

## How to program Adapt812DX with HyperTerm

To program External memory the internal EEPROM of MCU are program with Flash Loader utility called MXFlash.

<http://www.interlog.com/~techart/myfiles/files/mxflash.zip>  
<http://www.technologicalarts.com/myfiles/812expanded.html>

The MCU must boot up in single chip mode in order for the Loader to work. Set the various jumpers as indicated below.

Jumper settings for MODA and MODB for single chip mode.

JB1 – pin 2 and 3 shunted – MODB  
JB5 – shunted – MODA  
SW3 to SGL (single chip) position.

Jumper settings for FLASH and RAM with REV0 to REV2A

JB3 – pin 1 and 2 shunted  
JB4 – CSP0 and F shunted  
JB6 – CSD and R shunted

Jumper settings for FLASH and RAM with REV3

JB3 – pin 1 and 2 shunted (CSD\* RAM enable)  
JB6 – pin 1 and 2 shunted (CSP0\* FLASH select)  
JB7 – pin 1 and 2 shunted (RAM write enable)

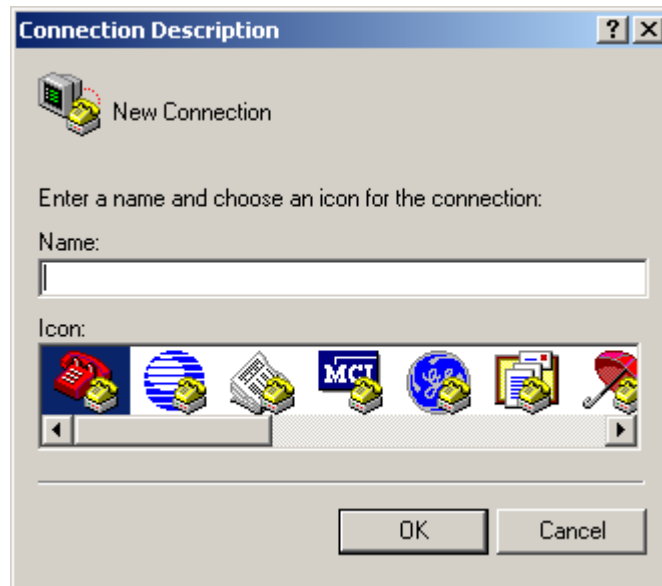
Once the jumpers are set one can proceed to programming the external FLASH. Other terminal program can be used just as long as one is familiar with how to use that program.

In this example the HyperTerm Terminal program will be use because it is already (usually) as part of Windows OS installation.

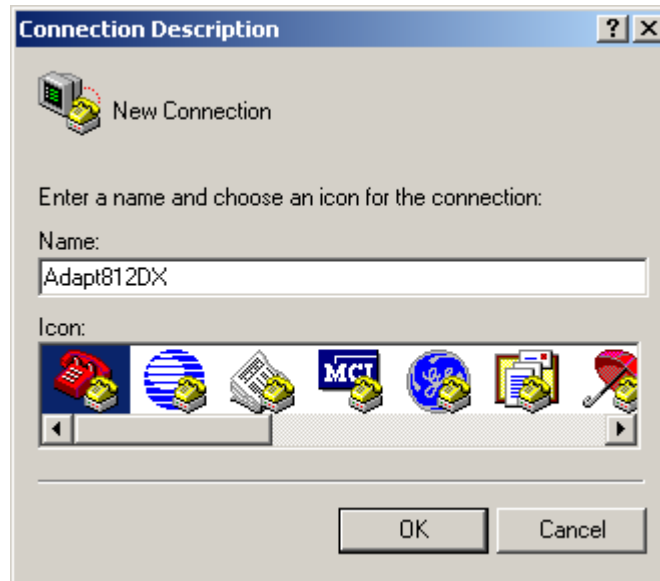
Setting HyperTerm:

The procedures are for WinXP, it may differ with different OS.

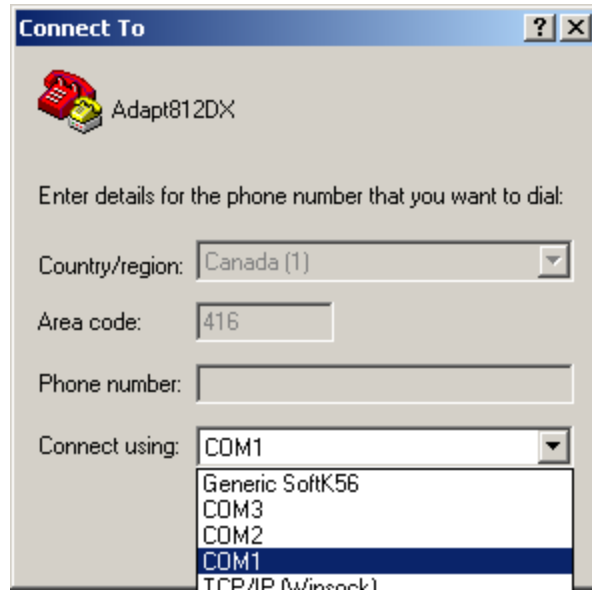
Start – Programs – Accessories – Communications – HyperTerminal



