300 d564    * CONSUL ERROR HANDLER
C300 d565    *
C300 d566    * GET SCAN ADDRESS
C300 C0 3F    d567    BRAZ    AVI    M-?    PUT A QUESTION MARK IN HERE
C300 C1 04 C1    d568    JNP    COMM0
C300 d569    *
C300 d570    * READ THE HEADER
C300 d571    *
C300 C0 BA    d572    XREAD    AVI    B1b    FIND 10 NULLS
C300 C0 BA    d573    IN    START    GET A BYTE
C300 C0 1D    d574    LD    0000
C300 CA 95 C3    d575    JE    RELAI
C300 C0 FM    d576    IM    TOATA    IGNORE ERROR CONDITIONS
C300 C0 77    d577    JNA    A    ZERO?
C300 C2 94 C3    d578    JNE    RELAD
C300 V5 4    d579    DCA    B
C300 C2 96 C3    d580    JNE    RELAI
C300 d581    * LOOP UNTIL IN IN A ROM
C300
C300 d582    * WAIT FOR THE START CHARACTER
C300 d583    *
C300 C0 C6 C3    d584    BRL    TAPIN
C300 C0 1D    d585    DCA
C300 C2 A7 C3    d586    JNE    S001    WAIT FOR A '1'
C300 d587    *
C300 d588    * NOT GET THE HEADER
C300 C1 05 C6    d589    LXI    H,TMLOAD    POINT TO BUFFER
C300 W1 1D    d590    LXI    W,LEN=756    LENGTH OF HEADER IN 'B', CCE
C300 d591    *
C300 C0 C6 C3    d592    BRL    TAPIN
C300 C0 77    d593    BRAZ    AVI    GET BYTE
C300 C0 23    d594    INX    H-A    INCREMENT ADDRESS
C300 C0 29    d595    BRAZ    C    NOW CALCULATE THE CRC
C300 Y1    d596    BRAZ    C    INDEX DOWN AND UPSIDE DOWN
C300 49    d597    BRAZ    C    SQUISH IT
C300 05    d598    ADV    C,A    AND SAVE AGAIN
C300 C2 44 C3    d599    DCR    B    WHILE HEADER YET?
C300 d600    JNE    S001
C300
C300 d601    * THIS ROUTINE GETS THE NEXT BYTE FROM THE
C300 C0 1D    d602    TO THE VALUE IN REGISTER C. THE FLAGS ARE SET ON
C300 C0 1D    d603    RET    RETURN
C300 C0 C6 C3    d604    BRL    TAPIN    CALL    TAPIN    GET CRC DATA
C300 C0 93    d605    CMP    C    COMPARE IT WITH CALCULATED
C300 d606    RET
C300 C0 C6 C3    d607    BRL    TAPIN    CALL    TAPIN    GET CRC DATA
C300 C0 93    d608    CMP    C    COMPARE IT WITH CALCULATED
C300 d609    RET
C300 C0 C6 C3    d610    BRL    TAPIN    CALL    TAPIN    GET CRC DATA
C300 C0 93    d611    CMP    C    COMPARE IT WITH CALCULATED
C300 d612    RET
C300 C0 C6 C3    d613    BRL    TAPIN    CALL    TAPIN    GET CRC DATA
C300 C0 93    d614    CMP    C    COMPARE IT WITH CALCULATED
C300 d615    RET
C300 C0 C6 C3    d616    BRL    TAPIN    CALL    TAPIN    GET CRC DATA
C300 C0 93    d617    CMP    C    COMPARE IT WITH CALCULATED
C300 d618    RET
C300 C0 C6 C3    d619    BRL    TAPIN    CALL    TAPIN    GET CRC DATA
C300 C0 93    d620    CMP    C    COMPARE IT WITH CALCULATED
C300 d621    RET
C300 C0 C6 C3    d622    BRL    TAPIN    CALL    TAPIN    GET CRC DATA
C300 C0 93    d623    CMP    C    COMPARE IT WITH CALCULATED
C300 d624    RET
C300 C0 C6 C3    d625    BRL    TAPIN    CALL    TAPIN    GET CRC DATA
C300 C0 93    d626    CMP    C    COMPARE IT WITH CALCULATED
C300 d627    RET
C300 C0 C6 C3    d628    BRL    TAPIN    CALL    TAPIN    GET CRC DATA
C300 C0 93    d629    CMP    C    COMPARE IT WITH CALCULATED
C300 d630    RET
C300 C0 C6 C3    d631    BRL    TAPIN    CALL    TAPIN    GET CRC DATA
C300 C0 93    d632    CMP    C    COMPARE IT WITH CALCULATED
C300 d633    RET
C300 C0 FF FF    d634    LXI    H,-1
C300 C0 00 00    d635    DAU    B    COMPLEMENT NL
C300 C0 23    d636    INX    H    -----TWO'S
C300 C0 45    d637    MOV    B,L    LENGTH TO HIGHEST B
C300 C0 00 00 00    d638    LXI    H,D    TELL ME HE DOWN
C300 C0 C1 00 00    d639    JMF    ADDRESS    FORWARD TO THE END
C300 C0 00 00    d640    *
C300 C0 00 00    d641    * THIS ROUTINE TURNS THE TAPE UNITS OFF
C300 C0 00 00    d642    OUT    A    CHECK STATUS
C300 C0 00 00    d643    OUT    T0A    DATA ERROR?
C300 C0 00 00    d644    OUT    T0B    IF FRANTIC OR OVERBURN ERROR
C300 C0 00 00    d645    OUT    T0A    GET THE DATA
C300 C0 00 00    d646    RET
C300 C0 00 00    d647    * THIS ROUTINE CALCULATES THE LENGTH OF THE LAST BLOCK
C300 C0 00 00    d648    LXI    H,-1
C300 C0 00 00    d649    BRAZ    A    DATA ERROR?
C300 C0 00 00    d650    BRAZ    T0B    IF FRANTIC OR OVERBURN ERROR
C300 C0 00 00    d651    BRAZ    T0A    GET THE DATA
C300 C0 00 00    d652    RET
C300 C0 00 00    d653    *
C300 C0 00 00    d654    *
C300 C0 00 00    d655    *
C300 C0 00 00    d656    *
C300 C0 00 00    d657    * < SOL SYSTEM LOCATES >
C300 C0 00 00    d658    *
C300 C0 00 00    d659    *
C300 C0 00 00    d660    DISPLAY PARAMETERS
C300 C0 00 00    d661    *
C300 C0 00 00    d662    VDUMEM    EQU    000000    SCREEN MEMORY
C300 C0 00 00    d663    *
C300 C0 00 00    d664    *
C300 C0 00 00    d665    KEYBOARD SPECIAL KEY ASSIGNMENTS
C300 C0 00 00    d666    *
C300 C0 00 00    d667    UP-H    EQU    9AH
C300 C0 00 00    d668    UP-V    EQU    97H
C300 C0 00 00    d669    LEFT    EQU    81H
C300 C0 00 00    d670    RIGHT   EQU    93H
C300 C0 00 00    d671    DOWN    EQU    8DH
C300 C0 00 00    d672    HOME    EQU    80H
C300 C0 00 00    d673    CLEAR    EQU    84H
C300 C0 00 00    d674    END-E    EQU    82H
C300 C0 00 00    d675    PAGE-D    EQU    8FH
C300 C0 00 00    d676    PAGE-U    EQU    8EH
C300 C0 00 00    d677    CF    EQU    11H
C300 C0 00 00    d678    LOU    EQU    80H
C300 C0 00 00    d679    BLANK    EQU    00H
C300 C0 00 00    d680    CR-AFH    EQU    00H
C300 C0 00 00    d681    *
C300 C0 00 00    d682    PORT ASSIGNMENTS
C300 C0 00 00    d683    STATT    EQU    #FFH    STATUS PORT GENERAL
C300 C0 00 00    d684    SSTAT    EQU    #FFH    SERIAL STATUS PORT
C300 C0 00 00    d685    SDATA    EOU    #FFH    SERIAL DATA
C300 C0 00 00    d686    RDATA    EOU    #FFH    SERIAL DATA
C300 C0 00 00    d687    KDATA    EOU    #FFH    KEYBOARD DATA
C300 C0 00 00    d688    PDATA    EOU    #FFH    PARALLEL DATA
C300 C0 00 00    d689    PSTAT    EOU    #FFH    DISPLAY PARAMETER PORT
C300 C0 00 00    d690    SESSGE    EOU    #FFH    SENSOR SWITCHES
C300 C0 00 00    d691    *
C300 C0 00 00    d692    *
C300 C0 00 00    d693    *
C300 C0 00 00    d694    BIT ASSIGNMENT MASKS
C300 C0 00 00    d695    *
C300 C0 00 00    d696    LCD    EOU    1    SERIAL CARRIER DETECT
C300 C0 00 00    d697    LCD    EOU    2    SERIAL DATA NOT READY
C300 C0 00 00    d698    SCK    EOU    4    SERIAL TRANSMITTER SHOT
C300 C0 00 00    d699    SCK    EOU    6    SERIAL FRAMING ERROR
C300 C0 00 00    d700    SCK    EOU    8    SERIAL OVERRUN ERROR
C300 C0 00 00    d701    SCK    EOU    16    SERIAL CLEAN TO BAND
C300 C0 00 00    d702    SCK    EOU    32    SERIAL CARRIER RESTART
C300 C0 00 00    d703    SCK    EOU    44    SERIAL TRANSMITTER
C300 C0 00 00    d704    SCK    EOU    128    SERIAL TRANSMITTER BUFFER EMPTY
C300 C0 00 00    d705    *
C300 C0 00 00    d706    EDR    EOU    1    PARALLEL DATA READY
C300 C0 00 00    d707    EDR    EOU    2    PARALLEL DATA NOT READY
C300 C0 00 00    d708    PDATA    EOU    4    PARALLEL DEVICE READY
C300 C0 00 00    d709    PDATA    EOU    8    TAPE FRAMING ERROR
C300 C0 00 00    d710    TUE    EOU    16    TAPE OVERFLOW ERROR
C300 C0 00 00    d711    TUE    EOU    64    TAPE DATA READY
C300 C0 00 00    d712    TUE    EOU    128    TAPE TRANSMITTER BUFFER EMPTY
C300 C0 00 00    d713    *
C300 C0 00 00    d714    SCK    EOU    1    SCROLL OR FLAG
C300 C0 00 00    d715    SCK    EOU    1    TAPE ONE 'ON' BIT
C300 C0 00 00    d716    TAPE1    EOU    64    TAPE ONE
C300 C0 00 00    d717    TAPE2    EOU    128    TAPE TWO
C300 C0 00 00    d718    *
C300 C0 00 00    d719    *
C300 C0 00 00    d720    *
C300 C0 00 00    d721    *
C300 C0 00 00    d722    * SOL SYSTEM GLOBAL AREA
C300 C0 00 00    d723    *
C300 C0 00 00    d724    ORG    0C800H    START OF 1K RAM AREA
C300 C0 00 00    d725    S124    S    START OF SYSTEM RAM
C300 C0 00 00    d727    STATT    EOU    $1d44    STACK IS AT THE TOP

```

## MICROCOMPUTER DEVELOPMENT SOFTWARE

