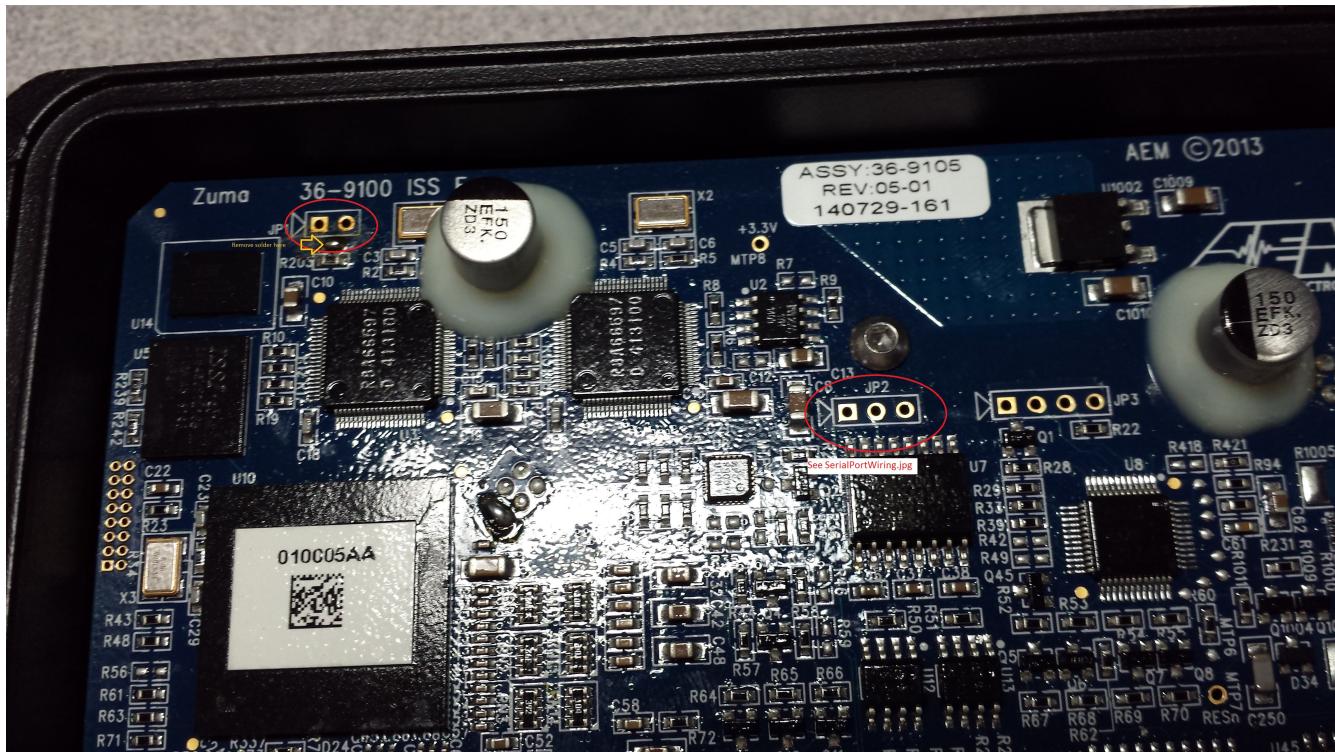


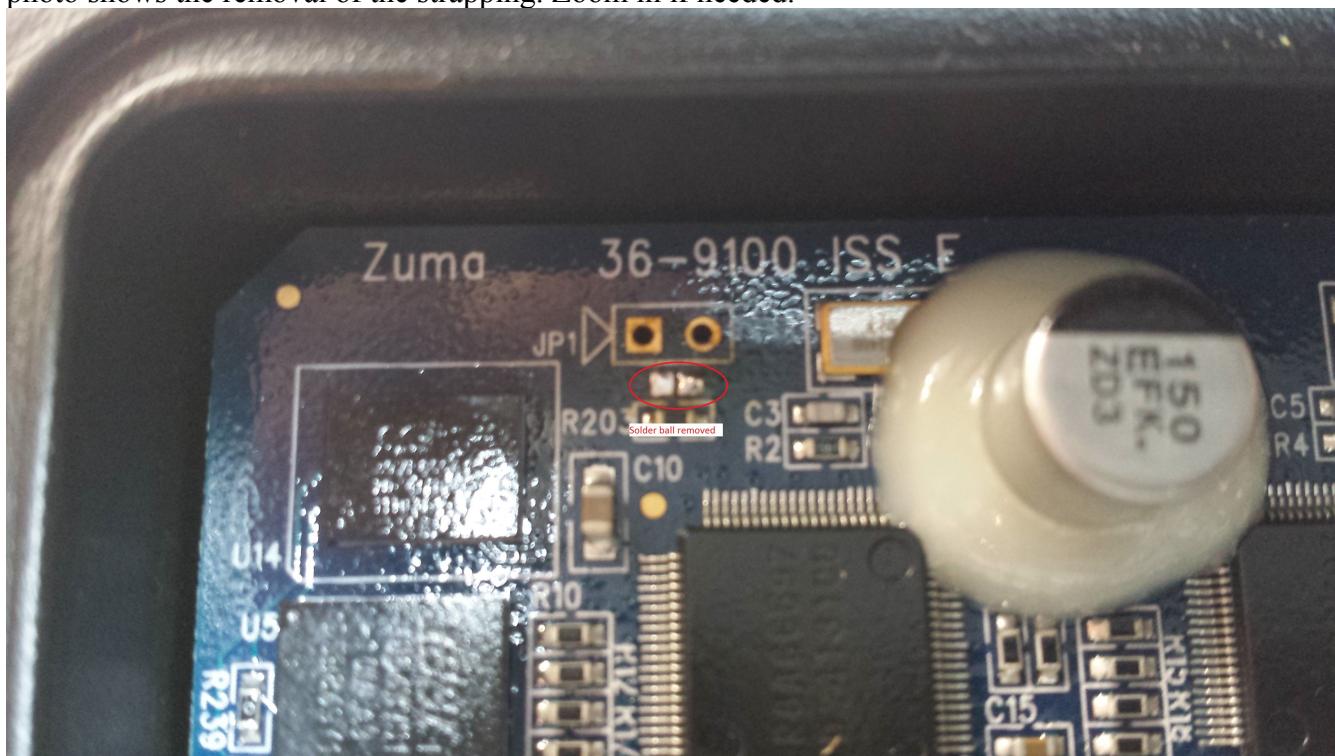
FDT Programming instructions

During development a user may encounter a hardware platform that becomes unresponsive. The first counter measure would be to enter the boot loader mode and erase the internal data of the system. If the hardware cannot connect through the USB port in either main running mode or boot loader mode, the user will have to program the hardware at a lower level. The default method is to access the hardware through the serial port.

