University of British Columbia
Electrical and Computer Engineering

CV-8052 Soft Processor in the DE10-Lite Board: Getting Started Guide

Introduction

This document provides step by step instructions on how to build, load, and run user code in an 8052 soft processor configured for the Terasic DE10-Lite board outfitted with a MAX10 FPGA.

Configuration Steps

These configuration steps need to be executed **only once**:

1) Download the Quartus Prime Lite project files for the 8052 soft processor from the course WebCT page. All the required files are compressed in the file ***'MAX10_8025_FD_ADC.zip'***. Decompress all files in a folder of your choice.

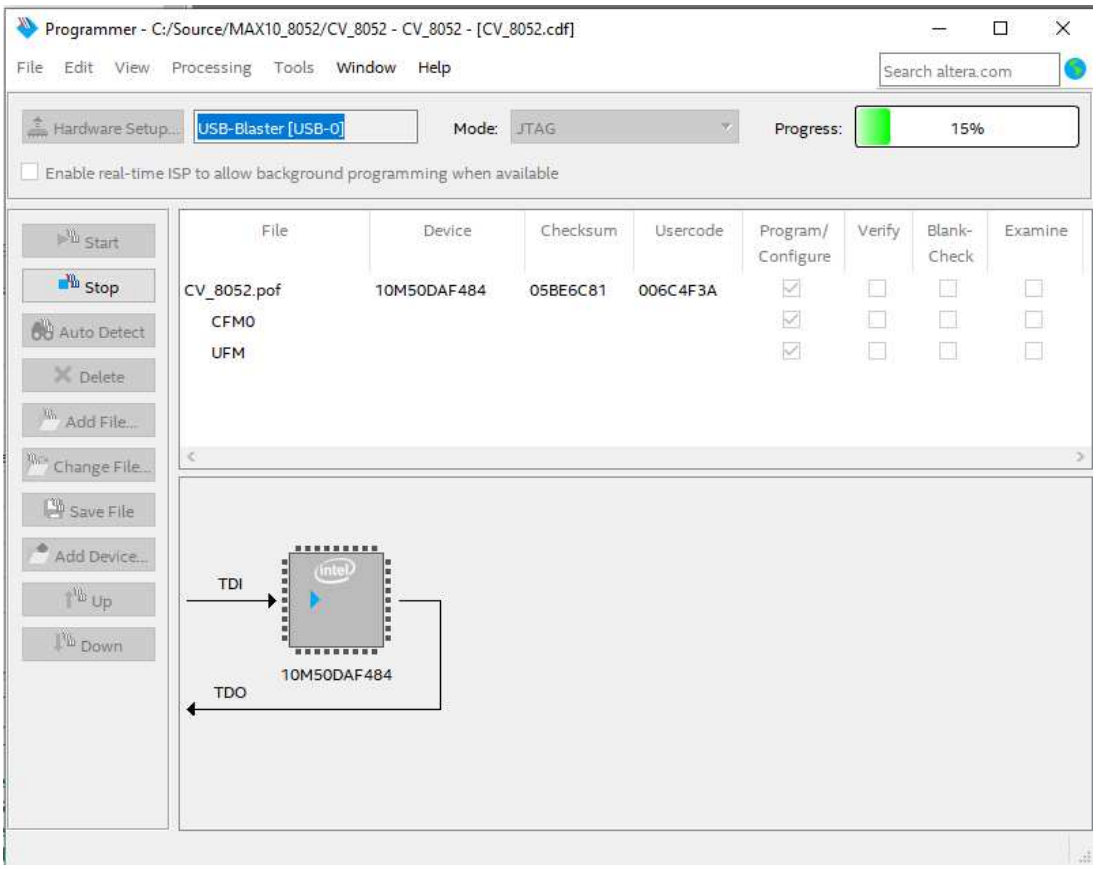
2) Start Quartus Prime Lite (version 16.1 or up[1]). Open the project ***'CV_8052.qpf'***.

3) You may skip this step if you want, as the configuration file '***CV_8052.pof'*** is already included in the project. Otherwise, in Quartus Prime Lite click ***'Processing'*** → ***'Start Compilation'***. It may take a few minutes for Quartus Prime Lite to finish synthesizing the CV-8052soft processor.

4) Connect the Altera DE10-Lite to the computer using the USB cable.

5) To download the configuration file to the Altera DE0-CV board click ***'Tools'*** → ***'Programmer'***. In the line where the file ***'CV_8052.pof'*** is, make sure the selection box ***'Program/Configure'*** is checked. Then press the ***'Start'*** button. A moving progress bar shows that the Altera DE10-Lite board is being configured with the 8052 soft-core. Be patient. This usually takes several seconds.

[1] Tested successfully with Quartus Prime Lite versions 16.1 and 23.1. Support for the MAX10 family of FPGA devices must be installed in Quartus Prime Lite edition.

