

Programming the Actel® M1A3P evaluation board with the Cortex™-M1 processor and using the RealView® Microcontroller Development Kit MDK featuring the Keil µVision® 3 IDE:

version 1.0



Introduction:

This application note describes the techniques to install the ARM Cortex-M1 soft core into the Actel ProASIC3® M1A3P-DEV-KIT-SCS evaluation board using the Actel FlashPro® utility.

Once this is accomplished, you can write, run and debug your C and C++ programs for the Cortex-M1 using RealView MDK featuring the Keil µVision3 IDE. MDK version 3.22e and later contains support for this board. You can obtain the evaluation version of this software from www.keil.com. The evaluation or the full version of MDK® contains two example projects to run on the M1A3P. These project files are contained in C:\Keil\ARM\Boards\Actel\M1A3P.

Purpose of this application note: why we are doing this.....

The M1A3P® board comes from the factory pre-programmed with a proprietary JTAG configuration. This means that standard JTAG debuggers such as used by Keil µVision3 will not work. The solution is to reload the FPGA with a slightly different version of the Cortex-M1 core that supports standard JTAG debugging. This is easy to accomplish and works perfectly.

Once this is done, the Realview tools will then run “out of the box”. You can then run the two Keil examples as described starting on page 4.

The Necessary Steps:

- 1) Get the new Cortex-M1 image from Actel that has the standard JTAG interface implemented.
- 2) Get FlashPro™ from Actel. This is Actel’s free FLASH programmer program.
- 3) Install the USB drivers for FlashPro.
- 4) Program new image into the Actel FPGA
- 5) Test with RealView MDK and Keil µVision3 running the example program Blinky and TrafficLight. You can use the evaluation version of MDK.

1) Get new Cortex-M1 image:

The current image filename is socTop_KEIL.stp and is available with this document. If you wish to see if there is a later version, please contact your Actel representative. This is the image file you program the ARM Cortex-M1 soft core into the M1A3P board using the PC program FlashPro available for free from the Actel website. FlashPro connects to the M1A3P via a USB port.

2) Get FlashPro from Actel:

FlashPro is a free program provided by Actel and is used to program their FPGAs. The filename is FlashPro83.exe and is available from the Actel website. www.actel.com and search for *flashpro* and install it on your computer. You will need to register yourself on the Actel website to get it but there is no charge. If you installed the Actel Libero® software on your computer – FlashPro is included and can be accessed from there. You can use an older version than 8.3. FlashPro by itself does not need a license key.

3) Install the USB drivers:

- 1) Power the Actel M1A3P board
- 2) Connect the USB port on the M1A3P labeled **USB PROG** to your computer.
- 3) Install the USB drivers in the usual way. There will be two drivers loaded: one for the FlashPro firmware loader and the second for the programmer. Your FlashPro installation already contains the USB driver software.
- 4) Leave the USB cable connected for the next steps.



4) Program the updated Cortex-M1 image into the Actel FPGA:



- 1) **Start FlashPro:** Click on the FlashPro icon. FlashPro will start up.
- 2) **Create a new project:** In the FlashPro window click on New Project and enter a project name. I chose **Keil** as shown in Figure 1. Click on OK.