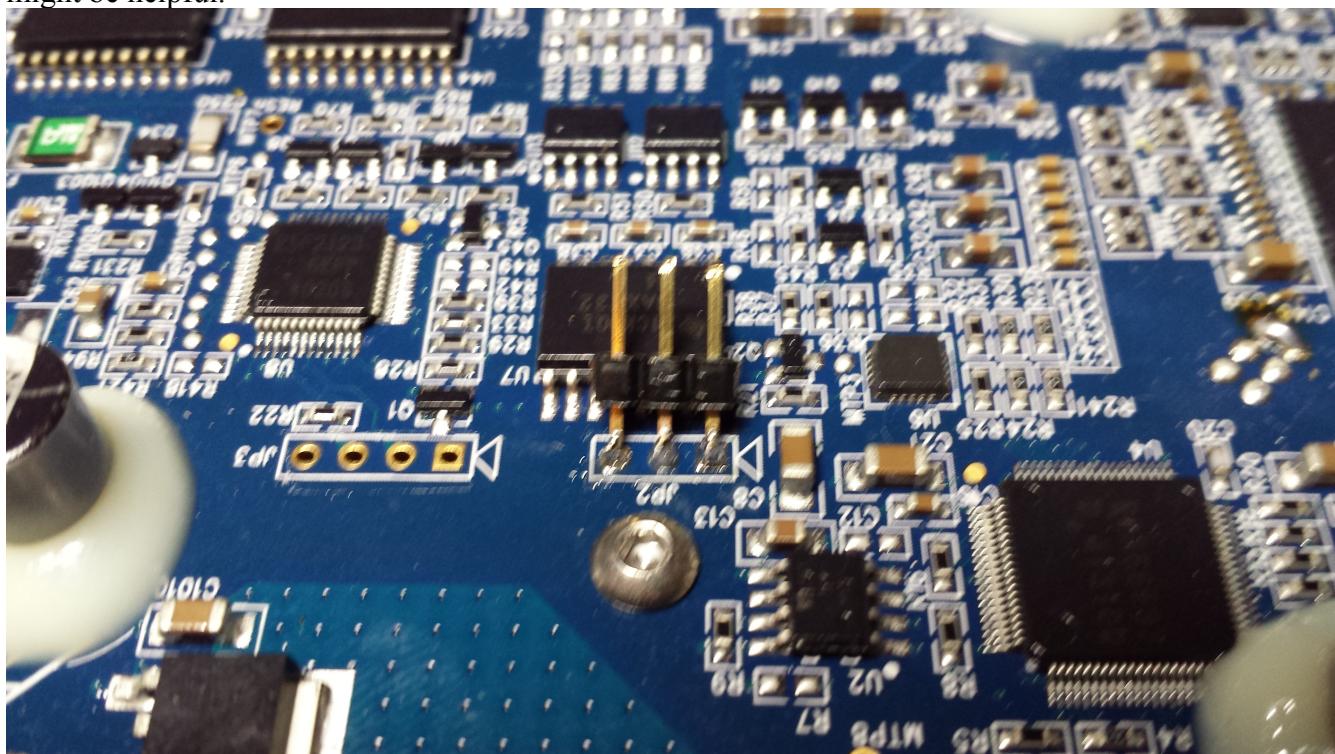
In order to place the hardware in a state where serial port access is possible, the JP1 strapping must be removed. Remove the backing plate from the unit and find the following circuit diagram. Zoom in if needed.

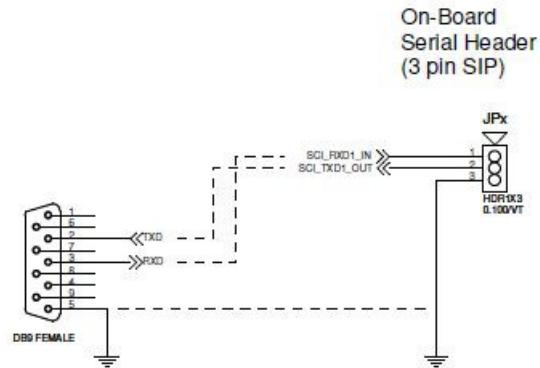


The 2 locations of interest are JP1 and JP2. JP1 is the strapping that controls the FDT mode. JP2 are the pins of the serial port. In order to program the hardware, one must remove the strapping. The following photo shows the removal of the strapping. Zoom in if needed.