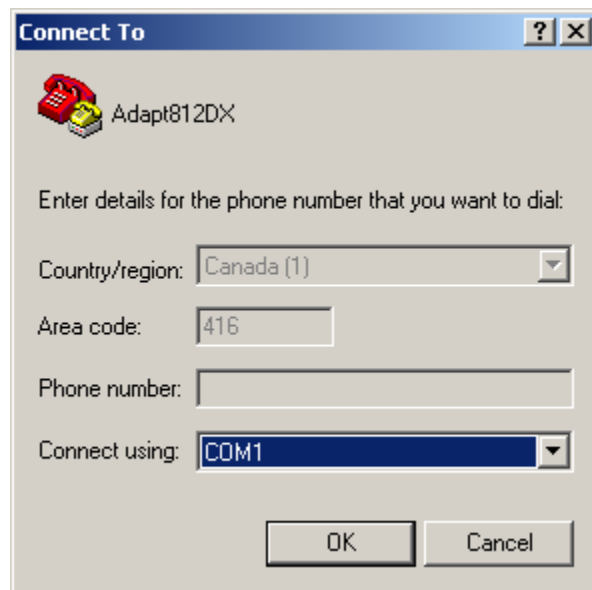
Give it a name.



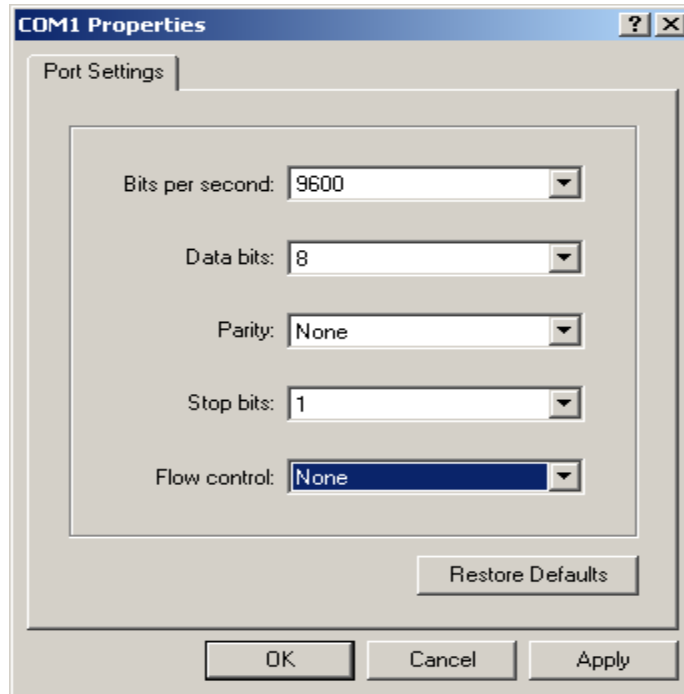
Choose a COM port to use.



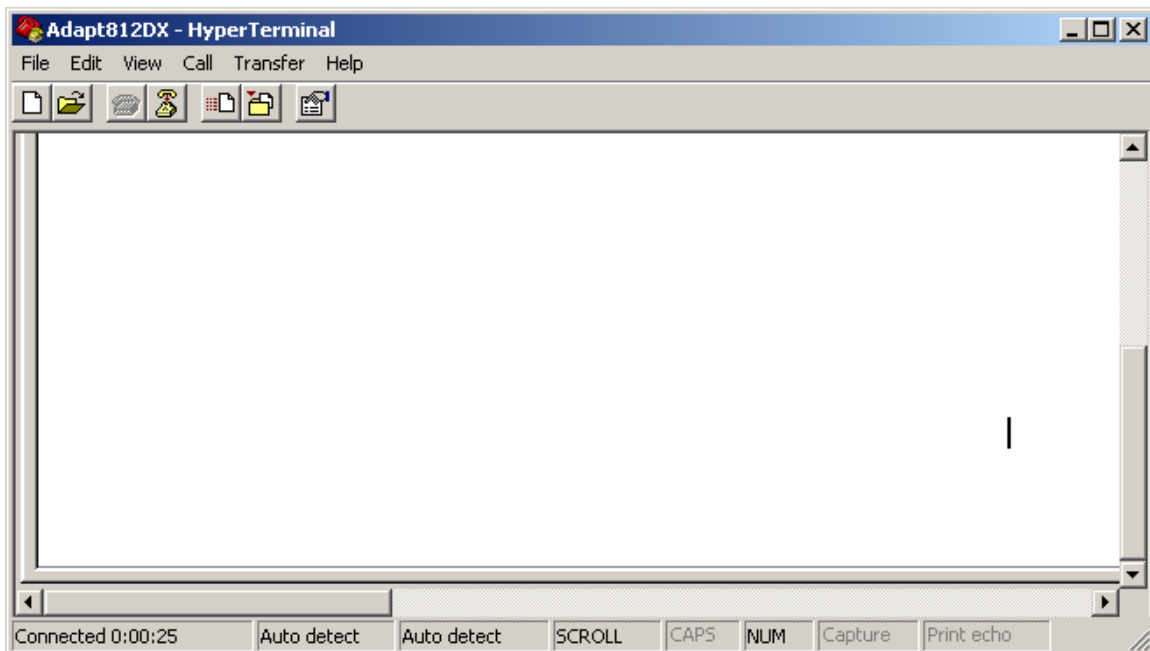
In this example, COM 1 is selected.



Set the BAUD parameters as shown to **9600**, **8**, **N**, **1** and set Flow control to **None**.