```

Code    1020    *
Code    1029    *
Code    1030    *
Code    1031    * CONSUL PARAMETER AREA
Code    1032    MCHAR    DS    1    CURRENT CHARACTER POSITION
Code    1033    LINE    DS    1    CURRENT LINE POSITION
Code    1034    M2L    DS    1    BEGINNING OF TEXT DISPLACEMENT
Code    1035    M2R    DS    1    OUTPUT PORT
Code    1036    IPUART    DS    1    INPUT PORT
Code    1037    *
Code    1038    *
Code    1039    * ImAD    DS    5    NAME
Code    1040    * LSTN    DS    1    THIS BYTE MUST BE ZERO
Code    1041    * TYPE    DS    1    TYPE
Code    1042    * LBL    DS    2    BLOCK SIZE
Code    1043    * BLACK    DS    2    ADDRESS
Code    1044    * IN    DS    2    AUTO EXECUTE ADDRESS
Code    1045    * ILOAD    DS    2    SPANL
Code    1046    * MWD    DS    3    LENGTH OF HEADER
Code    1047    *
Code    1048    * LBL    EQU    1-THEAD    LENGTH OF HEADER
Code    1049    *
Code    1050    * 1450    EQU    *
Code    1051    * 1850    *
Code    1052    *

```

## LET'S HEAR FROM YOU

Your letters are the lifeblood for us

We really do appreciate your letters and comments about INTERFACE AGE. We need to know what you think about the publication so that we know in what ways we can make it better.

If you like it, tell us, and if there are things you think could be improved, tell us that, too.

We do read all our mail and we appreciate your feedback.

The Editors

## Bits and Bytes COMPUTER SHOP IN

### PHOENIX

AFFORDABLE — OFF THE SHELF

**WE FEATURE** the **EPA MICRO-68** and **EPA Expanded-68** microcomputers. These fine systems are based on the popular Motorola 6800.

**WE FEATURE** the **Micro-Term ACT-I** keyboard and **Sanyo** high resolution Video Monitor as a TTY replacement, for the lowest possible cost CRT.

**ALSO . . .** we sell a full line of EPA boards and computer peripherals as well as a large selection of books and magazines.

OPEN EVENINGS AND WEEKENDS

6819 North 21st Avenue  
Phoenix, Arizona 85015  
(602) 242-2507

CIRCLE INQUIRY NO. 37

ASCII CHARACTER	OUTPUT 87654321	ASCII CHARACTER	OUTPUT 87654321	ASCII CHARACTER	OUTPUT 87654321
NULL	00000000	7 <sup>w</sup>	10110111	o	01101111
SOH	10000001	8 <sup>F</sup>	10111000	p	11110000
STX	10000010	<sup>FB Home</sup>	9 <sup>y</sup>	q	01110001
ETX	00000011	:	00111010	r	01110010
EOT	10000100	<sup>FD CLEAR SCREEN</sup>	{	s	11110011
ENQ	00000101	LESS THAN :	00111100	t	01110100
ACK	00000110	<sup>↑ F MODE</sup>	EQUALS }	u	11110101
BELL <sup>Sol on beam</sup>	00001111	MORE THAN ~	10111110	v	11110110
BS <sup>Backspace</sup>	00010000	?	00111111	w	01110111
HT <sup>TAB</sup>	00001001	@	11000000	x	01111000
LF <sup>LineFeed</sup>	00001010	<sup>φ A<sub>H</sub> 10<sub>D</sub></sup>	A	y	11111001
VT	10001011		B	z	11111010
FF	00001100		C		01111011
CR <sup>Carriage Return</sup>	00011011	<sup>φ D<sub>H</sub> 13<sub>D</sub></sup>	D		11111100
SO	10001110		E		01111101
SI	00001111		F		01111110
DLE	10010000		G		~
DC1	00010001		H		DEL Delete
DC2	00010010		I		11111111
DC3	10010011		J		
DC4	00010100		K		
NAK	10010101		L		
SYN	10010110		M		
ETB	00010111		N		
CAN	00011000		O		
EM	10011001		P		
SUB	10011010		Q		
ESC	00011011		R		
FS	10011100		S		
GS	00011101		T		
RS	00011110		U		
US	10011111		V		
SPACE	10100000		W		
! <sup>a</sup>	00100001		X		
" <sup>b</sup>	00100010		Y		
# <sup>c</sup>	01000111		Z		
\$ <sup>d</sup>	00100100				
% <sup>e</sup>	10100101				
& <sup>f</sup>	10100110				
' <sup>g</sup>	00100111				
( <sup>h</sup>	00101000				
) <sup>i</sup>	01010001				
* <sup>j</sup>	01010100				
PLUS <sup>k</sup>	00101011				
, <sup>l</sup>	01011100				
MINUS <sup>m</sup>	00101101				
. <sup>n</sup>	00101110				
/ <sup>o</sup>	10101111				
0 <sup>p</sup>	00110000				
1 <sup>q</sup>	10110001				
2 <sup>r</sup>	10110010				
3 <sup>s</sup>	00110011				
4 <sup>t</sup>	10110100				
5 <sup>u</sup>	00110101				
6 <sup>v</sup>	00110110				

ASCII CHARACTER	OUTPUT 87654321	ASCII CHARACTER	OUTPUT 87654321	ASCII CHARACTER	OUTPUT 87654321
NULL	00000000	7	10110111	o	01101111
SOH	10000001	8	10111000	p	11110000
STX	10000010	FB HONE	00111001	q	01110001
ETX	00000011	:	00111010	r	01110010
EOT	10000100	FD CLEAR SCREEN	10111011	s	11110011
ENQ	00000101	LESS THAN	00111100	t	01110100
ACK	00000110	↑F NODE	10111101	u	11110101
BELL	SOL on board	EQUALS	10111110	v	11110110
BS	Backspace	MORE THAN	00111111	w	01110111
HT	TAB	?	01100000	x	01111000
LF	LineFeed	@	01000001	y	11111001
VT		A	01000010	z	11111010
FF		B	11000011	{	01111011
CR	Return	C	01000100	}	11111100
SO		D	11000101	~	01111101
SI		E	11000110	DEL	Delete
DLE		F	01000111		11111111
DC1		G	01001000		
DC2		H	11001001		
DC3		I	11001010		
DC4		J	01001011		
NAK		K	11001100		
SYN		L	01001101		
ETB		M	01001110		
CAN		N	11001111		
EM		O	01010000		
SUB		P	11010001		
ESC		Q	11010010		
FS		R	01010011		
GS		S	11010100		
RS		T	01010101		
US		U	01010110		
		V	11010111		
SPACE	10100000	X	11011000		
!	00100001	Y	01011001		
"	00100010	Z	01011010		
#	10100011	[	11011011		
\$	00100100	/	01011100		
%	10100101	]	11011101		
&	10100110	^	11011110		
-	00100111	-	01011111		
(	00101000	\	01100000		
)	10101001	a	11100001		
*	10101010	b	11100010		
PLUS	00101011	c	01100011		
,	10101100	d	11100100		
MINUS	00101101	e	01100101		
.	00101110	f	01100110		
/	10101111	g	11100111		
0	00110000	h	11101000		
1	10110001	i	01101001		
2	10110010	j	01101010		
3	00110011	k	11101011		
4	10110100	l	01101100		
5	00110101	m	11101101		
6	00110110	n	11101110		

lower case = '00100001' B OR lowercase