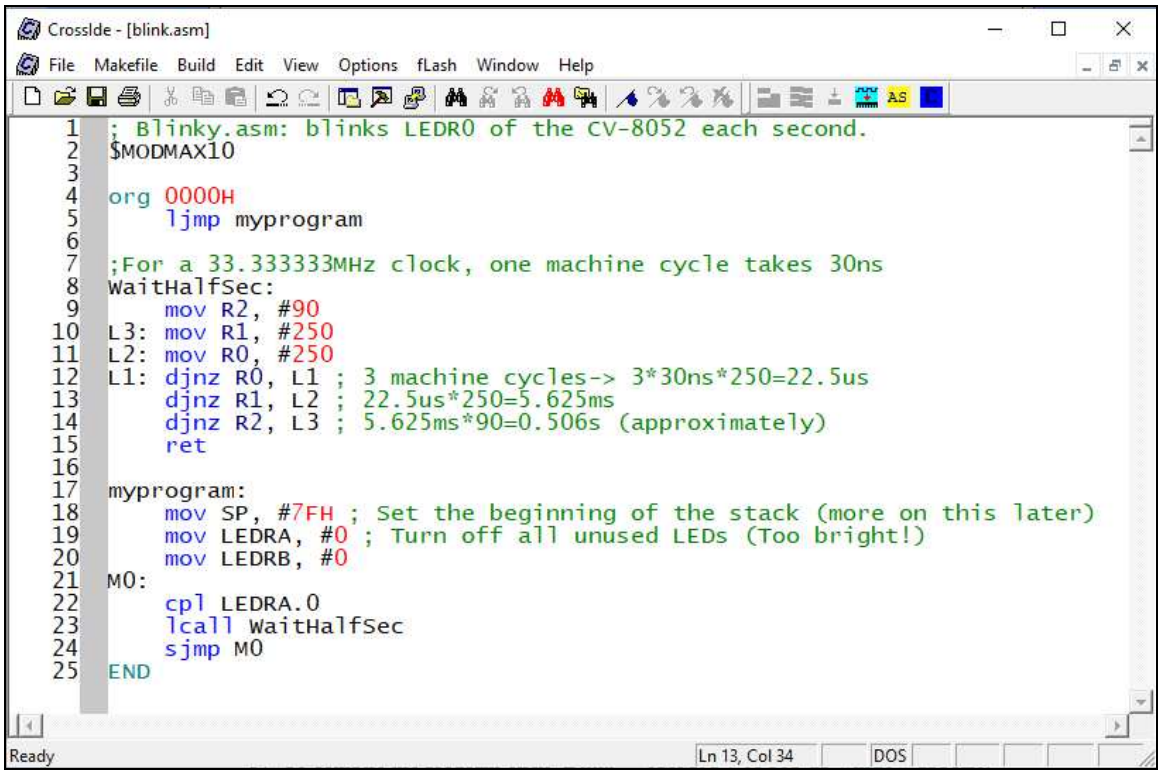


Compiling and Loading Programs into the CV-8052

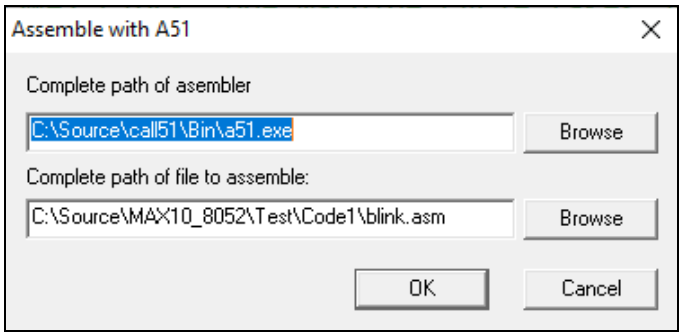
The CV-8052soft processor includes a boot loader that can be used to load and run user programs with the MAX10-8052. Before doing that, we need to write and compile our program:

1) Download and install Crosside from the course web page. Crosside is a text editor with assembly and C syntax highlighting. It also includes a module to transmit files to the CV-8052processor.

2) Start Crosside. Create a new assembly file by clicking ***'File'->'New'->'Asm source file'***. Type your assembly program and save it. The figure below shows an assembly program saved as ***'blinky.asm'***.