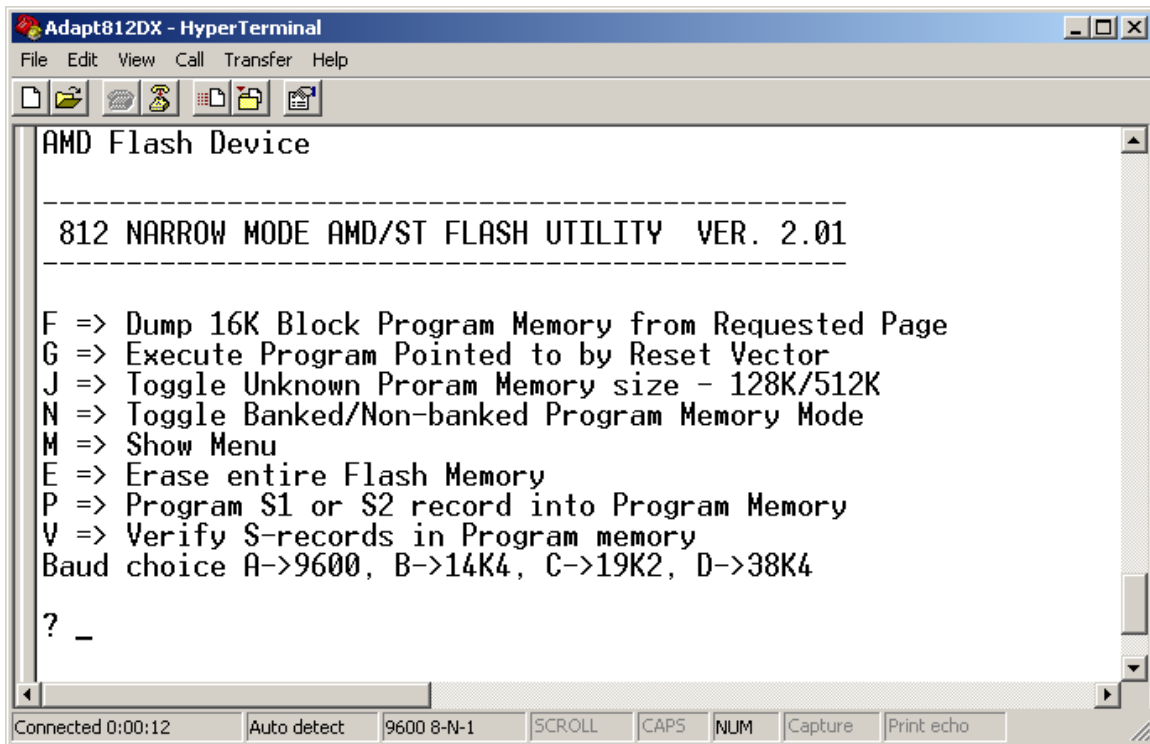
Once the above are set we can proceed to connect the Adapt812DX to COM 1. Below, HyperTerm is ready to connect to COM 1.



Connect Serial Cable to COM 1 at the back of PC and to the Adapt812DX. Slide SW2 Run/Boot switch to Run position. Make sure SW3 is in SGL position.

Power up the Adapt812DX board with a known good power supply, making sure the PWR LED is on. Once the board is powered, HypeTerm should display the MXFlash Loader menu as shown.

Notice that it detected the FLASH to be *AMD Flash Device*. If the Loader cannot detect a FLASH memory it will display unknown Memory and will assume that the memory is RAM.



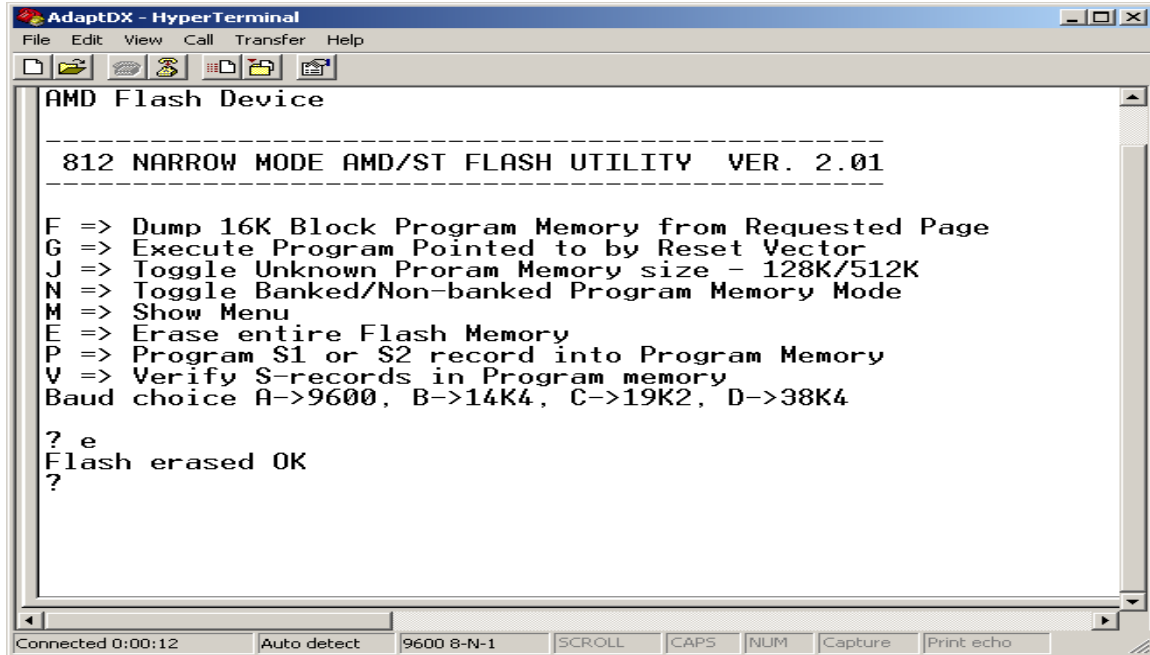
```
Adapt812DX - HyperTerminal
File Edit View Call Transfer Help
AMD Flash Device
-----
812 NARROW MODE AMD/ST FLASH UTILITY VER. 2.01
-----
F => Dump 16K Block Program Memory from Requested Page
G => Execute Program Pointed to by Reset Vector
J => Toggle Unknown Program Memory size - 128K/512K
N => Toggle Banked/Non-banked Program Memory Mode
M => Show Menu
E => Erase entire Flash Memory
P => Program S1 or S2 record into Program Memory
V => Verify S-records in Program memory
Baud choice A->9600, B->14K4, C->19K2, D->38K4
? _
Connected 0:00:12 Auto detect 9600 8-N-1 SCROLL CAPS NUM Capture Print echo
```