```

C000          org  0c000h
C000 00        DB    0
C001 C365C2    BGIN: JMP  STRTA
*
*   JUMP TABLE I/O ROUTINES
*   0 - DISPLAY SCREEN
*   1 - SERIAL OUTPUT
*   2 - PARALLEL OUTPUT
*   3 - ERROR HANDLER
* ENTRY AT: SOUT SELECTS CURRENT OUTPUT DEVICE
*           AOUT SELECTS DEVICE IN REGISTER "A"
C004 3A03C8    SOUT: LDA   OPORT
C007 E603      AOUT: ANI   3
C009 E5        PUSH  H
C00A 2184C1    LXI   H,OTAB
C00D 07        RLC
C00E 85        ADD   L
C00F 6F        MOV   L,A
C010 C371C0    JMP   DISPT
*
*   ENTRY POINTS ARE DEFINED:
*   0 - KEYBOARD IN
*   1 - SERIAL IN
*   2 - PARALLEL IN
*   3 - ERROR HANDLER
C013 3A04C8    SINP: LDA   IPORT
C016 E603      AINP: ANI   3
C018 E5        PUSH  H
C019 218CC1    LXI   H,ITAB
C01C 07        RLC
C01D 85        ADD   L
C01E 6F        MOV   L,A
C01F C371C0    JMP   DISPT
*
*   KEYBOARD INPUT STATUS CHECK
C022 DBFA      KSTAT: IN    STAPT
C024 E601      ANI   KDR
C026 C9        RET
*
*   KEYBOARD DATA INPUT
C027 E1        KREAD: POP   H
C028 CD22C0    CALL  KSTAT
C02B C228C0    JNZ   KREAD
C02E DBFC      IN    KDATA
C030 C9        RET
*
*   SERIAL INPUT STATUS CHECK
C031 DBF8      SSTAT: IN    SERST
C033 E640      ANI   SDR
C035 C9        RET
*

```

```

* SERIAL DATA INPUT
C036 E1      SREA1: POP H
C037 CD31C0   SREAD: CALL SSTAT
C03A CA37C0   JZ    SREAD
C03D DBF9     IN    SDATA
C03F C9       RET

*
* SERIAL DATA OUTPUT
C040 E1      SEROT: POP H
C041 DBF8     SDROT: IN  SERST
C043 17       RAL
C044 D241C0   JNC  SDROT
C047 78       MOV  A,B
C048 D3F9     OUT  SDATA
C04A C9       RET

*
* VIDEO DISPLAY DRIVER
C04B E5      VDMOT: PUSH H
C04C D5      VDMO1: PUSH D
C04D C5      PUSH B
C04E F5      PUSH PSW
C04F 78      MOV  A,B
C050 2165C1   LXI  H,TBL
C053 CD62C0   CALL TSRCH

*
C056 CD21C1   GOBACK: CALL VDADD
C059 7E        MOV  A,M
C05A F680     ORI  80H
C05C 77        MOV  M,A
C05D F1        GOBK:  POP  PSW
C05E C1        POP  B
C05F D1        POP  D
C060 E1        POP  H
C061 C9        RET

*
C062 7E        TSRCH: MOV  A,M
C063 B7        ORA  A
C064 CA7BC0   JZ   CHAR
C067 B8        CMP  B
C068 23        INX  H
C069 C276C0   JNZ  NEXT
C06C E5        PUSH H
C06D CD3DC1   CALL CREM
C070 E1        POP  H

*
* DISPATCH
C071 7E        DISPT: MOV  A,M
C072 23        INX  H
C073 66        MOV  H,M
C074 6F        MOV  L,A
C075 E9        PCHL

