GCPZ

CPZ-48000 SBC Z80 Computer Emulator.

4th December 2020

This Page is left Blank

TABLE OF CONTENTS

0.TABLE OF CONTENTS

1. INTRODUCTORY INFORMATION

2. OBTAINING AND BUILDING GCPZ

3. The ‘OPTIONS’ MENU

4. GCPZ DEVELOPMENT ASSISTANCE

5. HELPER PROGRAMS

6. VARIOUS THINGS

GCPZ APPENDICES

IMAGES AND PRINTOUTS

1. INTRODUCTORY INFORMATION

The Inter Continental Microsystems' CPZ-48000 single-board computer from around 1982 was quite advanced for its day. It was designed as a base system with 'personality boards' (daughter-circuitboards) which took over the interfaces between different peripherals, such as serial printers, parallel printers, 5" floppy drives, 8" floppy drives, etc. The CPZ emulator can deal with most of the everyday operation of the CPZ-48000 and runs up to 4 eight-inch floppy drives. These drives simulate an 8" double-sided double-density floppy. On track 0, side 0, the sectors are formatted to be 26 sectors of 128 byes, all the rest of the tracks on both sides are formatted to 16 sectors of 512 bytes. This gives a total of about 1.2megs per disk.

GCPZ uses disk-image files which may contain Intercontinental Micro Systems Operating Systems of the period: mainly CP/M. Other IMS Operating Systems may work, but haven't been tested.

The CPZ-48000 SBC system used one or more floppy disks in either 5-inch or 8-inch. GCPZ can access four 8-inch CP/M virtual disk drives.

GCPZ is constructed from two modules. The first module, the GXE Z80 Emulator Toolkit, contains the 64K of RAM, a Z80 microprocessor emulator, and a display screen. Drop-down menus emulate the operator's interaction with the hardware, such as inserting or removing floppy disks, and organizing the interaction between the host linux machine and the virtual Z80 machine. This first module is installed as a Graphical User Interface (GUI) and can be called by GCPZ or any other Z80-based emulator software, such as GMD128 (Morrow MDxx Series emulator) or NSE (North Star HORIZON emulator).

The second module consists of the CPZ-48000 system-specific components, such as the data and control ports of the serial and parallel I/O, the 1793 DMA floppy-disk controllers, with the associated boot PROMS containing a minimal Monitor program.

1.2 ATTRIBUTIONS FOR OTHERS' CODE in GCPZ

GCPZ's Z80 emulation code pretty much comes from yaze, a CP/M emulator written by Frank Cringle. Morrow-specific code amendments such as the floppy-disk I/O, and a few other additions such as Mode 2 interrupt code were made by Jack Strangio.

GCPZ's Z80 disassembly code comes from Marat Fayzullin's 1999 DAsm code with some local alterations.

The rest of GCPZ cannot be blamed on anyone else but myself.
Jack Strangio, November 2020

1.3 THANKS

I have only the greatest appreciation for all those who have helped me in my rather idiosyncratic quest to write emulators of the North Star Horizon, and several other ‘home computers’ as they were called in the late 1970s and early 1980s.

The North Star Horizon was my first computer which took more than 40 hours to build over the course of several weeks in late 1978. The thousands of solder-joints literally burned-out a new soldering iron. It says a lot for the quality of the instruction manual that most of the time I really had no idea what each step did but at the end (once my half-dozen wiring mistakes were fixed) I had assembled a computer which worked perfectly.

Often, just a little bug in the emulators would get people off writing to me and setting off another round of debugging and coding. Thanks to you all.

1.4 FLOPPY DISKS SUPPLIED WITH GCPZ

Several floppy disks are supplied with GCPZ to get you up and going quickly. They are stored in the 'disks' subdirectory. These archive disks have been renamed to allow their uses to be self-explanatory.

cpzboot.fd8

8-inch boot disk. The associated CP/M BIOS is configured for four 8-inch double-sided, double-density floppies A:, B:, C:, and D:.

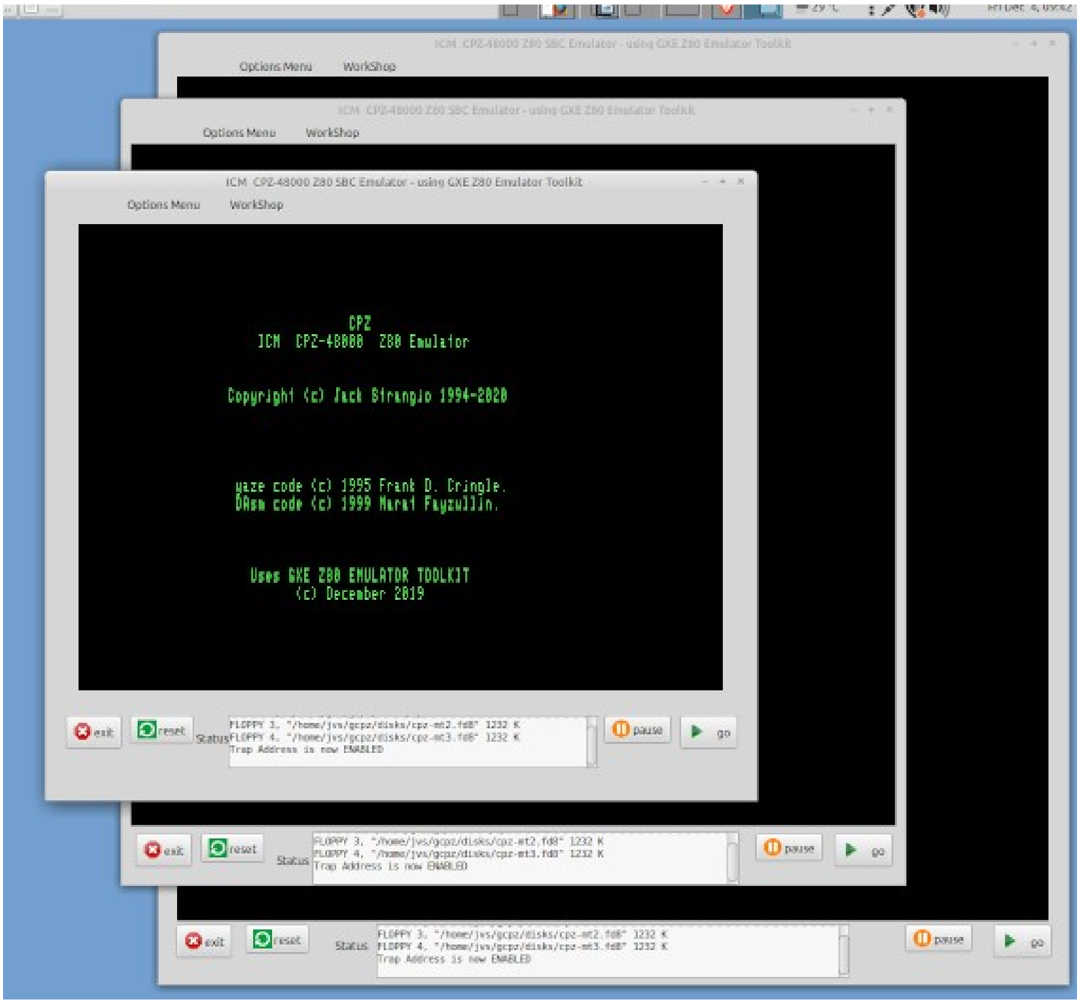
cpz-mt1.fd8
cpz-mt2.fd8
cpz-mt3.fd8

8-inch blank floppy disks.

1.5 SCREENSHOTS

(Note: Most of the screen images included in this User Guide do not render well. They look better when displayed dot-for-dot as screenshots. Find them in the ‘screenshots’ directory of the downloaded tarball.)

GCPZ is GTK+ based. When GCPZ starts up it will look at the screen-resolution and display a ‘terminal’ of a size that is suitable for that resolution. The ‘large’ terminal (110 chars wide, 45 lines) will fit on a 1920x1080 screen. The ‘medium’ terminal (96 chars wide, 36 lines) will fit on an intermediate resolution laptop. Then there is the ‘small’ terminal which is actually the historical standard-sized serial terminal of 80 chars wide and 25 lines high. The ‘small’ display (80 x 24) is the default display. Both the medium and large terminal displays can be specified with a command-line option. I generally use the ‘large’ display option.