There are several things to check if the Menu will not show up as shown above.

- Do not use NULL cable. Use a straight thru cable.
- Check and verify the cable is connected to the correct COM port.
- Re-boot the PC if necessary as other application may have corrupted the OS.
- Use known good serial cable.
- If there are other problems, test the COM port with another device that is known to work.
- If the FLASH type is not recognized re-check jumper settings. If still unrecognized then the board has a problem.

### Erasing:

To erase the FLASH the command is *E*.



```
AdaptDX - HyperTerminal
File Edit View Call Transfer Help
AMD Flash Device
-----
812 NARROW MODE AMD/ST FLASH UTILITY  VER. 2.01
-----
F => Dump 16K Block Program Memory from Requested Page
G => Execute Program Pointed to by Reset Vector
J => Toggle Unknown Program Memory size - 128K/512K
N => Toggle Banked/Non-banked Program Memory Mode
M => Show Menu
E => Erase entire Flash Memory
P => Program S1 or S2 record into Program Memory
V => Verify S-records in Program memory
Baud choice A->9600, B->14K4, C->19K2, D->38K4

? e
Flash erased OK
?
```

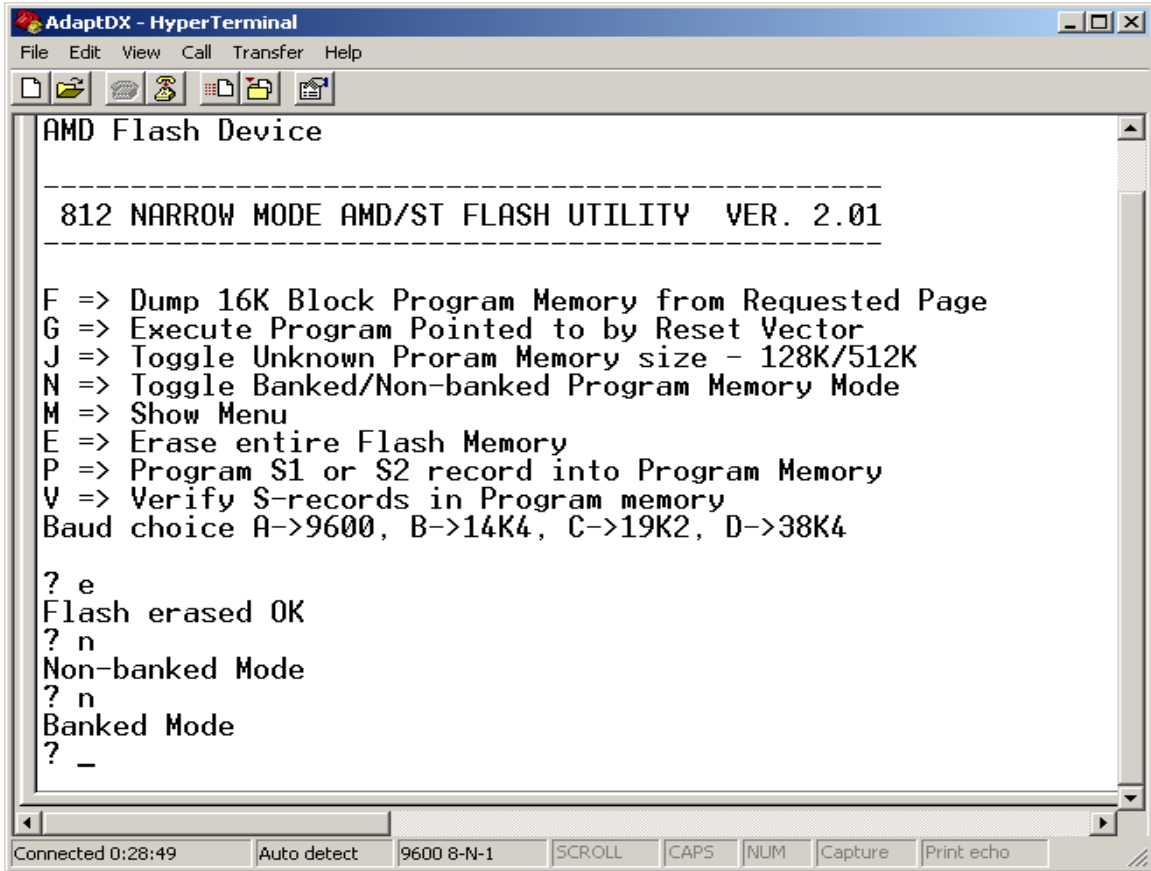
### Programming:

There are 2 options for programming the FLASH. The 1<sup>st</sup> is whether the S-record is BANKED. The command is *N* to toggle from BANKED to NON-BANKED. ICC12 creates mixed S1 and S2 record. This is generally called BANKED mode. One should always check the S19 record to see if there are S1 and S2 mixture. Other compiler creates BANKED S2 too. One should experiment whether to program the S-record as BANKED or NON-BANKED.