*
C076 23        NEXT: INX  H

```

C077	23		INX	H
C078	C362CO		JMP	TSRCH
*				
C07B	78	CHAR:	MOV	A,B
C07C	B7		ORA	A
C07D	C8		RZ	
C07E	FE7F		CPI	7FH
C080	C8		RZ	
*				
C081	CD21C1	OCHAR:	CALL	VDADD
C084	78		MOV	A,B
C085	E67F		ANI	7FH
C087	77		MOV	M,A
C088	3A00C8		LDA	NCHAR
C08B	FE3F		CPI	63
C08D	DAADCO		JC	OK
C090	3A01C8		LDA	LINE
C093	FE0F		CPI	15
C095	C2ADCO		JNZ	OK
*				
* END OF SCREEN. ROLL UP ONE LINE				
C098	AF	SCROLL:	XRA	A
C099	3200C8		STA	NCHAR
C09C	4F	SROL:	MOV	C,A
C09D	CD28C1		CALL	VDAD
COAO	AF		XRA	A
COA1	CDE9C0		CALL	CLIN1
COA4	3A02C8		LDA	BOT
COA7	3C		INR	A
COA8	E60F		ANI	OFH
COAA	C3DDCO		JMP	ERAS3
*				
* INCREMENT LINE COUNTER, IF NECESSARY				
COAD	3A00C8	OK:	LDA	NCHAR
COBO	3C		INR	A
COB1	3200C8		STA	NCHAR
COB4	FE40		CPI	64
COB6	D8		RC	
COB7	AF		XRA	A
COB8	3200C8		STA	NCHAR
COBB	3A01C8		LDA	LINE
COBE	3C		INR	A
COBF	E60F		ANI	OFH
COC1	3201C8	CUR:	STA	LINE
COC4	C9		RET	
*				
* ERASE SCREEN				
COC5	2100CC	PERSE:	LXI	H,VDMEM
COC8	36AO		MVI	M,80H+' ' ;CURSOR
*				
COCA	23	ERAS1:	INX	H
COCB	7C		MOV	A,H
COCC	FEDO		CPI	ODOH

COCE	D2D6C0	jnc	eras2
COD1	3620	MVI	M, ' '
COD3	C3CAC0	JMP	ERAS1
*			
COD6	AF	ERAS2:	XRA A
COD7	3201C8		STA LINE
CODA	3200C8		STA NCHAR
*			
CODD	D3FE	ERAS3:	OUT DSTAT
CODF	3202C8		STA BOT
COE2	C9		RET
*			
COE3	CD21C1	CLINE:	CALL VDADD
COE6	3A00C8		LDA NCHAR
COE9	FE40	CLIN1:	CPI 64
COEB	DO		RNC
COEC	3620		MVI M, ' '
COEE	23		INX H
COEF	3C		INR A
COFO	C3E9C0		JMP CLIN1
*			
* HOME CURSOR			
COF3	AF	PHOME:	XRA A
COF4	3200C8		STA NCHAR
COF7	C3C1C0		JMP CUR
*			
* MOVE CURSOR DOWN ONE LINE			
COFA	3A01C8	PDOWN:	LDA LINE
COFD	FE0F		CPI 15
COFF	C8		RZ
C100	3C		INR A
C101	C3C1C0		JMP CUR
*			
* MOVE CURSOR UP ONE LINE			
C104	3A01C8	PUP:	LDA LINE
C107	B7		ORA A
C108	C8		RZ
C109	3D		DCR A
C10A	C3C1C0		JMP CUR
*			
* MOVE CURSOR LEFT ONE COLUMN			
C10D	3A00C8	PLEFT:	LDA NCHAR
C110	B7		ORA A
C111	C8		RZ
C112	3D		DCR A
C113	3200C8	PCUR:	STA NCHAR
C116	C9		RET
*			
* MOVE CURSOR RIGHT ONE COLUMN			
C117	3A00C8	PRIT:	LDA NCHAR
C11A	FE3F		CPI 63
C11C	C8		RZ
C11D	3C		INR A

C11E C313C1	JMP PCUR
	*
	* CALCULATE SCREEN ADDRESSES
	* VDADD=CURRENT SCREEN ADDRESS
	* VDAD2=ADDRESS OF LINE
	* VDAD =LINE IN 'A', CHARACTER IN 'C'
C121 3AOOC8	VDADD: LDA NCHAR
C124 4F	MOV C,A
C125 3AO1C8	VDAD2: LDA LINE
C128 6F	VDAD: MOV L,A
C129 3AO2C8	LDA BOT
C12C 85	ADD L
C12D OF	RRC
C12E OF	RRC
C12F 6F	MOV L,A
C130 E603	ANI 3
C132 57	MOV D,A
C133 3ECC	MVI A,VDMEM SHR 8
	*
	Mvi A,<VDMEM
C135 82	ADD D
C136 67	MOV H,A
C137 7D	MOV A,L
C138 E6CO	ANI OCOH
C13A 81	ADD C
C13B 6F	MOV L,A
C13C C9	RET
	*
	* REMOVE CURSOR
C13D CD21C1	CREM: CALL VDADD
C140 7E	MOV A,M
C141 E67F	ANI 7FH
C143 77	MOV M,A
C144 C9	RET
	*
	* BACKSPACE
C145 CDODC1	PBACK: CALL PLEFT
C148 CD21C1	CALL VDADD
C14B 3620	MVI M,' '
C14D C9	RET
	*
	* CARRIAGE RETURN
C14E CDE3CO	PCR: CALL CLINE
C151 AF	XRA A
C152 C313C1	JMP PCUR
	*
	* LINEFEED
C155 3AO1C8	PLF: LDA LINE
C158 FEOF	CPI 15
C15A D261C1	JNC SC
C15D 3C	INR A
C15E C3C1CO	JMP CUR
	*
C161 AF	SC: XRA A

C162 C39CC0	JMP	SROL
	*	
	* TABLE OF SPECIAL CHARACTERS	