ICM CPZ-48000 Z80 SBC Emulator - using GXE Z80 Emulator Toolkit
Options Menu
WorkShop
exit
reset
Status
pause
go

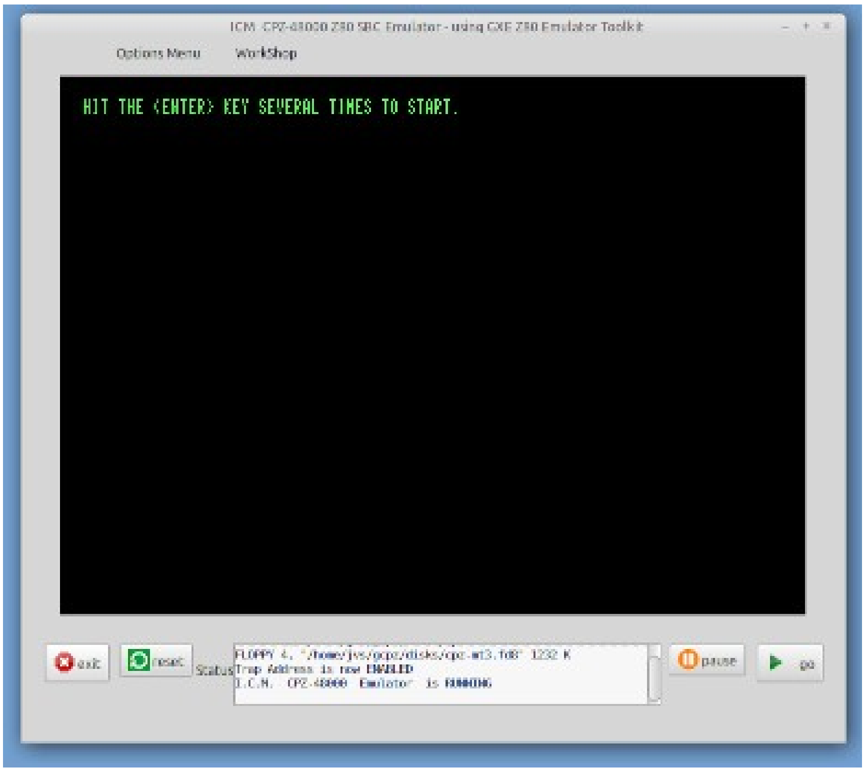
```
CPZ
ICM  CPZ-48000  Z80 Emulator

Copyright (c) Jack Strangio 1994-2020

yaze code (c) 1995 Frank D. Cringle.
DAsm code (c) 1999 Marat Fayzullin.

Uses GXE Z80 EMULATOR TOOLKIT
(c) December 2019
```

```
FLOPPY 3. "/home/jvs/gcpz/disks/cpz-mt2.fd8" 1232 K
FLOPPY 4. "/home/jvs/gcpz/disks/cpz-mt3.fd8" 1232 K
Trap Address is now ENABLED
```

```
FLOPPY 3. "/home/jvs/gcpz/disks/cpz-mt2.fd8" 1232 K
FLOPPY 4. "/home/jvs/gcpz/disks/cpz-mt3.fd8" 1232 K
Trap Address is now ENABLED
```

```
FLOPPY 3. "/home/jvs/gcpz/disks/cpz-mt2.fd8" 1232 K
FLOPPY 4. "/home/jvs/gcpz/disks/cpz-mt3.fd8" 1232 K
Trap Address is now ENABLED
```

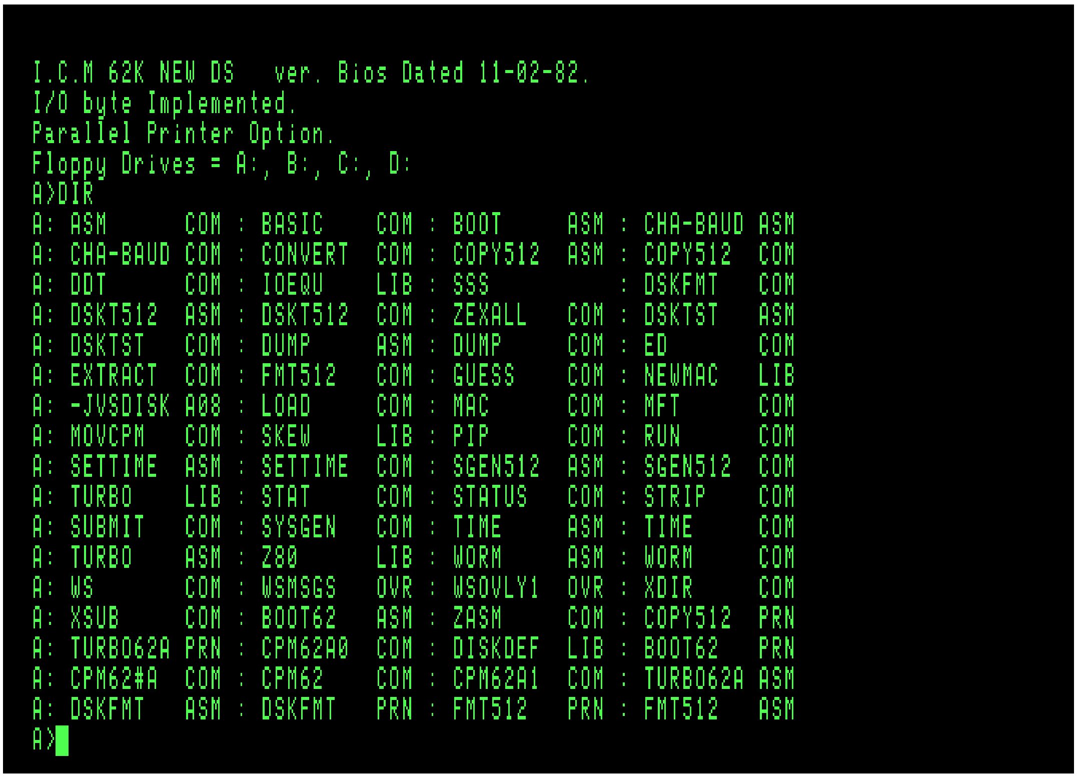
Fig 1. Three GCPZ startup screens, running on a 1920x1080 resolution screen in 110x45 display, 96x36 display, and in 80x25 character screen formats.


ICM CPZ-48000 Z80 SBC Emulator - using GXE Z80 Emulator Toolkit
Options Menu
WorkShop
exit
reset
Status
pause
go

```
HIT THE <ENTER> KEY SEVERAL TIMES TO START.
```

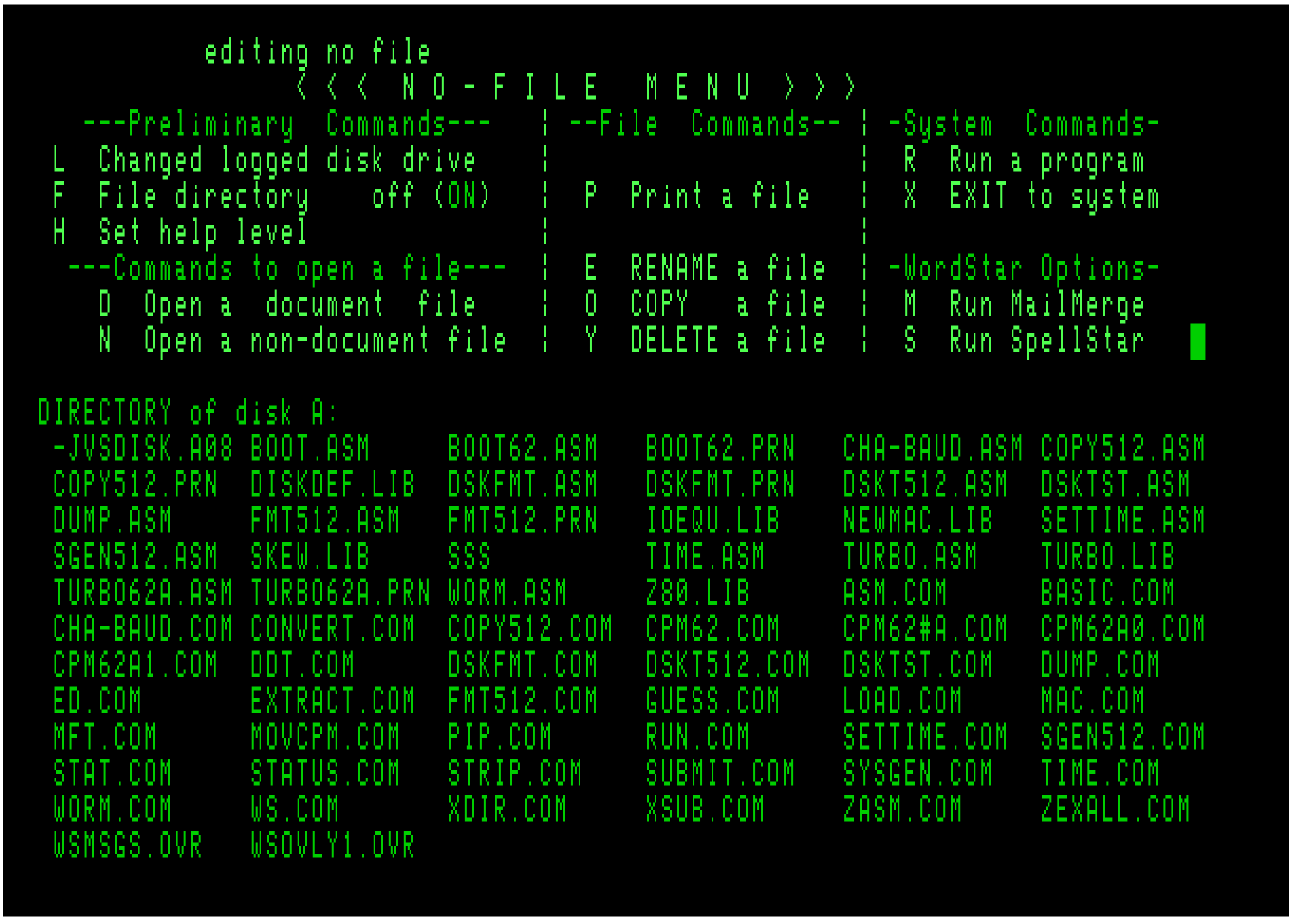
```
FLOPPY 4. "/home/jvs/gcpz/disks/cpz-mt3.fd8" 1232 K
Trap Address is now ENABLED
I.C.M.  CPZ-48000  Emulator  is RUNNING
```