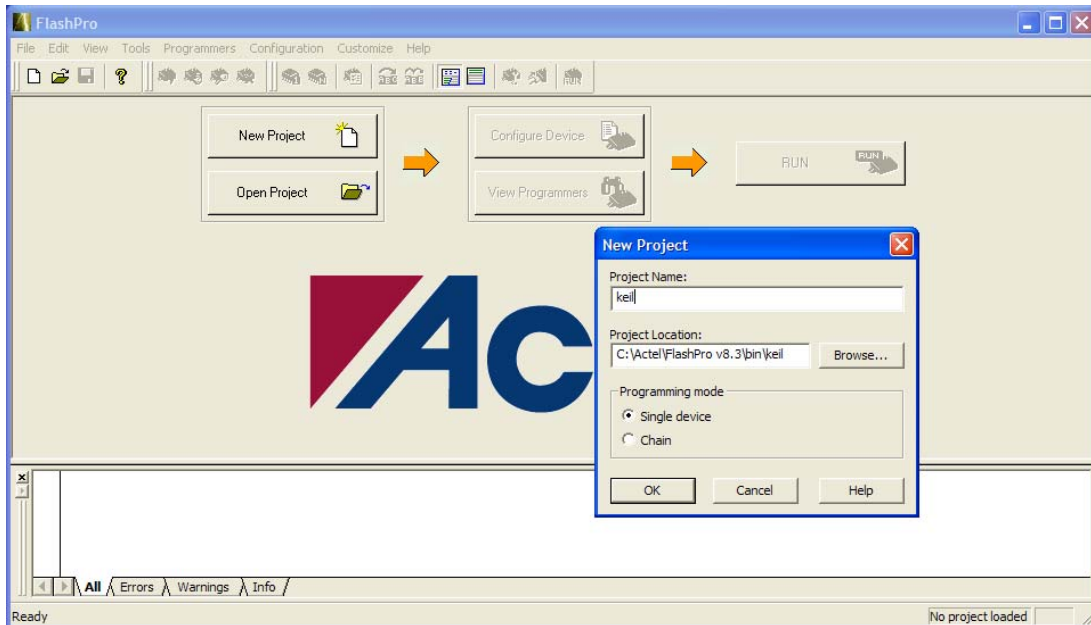


Figure 1 FlashPro “New Project”

- 3) Once the project has been created the FlashPro window will look like Figure 2. Note FlashPro has detected the M1A3P board and its USB port.

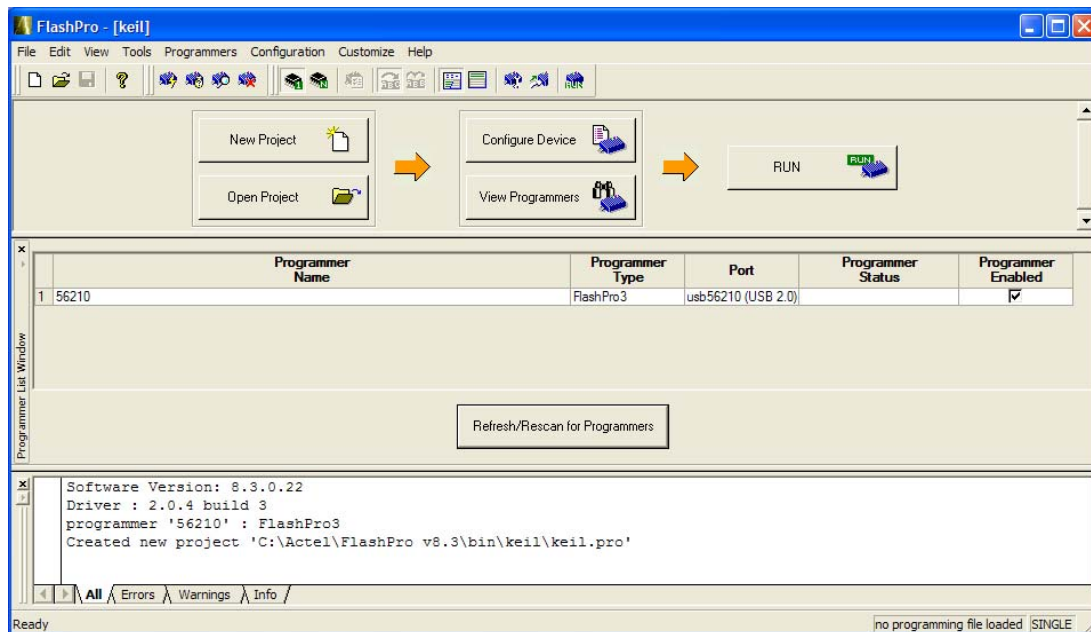


Figure 2 “New Project” completed and programmer detected.

- 4) **Enter the new Cortex-M1 image:** Click on Configure Device.
- 5) Select Browse and find and select the file socTop_KEIL.stp. Figure 3 will be the result.