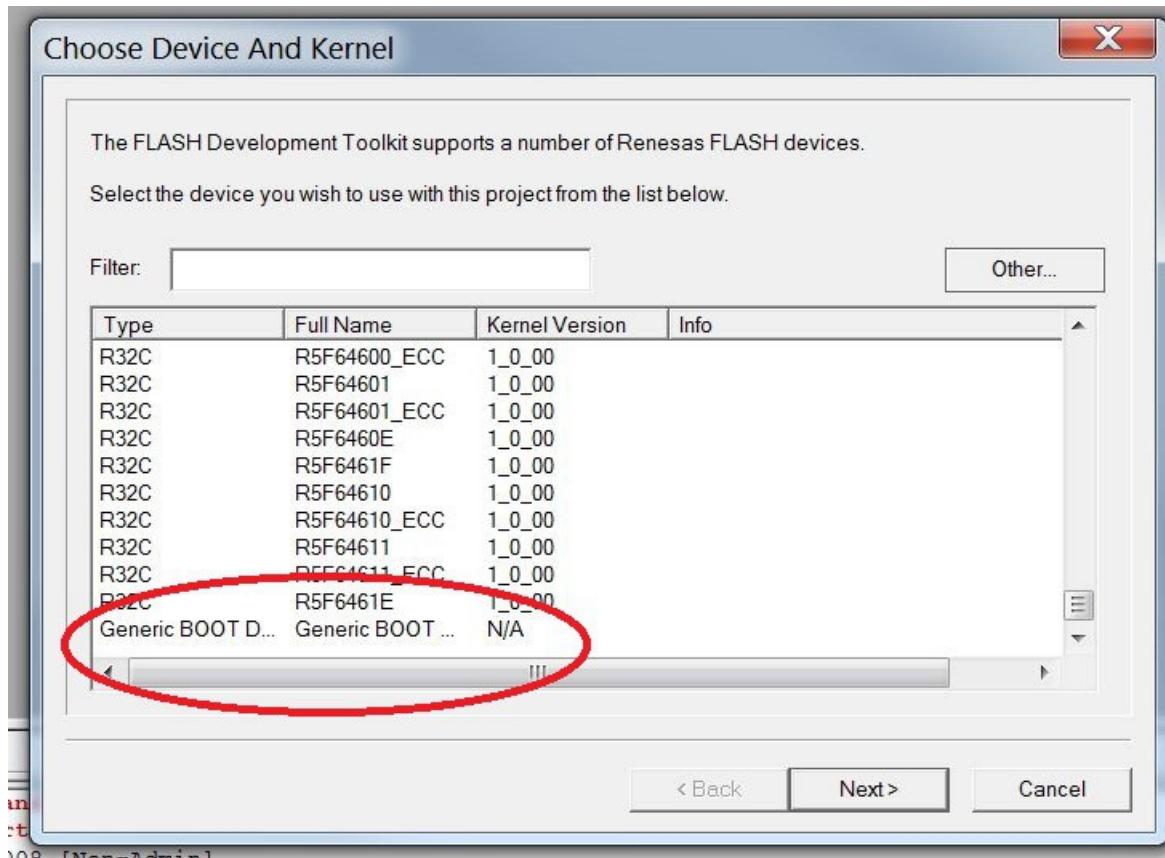


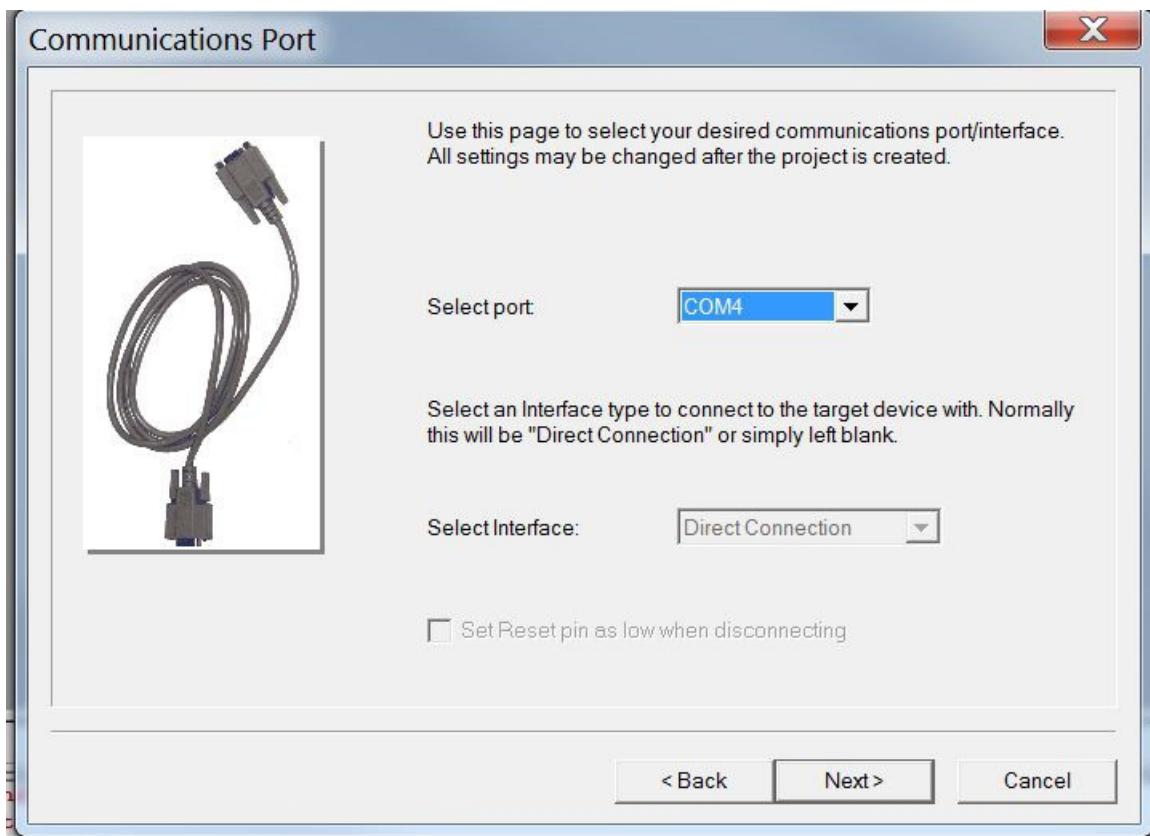
Connect the wiring of the serial port to the hardware. Soldering wires is OK but a connector might be helpful.