Fig 2.   GCPZ startup screen. Hit <ENTER> several times to boot CP/M



```
I.C.M 62K NEW DS   ver. Bios Dated 11-02-82.
I/O byte Implemented.
Parallel Printer Option.
Floppy Drives = A:, B:, C:, D:
A>DIR
A: ASM      COM : BASIC    COM : BOOT     ASM : CHA-BAUD ASM
A: CHA-BAUD COM : CONVERT  COM : COPY512  ASM : COPY512  COM
A: DDT      COM : IOEQU    LIB : SSS          : DSKFMT   COM
A: DSKT512  ASM : DSKT512  COM : ZEXALL   COM : DSKTST   ASM
A: DSKTST   COM : DUMP     ASM : DUMP     COM : ED       COM
A: EXTRACT  COM : FMT512   COM : GUESS    COM : NEWMAC   LIB
A: -JVSDISK A08 : LOAD     COM : MAC      COM : MFT      COM
A: MOVCPM   COM : SKEW     LIB : PIP      COM : RUN      COM
A: SETTIME  ASM : SETTIME  COM : SGEN512  ASM : SGEN512  COM
A: TURBO    LIB : STAT     COM : STATUS   COM : STRIP    COM
A: SUBMIT   COM : SYSGEN   COM : TIME     ASM : TIME     COM
A: TURBO    ASM : Z80      LIB : WORM     ASM : WORM     COM
A: WS       COM : WSMSGS   OVR : WSOVLY1  OVR : XDIR     COM
A: XSUB     COM : BOOT62   ASM : ZASM     COM : COPY512  PRN
A: TURBO62A PRN : CPM62A0  COM : DISKDEF  LIB : BOOT62   PRN
A: CPM62#A  COM : CPM62    COM : CPM62A1  COM : TURBO62A ASM
A: DSKFMT   ASM : DSKFMT   PRN : FMT512   PRN : FMT512   ASM
A>
```

Fig 3. GCPZ splash screen followed by CP/M directory listing. GCPZ looks like a typical "green-screen" terminal of the 70's-80's period, in particular it will default to be a terminal which acts very similar to Televideo 920/ Soroc 120/ ADM3A terminals.



```
          editing no file
               < < <  N O - F I L E   M E N U  > > >
   ---Preliminary  Commands---   | --File  Commands-- | -System  Commands-
 L  Changed logged disk drive    |                    |  R  Run a program
 F  File directory   off (ON)    |  P  Print a file   |  X  EXIT to system
 H  Set help level               |                    |
  ---Commands to open a file---  |  E  RENAME a file  | -WordStar Options-
    D  Open a  document  file    |  O  COPY   a file  |  M  Run MailMerge
    N  Open a non-document file  |  Y  DELETE a file  |  S  Run SpellStar

DIRECTORY of disk A:
 -JVSDISK.A08 BOOT.ASM     BOOT62.ASM   BOOT62.PRN   CHA-BAUD.ASM COPY512.ASM
 COPY512.PRN  DISKDEF.LIB  DSKFMT.ASM   DSKFMT.PRN   DSKT512.ASM  DSKTST.ASM
 DUMP.ASM     FMT512.ASM   FMT512.PRN   IOEQU.LIB    NEWMAC.LIB   SETTIME.ASM
 SGEN512.ASM  SKEW.LIB     SSS          TIME.ASM     TURBO.ASM    TURBO.LIB
 TURBO62A.ASM TURBO62A.PRN WORM.ASM     Z80.LIB      ASM.COM      BASIC.COM
 CHA-BAUD.COM CONVERT.COM  COPY512.COM  CPM62.COM    CPM62#A.COM  CPM62A0.COM
 CPM62A1.COM  DDT.COM      DSKFMT.COM   DSKT512.COM  DSKTST.COM   DUMP.COM
 ED.COM       EXTRACT.COM  FMT512.COM   GUESS.COM    LOAD.COM     MAC.COM
 MFT.COM      MOVCPM.COM   PIP.COM      RUN.COM      SETTIME.COM  SGEN512.COM
 STAT.COM     STATUS.COM   STRIP.COM    SUBMIT.COM   SYSGEN.COM   TIME.COM
 WORM.COM     WS.COM       XDIR.COM     XSUB.COM     ZASM.COM     ZEXALL.COM
 WSMSGS.OVR   WSOVLY1.OVR
```

Fig 4. Standard version of Word Star running in default screen format (80 chars, 24 lines)

1.6 GCPZ COMMAND-LINE START-UP OPTIONS

gcpz [-c config-file] [-m or -l]

-c <config-file>

Use an alternate config-file instead of the default '/home/username/gcpz/gcpz.conf' file. The alternate file should also be placed in the '/home/username/gcpz' directory.

-m

Specify the use of a display 96 characters wide by 36 lines.

-l

Specify the use of a display 110 characters wide by 45 lines.

1.7 GCPZ COMMAND-LINE STARTUP EXAMPLES:

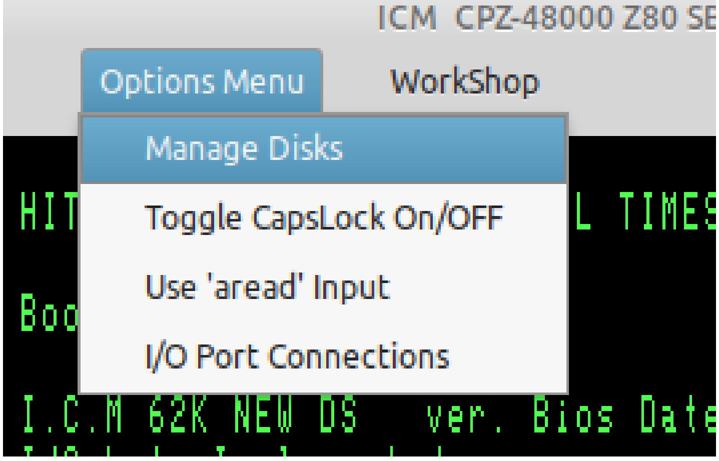
gcpz -c gcpz.zzz

Start GCPZ using the floppy or hard disk controller, booting from the disk-image file which is specified in the '/home/username/gcpz/gcpz.zzz' configuration file.

gcpz

Start GCPZ using the floppy or hard disk controller, booting from the disk-image specified in the default configuration file.

NOTE: The default boot device is the CP/M A: drive. To boot the with a different floppy disk, open the Options Menu, and startup the 'Manage Disks' menu item. Using the file-selection dialogs, select your new boot floppy, then 'Exit' and restart the emulator. Alternatively, hit the 'Reset' button.



```
HIT                    L TIMES
Boo
I.C.M 62K NEW DS   ver. Bios Date
```

Fig. 5. Options Menu: 'Manage Disks'

1.8 GETTING THE EMULATOR’S START-UP CONFIGURATION

1.8 THE WORK DIRECTORY

All of the emulators do their work in a directory which has the general look of

```
/home/user_name/emulator_name/
```

thus if user ‘fred’ is working with the ‘gcpz’ emulator, the gcpz work-directory is installed at

```
/home⁄fred/gcpz
```

In this work-directory will be found any logfiles, such as the **screenlog** which contains a record of all the output that was displayed by the screen during the emulator’s activity. If any debug logging was required there will be a debug log called **xlog** written into the work-directory.

Also found in the work-directory are any configuration files which specify which floppy and hard disk images will be used while the emulator is working.

Subdirectories in the work directory are **disks**, **documentation**, and **info**.

The **disks** subdirectory is where floppy and hard disk images may be found. It is a good idea to put any other disk-images in there also. That is the first place that the emulator will usually look for disks.

The **documentation** subdirectory is where official Manufacturer and CP/M documentation is found.

The **info** subdirectory is where other useful information may be placed.

1.9 CONFIGURATION FILES:

A user's default configuration file is found at

```
⁄home/username/work_directory/emulator_name.conf
```

thus user **fred** will find his *default* **gcpz** configuration file at

```
/home/fred/gcpz/gcpz.conf
```

Bear in mind, though, that any other configuration filename can be specified on the command-line by using the **-c** option, as in

```
/home⁄fred/.local/bin/gcpz -c gcpz-001.conf
```

or even just simply

```
gcpz -c gcpz-001.conf
```

if gcpz is located in one of your $PATH directories, and gcpz-001.conf is located in the work-directory.

1.10 USER CONFIGURATION FILES: gcpz.conf

This is an actual configuration file.

```
###
### Configuration File for GXE-Based Emulators (c) 201204
###
###             Avoid Editing This File Manually.
###
### Any Changes You Make Are Liable To Be Overwritten at Any Time.
###

fd1           /home/jvs/gcpz/disks/cpzboot.fd8
fd2           /home/jvs/gcpz/disks/cpz-mt1.fd8
fd3           /home/jvs/gcpz/disks/cpz-mt2.fd8
fd4           /home/jvs/gcpz/disks/cpz-mt3.fd8
disk_dir      /home/jvs/gcpz/disks/
hd_delay      off
capslock      on
=======
log           /home/jvs/gcpz/xlog
screenlog     /home/jvs/gcpz/screenlog
debug_level   FFFF
break_addr    0000
break_on      off
trap_addr     0100
trap_on       off
```

Note that there is provision for 4 floppies. This config file specifies all four floppy drives. The '.fd8' extension in the floppy filenames signifies that these are 8-inch disks holding about 1.2 megabytes of data.