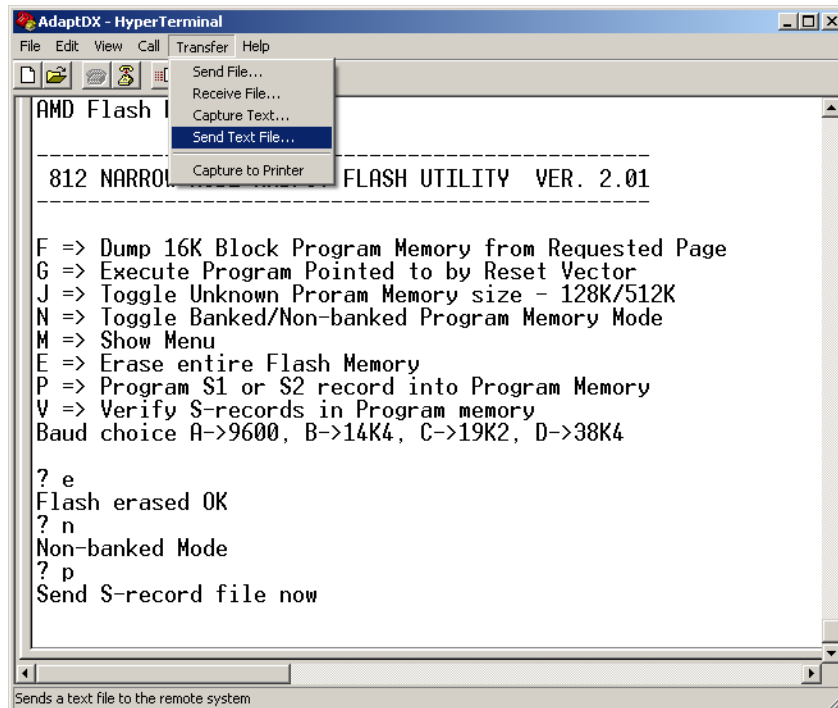
By choosing the *N* command the loader will always toggle from previous state to the next. By default it is set for BANKED mode.

If the S-record contain only addresses from \$C000 to \$FFFF then it should always be programmed as NON-BANKED. Linear S2 record must always be programmed as NON-BANKED.