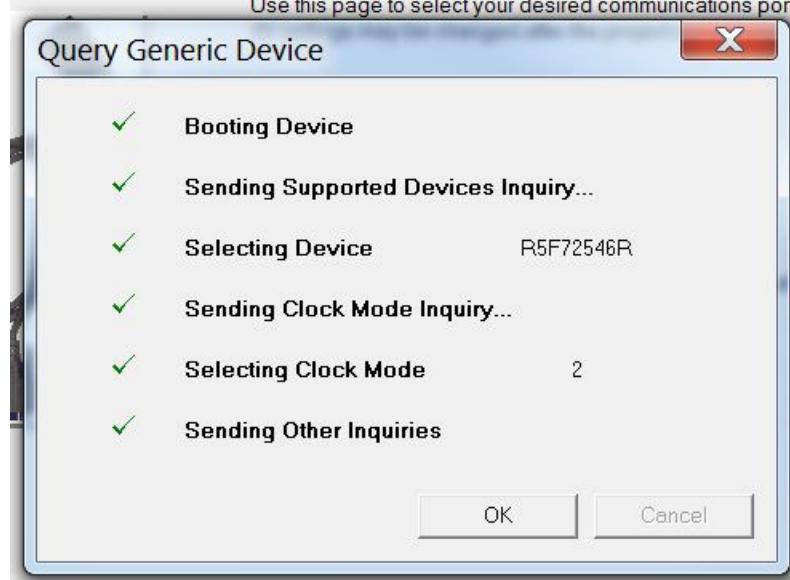


Run the FDT program and connect to the system using the “Generic Protocol C”.





If successful, you should observe green checks in the following dialog.

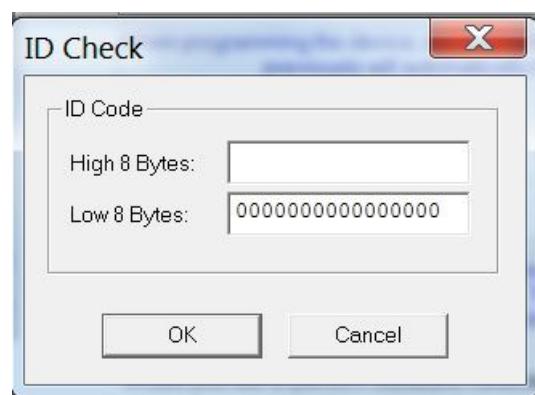
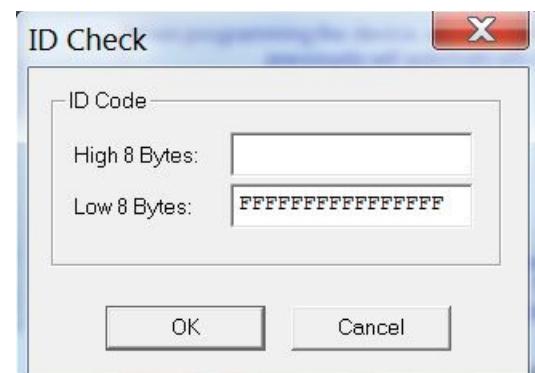


The device may be different and this is not a problem.

The FDT may ask you for an ID check with the following dialog:



If this is a new system, the all F's in the Low 8 bytes will generally work. If the hardware had a programming problem, all F's should work as well. For a programmed system, the ID check would be all 0's. In all cases, delete the "High 8 Bytes" and use all F's or all 0's in the "Low 8 Bytes". Try both all 0's or all F's until you have a successful connection.



A successful connection is expected before you can continue.

