Setting Up CP/NET on the MinZ-U
April 9, 2021

This document is located in a github repository and the most-recent version may be obtained at https://github.com/durgadas311/minimalCPM/blob/master/doc/MinZ-U-CPNET.pdf.

Table of Contents

Introduction

The base model MinZ-U is a (approx.) 2”x2” PCB with a Z180 MCU, 512K SRAM, and 512K EEPROM. The Z180 includes two UARTs, which are connected to a dual serial-to-USB chip which provides access to the serial ports over a single USB cable to a host PC.

The MinZ-U powers-up (auto boots) into CP/M, using the rest of the SRAM as a ramdisk (A:) and the rest of the EEPROM as a romdisk (P:).

One of the serial ports is the system console. The other is available to be used for other purposes. This document describes how to use the second serial port as a CP/NET link to a host PC, where a CP/NET server provides extended disk services to the MinZ-U.

Downloads Required

cpnet12-minz.tgz: may be downloaded from http://sebhc.durgadas.com/cpnet12-minz.tgz or may be created from the github repository https://github.com/durgadas311/cpnet-z80 as described in section “Building CP/NET 1.2 for the MinZ-U”.

CpnetSerialServer.jar: The only known-working server currently available is https://github.com/durgadas311/cpnet-z80/blob/master/contrib/CpnetSerialServer.jar with documentation located at https://github.com/durgadas311/cpnet-z80/blob/master/doc/CpnetSerialServer.pdf.

jSerialComm-2.6.2.jar: This provides access to serial ports to **CpnetSerialServer.jar**. It is available at https://fazecast.github.io/jSerialComm/.

dri-cpnet.pdf (optional): The original Digital Research CP/NET documentation, updated for CP/M 3 support, is available at https://github.com/durgadas311/cpnet-z80/blob/master/doc/dri-cpnet.pdf. There is a lot of good information there, but it is a long document.

MinZ-U Setup

Connect the USB cable to the MinZ-U and to the host computer. The MinZ-U will power-up and boot, but without a console connected nothing will not be visible. Two sequentially-numbered serial (COM) ports will appear on the PC. On Linux, they are typically /dev/ttyUSB*N*, where *N* are numbers like "0" and "1". On the host PC, start a terminal program (such as PuTTY or minicom) on the first serial port, setup for 115200 baud and 8N1. Pressing the Enter key (known as "Return" on legacy systems) should cause the "A>" prompt to appear, or you can press the RESET button on the MinZ-U and see the whole startup/boot sequence.
If your MinZ-U has the newest ROM (romdisk image), then it contains the program CPNBOOT.COM in the romdisk, which is required to boot CP/NET. Otherwise, follow the MinZ instructions for writing this file to the romdisk. This file is located in the "cpnet12-minz.tgz" archive.

CP/NET Server Setup

In order to run CP/NET, you need to have a CP/NET server attached to the second serial port. This server needs to support network booting.

A compatible server is available from https://github.com/durgadas311/cpnet-z80 in contrib/CpnetSerialServer.jar. Documentation on this server is in doc/CpnetSerialServer.pdf. The server requires that JAVA be installed on the host computer, which is usually already installed. The server is configured using a file that is plain text and contains properties and values. Here is a typical configuration file for the MinZ-U:

```
# CpnetSerialServer for use with real serial port
cpnet_tty = /dev/ttyUSB1 115200
cpnet_cid = f1
cpnet_server00 = HostFileBdos /home/you/MinCPM
cpnet_protocol = dri
dri_ack_timeout = 100
dri_char_timeout = 100
cpnet_flow_control = rts/cts
netboot_dir = /home/you/MinCPM
netboot_default = cpnos.sys
netboot_org0 = none
```

The properties that require modification for your system are:

cpnet_tty: the tty name needs to match the second serial port name on your system. This example is for Linux, and Windows device names are different (e.g. "COM1:").
cpnet_server00: the path needs to match the location you have chosen for your CP/NET drives. This directory needs to have 16 subdirectories named "a" through "p". This defines server node ID "00", which is the default. Other servers may be defined for other node IDs, but each must use a different directory path (each with subdirectories "a" through "p"). Servers may also be defined to use a TCP/IP socket connection for access to remote machines.
netboot_dir: the path needs to match the location you have chosen for your CP/NET boot files. This may be the same as the path for **cpnet_server00** or it may be a different directory.

Other properties that you probably want to customize are:

cpnet_cid: the client node ID used by CP/NET to identify this MinZ-U. It should be unique, however if this server does not connect to other servers on the internet, uniqueness does not matter beyond the defined servers.

Set up the server as follows:

1. Create the directories specified by the **cpnet_server00** and **netboot_dir** properties.
2. Create subdirectories named “a” through “p” below the **cpnet_server00** directory.
3. From the “cpnet12-minz.tgz” archive, copy the files from the “cpnet12” subdirectory into the “a” subdirectory of the **cpnet_server00** path. Not all files are needed, but they may all be copied without causing any problems.
4. From a standard CP/M 2.2 distribution, copy the desired utilities to the “a” subdirectory. [TODO: suggest a URL to download these files] Add any other CP/M programs or utilities.
5. Copy the “snios.spr” and “ndos.spr” files to the **netboot_dir** directory. These files do not need to remain in the “a” subdirectory (i.e. they may be moved).
6. Place the configuration file created above, and the two JAR files, in a convenient directory for running the server. Start the server from this directory.