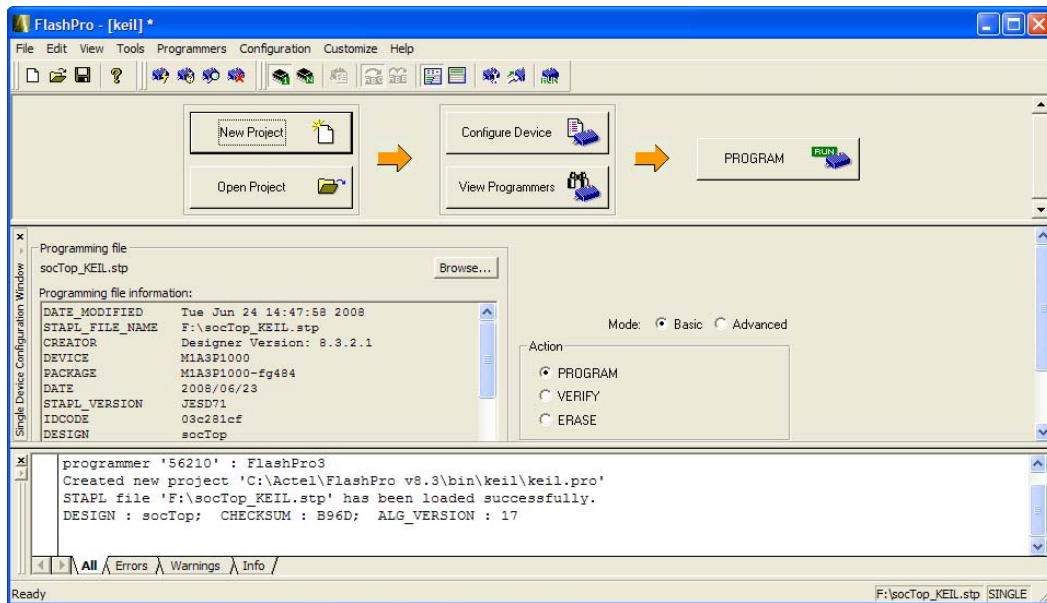


Figure 3 Cortex-M1 image loaded into FlashPro

- 6) **Program the FPGA:** Click on PROGRAM and FlashPro will start programming the FPGA. Ignore the error about the VPUMP.
- 7) When the programming is successfully completed Figure 4 will be shown. The green RUN PASSED is a very good sign as is "Program Passed" in the comments window.
- 8) The FPGA is now loaded with the Cortex-M1 and ready to have a program loaded and run. We will use RealView MDK with Keil μ Vision3 to do this. Close FlashPro, remove the USB cable and power cycle the Actel board. You will not have to repeat these steps again unless you load another image into the FPGA.

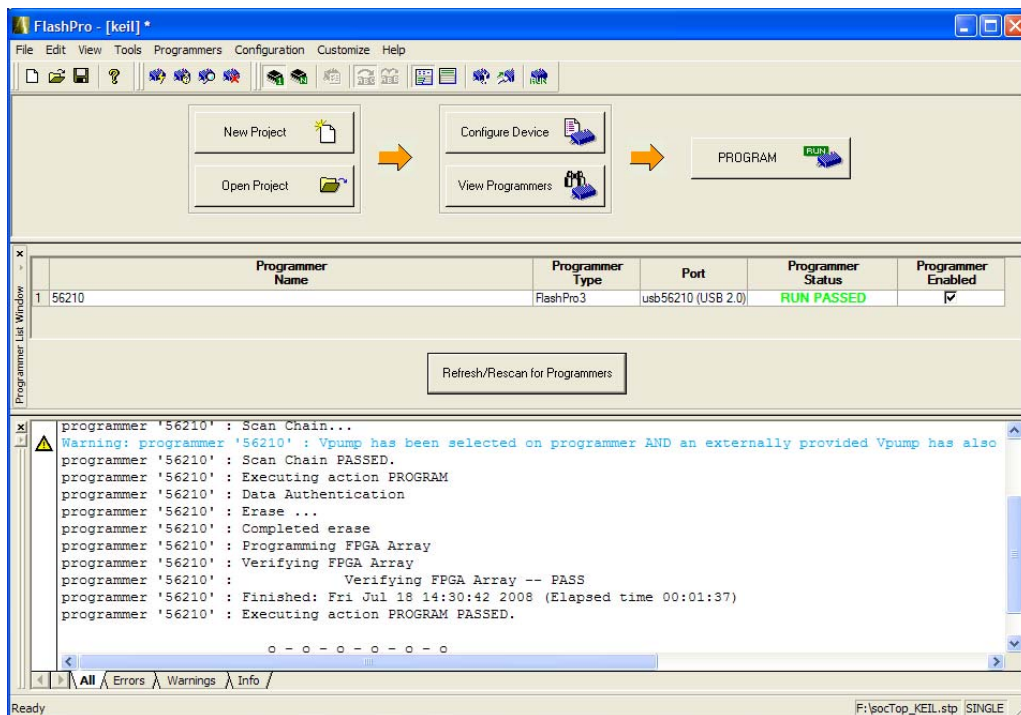



Figure 4 Cortex-M1 image loaded into the Actel FPGA Flash memory.