	* AND WHERE TO PROCESS	
	* IF CHAR IS NOT IN TABLE,	
	* THEN IT GOES TO THE SCREEN	
C165 04	TBL:	DB CLEAR
C166 C5CO		DW PERSE
C168 97		DB UP
C169 04C1		DW PUP
C16B 9A		DB DOWN
C16C FAC0		DW PDOWN
C16E 81		DB LEFT
C16F ODC1		DW PLEFT
C171 93		DB RIGHT
C172 17C1		DW PRIT
C174 02		DB HOME
C175 F3CO		DW PHOME
C177 0D		DB CR
C178 4EC1		DW PCR
C17A 0A		DB LF
C17B 55C1		DW PLF
C17D 08		DB BACKS
C17E 45C1		DW PBACK
C180 06		DB MODE
C181 B4C1		DW COMND
C183 00		DB O
	*	
	* TABLE OF OUTPUT DEVICES	
C184 4CC0	OTAB:	DW VDMO1
C186 40CO		DW SEROT
C188 ADC1		DW ERROT
C18A ADC1		DW ERROT
	*	
	* TABLE OF INPUT DEVICES	
C18C 27CO	ITAB:	DW KREA1
C18E 36CO		DW SREA1
C190 ADC1		DW ERROT
C192 ADC1		DW ERROT
	*	
	* TABLE OF COMMANDS	
C194 5445	COMTAB:	DB 'TE'
C196 70C2		DW TERM
C198 4455		DB 'DU'
C19A A1C2		DW DUMP
C19C 454E		DB 'EN'
C19E 07C3		DW ENTER
C1A0 4558		DB 'EX'
C1A2 35C3		DW EXEC
C1A4 544C		DB 'TL'
C1A6 39C3		DW TLOAD
C1A8 4241		DB 'BA'
C1AA 0000		DW O

C1AC 00	DB 0
	*
	* CONSOL PORT ERROR HANDLER
C1AD AF	ERROT: XRA A
C1AE 3204C8	STA IPORT
C1B1 3203C8	STA OPORT
	*
	* GET AND PROCESS COMMANDS
C1B4 3100CC	COMND: LXI SP,SYSTP
C1B7 CD12C2	CALL PROMPT
C1BA CDC3C1	CALL GCLIN
C1BD CDE1C1	CALL COPRC
C1CO C3B4C1	JMP COMND
	*
	* READ COMMAND FROM KEYBOARD
	* AND SEND IT TO THE SCREEN
	* CR ENDS COMMAND AND ERASES ALL CHARS TO RIGHT
	* OF THE CURSOR
	* LF ENDS COMMAND
	* MODE RESTARTS THE COMMAND LINE
C1C3 CD28C0	GCLIN: CALL KREAD
C1C6 FE20	CPI 20H
C1C8 47	MOV B,A
C1C9 DAD2C1	JC PROCS
C1CC CD4BC0	CONT: CALL VDMOT
C1CF C3C3C1	JMP GCLIN
	*
	* PROCESS CONTROL KEYS
C1D2 FE0D	PROCS: CPI CR
C1D4 CADD1C1	JZ CRPRC
C1D7 FEOA	CPI LF
C1D9 C8	RZ
C1DA C3C3C1	JMP GCLIN
	*
C1DD CDE3C0	CRPRC: CALL CLINE
C1E0 C9	RET
	*
	* FIND AND PROCESS COMMAND
C1E1 CD3DC1	COPRC: CALL CREM
C1E4 OE01	MVI C,1
C1E6 CD25C1	CALL VDAD2
C1E9 EB	XCHG
C1EA CD32C2	CALL SCHR
C1ED CA8EC3	JZ ERR1
C1FO EB	XCHG
C1F1 1194C1	LXI D,COMTAB
	*
	* SEARCH TABLE POINTED TO BY "DE"
	* FOR TWO CHAR MATCH OF
	* "HL" MEMORY. IF NO MATCH
	* THEN INSERT "?"
C1F4 1A	FDCOM: LDAX D
C1F5 B7	ORA A

C1F6	CA8FC3	JZ	ERR2
C1F9	E5	PUSH	H
C1FA	BE	CMP	M
C1FB	13	INX	D
C1FC	C20BC2	JNZ	NCOM
*			
C1FF	23	INX	H
C200	1A	LDAX	D
C201	BE	CMP	M
C202	C20BC2	JNZ	NCOM
*			
C205	C1	POP	B
C206	EB	XCHG	
C207	23	INX	H
C208	C371CO	JMP	DISPT
*			
C20B	13	NCOM:	INX D
C20C	13		INX D
C20D	13		INX D
C20E	E1		POP H
C20F	C3F4C1		JMP FDCOM
*			
* PUT CR, LF, PROMPT;			
C212	CD1AC2	PROMPT:	CALL CRLF
C215	063E		MVI B,'>'
C217	C34BC0		JMP VDMOT
*			
C21A	060A	CRLF:	MVI B,LF
C21C	CD4BC0		CALL VDMOT
C21F	060D		MVI B,CR
C221	C34BC0		JMP VDMOT
*			
* scan over up to 12 characters looking for a blank			
C224	OEOC	SBLK:	MVI C,12
C226	1A	SBLK1:	LDAX D
C227	FE20		CPI BLANK
C229	CA32C2		JZ SCHR
C22C	13		INX D
C22D	0D		DCR C
C22E	C226C2		JNZ SBLK1
C231	C9		RET
*			
* SCAN FOR NON BLANK, OVER UP TO 10 BLANKS			
C232	OEOA	SCHR:	MVI C,10
C234	1A	SCHR1:	LDAX D
C235	FE20		CPI SPACE
C237	C0		RNZ
C238	13		INX D
C239	0D		DCR C
C23A	C8		rz
C23B	C334C2		JMP SCHR1
*			
* SCAN PAST CHARS AND BLANKS FOR ADDRESS			

