CP/M 2.2 for the ***N8VEM*** SBC

SYSTEM CONFIGURATION

CP/M IS A PRODUCT OF
DIGITAL RESEARCH INCORPORATED

ORIGINAL CBIOS CONVERSION FOR THE N8VEM
BY ANDREW LYNCH

I. CP/M Files

The ZIP archive contains the following files necessary to build a ROM image for the N8VEM SBC:

BDOSB01.ASM	CP/M BDOS
BLOADER.ASM	SYSTEM BOOTSTRAP PROGRAM
BOOTGEN.ASM	BOOT DISK UTILITY
BUILDROM2.BAT	WINDOWS BUILD SCRIPT
CBIOS.ASM	CP/M CBIOS
CCPB03.ASM	CP/M CCP
DBGMON.ASM	ROM DEBUGGER/MONITOR
LOADER-B.ASM	CP/M ".COM" FILE LOADER
VDUDRIVE.ASM	VDU ROM DEVICE DRIVERS

BDOSB01.ASM CP/M BDOS

This is the CP/M **B**asic **D**isk **O**perating **S**ystem or **BDOS**, source assembly file. This is a standard part of CP/M and has not been specifically modified for the N8VEM. There are no conditional assembly statements in this file, and can be used with no modifications.

BLOADER.ASM SYSTEM BOOTSTRAP PROGRAM

This is the ROM bootstrap program. It is located in the first 1000h bytes of the ROM, and controls the initial startup of the system. It can currently be configured to allow boot device selection from either the DSKY or the on board serial console, using conditional assembly. Support for the VDU, PropIO, and Zilog Peripheral boards is planned.

BOOTGEN.ASM **BOOT DISK UTILITY**

This is the source code for the BOOTGEN.COM CP/M utility. This utility is used to copy the running version of CP/M into the boot tracks of a Floppy Disk, a Hard Disk Partition, or a ZIP disk.

BUILDROM2.BAT **WINDOWS BUILD SCRIPT**

This is a windows script file that will build the CP/M rom image and debug loader, using the TASM assembler.

CBIOS.ASM **CP/M CBIOS**

This is the CP/M **C**ustomized **B**asic **I**nput/**O**utput **S**ystem or **CBIOS** assembly source file. It contains most of the custom code to support CP/M on the N8VEM. There are configuration lines in this file to enable support for the DSKY, VDU, IDE Hard Drive, ATAPI Zip Drive, and Floppy Drive in this file.

CCPB03.ASM **CP/M CCP**

This is the CP/M **C**onsole **C**ommand **P**rocessor or **CCP** assembly source file. This is a standard part of CP/M and has not been specifically modified for the N8VEM. There are no conditional assembly statements in this file, and can be used with no modifications.

DBGMON.ASM **ROM DEBUGGER/MONITOR**

This is the source code file for the “RAM” monitor. It gives some low level control and debugging facilities for the N8VEM system.

LOADER-B.ASM CP/M “.COM” FILE LOADER

This is the “COM” file loader assembly source file. Its purpose is to assist in CP/M CBIOS development and customization. It allows starting and testing a different version of CP/M on a running N8VEM system without burning a new EPROM.

ROMDSK.BIN ROM DISK IMAGE

This is the ROM disk image for the “F” and “G” drives.

VDUDRIVE.ASM VDU ROM DEVICE DRIVERS

This is the assembly source file for the VDU card device drivers (VT-52 emulation). These drivers are loaded at the top page of the ROM image, and then “paged” in during VDU calls. This way it is possible to take advantage of the VDU card without giving up valuable “TPA” or transient program area for CP/M.

II. N8VEM ROM image Memory Map

The ROM image map is as follows:

00000 -- 009FF	Loader Program
00A00 – 037FF	CP/M Image
03800 – 04FFF	Monitor Image
05000 – 07FFF	ROM Disk (G:)
08000 – F808F	ROM Disk (F:)
F8100 –FFFFF	VDU Card Drivers

III. Configuring CP/M

Configuring the CP/M image to work with the supported N8VEM hardware options is just a matter of making a few small changes to the **CBIOS.ASM** and **BLOADER.ASM** files.

