

CUBIX for the N8VEM HOST PROCESSOR and BACKPLANE

SYSTEM CONFIGURATION for STAND ALONE
OPERATION

CUBIX IS A PRODUCT OF
DUNFIELD DEVELOPMENT SERVICES

WWW.DUNFIELD.COM

ORIGINAL CUBIX SOURCES LOCATED AT

<http://www.classiccmp.org/dunfield/d6809/cubix.htm>

CONVERSION FOR THE N8VEM
BY DAN WERNER

I. 6809 HOST PROCESSOR ROM

The 6809 ROM image is located in the m6809 folder in the zip file. There are currently three versions of the ROM available for different “stand alone” configurations:

- R09SASE – IO board 6551 Serial Console
- R09SAVD – VDU board console
- R09SADS – DSKY based (front panel) Console

The “SA” and “VD” ROMs also support displaying disk IO track and sector information via the DSKY. This support can be deactivated by commenting out the following line in the beginning of either source file.

```
BDSKY      EQU      1                      USE DSKY TO DISPLAY SECTOR INF
```

Serial ROM

The serial ROM supports the following monitor commands

Dxxxxyyyy	Dump memory from xxxx (in hex) to yyyy (in hex)
L	Load a Motorola format image file
Mxxxxyy	Change Memory byte at location xxxx (in hex) to value yy
P	Print stack contents
Gxxxx	Run program at xxxx
B	Boot CUBIX from a device

VDU ROM

The VDU ROM does not support any monitor functions, upon system start the user is prompted to boot CUBIX from either the (F)loppy or (H)ard drives.

DSKY ROM

[DE] function – Deposit to Memory

- At the -CPU UP- prompt the user presses the [DE] key.
- System prompts for the address to write to: ADDR
- The user enters the four digit address on the keypad, then press [EN] (enter).
- The [CL] key will clear the input if the user enters the address incorrectly (before pressing [EN])
- System will the prompt for the value to write to the address: xxxx .. (xxxx is the address that was entered)
- The user then enters the byte value to write to the address, then press [EN] (enter). The [CL] key will clear the input if the user enters the port incorrectly (before pressing [EN])
- The user can then press the [CL] key to return to the -CPU UP- prompt, or can press [EN] and the system will prompt the user to enter a value to the next consecutive address. The user can also press the [DE] key to enter a new address.

[EX] function – Examine Memory

- At the -CPU UP- prompt the user presses the [EX] key.
- System prompts for the address to write to: ADDR
- The user enters the four digit address on the keypad, then presses [EN] (enter).
- The [CL] key will clear the input if the user enters the address incorrectly (before pressing [EN])
- System will then display the address and the value.: xxxx yy (xxxx is the address that was entered, yy is the value at that address)
- The user can then press the [CL] key to return to the -CPU UP- prompt, or can press [EN] and the system will display the next consecutive address and value. The user can also press the [EX] key to enter a new address.

[GO] function – Execute Program

- At the -CPU UP- prompt the user presses the [GO] key.
- System prompts for the address to write to: ADDR
- The user enters the four digit address on the keypad, then press [EN] (enter).

- The [CL] key will clear the input if the user enters the address incorrectly (before pressing [EN])
- System will then execute the code at that address

[BO] Function – Boot System

- At the –CPU UP– prompt the user presses the [BO] key.
- System prompts for the boot device/location: boot . .
- The user enters the one digit boot identifier
- System will then boot up using that device/location

Currently supported:

0 Boot from floppy unit 0

1 Boot from Primary IDE

II. CUBIX OS

The source files for the CUBIX OS are stored in the CUBIX OS folder and built with the BUILD CUBIX script. All of the N8VEM configurable code is stored in the DRIVERS.ASM file. Very little modification is required to customize CUBIX for a specific configuration.