```

* THEN COVERTS IT TO HEX
C23E CD24C2 SCONV: CALL SBLK
C241 CA8EC3      JZ   ERR1
*
* CONVERT ASCII DIGITS TO BINARY FOLLOWING
* STANDARD HEX CONVERSION. IT STOPS WHEN A SPACE
* IS ENCOUNTERED.
* PARAMETER ERRORS REPLACE THE ERROR CHAR ON THE SCREEN
* WITH A ?.
C244 210000 SHEX: LXI H,0
C247 1A SHE1: LDAX D
C248 FE20 CPI  ' '
C24A C8 RZ
*
C24B 29 HCONV: DAD H
C24C 29 DAD H
C24D 29 DAD H
C24E 29 DAD H
C24F CD5BC2 CALL HCOV1
C252 D28EC3 JNC ERR1
C255 85 ADD L
C256 6F MOV L,A
C257 13 INX D
C258 C347C2 JMP SHE1
*
C25B D630 HCOV1: SUI '0'
C25D FE0A CPI 10
C25F D8 RC
C260 D607 SUI 7
C262 FE10 CPI 10H
C264 C9 RET
*
* SYSTEM START
* CLEAR PART OF RAM, SET STACK POINTER,
* AND FALL THROUGH TO TERMINAL MODE
C265 AF STRTA: XRA A
C266 4F MOV C,A
C267 2100C8 LXI H,SYSRAM
*
C26A 77 CLERA: MOV M,A
C26B 23 INX H
C26C 0C INR C
C26D C26AC2 JNZ CLERA
*
* TERMINAL MODE
* GET CHAR FROM KEYBOARD, AND PUT TO MONITOR
* COMMAND KEYS ARE NOT PUT TO TERMINAL,
* BUT ARE INTERPRETED AS DIRECT SOL COMMANDS
* THE MODE COMMAND PUTS SOL IN THE COMMAND MODE
C270 3100CC TERM: LXI SP,SYSTP
C273 CDEDC3 CALL TOFF
C276 CDC5C0 CALL PERSE
*

```