The **disk_dir** indicates where the last disk used was located, and where the emulator will look first for any other disks that are wanted.

Capslock is as described. Most people using CP/M will want the capslock on, but won't want it 'on' for their host machine. See section XX to vary the condition.

All the items under the **=======** separator are normally used only during the development of the emulator itself and so will rarely be used (if ever) by most **gcpz** emulator users.

NOTE: While it is possible to edit the configuration file manually, your changes will be overwritten when any of the 'Options Menu' or 'WorkShop' menu items are used.

2. Obtaining and Building ‘GCPZ’

2.1 Linux Libraries required

Very few Linux libraries are required, apart from the standard packages installed on most Linux Distros.

The GUI Toolkit used is GTK+ Version 3, apart from a few deprecated functions from GTK+ Version 2.

This Toolkit can be installed using your Package Manager. If you’re using one of the Debian derivatives such as Debian itself, Mint, or Ubuntu, this can be done by installing **libgtk-3-dev** and **libglib2.0-dev** using Synaptic or even just

```
sudo apt install libgtk-3-dev  libglib2.0-dev
```

from the command line.

2.2 Get the source files

Download the GCPZ source code from https://itelsoft.com.au/code/gcpz_gtk_latest.tar.gz and move it to any convenient work directory. Untar and decompress the tarfile:

```
tar xvfz gcpz_latest.tar.gz
```

This will produce a subdirectory called gcpz. Move there.

```
cd gcpz
```

Compilation should be initiated with a simple **make** on the command-line**.**

If all goes well and the compile completes successfully, install the gcpz package with

```
make install
```

This will install the package in the **/home/*username*/gcpz** work directory. So user ‘fred’ will find a directory called **/home/fred/gcpz**.

A launcher icon will appear on the username’s Desktop. Clicking on that should launch the emulator. It can be ‘Drag n Dropped’ to the Desktop Panel with most Desktops. Alternatively, GCPZ can be invoked from the command-line if the **gcpz** executable file is to be found somewhere within your $PATH list. The default location for the **gcpz** program file is **/home/username/.local/bin**

2.3 What’s in the /home/username/gcpz work-directory?

The /home/username/gcpz directory has several important files:

gcpz.conf the default configuration file for GCPZ which holds most of your personal preferences and:

designates which CP/M disk image-files are mounted.
specifies what I/O files will be attached to the machine’s I/O ports.
preferred settings for capslock.
preferred gcpz-development settings.

Avoid editing the gcpz.conf file manually. It gets updated automatically every time you make different choices on the Menu Options and WorkShop menus, and will hold those settings indefinitely over more than one session.

pio_out destination of text from the parallel-out port: the ‘LST:’ device in CP/M

sio_out destination of text from the serial-out port: the ‘PUN:’ device in CP/M

2.4 Starting up GCPZ

Starting GCPZ can be done from the Desktop with one of the emulator icons or from the command-line. On start-up, the program will show the title (splash screen) and will then wait for user input. Usually, the user will then just hit the ‘go’ button because the installation process also provides the default configuration file, **gcpz.conf**, which will be found in the GCPZ work directory, **/home/username/gcpz**

gcpz.conf contains the default settings which are expected by the Morrow Micro Decision 128 computer:

A boot floppy *must* be in ‘floppy drive’ 1 at the minimum when booting from the floppy.

Several GCPZ settings are also stored in the gcpz.conf file. Such as Capslock ON/OFF. The **gcpz.conf** file should not be edited manually. While that can actually be done, any changes you make may not be permanent.

If for some reason, the default configuration is not present in the top directory, then a new configuration file needs to be made. This is simply done by providing the user’s settings with the ‘Options Menu’, and/or the WorkShop menu. See Section 3, page 18. Any time a setting is altered with these two menus, the new setting is saved automatically into the **/home/username/gcpz/gcpz.conf** file.

‘Options Menu’: Things to be changed by the everyday user.

‘WorkShop’ menu: Settings for use during GCPZ development. Most users won’t need to bother with these.

2.5 Running CP/M. The ‘go’ button.

Now hit the ‘go’ button. The screen will clear, followed almost immediately by the GCPZ blank (almost) startup screen which just has the command

```
HIT THE <ENTER> SEVERAL TIMES TO START.
```

After you hit the <ENTER> key twice (usually), the emulator will boot up and display the CPZ-48000 CP/M banner:

```
I.C.M 62K NEW DS   ver. Bios Dated 11-02-82.
I/O byte Implemented.
Parallel Printer Option.
Floppy Drives = A:, B:, C:, D:@
A>
```

indicating that CP/M has been loaded from the floppy disk, with the ‘A> ‘ prompt indicating that it is waiting for a normal CP/M command-line as user input.

If there is no boot-floppy in Drive A: on startup, you will find the banner for the boot-PROM Monitor:

```
I.C.M. System Monitor Ver. 2.1

8" Floppy Ver. Dated 08-03-82

Cmnd:
```

Use the Monitor as desired. To boot into CP/M from the Monitor, use the ‘Menu Options’ menu item ‘Manage Disks’ to insert a boot-floppy into the A: Drive, and then enter ‘B’ at the monitor prompt:

```
Cmnd: B
```

CP/M will then load normally.

2.6. Pausing the Emulator. The ‘pause’ button.

In most cases you won’t usually need to use the pause button unless things happen to move too fast for you, for instance to change floppies before the software moves on. Otherwise, using the Emulator is just like using a normal computer.

2.7 Rebooting/Resetting the Computer. The ‘reset’ button.

Just like the real thing, a reset will wipe the screen and reboot the emulator from scratch. Use this button sparingly, your work may be lost.

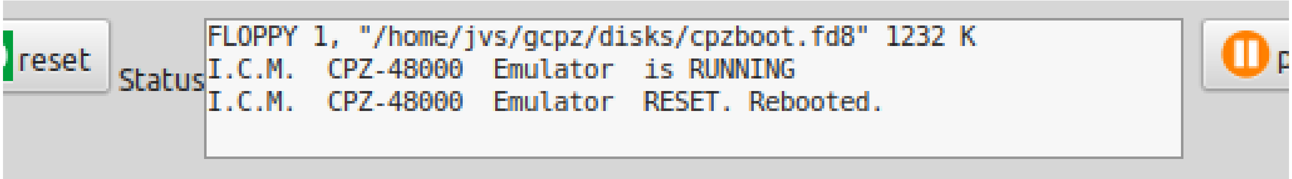
2.8. Finishing Up. The ‘exit’ button.

Pack it up and put it away. The GCPZ program closes down and the emulator window is closed. Settings in gcpz.conf will remain for next session.

2.9 The ‘Status’ window

In between the two pairs of buttons, left and right, is a small window which displays short one-line messages. This is used to show information or warnings regarding the progress of the emulator. A short beep may be heard when some messages are shown. Examples:

I.C.M CPZ-48000 Emulator is RUNNING
Capslock is now ON
FLOPPY 1 “/home/fred/gcpz/disks/cpzboot.fd8” 1232 K
I.C.M. CPZ-48000 Emulator   RESET. Rebooted.



```
FLOPPY 1, "/home/jvs/gcpz/disks/cpzboot.fd8" 1232 K
I.C.M.  CPZ-48000  Emulator  is RUNNING
I.C.M.  CPZ-48000  Emulator  RESET. Rebooted.
```