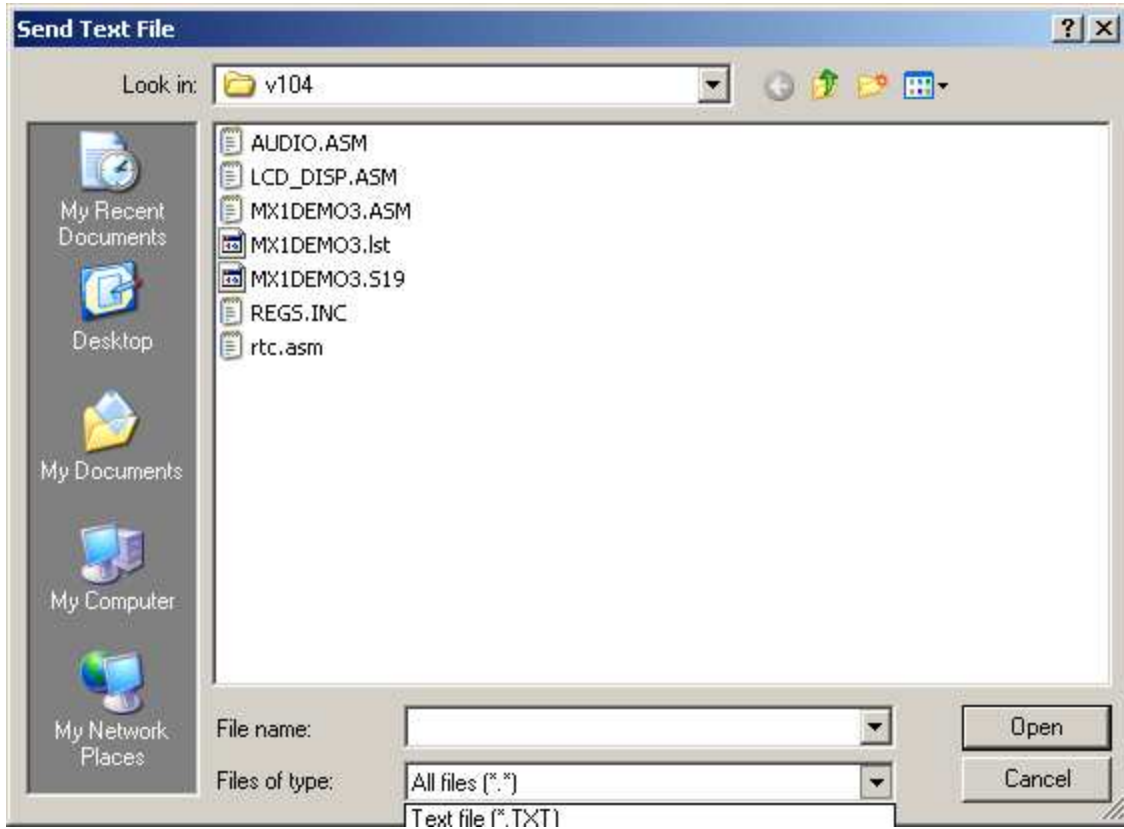


To program press **P** once press click Transfer menu – Send Text File

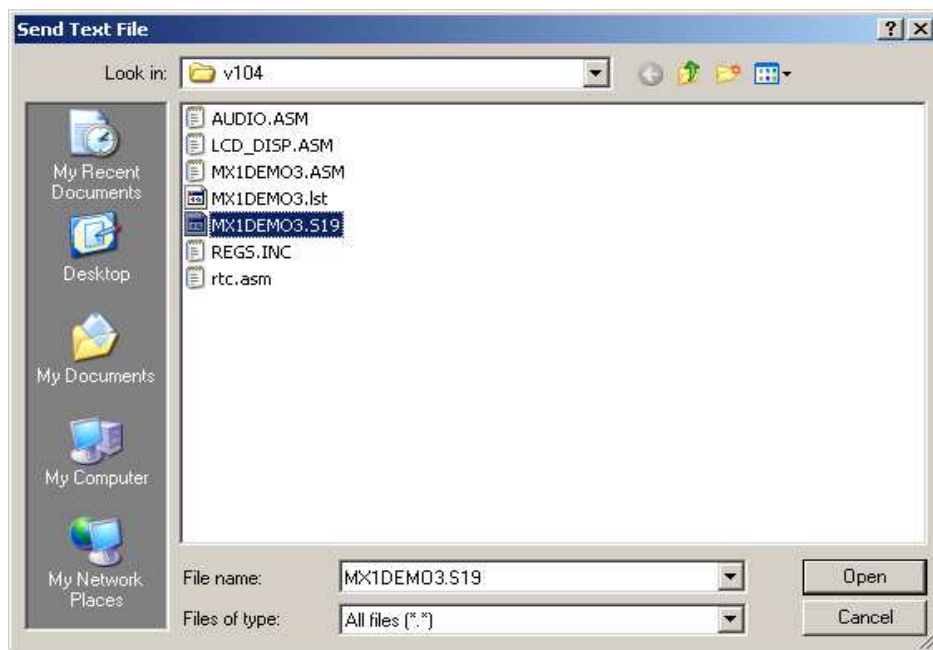




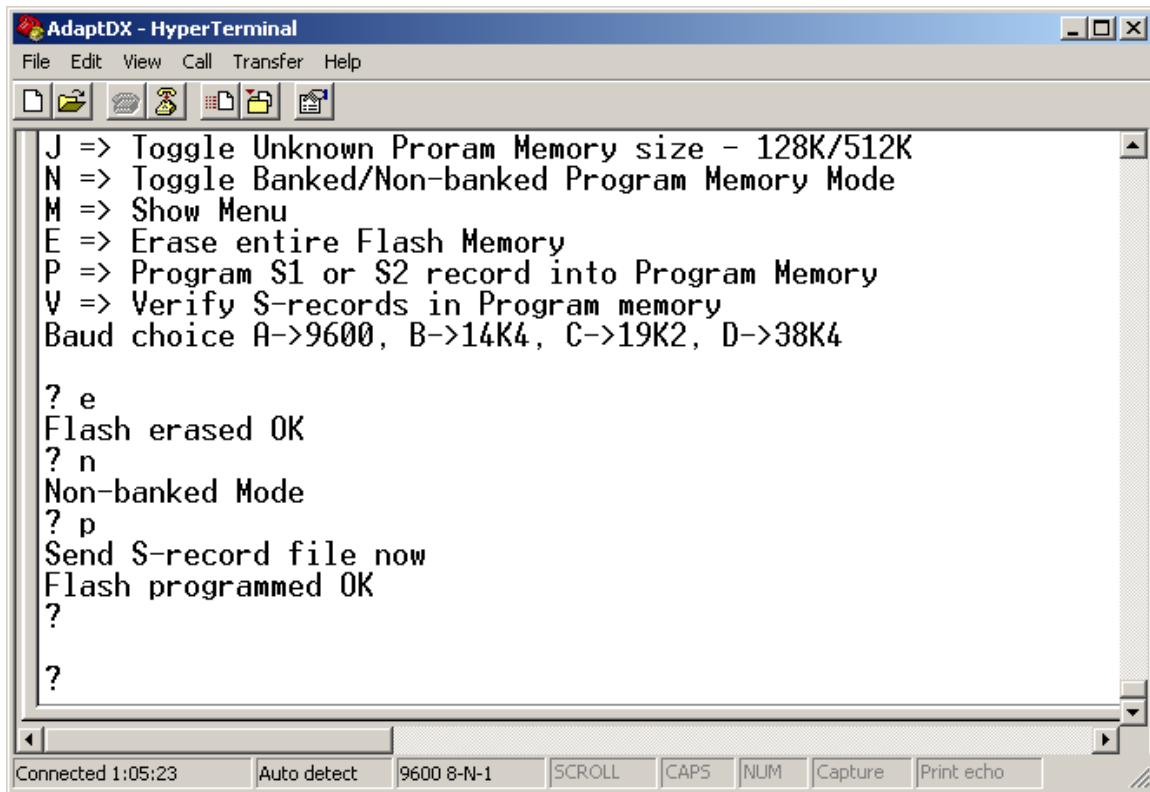
HyperTerm will open an explorer window to help locate the file to be uploaded. Change file type to All files.



Click on the S19 record to be uploaded and press the *Open* button.