```

C279 CD22C0      KIN:    CALL KSTAT
C27C C290C2      JNZ     TIN
C27F DBFC         IN      KDATA
C281 47          MOV     B,A
*                 ANI     80H
*                 JZ      TOUT
C282 FE06         cpi    06   ;Command Key?
C284 C28DC2       jnz    TOUT
C287 CD4BC0       CALL   VDMOT ;Process it
C28A C390C2       JMP    TIN
*
C28D CD41C0       TOUT:   CALL SDROT
C290 CD31C0       TIN:    CALL SSTAT
C293 CA79C2       JZ     KIN
C296 DBF9         IN      SDATA
C298 E67F         ANI    7FH
C29A 47          MOV     B,A
C29B CD4BC0       CALL   VDMOT
C29E C379C2       JMP    KIN
*
*
* DUMP COMMAND
* DUMP ALL CHARS FROM MEMORY TO THE
* CURRENT OUTPUT DEVICE
* ALL VALUES ARE DISPLAYED AS HEX
*
* THE COMMAND IS AS FOLLOWS:
* DUMP ADDR1 ADDR2
* THE VALUES FROM ADDR1 TO ADDR2 ARE THEN OUTPUT TO THE
* OUTPUT DEVICE. IF ONLY ADDR1 IS GIVEN, THEN THE VALUE AT TH
C2A1 CD3EC2       DUMP:   CALL SCONV
C2A4 E5            PUSH   H
C2A5 CD32C2       CALL   SCHR
C2A8 E1            POP    H
C2A9 CAB4C2       JZ     POVER
C2AC E5            PUSH   H
C2AD CD44C2       CALL   SHEX
C2B0 D1            POP    D
C2B1 C3B6C2       JMP    NPASS
*
C2B4 54            POVER: MOV   D,H
C2B5 5D            MOV   E,L
C2B6 EB            NPASS: XCHG
*
C2B7 CD1AC2       DLOOP:  CALL  CRLF
C2BA DBFC          IN    KDATA
C2BC FE06          CPI   MODE
C2BE CAB4C1        JZ    COMND
C2C1 CDDDC2        CALL  ADOUT
C2C4 CDE5C2        CALL  BOUT
C2C7 OE10          MVI   C,16
*
C2C9 7E            DLP1:   MOV   A,M

```

C2CA	C5	PUSH B
C2CB	CDE2C2	CALL HBOUT
C2CE	CDO2C3	CALL ACOMP
C2D1	D2B4C1	JNC COMND
C2D4	C1	POP B
C2D5	23	INX H
C2D6	0D	DCR C
C2D7	C2C9C2	JNZ DLP1
C2DA	C3B7C2	JMP DLOOP
*		
* PUT HL AS 16 BIT HEX		
C2DD	7C	ADOUT: MOV A,H
C2DE	CDEAC2	CALL HeOUT
C2E1	7D	MOV A,L
*		
C2E2	CDEAC2	HOUT: CALL HEOUT
C2E5	0620	BOUT: MVI B,' '
C2E7	C34BC0	JMP VDMOT
*		
C2EA	4F	HEOUT: MOV C,A
C2EB	OFOFOFOF	RRC!RRC!RRC!RRC
C2EF	CDF3C2	CALL HEOU1
C2F2	79	MOV A,C
*		
C2F3	E60F	HEOU1: ANI OFH
C2F5	C630	ADI '0'
C2F7	FE3A	CPI '9'+1
C2F9	DAFEC2	JC OUTH
C2FC	C607	ADI 7
C2FE	47	OUTH: MOV B,A
C2FF	C34BC0	JMP VDMOT
*		
* COMPARE DE AND HL		
C302	7D	ACOMP: MOV A,L
C303	93	SUB E
C304	7C	MOV A,H
C305	9A	SBB D
C306	C9	RET
*		
* ENTER COMMAND		
* GET VALUES FROM KEYBOARD AND PUT INTO MEMORY		
* THE INPUT VALUES ARE SCANNED FOLLOWING,		
* A STANDARD 'GOLIN' INPUT SO ON SCREEN EDITING MAY TAKE		
* PLACE PRIOR TO THE LINE TERMINATOR. A SLASH, '/',		
* ENDS THE ENTER COMMAND AND RETURNS CONTROL TO		
* THE COMMAND MODE		
C307	CD3EC2	ENTER: CALL SCONV
C30A	E5	PUSH H
*		
C30B	CD1AC2	ENLOP: CALL CRLF
C30E	063A	MVI B,':'
C310	CDCCC1	CALL CONT
C313	CD3DC1	CALL CREM

```

C316 0E01          MVI   C,1
C318 CD25C1        CALL  VDAD2
C31B EB            XCHG

*
C31C 0E03          ENL01: MVI   C,3
C31E CD34C2        CALL  SCHR1
C321 CA0BC3        JZ    ENLOP

*
C324 1A            ENL02: LDAX  D
C325 FE2F          CPI   '/'
C327 CAB4C1        JZ    COMND
C32A CD44C2        CALL  SHEX
C32D 7D            MOV   A,L
C32E E1            POP   H
C32F 77            MOV   M,A
C330 23            INX   H
C331 E5            PUSH  H
C332 C31CC3        JMP   ENL01

*
* EXECUTE COMMAND
* GET THE FOLLOWING PARAMETER AND JUMP TO THE LOCATION.
* IF PROPER STACK OPERATIONS ARE USED WITHIN THE EXTERNAL
* PROGRAM,
* IT CAN DO A STANDARD 'RET' TO THE CONSOL COMMAND MODE
C335 CD3EC2        EXEC:  CALL  SCONV
C338 E9            PCHL

*
* TAPE LOAD COMMAND
* READ FROM EITHER TAPE UNIT, PLACING THE DATA INTO MEMORY.
* WHILE SPACE WITHIN CONSOL DOES NOT ALLOW FOR 'STANDARD'
* TAPE ROUTINES, THIS COMMAND WILL LOAD SOL-BAS
C339 CD24C2        TLOAD: CALL  SBLK
C33C CA4AC3        JZ    DFLT
C33F CD44C2        CALL  SHEX
C342 7D            MOV   A,L
C343 E601          ANI   1
C345 3E20          MVI   A,' '
C347 C24BC3        JNZ   SETSP

*
C34A AF            DFLT:  XRA   A
C34B F6C0          SETSP: ORI   TAPE1+TAPE2
C34D D3FA          OUT   STAPT
C34F CDF1C3        CALL  DELAY
C352 DBFB          IN    TDATA