Fig 6. The Status Window

3.0 The Options Menu

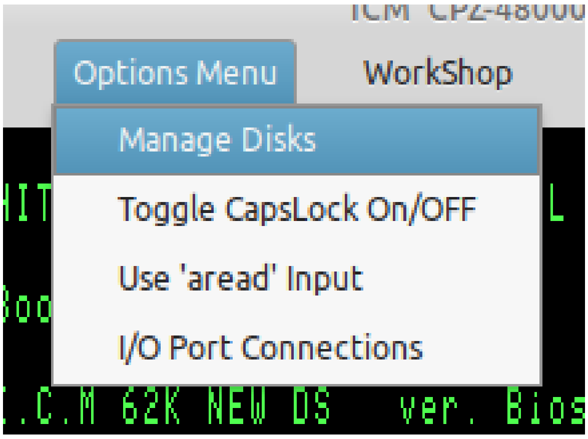


Fig. 7. The Options Menu

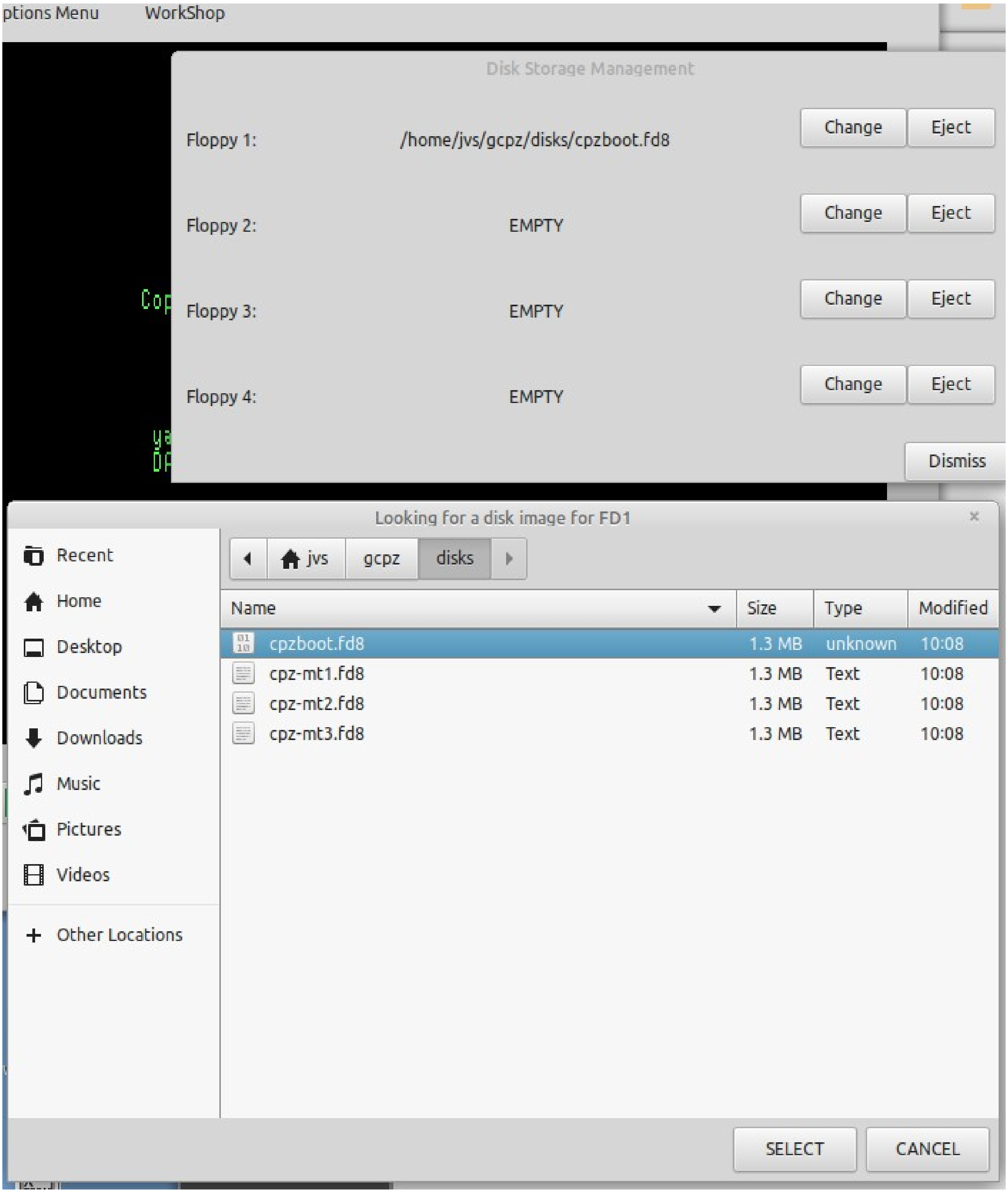
3.1 Disk Management.

The Disk Management menu item allows the user to ‘eject’ floppies and hard drives from the Emulator. The first displayed window shows what floppy disks happen to be ‘inserted’ in Floppy 1 or in Floppy 2. It also shows which hard drives were installed when the Emulator was booted.

Each disk-drive has two buttons: a ‘Change’ button which will install a different floppy-image or hard-disk image. And an ‘Eject’ button which removes any image-file which was previously installed.

If the ‘Change’ button is hit, a file-chooser dialog window opens and allows the user to browse through the whole file-system looking for a floppy-disk image to install. Once the file is selected, hit the ‘Select’ button to confirm your choice. The file-chooser window will close, the floppy-image is ‘inserted’ into the selected floppy-drive and is then ready for use.

The directory which the floppy-image came from will be used as the default disk directory in future disk-image searches. For this reason it is handy to store all your emulator floppy and hard-drive image-files in one or two directories.



Disk Storage Management

Floppy 1: /home/jvs/gcpz/disks/cpzboot.fd8

Floppy 2: EMPTY

Floppy 3: EMPTY

Floppy 4: EMPTY

Looking for a disk image for FD1

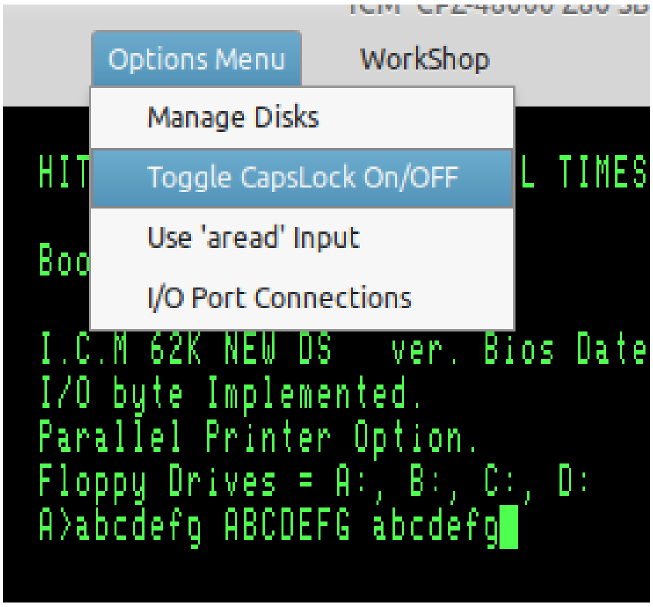
Name	Size	Type	Modified
cpzboot.fd8	1.3 MB	unknown	10:08
cpz-mt1.fd8	1.3 MB	Text	10:08
cpz-mt2.fd8	1.3 MB	Text	10:08
cpz-mt3.fd8	1.3 MB	Text	10:08

Fig. 8. Disk Selection Pop-Ups

3.2 Toggle Capslock ON and OFF

Many of the older Operating Systems will not recognise the use of lower-case characters. While one can use the actual Caps Lock Key to turn on the CapsLock, it would also turn on upper-case for the host Operating System as well. This can be a nuisance.

North Star DOS only understands uppercase commands, so it’s necessary to toggle Capslock ON when using DOS. CP/M automatically converts command-line lowercase to uppercase anyway, so the Capslock setting can be set to personal preference.



```
HIT                                  L TIMES

Boo

I.C.M 62K NEW DS   ver. Bios Date
I/O byte Implemented.
Parallel Printer Option.
Floppy Drives = A:, B:, C:, D:
A>abcdefg ABCDEFG abcdefg
```

Fig 9.   Capslock Toggled ON/OFF

In the screenshot above, Capslock starts out as being OFF, showing the ‘abcdefg’ in lowercase. Then Capslock is toggled to OFF, and shown in the Status Window, with the next set of characters being uppercase ‘ABCDEFG’. The Capslock is then toggled back OFF, again being shown in the Status Window, and the final ‘abcdefg’ is again lowercase

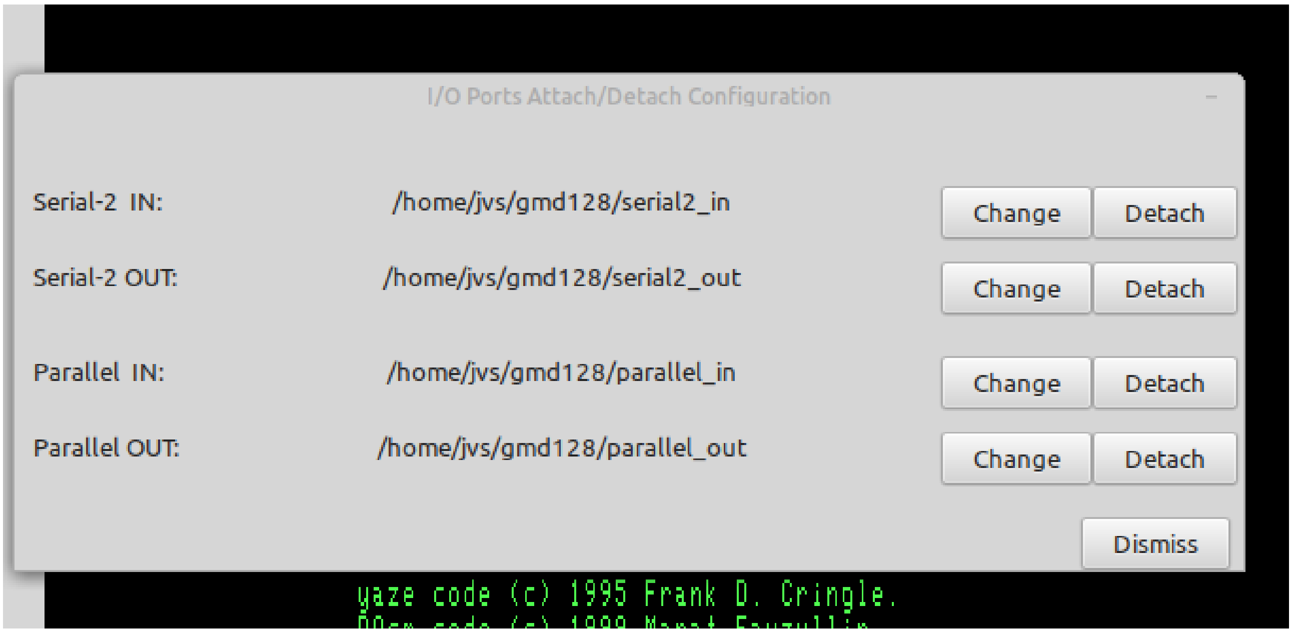
3.3 Use ‘aread’ Input.