After uploading the message *Flash programmed OK* will appear to show the programming is finished.



```
AdaptDX - HyperTerminal
File Edit View Call Transfer Help
J => Toggle Unknown Proram Memory size - 128K/512K
N => Toggle Banked/Non-banked Program Memory Mode
M => Show Menu
E => Erase entire Flash Memory
P => Program S1 or S2 record into Program Memory
V => Verify S-records in Program memory
Baud choice A->9600, B->14K4, C->19K2, D->38K4

? e
Flash erased OK
? n
Non-banked Mode
? p
Send S-record file now
Flash programmed OK
?
?
```

Connected 1:05:23   Auto detect   9600 8-N-1   SCROLL   CAPS   NUM   Capture   Print echo

After programming, slide SW3 to EXP (MODA = 1) then press the RESET button. The programmed application is now running in expanded NARROW mode.

### Checking DATA in Load mode:

The command is *F* for memory dumping DATA from FLASH. This is used to check certain parts of the memory.

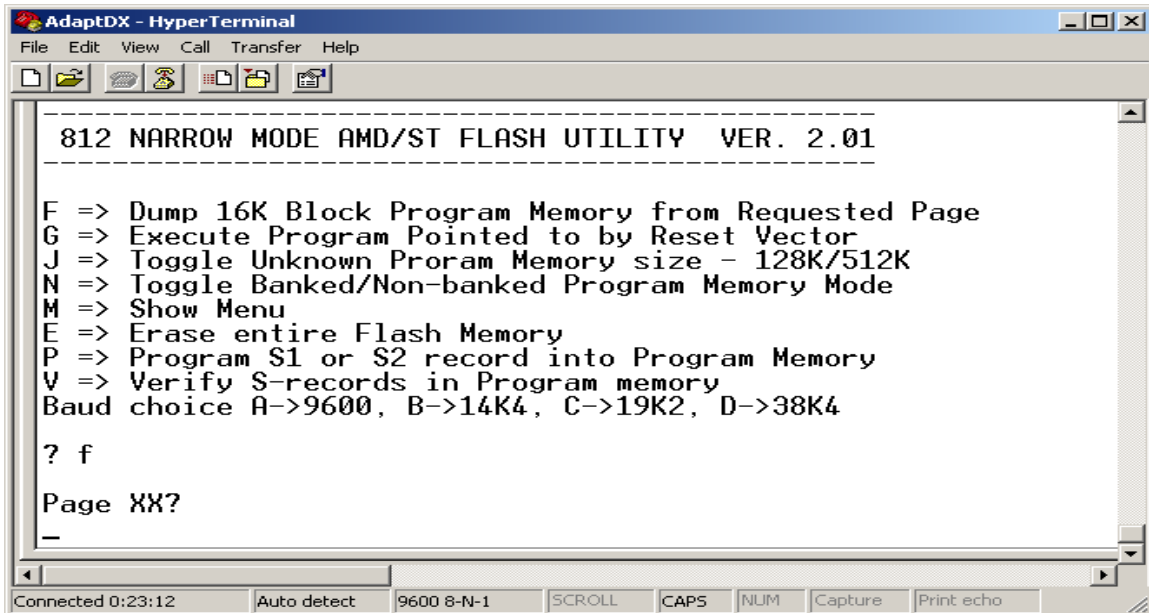
For example, one may want to check the address from \$C000 to \$FFFF. How is this possible? There are few things that need to be understood. The memory block \$C000 to \$FFFF is always present in the memory map. This is where the ISR and start of code resides. When servicing an ISR it is possible to access a different PPAGE. With 128Kbyte Flash, the \$C000 to \$FFFF can be accessed at PPAGE = \$07 in the page window \$8000 to \$BFFF.

### Calculating PPAGES:

To calculate how many PPAGE in 128Kbyte (AM29F010) is to have  $128K/16K = 8$ . Therefore there are 8 PPAGES beginning from \$00 to \$07. A page is always = 16K.

For 512K FLASH (AM29F040) there would be  $512K/16K = 32$ . Therefore there are 32 PPAGES beginning from \$00 to \$1F. Address \$C000 to \$FFFF is PPAGE = \$1F

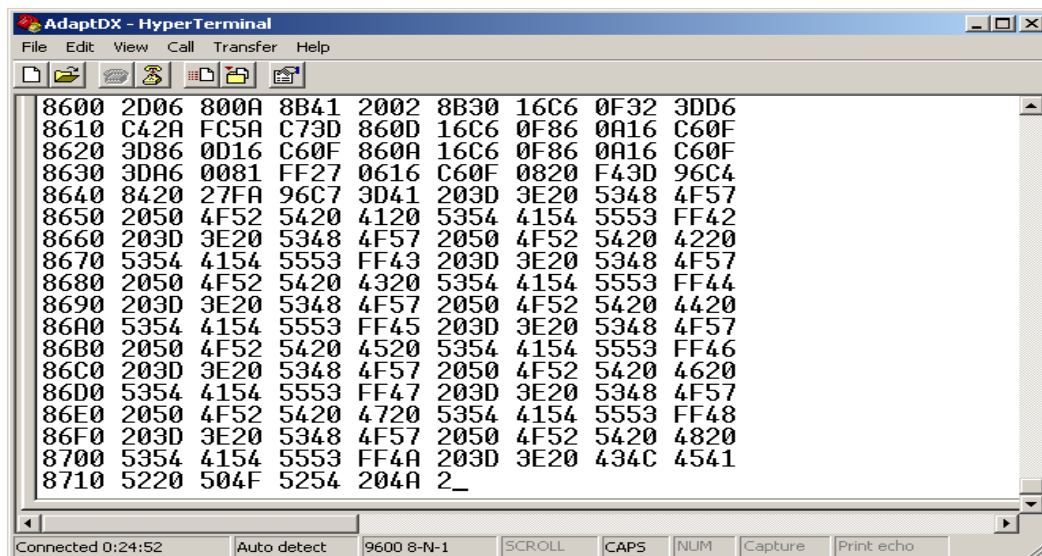
Here is an example to dump \$C000 to \$FFFF with a 128Kbyte Flash. Select *F* command then type the value *07*



```
AdaptDX - HyperTerminal
File Edit View Call Transfer Help
-----
812 NARROW MODE AMD/ST FLASH UTILITY VER. 2.01
-----
F => Dump 16K Block Program Memory from Requested Page
G => Execute Program Pointed to by Reset Vector
J => Toggle Unknown Program Memory size - 128K/512K
N => Toggle Banked/Non-banked Program Memory Mode
M => Show Menu
E => Erase entire Flash Memory
P => Program S1 or S2 record into Program Memory
V => Verify S-records in Program memory
Baud choice A->9600, B->14K4, C->19K2, D->38K4

? f
Page XX?
_
```