5) Running the Keil example programs “Blinky” and “TrafficLight”:

We will now program two example programs provided by Keil into the Actel board and run them on the Cortex-M1 device you programmed into the FPGA. You can use the free evaluation copy of the Keil tools to do this.

Starting and Configuring μ Vision3:

- 1) Install RealView MDK using the default directory of C:\Keil.
- 2) Click on the μ Vision3 icon.  μ Vision3 will start up.
- 3) Connect a ULINK2 or ULINK-ME between your computer and P3 on the Actel M1A3P board.
- 4) Open the Actel project: in μ Vision3 select Project/Open Project and go to the directory C:\Keil\ARM\Boards\Actel\M1A3P\Blinky. Open Blinky.Uv2 (this is the μ Vision3 project file).

NOTE: You can run the Blinky program in either the RAM or Flash devices on the M1A3P board.

This is selected by:

- a) the drop-down menu in μ Vision3 (see Figure 5) **AND**
- b) SWITCH 9 on SW2 on the Actel board. (ON = Flash and OFF = RAM). See Fig 6.

Please set both of these to Flash operation for this tutorial. You can try the RAM version later.

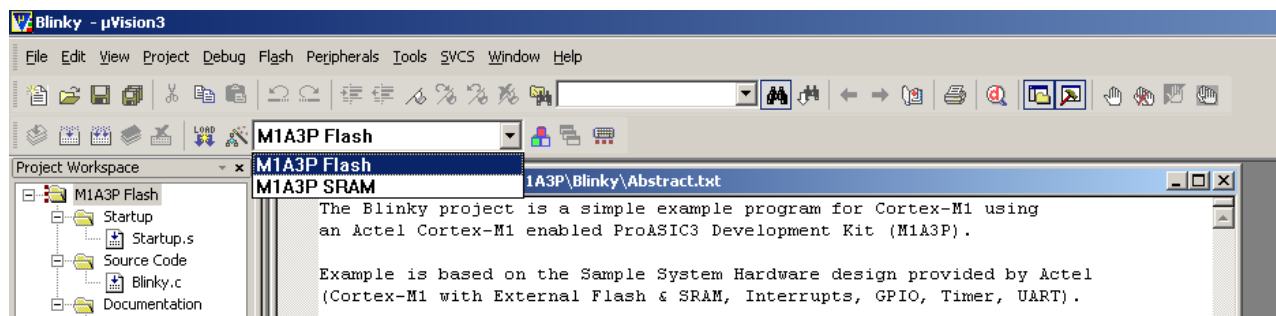


Figure 5 μ Vision3 front menu showing Flash and RAM selection menu.

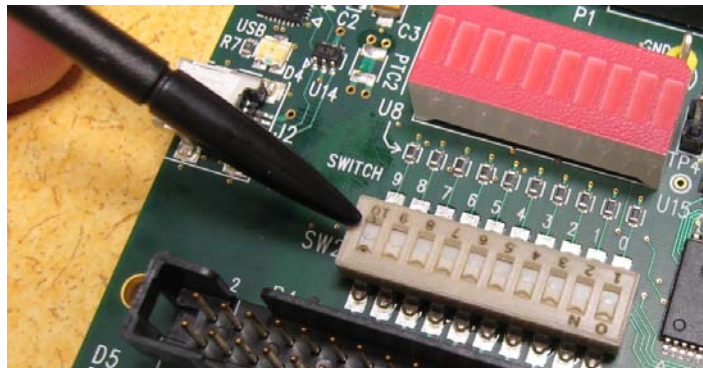



Figure 6 SW2 is set for Flash operation in this photo.

Note: SWITCH 9 is also labeled as “10” on the switch.

Checking the JTAG communication channel:

We can easily test the JTAG connection to make sure this is running correctly.

- 1) Have μ Vision3 running and make sure a UINK2 or ULINK-ME is connected to P3 on the Actel board.
- 2) Click on the Options for target icon.  Or open the Project menu and select “Options for target “M1A3P Flash”.
- 3) Select the Debug tab.
- 4) Select Settings in the upper right hand side next to ULINK CoreSight Debugger.
- 5) Figure 7 will open up showing the existence of the CoreSight debugging module inside the Cortex-M1.
- 6) If you do not see this or an error such as JTAG communication failure – check your connections, make sure the board is powered and all settings in μ Vision3 are correct. Try power cycling the board.
- 7) To refresh this window – close it and reopen it. Or change the Port: from JTAG to SW and then back again.
- 8) Close these windows and return to the μ Vision3 main window.