Read in an ASCII file from disk instead of having to type it all in manually. The ASCII file is read in line-by-line until it has all been entered. The keyboard then waits for user input, as it does normally

The Input File is selected with a file-chooser window. It is read in immediately after being selected.

Files read in with ‘aread’ will be processed in exactly the same way as they would if typed in at the keyboard. Excessively long lines will be rejected by the command-line processor of some operating systems, WordStar can ‘choke’ temporarily because it is unable to keep up with the faster input, but it usually recovers well.

3.4 Allocate I/O Port Files



I/O Ports Attach/Detach Configuration

Serial-2 IN: /home/jvs/gmd128/serial2_in

Serial-2 OUT: /home/jvs/gmd128/serial2_out

Parallel IN: /home/jvs/gmd128/parallel_in

Parallel OUT: /home/jvs/gmd128/parallel_out

```
yaze code (c) 1995 Frank D. Cringle.
```

Fig 10. Allocating I/O Files to the gcpz I/O Ports

Attach or detach a unix file to or from a gcpz I/O Port. There is a parallel I/O port. And there is a serial I/O port. In unix, everything is a file so one unix file or pipe is attached to the second serial-in port, and another to the second serial-out port.

Example: The ‘List’ device is allocated to the parallel port. Anything sent to the ‘List’ device will therefore show up as data in the file attached to the parallel output port.

3.5 TEXT COLOR OF THE EMULATOR OUTPUT

A selection of colors is available for the screen display. As this is pretty much a ‘set and forget forever’ option, it was decided against having a color-selection window as one of the ‘Options Menu’ items. To make a change it is simply a matter of selecting suitable values for the 24-bit RGB components, RED_LEVEL, BLUE_LEVEL, GREEN_LEVEL in the ‘gxe.h’ file and recompiling. Some examples -

Green on Black: (as default)

RED_LEVEL	0x3F
GREEN_LEVEL	0xFF
BLUE_LEVEL	0x3F

Amber on Black:

RED_LEVEL	0xFF
GREEN_LEVEL	0xBF
BLUE_LEVEL	0x3F

Yellow on Black:

RED_LEVEL	0xFF
GREEN_LEVEL	0xFF
BLUE_LEVEL	0x3F

White on Black:

RED_LEVEL	0xFF
GREEN_LEVEL	0xFF
BLUE_LEVEL	0xFF

4.0 GCPZ DEVELOPMENT ASSISTANCE

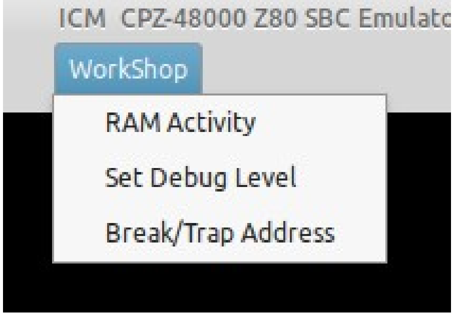
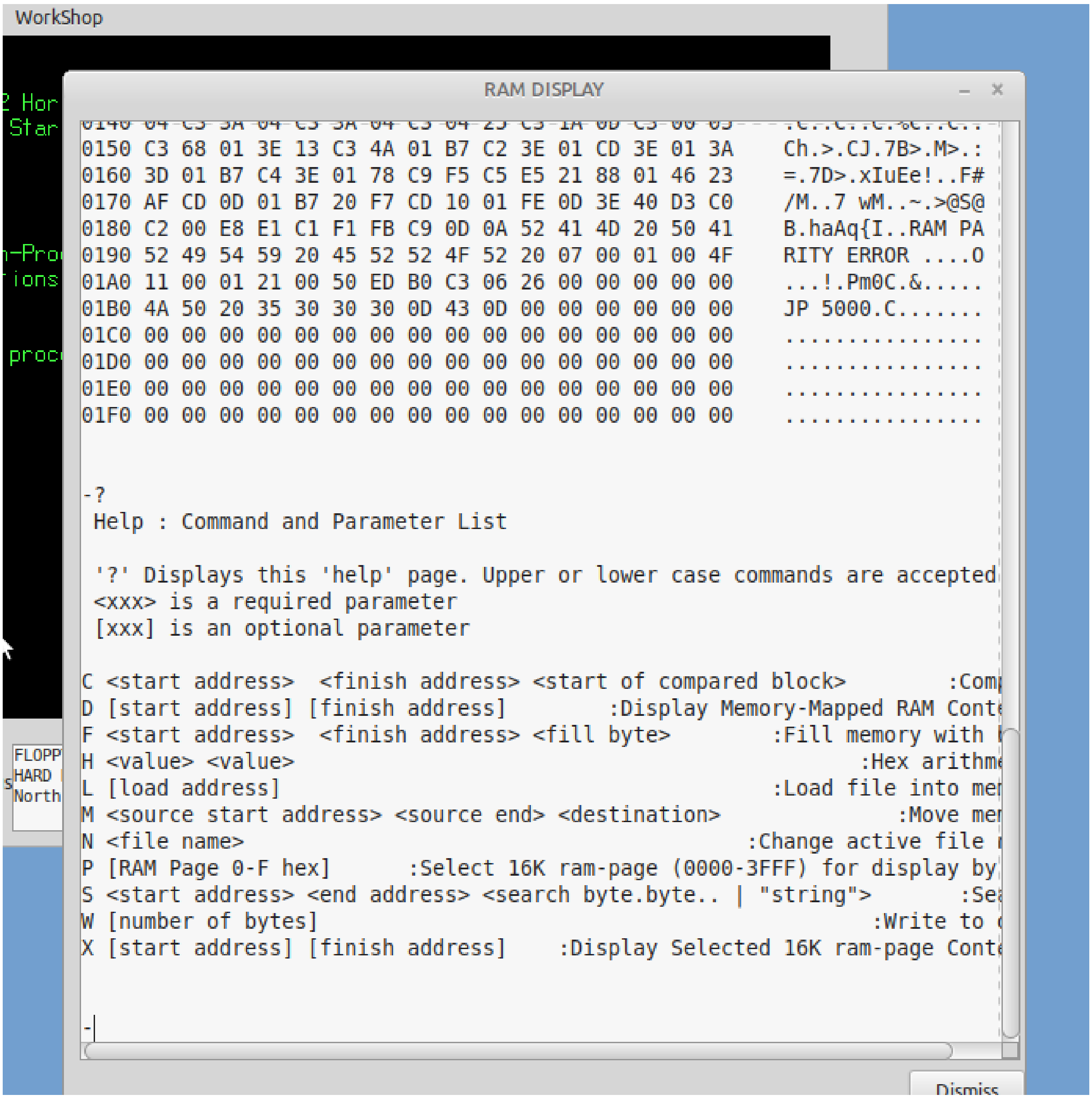


Fig 11. GCPZ Development menu: ‘WorkShop’

4.1 Display RAM in the Morrow gcpz virtual machine.



```
0140 04 C3 3A 04 C3 3A 04 C3 04 25 C3 1A 0D C3 00 03    .C..C..C.%C..C..
0150 C3 68 01 3E 13 C3 4A 01 B7 C2 3E 01 CD 3E 01 3A    Ch.>.CJ.7B>.M>.:
0160 3D 01 B7 C4 3E 01 78 C9 F5 C5 E5 21 88 01 46 23    =.7D>.xIuEe!..F#
0170 AF CD 0D 01 B7 20 F7 CD 10 01 FE 0D 3E 40 D3 C0    /M..7 wM..~.>@S@
0180 C2 00 E8 E1 C1 F1 FB C9 0D 0A 52 41 4D 20 50 41    B.haAq{I..RAM PA
0190 52 49 54 59 20 45 52 52 4F 52 20 07 00 01 00 4F    RITY ERROR ....O
01A0 11 00 01 21 00 50 ED B0 C3 06 26 00 00 00 00 00    ...!.Pm0C.&.....
01B0 4A 50 20 35 30 30 30 0D 43 0D 00 00 00 00 00 00    JP 5000.C.......
01C0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
01D0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
01E0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................
01F0 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00 00    ................


-?
 Help : Command and Parameter List

 '?' Displays this 'help' page. Upper or lower case commands are accepted
 <xxx> is a required parameter
 [xxx] is an optional parameter

C <start address>  <finish address> <start of compared block>        :Comp
D [start address] [finish address]        :Display Memory-Mapped RAM Conte
F <start address>  <finish address> <fill byte>        :Fill memory with
H <value> <value>                                              :Hex arithme
L [load address]                                       :Load file into mem
M <source start address> <source end> <destination>              :Move mem
N <file name>                                        :Change active file
P [RAM Page 0-F hex]      :Select 16K ram-page (0000-3FFF) for display by
S <start address> <end address> <search byte.byte.. | "string">       :Sea
W [number of bytes]                                           :Write to
X [start address] [finish address]    :Display Selected 16K ram-page Conte


-|
```