Immediately after it would begin to dump the DATA beginning at \$8000 to \$BFFF. Please note that PPAGE = \$07 at \$8000 to \$BFFF is the same as \$C000 to \$FFFF



```
AdaptDX - HyperTerminal
File Edit View Call Transfer Help
-----
8600 2D06 800A 8B41 2002 8B30 16C6 0F32 3DD6
8610 C42A FC5A C73D 860D 16C6 0F86 0A16 C60F
8620 3D86 0D16 C60F 860A 16C6 0F86 0A16 C60F
8630 3DA6 0081 FF27 0616 C60F 0820 F43D 96C4
8640 8420 27FA 96C7 3D41 203D 3E20 5348 4F57
8650 2050 4F52 5420 4120 5354 4154 5553 FF42
8660 203D 3E20 5348 4F57 2050 4F52 5420 4220
8670 5354 4154 5553 FF43 203D 3E20 5348 4F57
8680 2050 4F52 5420 4320 5354 4154 5553 FF44
8690 203D 3E20 5348 4F57 2050 4F52 5420 4420
86A0 5354 4154 5553 FF45 203D 3E20 5348 4F57
86B0 2050 4F52 5420 4520 5354 4154 5553 FF46
86C0 203D 3E20 5348 4F57 2050 4F52 5420 4620
86D0 5354 4154 5553 FF47 203D 3E20 5348 4F57
86E0 2050 4F52 5420 4720 5354 4154 5553 FF48
86F0 203D 3E20 5348 4F57 2050 4F52 5420 4820
8700 5354 4154 5553 FF4A 203D 3E20 434C 4541
8710 5220 504F 5254 204A 2_
```

Below is at the end of DATA dump.

```
AdaptDX - HyperTerminal
File Edit View Call Transfer Help
[Icons]
BEF0 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BF00 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BF10 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BF20 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BF30 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BF40 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BF50 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BF60 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BF70 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BF80 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BF90 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BFA0 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BFB0 FFFF FFFF FFFF FFFF FFFF FFFF FFFF FFFF
BFC0 FFFF FFFF FFFF FFFF FFFF FFFF FFFF C0D2
BFD0 C0D2 C0D2 D1BB D197 C0D2 C0D2 C0D2 C0D2
BFE0 C0D2 D21D C0D2 C0D2 C0D2 C0D2 C0D2 C0D2
BFF0 D149 C0D2 C0D2 D212 C0D2 C0D2 C0D2 C000
? _
Connected 0:27:39 Auto detect 9600 8-N-1 SCROLL CAPS NUM Capture Print echo
```

Note that at address \$BFFE:BFFF is equal to \$FFFE:\$FFFF. The value is seen to be \$C000 the start of code.

This tool is implemented to help diagnose where are the DATA being programmed too from an S-record file.

### Executing Program in Load Mode:

The command is **G** to execute the already program code. The Loader will fetch the address at \$BFFE:\$BFFF (PPAGE set at the Last page) then start executing the code at that address. For example, if the value was \$C000, the loader removes the EEPROM out of the memory map, read the vector address and make Register X equal \$C000 and jump to \$C000 with Register X.

### Note:

In load mode certain registers are initialized. The stack is set to \$0C00, MODA:MODB in expanded narrow mode including the various expanded register controls. Lastly the COP control is disabled.

### Problems:

Q. Programmed okay but will not appear to run.

A. Toggle from BANKED to NON-BANKED or vice-versa then erase and re-program again.

Q. Tried to program in BANKED or NON-BANKED, Flash programmed OK but will not work.

A. There are 3 things to check.

1. COP control needs to be disabled if not being serviced.
2. Check if the Stack is initialized
3. Check if the Power ON reset vector is correctly pointing to start of code.

Q. Programmed OK and runs with G command but not in Expanded mode.

A. Check the stack is initialized and COP control disabled if not being serviced. In load mode the stack is initialized and COP disabled.

Q. During programming the red LED comes on

A. The Loader is unable to program into a memory location. The reasons are many.

1. The S-record may contain code that are manipulating the registers.
2. The S-record may contain addresses that are outside the bound of the FLASH chip.
3. Lastly, the FLASH memory maybe faulty.

Q. Programming Rev3 is different from Rev2A.

A. Verify the jumper settings FLASH and RAM

JB3 – pin 1 and 2 shunted (CSD\* RAM enable)

JB6 – pin 1 and 2 shunted (CSP0\* FLASH select)

JB7 – pin 1 and 2 shunted (RAM write enable)

Please note that this problem is recent. We are investigating why there is a difference between these Revs.