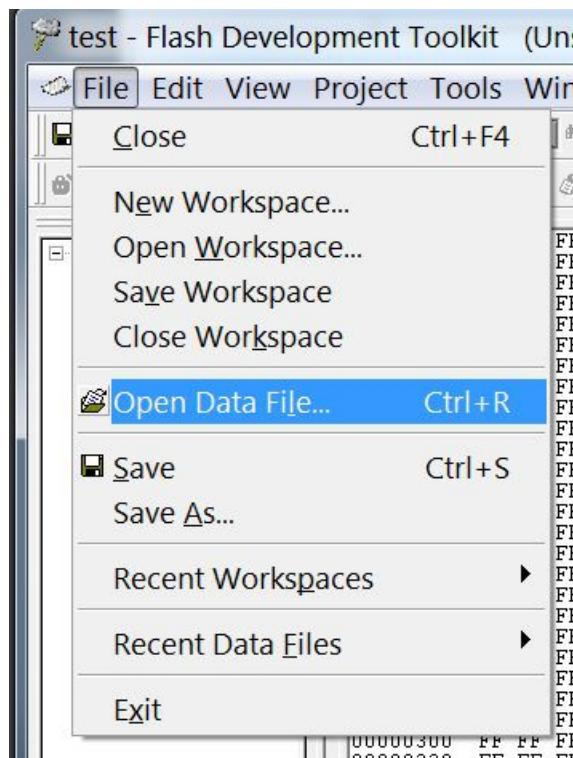
2 files must be downloaded (SH725xxB.mot and hardwareloader.bin).

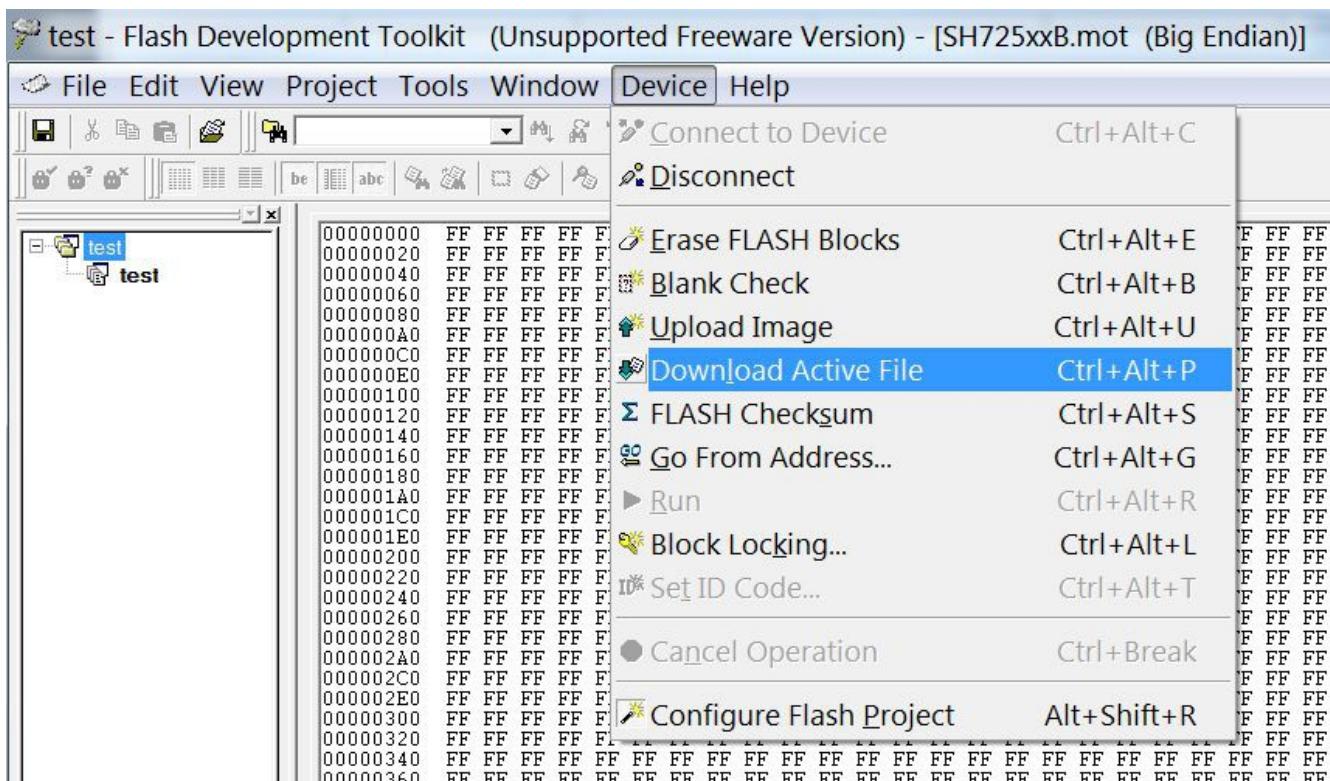
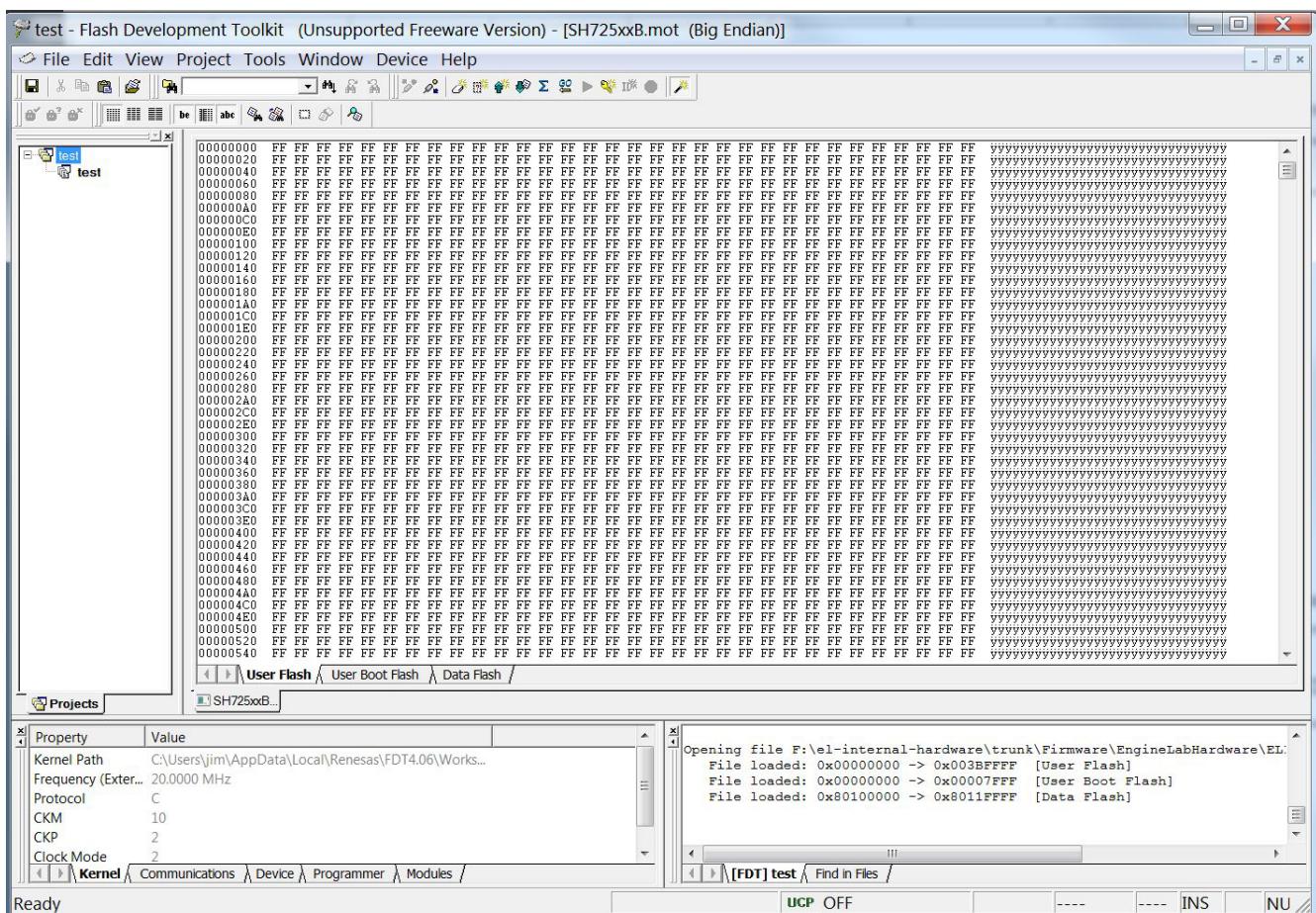
The location of the file may be different depending on the hardware platform and version that you have. Find JP1 on the board and you will see “ISS X”, where X=A,B,C,D,etc The X is the indication of the board revision. Zuma is the name for Infinity-10 and this system has 2 connectors with 129 pins. The Venice or Infinity-8 is the smaller system with one connector and 80 pins. The path to the files on the el-public-qa site are here: [\\el-public-qa\\trunk\\Software\\InfinityHardware\\Firmware\\Bootloader](http://el-public-qa/trunk/Software/InfinityHardware/Firmware/Bootloader)....