Fig 12. Display RAM Dialog

This subsystem has usage similar to CP/M 'DDT' or MSDOS 'DEBUG' Commands.
Upper or lower case commands are accepted
<xxx> is a required parameter, [xxx] is an optional parameter

compare
C <start address> <finish address> <start of compared block>
c 1a00 2000 2a00
Compare two equal-length blocks of memory. Only the bytes which are different will be displayed with location and values.

display
D [start address] [finish address]
d 0 12FF

Display the block of memory selected, showing bytes as hexadecimal and ASCII. If no start and end address specified, the command will continue for 100 H bytes from where it ended last.

examine/substitute
E <start address>
E 2CFF

Examine/change values at memory locations. The operation is stopped when no new value is entered, just a plain 'enter'.

fill
F <start address> <finish address> <fill byte>
f 1000 2000 55

Fill a block of memory with byte-value specified by <fill byte>.

hex
H <value> <value>
h 1267 abcd

Hex arithmetic results of the addition of two values and the subtraction of the second value from the first value.

load
L [load address]
l 2a00

Load the file (previously specified by the 'N' command) into memory. If a load-address is not specified the file will be loaded into location 0000 H.

move
M <source start address> <source end> <destination>
M 4d00 5000 6d00

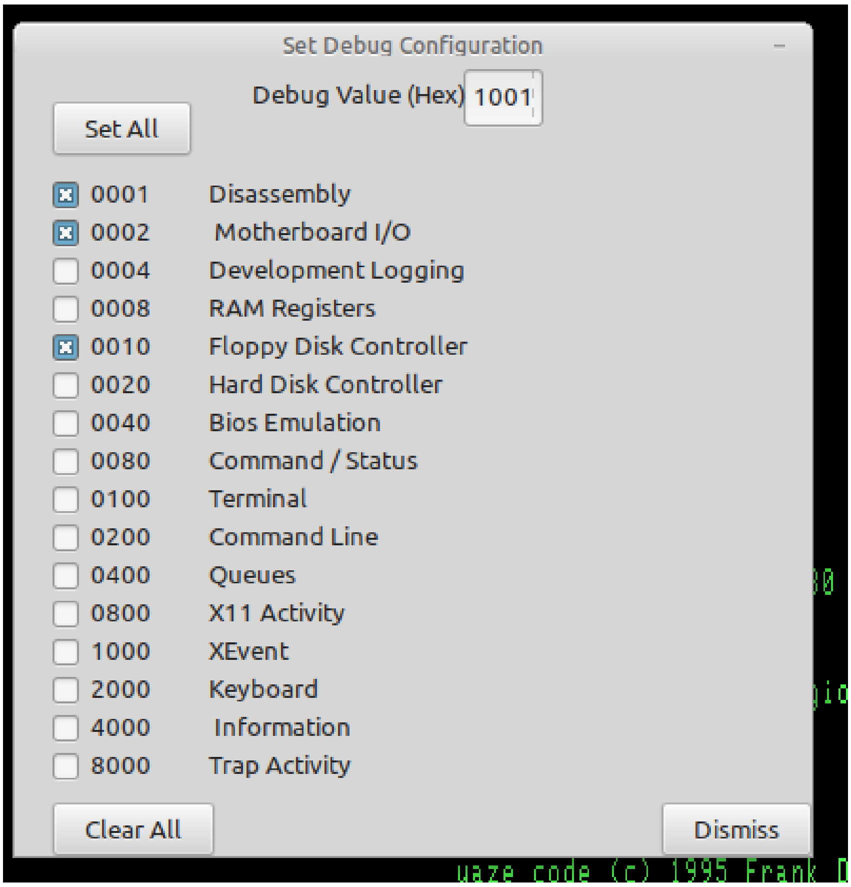
Move the block of memory specified by the block's start and end into memory beginning at the destination address.

name
N <file name>
N xtest.bin.bas

Change active file-name which specifies which unix file will be used for 'load' and 'write' operations.

quit
Q
Quit from the RAM display subsystem back to the emulator's control console.

4.2 Setting the Debug Parameters for the ‘xlog’ Debugging File Output



Set Debug Configuration

Debug Value (Hex) 1001

Set All

☒ 0001 Disassembly
☒ 0002 Motherboard I/O
☐ 0004 Development Logging
☐ 0008 RAM Registers
☒ 0010 Floppy Disk Controller
☐ 0020 Hard Disk Controller
☐ 0040 Bios Emulation
☐ 0080 Command / Status
☐ 0100 Terminal
☐ 0200 Command Line
☐ 0400 Queues
☐ 0800 X11 Activity
☐ 1000 XEvent
☐ 2000 Keyboard
☐ 4000 Information
☐ 8000 Trap Activity

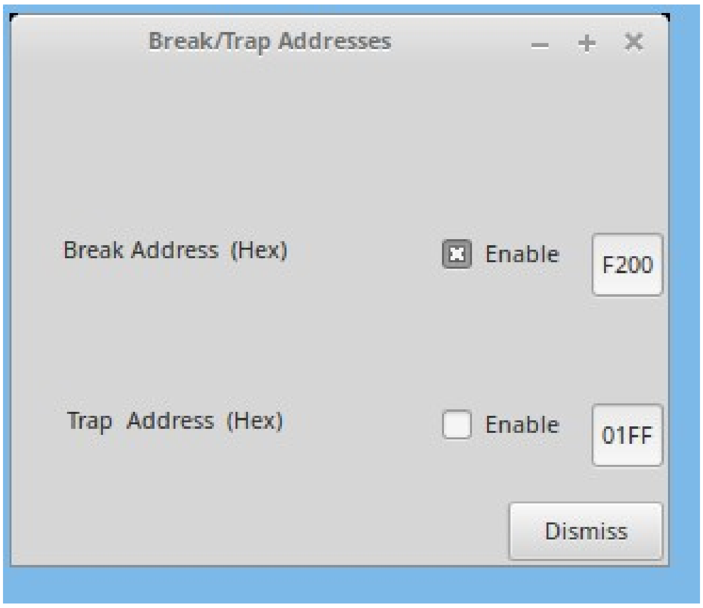
Clear All

Dismiss

Fig 13. Setting Debug Logging Parameters

Depending on which items are selected for debug logging, *a lot* of logging output can be produced. Take care that your filesystem does not get over-filled.

4.3 Setting Execution Breakpoint Address, and Trap Address



Break/Trap Addresses

Break Address (Hex) ☒ Enable F200

Trap Address (Hex) ☐ Enable 01FF

Dismiss

Fig 14. Enabling and Setting Break And Trap Addresses
break

Set a breakpoint address to stop the emulator at a pre-specified address. This is equivalent to the “PAUSE” button, but it occurs at a desired execution address. The contents of the RAM can then be examined by using the Ram Display functions. Hitting the ‘go’ button will resume execution from that breakpoint address and it will continue until that breakpoint address is again reached, unless the breakpoint is disabled while execution is stopped.

trap

Set a trap address to stop the emulator, perform a user-specified unix operation, return to the emulator and continue.

A dummy ‘trap’ function is included in the emulator source (trap.c) which merely prints the trap address and the register values. The trap function could be used to access parts of the host unix system or perform any other required operation.

Both the ‘break’ and ‘trap’ functions are enabled and disabled by the Check Buttons associated.

4.4 Log the debug information to Unix Disk File.

Automatically sends debugging/information output to the ‘xlog’ unix file.Take care, because the quantity of information sent to the log file can reach the maximum size (2 Gig in 32-bit systems, whole disk or whole filesystem in 64-bit systems) within a fairly short time.

Unless you're doing development on the GCPZ Emulator itself, it probably will not be useful to use any debug logging at all.

4.5 Log the Screen Output to Unix Disk File.

Automatically sends all ASCII screen text output to the ‘screenlog’ unix file. This can be handy to refer to if text output scrolls off the top of the screen before you can read it.

5. HELPER PROGRAMS

5.1 jdz80 (Z80 disassembler)

jdz80 is a slightly improved version of Marat Fayzullin's 1999 DAsm, in which relative jump destination addresses are calculated and displayed rather than just displaying the relative jump offsets.

5.2 cpmtools

Life is simpler with cpmtools-2.7 (or later) which can be obtained from most linux repositories. This set of utilities can be used to copy files directly between CP/M disk-images in the .fd8 and JX32 formats and the unix/linux file space. It will be necessary to add the disk definitions specified in the **gcpz_diskdefs** file to the cpmtools config-file **diskdefs** which is usually at /etc/cpmtools/diskdefs.

The added disk-definitions will enable cpmtools to understand the GCPZ CP/M disk formats, both the floppy-disk images and the larger CP/M Virtual Disk Images on the hard disk.

The utilities in cpmtools include:

cpmls	list files in the North Star CP/M disk-image
cpmcp	copy files to and from the North Star CP/M disk-image
cpmrm	delete files from the North Star CP/M disk image
mkfs.cpm	prepare stub disk for CP/M. In my experience, this does not work properly. Instead, use **mkmd128floppy** to produce a floppy disk then FORMAT it for CP/M.

5.3 screenlog

screenlog is not a tool as such but a record of GCPZ's screen output.

5.4 xlog

xlog is not a tool but is a record of all debugging information. Can make very large log files.

6.1 OTHER FILES REQUIRED

Various floppy-disk image files:

These are available from various sources on the Internet.

6.2 COMPILING LIBRARIES REQUIRED

The linux libraries required are GTK+ version 3

6.3 VARIOUS USEFUL MANUALS

Most of the manuals are available from https://itelsoft.com.au.