*
C354 CD94C3        TL001: CALL  RHEAD
C357 C254C3        JNZ   TL001

*
C35A 2AOCC8        LHLD  BLOCK
C35D EB            XCHG
C35E 2AOEC8        LHLD  LOADR

*
C361 7A            LOLOOP: MOV   A,D

```

```

C362 B3          ORA   E
C363 CAEDC3      JZ    TOFF
C366 0100FF      LXI   B,-256
C369 EB          XCHG
C36A 09          DAD   B
C36B D2E1C3      JNC   LBLK
C36E 0600      MVI   B,0
*
C370 0E00      RDBLK: MVI   C,0
C372 EB          XCHG
*
C373 CDC6C3      RTBYT: CALL  TAPIN
C376 77          MOV   M,A
C377 23          INX   H
C378 A9          XRA   C
C379 2F          CMA
C37A 91          SUB   C
C37B 4F          MOV   C,A
C37C 05          DCR   B
C37D C273C3      JNZ   RTBYT
*
C380 CDC1C3      CALL  CRCCK
C383 CA61C3      JZ    LOLOOP
*
C386 0607      TERR:  MVI   B,'G'-40H
C388 CD4BC0      CALL  VDMOT
C38B C3B4C1      JMP   COMND
*
*   CONSOL ERROR HANDLER
C38E EB          ERR1:  XCHG
C38F 363F      ERR2:  MVI   M,'?'
C391 C3B4C1      JMP   COMND
*
*   READ THE HEADER
C394 060A      RHEAD: MVI   B,10
C396 DBFA      RHEA1: IN    STAPT
C398 E640      ANI   TDR
C39A CA96C3      JZ    RHEA1
C39D DBFB      IN    TDATA
C39F B7          ORA   A
C3AO C294C3      JNZ   RHEAD
C3A3 05          DCR   B
C3A4 C296C3      JNZ   RHEA1
*
*   WAIT FOR START CHAR
C3A7 CDC6C3      SOHL:  CALL  TAPIN
C3AA 3D          DCR   A
C3AB C2A7C3      JNZ   SOHL
*
*   NOW GET THE HEADER
C3AE 2105C8      LXI   H,THEAD
C3B1 010010      LXI   B,HLEN*256
*

```

C3B4 C3C6C3	RHED1:	CALL TAPIN
C3B7 77		MOV M,A
C3B8 23		INX H
C3B9 A9		XRA C
C3BA 2F		CMA
C3BB 91		SUB C
C3BC 4F		MOV C,A
C3BD 05		DCR B
C3BE C2B4C3		JNZ RHED1
	*	
	*	GET THE NEXY BYTE AND COMPARE IT
	*	TO THE VALUE IN REGISTER C.
	*	THE FLAGS ARE SET ON RETURN
C3C1 C3C6C3	CRCCK:	CALL TAPIN
C3C4 B9		CMP C
C3C5 C9		RET
	*	
	*	GET THE NEXT AVAILABLE BYTE FROM THE TAPE.
	*	WHILE WAITING, TEST THE KEYBOARD FOR A "MODE" COMMAND
	*	IF RECEIVED, STOP THE TAPE LOAD AND RETURN TO THE
	*	COMMAND MODE
C3C6 DBFA	TAPIN:	IN STAPT
C3C8 E640		ANI TDR
C3CA C2D7C3		JNZ TREDY
C3CD DBFC		IN KDATA
C3CF FE06		CPI MODE
C3D1 CAB4C1		JZ comnd
C3D4 C3C6C3		JMP TAPIN
	*	
C3D7 DBFA	TREDY:	IN STAPT
C3D9 E618		ANI TFE+TOE
C3DB C286C3		JNZ TERR
C3DE DBFB		IN TDATA
C3EO C9		RET
	*	
	*	CALCULATE THE LENGTH OF THE LAST BLOCK
C3E1 01FFFF	LBLK:	LXI B,-1
C3E4 09		DAD B
C3E5 23		INX H
C3E6 45		MOV B,L
C3E7 210000		LXI H,0
C3EA C370C3		JMP RDBLK
	*	
	*	TURN THE TAPES OFF
C3ED AF	TOFF:	XRA A
C3EE D3FA		OUT STAPT
C3FO C9		RET
	*	
C3F1 110000	DELAY:	LXI D,0
C3F4 1B	DLOP1:	DCX D
C3F5 7A		MOV A,D
C3F6 B3		ORA E
C3F7 C2F4C3		JNZ DLOP1

C3FA C9

RET

\*  
\* SOL SYSTEM EQUATES  
\* DISPLAY PARAMETERS  
CCOO = VDMEM EQU OCCOOH  
\*  
\* KEYBOARD SPECIAL KEYS  
009A = DOWN EQU 9AH  
0097 = UP EQU 97H  
0081 = LEFT EQU 81H  
0093 = RIGHT EQU 93H  
008C = LOADK EQU 8CH  
0006 = MODE EQU 06H  
0004 = CLEAR EQU 04H  
0002 = HOME EQU 02H  
0008 = BACKS EQU 08H  
000A = LF EQU 10  
000D = CR EQU 13  
0020 = BLANK EQU ' '  
0020 = SPACE EQU BLANK  
0030 = CX EQU 'X'-40  
\*  
\* PORT ASSIGNMENTS  
00FA = STAPT EQU OFAH  
00F8 = SERST EQU Of8H  
00F9 = SDATA EQU OF9H  
00FB = TDATA EQU OFBH  
00FC = KDATA EQU OfCH  
00FD = PDATA EQU OFDH  
00FE = DSTAT EQU OFEH  
00FF = SENSE EQU OFFH  
\*  
\* BIT ASSIGNMENT MASKS  
0001 = SCD EQU 1  
0002 = SDSK EQU 2  
0004 = SPE EQU 4  
0008 = SFE EQU 8  
0010 = SOE EQU 16  
0020 = SCTS EQU 32  
0040 = SDR EQU 64  
0080 = STBE EQU 128  
\*  
0001 = KDR EQU 1  
0002 = PDR EQU 2  
0004 = PXDR EQU 4  
0008 = TFE EQU 8  
0010 = TOE EQU 16  
0040 = TDR EQU 64  
0080 = TTBE EQU 128  
\*  
0001 = SOK EQU 1  
\*  
0040 = TAPE1 EQU 64

```
0080 =      TAPE2    EQU 128
*
*   SOL SYSTEM GLOBAL AREA
C800          ORG  OC800H
*
C800 =      SYSRAM   EQU  $
CC00 =      SYSTP    EQU  $+1024
*
*   CONSOL PARAMETER AREA
C800          NCHAR    DS 1
C801          LINE     DS 1
C802          BOT      DS 1
C803          OPORT    DS 1
C804          IPORT    DS 1
*
C805          THEAD    DS 5
C80A          DS 1
C80B          HTYPE    DS 1
C80C          BLOCK    DS 2
C80E          LOADR   DS 2
C810          XEQAD   DS 2
C812          ASPR    DS 3
*
0010 =      HLEN     equ $-THEAD
*
*
```