Assuming I wanted the bootloader for a Infinity-10 (2 connector 129 pin) “ISS E” I would find version 8240 here: \\el-public-qa\trunk\Software\InfinityHardware\Firmware\Bootloader\Infinity_10 (Zuma2)\SH72544R\Bootloader\RevE\8248\SH725xxB.mot.

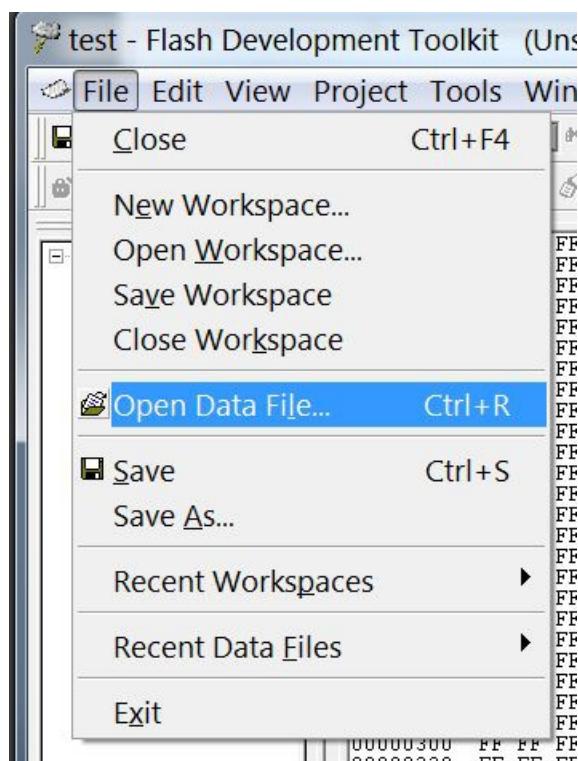
The hardware loader must also be programmed for the board. This would be found here: \\el-public-qa\\trunk\\Software\\InfinityHardware\\Firmware\\Bootloader\\Infinity_10 (Zuma2)\\SH72544R\\HardwareLoader\\HardwareLoader.bin.