Probably the most useful are:

This gcpz User Guide
CPM 2.2 Manual

These are all included in the 'documentation' directory

6.4 BUGS

I feel I have got many bugs out, which makes GCPZ very usable. But there are still a few to go, apart from the things that could be done to make GCPZ not quite so rough-edged. It certainly is not yet anywhere near as elegant as I would like, and the fault-lines between the several programs that GCPZ is based upon are still very visible. Please inform me of any bugs that you discover. Email me at: jackstrangio@yahoo.com

6.5 AUTHOR and SUPPORT

Jack Strangio <jackstrangio@yahoo.com>

Website: `https://itelsoft.com.au`

APPENDIX A.

GCPZ & GMD128 EMULATORS:
STRUCTURE OF THE .fd8 FLOPPY-DISK IMAGE FILE

These emulators can run up to four eight-inch floppy drives. These drives simulate an 8" double-sided double-density floppy. They contain 77 tracks on each side. On track 0, side 0, the sectors are formatted to be 26 sectors of 128 byes, all the rest of the tracks on both sides are formatted to 16 sectors of 512 bytes. This gives a total of about 1.2megs per disk.

The disk image is laid out as follows: every track is allocated 8K (16x512 or 0x2000 bytes), whether it contains 26 sectors of 128 bytes (0xD00 bytes) , or 16 sectors of 512 bytes (0x2000 bytes). The disk-image is built up of Track0, side 0; followed by Track 0, Side 1; then Track 1, Side 0; Track 1, Side 1, etc.

```
                            Side 0                                                  Side 1
Track  0:   S S S S S S S S S S S S S S S S S S S S S S S S S S      DD  DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD
Track  1:   DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD          DD  DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD

   Tracks Omitted ........

Track 75:   DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD          DD  DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD
Track 76:   DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD          DD  DD DD DD DD DD DD DD DD DD DD DD DD DD DD DD
```

Because the disk-image is laid out regularly, the unix cpmtools can be used quite well to copy files to and from disk-images, using the cpmtools disk_defs supplied with the two emulators. Append the disk_defs supplied to the file /etc/cpmtools/diskdefs using any suitable text editor.

System tracks: With the .fd8 disk-images, the system tracks take up Track 0, Side 0 plus one other track. If the disk is single-sided the systems tracks are Track 0, Side 0 and Track 1, Side 0. If the disk is double-sided, the systems tracks are Track 0, Side 0 and Track 0, Side 1.

The CPZ-48000 software distinguishes between the various disk formats by inspecting the byte at 7EH on Track 0, Side 0, Sector 1. (The first sector on the disk.) The byte values signify:

DE	double-density 512-byte sectors single-sided
DC	double-density 512-byte sectors double-sided
E5	single-density 128-byte sectors single-sided
E7	single-density 128-byte sectors double-sided

The identification byte is used or implemented by the CPZ-specific programs:

1. Formatting programs:

DSKFMT - Formats single/double-sided 8" floppies with 26x128-byte sectors per track

FMT512 - Formats single/double-sided 8" floppies with 16x512-byte sectors per track,
except for Track 0, Side 0 which has 26 128-byte sectors.

DSKTST & DSKT512 - checks single & double-density disk sectors for errors.

2. Other programs which write at the raw-sector level:

COPY512 - Copies a complete disk.

SGEN512 - Writes system tracks to double-density (512-byte sector) tracks.

APPENDIX B.

CPZ-48000 CP/M Master Disk Files

Note that some of these will not work or will work differently when using the CPZ emulator.

From CPZ-48000 System Disk Explanatory Pages written September 15 1982:

Disk File Explanation 9-15-82

This document describes the contents of your CP/M system disk.

1. ASM.COM - This is the normal CP/M assembler. It only assembles 8080 code.

2. BASIC.COM - This is a public domain basic written by Gordon Eubanks for his Thesis. This Basic is UNSUPPORTED.

3. BOOT.ASM - This is your secondary bootstrap source file used to load your CP/M 2.2 deblocked operating system. You will use this file any time you need to make any changes in your bios file and when you rebuild your operating system tracks. See the User's Guide provided.

4. CHA-BAUD - This program allows you to change the baud rate of the A channel I/O port. The default baud rate for Channel A = 9600 Baud when the system is first brought up. CHA-BAUD will allow you to change it.

5. CONVERT.COM - This program is used to convert lower case letters to upper case letters in the label field and instruction field before using the ZASM.COM assembler. This program will check the balance of quoted strings and if an imbalance is found, will mark the source file with a @ character. Program syntax is as follows:
CONVERT [filename.ext] <cr>

6. COPY512.ASM and COPY512.COM - This program copies single-sided or double-sided deblocked 512-byte double-density disks for backup. This program checks the target disk to see if it matches the source disk before the copy operation takes place. If the two disks do not match, then an error message is given and the program restarts at the beginning. Improper disk means source disk is double-sided 512 byte, target disk is a single-sided disk, for example.

7. DDT.COM - This your CP/M Dynamic Debugger Program. This program does not support Z80 instructions.

8. DISKDEF.LIB - Library file for generating disk definition tables as explained in the Alteration Guide.

9 DSKFMT.ASM and DSKFMT.COM - This program formats disks in single-density IBM 3740 standard, or double-sided, single-density.

10. DSKT512.ASM and DSKT512.COM - This program checks 512-byte diskettes for errors. This is a read-only disktest program.

11. DSKTST.ASM and DSKTST.COM - This program checks single-density diskettes for errors. This is a read-only disktest program.

12. DUMP.ASM and DUMP.COM - Example program supplied by Digital Research.

13. ED.COM - This is your CP/M context editor.

14. *EXTRACT.COM - This program allows you to list source code from a .PRN file by using a Starting and Ending label in your program. This saves you from having to list your whole file just to get at a small area if you need to debug. Syntax is as follows:*

```
    EXTRACT [filename] <first label> <second label> <cr>        where:
          filename = name of file with extension .PRN assumed
         first label      = starting label
         second label = ending label
```

15. *FMT512.ASM and FMT512.COM - This program formats diskettes in either single-sided or double-sided 512-byte x 16 sectors. Track 0, Side 0 is always single-density 128-byte sectors for single-sided or double-sided disks.*

16. *GUESS.COM - This is a small computer game written in assembly language.*

17. *IOEQU.LIB - This file is used in conjunction with TURBO.ASM during assembly time.*

18. *LOAD.COM - Used to load .HEX files into .COM files for running in CP/M's TPA space.*

19. *MOVCPM.COM - This file is used to create a system image file which is used to rebuild your CP/M operating system. Do not destroy this file, as it is the only file which allows you to change your system memory size. See the CP/M User's Guide for further instructions.*

20. *NEWMAC.LIB - This is a collection of MACRO's which are used in some of the support programs provided on this diskette.*

21. *PIP.COM - This the CP/M interchange program.*

22. *RUN.COM - This is the runtime package used with the BASIC.COM program. This program is likewise NOT suported.*

23. *SETTIME.COM - Program used to set the 'REAL TIME CLOCK' date and time. This program is used with the auto start feature of CP/M if you set the AUTOTIM equate true in the BIOS file.* **** **CPZ will NOT change the system clock ****

24. *SGEN512.ASM and SGEN512.COM - This program is used to place your CP/M operating system onto the first two tracks of your diskette. This file will check to see if you are trying to place the wrong system type onto the wrong diskette type. You can not place a double-sided operating system onto a single-sided diskette. This program is only used to place systems onto 512-byte diskettes. It will not work with single-density disks. (See SYSGEN.COM)*

25. *SKEW.LIB - This is another MACRO file used with some of the support programs provided on this diskette.*

26. *STAT.COM - This is your CP/M status program.*

27. *STRIP.COM - This program will tear down a .PRN file back into an .ASM file. It is mainly provided so that in the event that you should lose a source file, but you have the .PRN file, you can reasonably recover back to your source file level. This package does not remove everything, such as macro expansions, but at least it may be of some help. Syntax is as follows:*

```
                    STRIP [filename1] [filename2] <cr>
                     where .PRN is assumed for filename1,
                     and .ASM is assumed for filename2.
```

28. *SUBMIT.COM - Your CP/M submit program.*

29. *SYSGEN.COM - Your CP/M sysgen program. This file only works on single-density diskettes.*

30 TIME.COM - This program is used to display the date and time at the console. Options include: TIME<cr> to display the date and time and return to CP/M.
TIME P <cr> causes the date and time to display until any key is pushed to abort.

31. TURBO.ASM - This your BIOS file for CP/M 2.2 for use with the Intercontinental Micro Systems Corp. CPU board. See the User's Guide for more details.

32. TURBO.LIB - This file contains the macros for generating a BIOS which uses extended memory as a Memory Disk. See the User's guide for more details.

33. WORM.ASM and WORM.COM - This program is an M1 memory check program used to check memory for proper operation.

34. XDIR.COM - This program displays a sorted disk directory with file sizes displayed in kilobytes.

35. XSUB.COM - This your CP/M extended submit program.

36. Z80.LIB - This is a macro file containing Z80 instructions for use with Digital Research's MAC macro assembler.

37. ZASM.COM - This a Z80 assembler used to assemble most of the support programs. The syntax for this assembler closely follows that of the CP/M assembler with some exceptions as noted in the ZASM users guide.