To set the default console device change the following lines to either 1 for serial or 2 for VDU.

```
* CONSOLE DEVICE ASSIGNMENTS
FCB      1      CONSOLE INPUT DEVICE
FCB      1      CONSOLE OUTPUT DEVICE
```

To set the default drive and system drive change the following lines.

```
* MISC FLAGS & VARIABLES
FCB      $FF      ERROR MESSAGES ENABLED
FCB      0      TRACE DISABLED
FCB      0      DEBUG DISABLED
FCB      2      DEFAULT DRIVE (C)
FCC      'MAIN'    DEFAULT DIRECTORY
FCB      0,0,0,0    (FILLER)
FCB      2      SYSTEM DRIVE (C)
FCC      'SYSTEM'  SYSTEM DIRECTORY
FCB      0,0      (FILLER)
```


If you are interested in further customizing CUBIX, all of the documentation to do so is available in the DOCS folder.

III. Creating Boot Disk

The easiest way to bring up a new CUBIX system and create a boot disk, is to use the serial ROM attached to a P.C. You can then use the “L” command to load the CUBIX.S19 file into main memory and manually boot CUBIX with the GC808 command. Once CUBIX has started up, use the FORMAT (drive:) command to format either the A: or C: drives. Then use the DOWNLOAD command to download the BOOTGEN program. Note that the CUBIX DOWNLOAD command expects plain text input of a Motorola format (.S19) file, no download protocol such as XMODEM or KERMIT is required.

Example:

DOWNLOAD 1 A:BOOTGEN.EXE

Once the program is saved to your disk, it can be ran with the first parameter being the disk that you want to record CUBIX on to. Note that the “RUN” command is not needed if you store the program in the CUBIX system directory. (see the CUBIX documentation for more information)

Example

RUN BOOTGEN A:

This command will transfer the version of CUBIX in memory to the specified disk, once this is complete the system can be booted from that disk.

IV. CUBIX Utilities

All of the CUBIX utilities are stored in the CUBIXUTIL folder, and can be built with the BUILD CUBIXUtil script. Once the programs are built, you can use the CUBIX DOWNLOAD internal command to download the Motorola format files and save the binary images to disk.

Example:

DOWNLOAD 1 C:[SYSTEM]DIR.EXE

For more information see the CUBIX system description and users guide in the DOCS folder.

V. VDU hardware modification

In order to use the N8VEM VDU card with the 6809 backplane the following simple hardware changes need to be made to the VDU card:

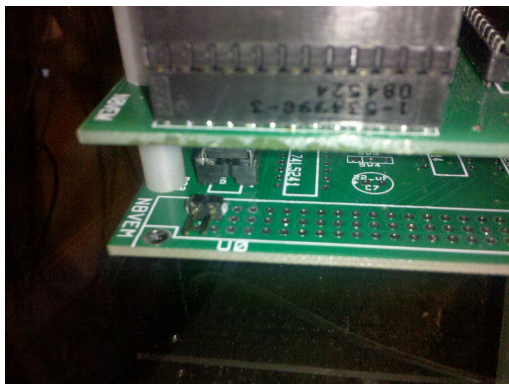
- Remove IC15, and replace it with a jumper from pin 4 to pin 6.
- Remove pin 6 from IC11

VI. 6809 host processor hardware

The 6809 host processor can be built with a reduced part count if the board will never be attached directly to the Z80 ECB bus. The following components may be omitted:

- 96 pin ECB connector
- U9 – 6821
- U6 – 8255
- RR2 – 10k resistor network
- IC1A – 74LS245
- U2 – 74LS241
- U4 – 74LS241
- U5 – 74LS14
- IC6 – 74LS32
- SW1,SW2,SW3,SW4 – address selection jumpers

Suggested Modifications:



Reset Modification: The reset switch on the IO card is inaccessible once the backplane level is installed. An easy way to add a reset switch is to solder a jumper onto the ECB connector on the host processor card. If this modification is used,

ensure U5 is not omitted from the build of the 6809 Host Processor.

Backplane Jumper: For proper operation of the backplane, a jumper is required from PIN 10 of P25 to C22 on any one of the slots.