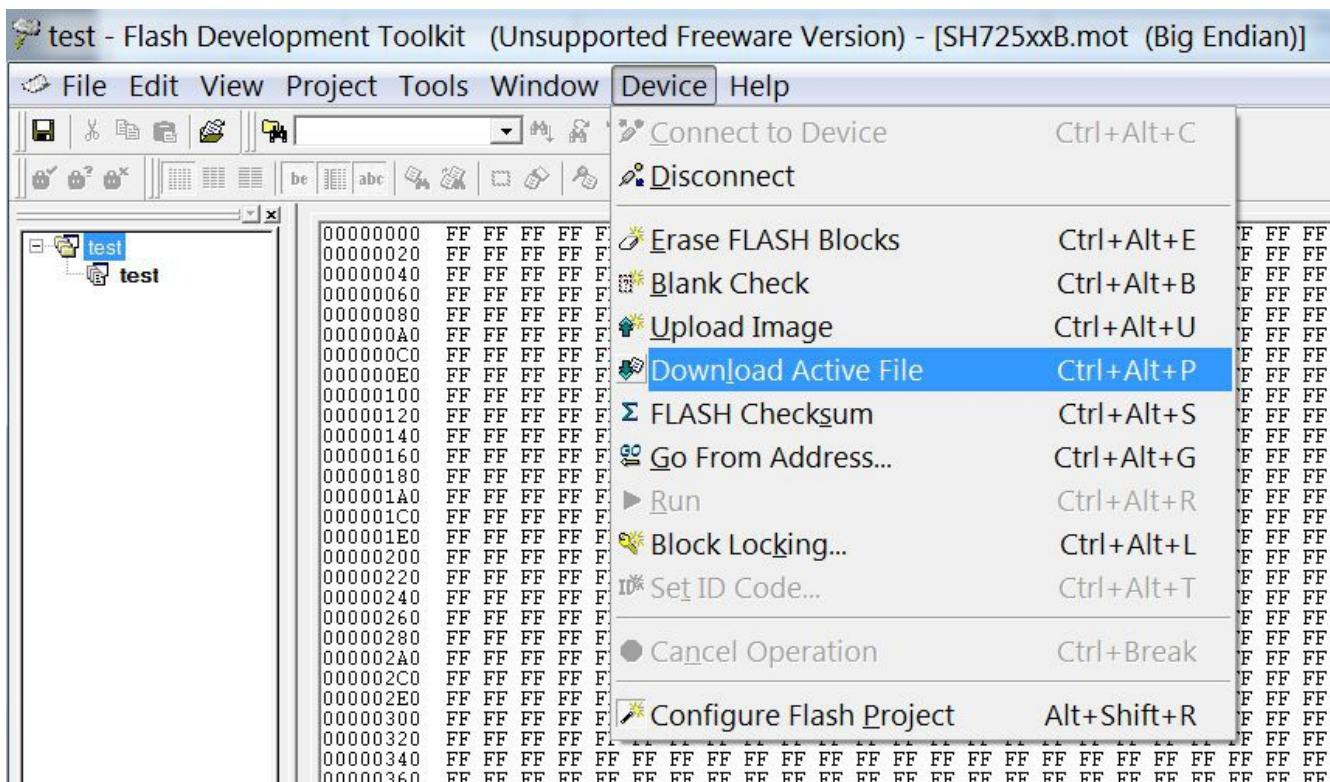
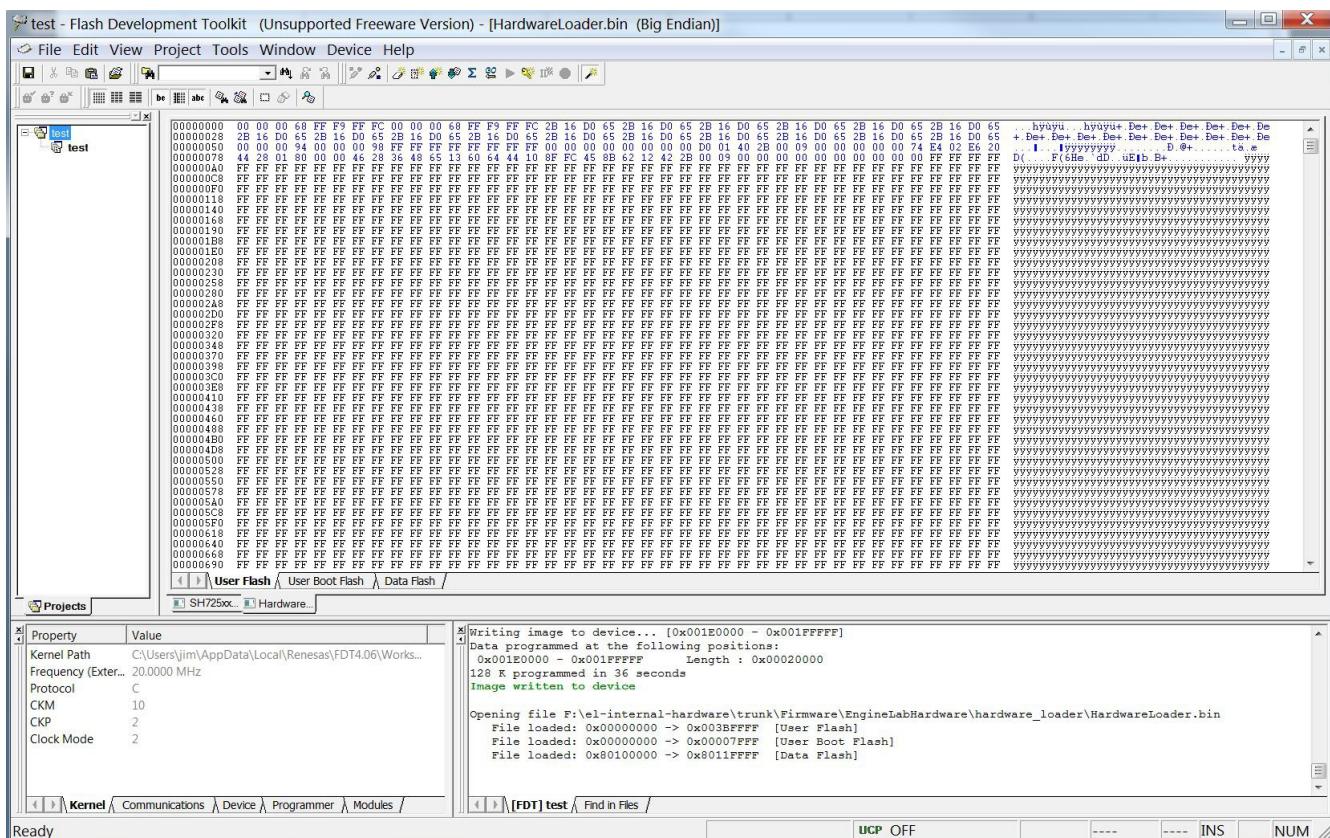
NOTE: If you program the wrong file, you will have problems. When you connect with the FDT the files will be deleted from the hardware and you will have to reprogram both files again.





```
Processing file :"F:\el-internal-hardware\trunk\Firmware\EngineLabHardware\EL129ECU\bootloader\8291\SH725xxB.mot"
[Data Flash] - No Data Loaded
Operation on User Flash
Loaded the Write operation module
Writing image to device... [0x001E0000 - 0x001FFFFF]
Data programmed at the following positions:
 0x001E0000 - 0x001FFFFF      Length : 0x00020000
128 K programmed in 36 seconds
Image written to device
```