There is a Linux shell script that may be used to start the server, however it must be customized for the locations of the JAR files, and the use of semi-colon if used under CYGWIN. This script is at https://github.com/durgadas311/cpnet-z80/blob/master/contrib/serialserver.

The server will print informative messages to the terminal, as well as errors – if encountered. That output may be directed to a log file if desired. See the server documentation for more information.

Windows Server

Windows also requires an additional property (no value is required) in the configuration file:

cpnet_nosys: this disables use of the CP/M “SYS” file attribute, which is not available on Window filesystems. Without this setting, none of the CP/NET shared files will appear on “DIR” listings.

Start the server with this command (run from the directory chosen in step 6):

```
java -cp CpnetSerialServer.jar;jSerialComm-2.6.2.jar CpnetSerialServer conf=config-file
```

Windows/CYGWIN Server

CYGWIN on windows also requires that the **cpnet_nosys** property be set, and uses the same JAVA command to start the server.

Linux Server

Start the server using the command (run from the directory chosen in step 6):

```
java -cp CpnetSerialServer.jar:jSerialComm-2.6.2.jar CpnetSerialServer conf=config-file
```

Starting CP/NET

Once the server is running on the second serial port, CP/NET may be started on the MinZ-U with the CPNBOOT.COM program. The general syntax of the command is:

```
CPNBOOT [sid [tag]]
```

Where *sid* and *tag* are optional. *sid* is the server node ID to boot from, and defaults to “00”. tag is an optional suffix to add to the boot files, and is only valid when there are multiple different copies of boot files provided in the **netboot_dir** directory. For the setup described here, the command “cpnboot” is sufficient.

Note that CPNBOOT will normally reside on the romdisk, which is drive P:.

By default, MinZ-U drive A: will be mapped to server 00 drive A:, which contains all the files and commands required by CP/NET. Before starting CP/NET, drive A: was the MinZ-U ramdisk, which will no longer be available [TODO: update MinZ-U BIOS to allow re-mapping of the ramdisk].

In addition, the CP/M LST: device will be mapped to server 00 LST 0. Since this device is currently not defined in the server configuration file, no LST: device is available.

Sample Session

```
A>dir
NO FILE
A>p:cpnboot
ser-dri:minz Network Loader

BIOS        F600 0A00
BDOS        E800 0E00
SNIOS   SPR E500 0300
NDOS    SPR D900 0C00

54K TPA

A>dir
A: DDT      COM : NETSTAT  COM : CPNETSTS COM : TPA      COM
A: MAIL     COM : RMAC     COM : DUMP     COM : Z80      LIB
A: LINK     COM : SRVSTAT  COM : DSKRESET COM : LOGOFF   COM
A: XSUBNET  COM : XREF     COM : SID      COM : Z180     LIB
A: CCP      SPR : STAT     COM : XEDIT    COM : TR       COM
A: NETDOWN  COM : ENDLIST  COM : LOGIN    COM : NETWORK  COM
A: ZSID     COM : PIP      COM : LOCAL    COM : MAC      COM
A: RDATE    COM : ASM      COM : LOAD     COM
A>netstat

CP/NET Status
=============
Requester ID = C8H
Network Status Byte = 10H
Device status:
  Drive A: = Drive A: on Network Server ID = 00H
  List Device = List #0 on Network Server ID = 00H
A>rdate
Remote Time is: Wed 03/10/21 18:01:03
A>netdown
Ready for RESET/power-off
(press any key to resume CP/NET)

Z8S180 @ 33.333MHz  512KB RAM (0ws)  512KB Flash "ROM" (1ws)

   A: RAMdisk 428 KB
   P: ROMdisk 496 KB

CP/M 2.2  63K (56K TPA) - BIOS V2.3   SAT SEP 1/18 0:00:00


A>
```

Setting Up A Printer

There are two basic types of printer that may be setup on the server. One is a simple capture file, where all raw output is stored in the file, without separation by job (ENDLIST.COM). The other uses a Diablo 630 emulation to produce post-script files, separated by job and potentially automatically printed.

Refer to CpnetSerialServer, and CpnetSocketServer, documentation for more details. The basic premise is to use the property:

cpnetserver_lst*X*: where *X* is a hexadecimal digit representing the desire List Device Number, 0-15. Values like "> *file*" will direct raw output to *file*. The more complicated "**Diablo630Stream** *args…*" is for the Diablo 630 emulation. *args…* may reference the same configuration file being used for CpnetSerialServer, allowing Diable630 properties to be inline with general server properties. Note that, by default, diablo630 will spawn a GUI for controlling the printer. This may be undesirable for certain server contexts. Consult the server documentation for full details.

Building CP/NET 1.2 for the MinZ-U

The repository https://github.com/durgadas311/cpnet-z80 contains all the source code used to build the MinZ-U CP/NET files. This build is setup for Linux systems.

1. Clone this repository as usual.
2. Instructs for setting up the build environment are in the BUILD.md file. This file is best viewed from the repository web interface, but may also be viewed directly as plain text.
3. The build command is:

   ```
   make NIC=ser-dri HBA=minz
   ```

   This will build the files in a "bld" subdirectory. The build may be set to a different location by adding a "BUILD=…" argument, for example:

   ```
   make  NIC=ser-dri HBA=minz BUILD=~/cpnet-bld
   ```

4. The contents of "cpnet12-minz.tgz" was taken from the "ser-dri/minz/bin/cpnet12" subdirectory of the build output.