The first configuration decision that will need to made, is to decide if the system will choose the boot device with the DSKY or the on board serial connection. (Other options will be supported in later releases) This configuration option is set in the **BLOADER.ASM** file. To make this selection, open up the **BLOADER.ASM** file in a text editor (like NOTEPAD.EXE in Windows). Scroll down to the following point in the text file:

```
;___________________________________________________________________________________
;
; SELECT ONE OF THE FOLLOWING TO BE THE ROOT CONSOLE DEVICE, ONLY ONE CAN BE ACTIVE
;___________________________________________________________________________________
;
DSKY:                .EQU      1                  ; 1= ACTIVE, 0=NOT ACTIVE
UART:                .EQU      0                  ; 1= ACTIVE, 0=NOT ACTIVE
```

Once this section is found, place a “1” after the EQU of the device that is to be used, and a “0” after the other. Only one device may be selected.

If the DSKY is used as the boot selection device, when the system starts up the “Boot” prompt will be shown on the DSKY.

At the “Boot” prompt the user can select:

0 Boot to DSKY Debug Monitor

1 Boot to Serial Console Debug Monitor

2 Boot CPM from Rom

3 Boot from floppy unit 0

4 Boot from floppy unit 1

5 Boot from Primary IDE

6 Boot from Secondary IDE

7 Boot from Primary ATAPI

8 Boot from Secondary ATAPI

More information on the DSKY device can be found in the DSKY folder on the N8VEM WIKI.

If the UART is chosen (on board serial port), the following menu will be displayed on the terminal device:

```
N8VEM BOOT LOADER

0->DSKY MONITOR
1->SERIAL MONITOR
2->BOOT CP/M ROM IMAGE
3->BOOT FLOPPY UNIT 0
4->BOOT FLOPPY UNIT 1
5->BOOT IDE PRIMARY
6->BOOT IDE SECONDARY
7->BOOT ATAPI PRIMARY
8->BOOT ATAPI SECONDARY
```

The rest of the configuration options are located in the **CBIOS.ASM** file. Open up the **CBIOS.ASM** file in a text editor, and find the following section:

```
; LIST OF CONDITIONAL ASSEMBLY INSTRUCTIONS

CONDIDESOFT:   .EQU    TRUE           ; IF NO IDE DRIVE, HAS A SIGNIFICANT DELAY
CONDSHORTMSG   .EQU    TRUE           ; TRUE FOR ORIGINAL WARM BOOT SIGNON, FALSE
CONDSUPERSUB   .EQU    TRUE           ; TRUE FOR NO SUPERSUB AUTOEXEC, FALSE TO
CONDABONLY     .EQU    TRUE           ; TRUE FOR ORIGINAL, FALSE TO ONLY HAVE

CONDUSEVDU     .EQU    FALSE          ; TRUE FOR USE VDU CARD, FALSE TO USE SERIAL
CONDUSEFLOPPY  .EQU    FALSE          ; TRUE FOR USE FLOPPY, FALSE FOR NO FLOPPY
CONDUSEATAPI   .EQU    FALSE          ; TRUE FOR USE ZIP DISK, FALSE FOR NO ZIP
CONDUSEDSKY    .EQU    FALSE          ; TRUE FOR USE DSKY, FALSE FOR NO DSKY
```

The options are:

CONDIDESOFT:

If there is an IDE drive in the system set this parameter to “TRUE”, otherwise set it to “FALSE”. Setting this parameter to “FALSE” will bypass the IDE initialization on boot up, giving a faster boot up time if there is no IDE drive attached.

CONDSHORTMSG:

Set this parameter to “TRUE” for the original boot up message, or “FALSE” for a shorter message.

CONDSUPERSUB:

Set this parameter to “TRUE” to disable SUPERSUB, or false to enable it.

CONDABONLY:

Set this parameter to “TRUE” to enable all CP/M drives, or “FALSE” to only allow drives “A” & “B”

CONDUSEVDU:

Set this parameter to “TRUE” to use the VDU Card as the primary console, or “FALSE” to use the on-board serial port.

CONDUSEFLOPPY:

Set this parameter to “TRUE” to enable the DISK IO Card floppy drive as drive “A:”, or “FALSE” to allow drive “A:” to default to the system RAM disk. CP/M is currently configured for 3.5” 720K DD floppies. Other configurations are possible, see the CP/M 2.2 Customization Guide for more information.