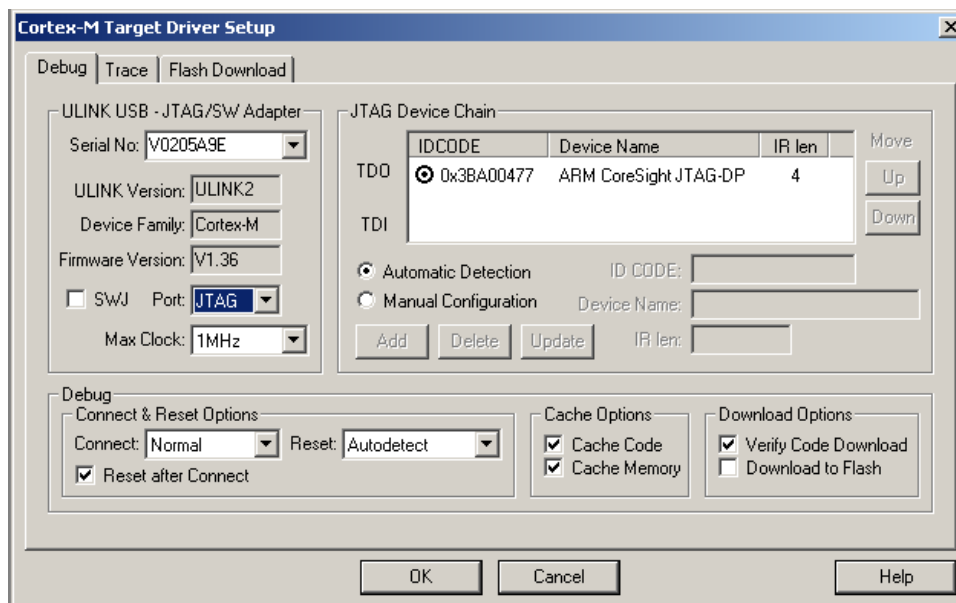








Figure 7 Testing the JTAG communication channel.

Compiling, loading and running the Example program Blinky:





- 1) **Compile:** Click on the Build All icon to compile the ready made Blinky program.  The source files will compile with no errors.
- 2) **Load:** Click on the load icon to program the executable into the Actel board's Flash. 
Note: you do not need this step if you plan on running the program in RAM.
- 3) **Debugger:** Start the debugger by clicking on its icon.  The Actel Flash will program and verify.
- 4) **Run:** Click on the Run icon. 

The LED array U8 will sweep from one end to the other.

SW2: SWITCH 1 starts and stops the LEDs and SWITCH 2 changes the speed.

Click on STOP icon  and exit debug mode by clicking on the debug icon. 

Compiling, loading and running the Example program TrafficLight:

- 1) **Open the Project:** Click on Project/Open Project and select TrafficLight.Uv2 in the directory Actel\M1A3P\TrafficLight.
- 2) **Compile:** Click on the Build All icon to compile the ready made TrafficLight program.  This will compile with no errors.
- 3) **Load:** Click on the load icon to program the executable into the Actel board's Flash.  The Actel Flash will program and verify. **Note:** you do not need this step if you plan on running the program in RAM.
- 4) **Debugger:** Start the debugger by clicking on its icon. 
- 5) **Run:** Click on the Run icon. 

The LEDS will slowly change simulating a traffic light.

The Actel Quick Start Guide in Step 4 describes a graphical program that interfaces to the traffic light example. This PC executable is traffic_Light.exe and also be found on the Actel CD that came with the kit.

6) What else can I do with this system ?

Plenty – you can use the examples as a template for your own project. RealView MDK is very easy to use and has the most advanced ARM compiler “under the hood” capable of the most challenging C or C++ programs.

8051: you can also install the Actel 8051 soft core on this board. Keil offers PK51 – the industry leading development suite to use. PK51 uses the same μ Vision3 IDE as in RealView MDK. You can put the Cortex-M1 and the 8051 core into the same FPGA and have them communicate with each other and the outside world.

Copyright © 2008 by ARM, Ltd.

Comments: bob.boys@arm.com 408-548-3023 (California)

Resources: www.actel.com/products/mpu/CortexM1/
www.keil.com/arm/mdk310.asp
www.arm.com/products/CPUs/ARM_Cortex-M1.html