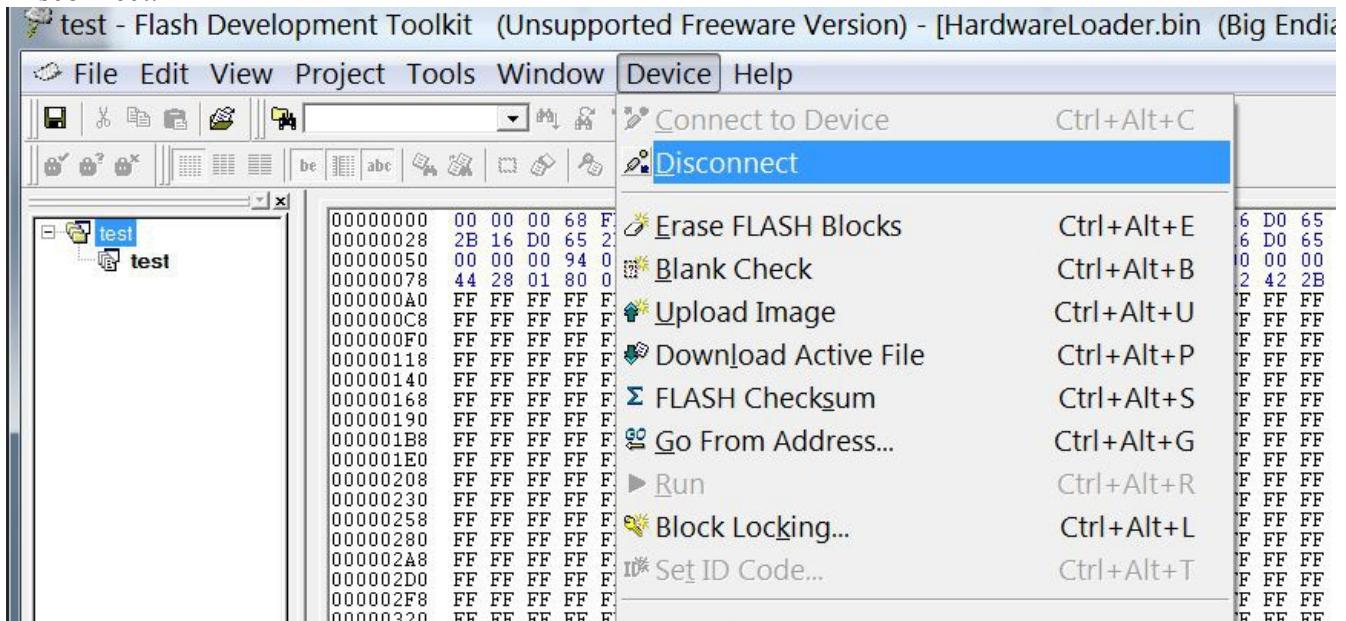
```

x| Loading image file : 'F:\el-internal-hardware\trunk\Firmware\EngineLabHardware\hardware_loader\HardwareLoader.bin'
x| Loading image file : 'F:\el-internal-hardware\trunk\Firmware\EngineLabHardware\hardware_loader\HardwareLoader.bin'
[Data Flash] - No Data Loaded
Operation on User Flash
Loaded the Write operation module
Writing image to device... [0x00000000 - 0x000000FF]
Data programmed at the following positions:
0x00000000 - 0x000000FF Length : 0x00000100
256 Bytes programmed in 0 seconds
Image written to device

```

[FDT] test / Find in Files / UCP OFF ---- INS NUM

Disconnect.



Make sure to solder the JP1 strapping again. If all goes well, a USB connection should be possible.