CONDUSEATAPI:

Set this parameter to “TRUE” to enable an ATAPI Zip device on the DISK IO IDE port as an 8MB drive “D:”. The ZIP drive will need to be jumpered as “Secondary”, and the ZIP media will need to be prepared for use with CP/M. (See the section on ZIP Media preparation)

CONDUSEDSKY:

Set this parameter to “TRUE” to initialize the on-board parallel port to use the DSKY board to display disk access information, or “FALSE” to disable this functionality. For the IDE and ATAPI devices, the DSKY will display the logical block address of any sector being accessed, and for floppy devices it will display the track, sector, and status information for any sectors accessed. This will be displayed in the format TTSSAABB, where TT is the current track, SS is the current Sector, AA is the primary status code and BB is the secondary status code.

IV. Building the ROM Image

To build the ROM Image, extract the ZIP file into a folder named ROM_IMAG. Place the **BUILDROM2**.BAT script file in the **parent** folder of the ROM_IMAG folder. Make sure that the TASM assembler s in the windows path, and execute the BUILDROM2 script. The script should assemble all of the source files, pausing after every other assemble to allow verification that there are no errors. Assuming that there are no errors there should be three .BIN files after the build. **ROMIMAGE.BIN** is image file to burn on to the EPROM**.** **LOADER.COM** is a test version of the CP/M build that can be transmitted to the N8VEM (via XMODEM) and ran like a standard CP/M command (assuming that the CCP image on the ROM is at the same location as the CCP image on this build). **BOOT.BIN** is a complete image of CP/M that can be copied to a floppy or zip disk for booting from that media.

V. CP/M Drive Letter Assignments

A:	Floppy Drive - or – Ram Disk
B:	Ram Disk
C:	IDE Partition
D:	Zip Disk
E:	Not used
F:	ROM DISK
G:	ROM DISK

VI. Preparing Hard Drive Partitions

The hard disk utility source code is located in the Disk IO folder in the N8VEM wiki and is called fmtpart2.ASM. It can be assembled into a CP/M .COM file with TASM. Upon execution fmtpart2 creates and formats a hard disk partition on the primary ide device attached to the Disk IO controller card.

VII. Preparing Floppy Disks

The floppy disk utility source code is located in the Disk IO folder in the N8VEM wiki and is called FDCMON.ASM. It can be assembled into a CP/M .COM file with TASM. Upon execution FDCMON, produces the following menu:

```
FLOPPY TEST PROGRAM
1->START MOTOR
2->STOP MOTOR
3->SELECT TRACK
4->FORMAT TRACK
5->READ SECTOR
6->WRITE SECTOR
7->DUMP BUFFER
8->SENSE INT
9->CLEAR BUFFER
A->FORMAT DISK
B->READ DISK
Q->Quit
```

To prepare a floppy disk for CP/M, format the disk (option “A”).

VIII. Preparing Zip Disks

The ATAPI (Zip) disk utility source code is located in the Disk IO folder in the N8VEM wiki and is called ATAPI-T.ASM. It can be assembled into a CP/M .COM file with TASM. Upon execution ATAPI-T, produces the following menu:

```
ATAPI TEST PROGRAM

1->Request Sense
2->Get IDE ID
3->Get ATAPI ID
4->Eject Disk
5->Read Sector
6->read Format capacities
7->Write Sector
8->Low Level Format
9->SYSGEN disk
A->CP/M Format disk
B->Select Primary Device
C->Select Secondary Device
Q->Quit
```

To prepare a ZIP disk for CP/M, first make sure the “Secondary” Device is selected (option “C”), then CP/M format the disk (option “A”).

IX. Using BOOTGEN.COM

BOOTGEN is a very simple program that takes the running version of CP/M, and creates a bootable media using that image. The quickest way to use a new (compatible) CP/M build is to use the “loader.com” image that is generated by the BUILDROM2 script to load the new CP/M version into memory, and then use BOOTGEN to write that image to a drive for booting. Using BOOTGEN is quite simple, just type BOOTGEN X: (where “X” is the drive letter to boot from), and the image will be written to that drive. Note that it is important to make sure that the CBIOS warm boot routine is set to the correct CCP address in the ROM file otherwise the new image will not boot correctly. Refer to the warm boot code in the CBIOS.ASM file for more detailed information on how this process works.

X. Finding out more

Much more information on CP/M, including how to create a new customized version of CP/M, can be obtained from the original Digital Research documentation available:

http://www.cpm.z80.de/drilib.html

And of course, a wealth of information can be found on the N8VEM discussion group, and the N8VEM Wiki:

http://n8vem-sbc.pbworks.com/
http://groups.google.com/group/n8vem