3) To compile the program click ***'Build'->'ASM51'***. A pop-up window appears. Click the first ***'Browse'*** button and find the program ***'a51.exe'*** in the ***'bin'*** folder of your CrosIDE\Call51 installation:



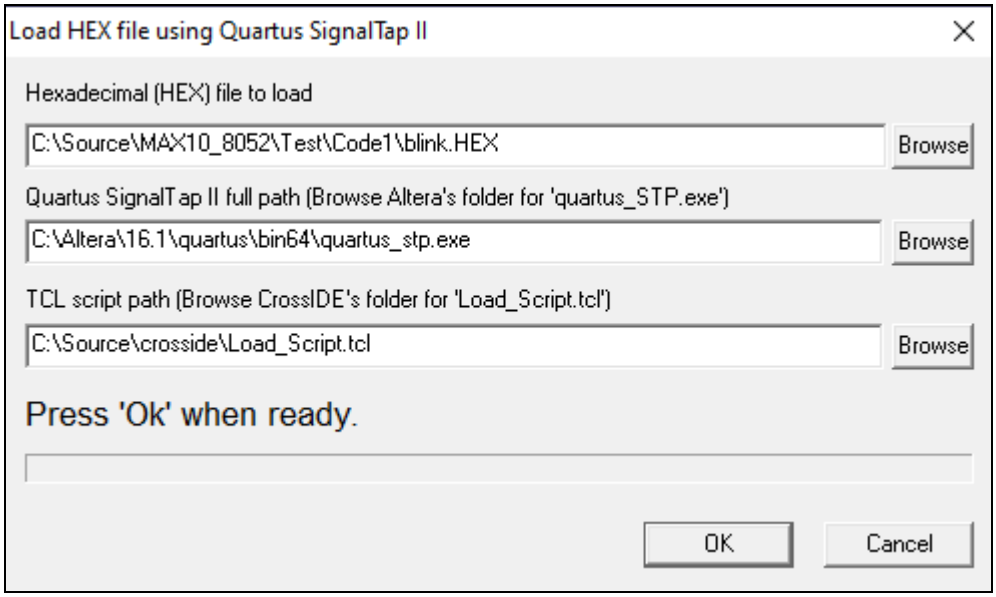
Click ***'OK'***. If your program has no errors you can proceed to the next step. Otherwise you need to fix the errors and try again.

4) After compiling your source file, a new file with the same name and extension ***'.HEX'*** is generated. This is the file we load or ***'flash'*** into the CV-8052 processor. To do that we need to activate the boot-loader in the CV-8052by following these steps:

1. Press and hold KEY1.
2. While holding KEY1, press and release KEY0.
3. Release KEY1.

You will know the boot-loader is running if LEDR0 is on and all other LEDs are off. Also, the message 'boot' will be briefely displayed on the 7-segment displays. In Crosside press ***'fLash'->'Quartus SignalTap II'***. There are three required fields you'll need to fill with valid information before proceeding further:

1. The hexadecimal (HEX) file created by the compiler.
2. The full path of the Quartus Prime Lite SignalTap program (quartus_STP.exe). This program is included with Quartus Prime Lite in a location similar to the one showed in the figure below.
3. The location of the Tcl script used by ***'quartus_STP.exe'***. This script is included with CrossIDE as shown in the figure below.



Additionally, the computer needs to be connected to the DE10-Lite board using a USB cable. Press ***'Ok'***. Wait until the program finishes. Check the report window in CrossIDE; the bottom of the message displayed should look like this:

```
.
.
Connecting to USB-Blaster [USB-0] @1: 10M50DA(.|ES)/10M50DC (0x031050DD)
Sending HEX file 'C:\Source\MAX10_8052\Test\Code1\blink.HEX'...
Sending command to copy hex file to 'flash' memory...
Done.
```

5) KEY0 is configured as the Reset button for the MAX10-8052. Press KEY0 and the loaded program starts running.

CV-8052Soft-Core Special Function Registers (SFRs)

The 8052 soft processor includes all the standard 8052 SFRs. Some additional SFRs were added to provide access to some of the resources in the Altera DE10-Lite board. These are the relevant SFRs:

SFR	Address	Description
HEX0	91H	Seven segment display 0
HEX1	92H	Seven segment display 1
HEX2	93H	Seven segment display 2
HEX3	94H	Seven segment display 3
HEX4	8EH	Seven segment display 4
HEX5	8FH	Seven segment display 5
LEDRA	E8H	LEDR0 to LEDR7 (bit addressable).
LEDRB	95H	LEDR8 and LEDR9.
SWA	E8H	Switches SW0 to SW7 (bit addressable).
SWB	95H	Switches SW8 to SW9.
KEY[2]	F8H	KEY1=KEY.1, KEY2=KEY.2, etc. (bit addressable).
P4	C0H	Input/Output port 4 (bit addressable).
P4MOD	DAH	Input/Output mode bits for port 4
P0MOD	9AH	Input/Output mode bits for port 0
P1MOD	9BH	Input/Output mode bits for port 1
P2MOD	9CH	Input/Output mode bits for port 2
P3MOD	9DH	Input/Output mode bits for port 3
P4MOD	C1H	Input/Output mode bits for port 4
ADC_C	A1H	Bit 7=1: Reset,0:Run; Bits 2 down to 0: Channel to read
ADC_L	A2H	ADC result low; bits 7 down to 0
ADC_H	A3H	ADC result high; bits 11 down to 8

Notice that LEDRA and SWA as well as LEDRB and SWB share the same address. Any writing to the corresponding address turns on/off the LEDs. Any reading from the corresponding address returns the state of the switches.

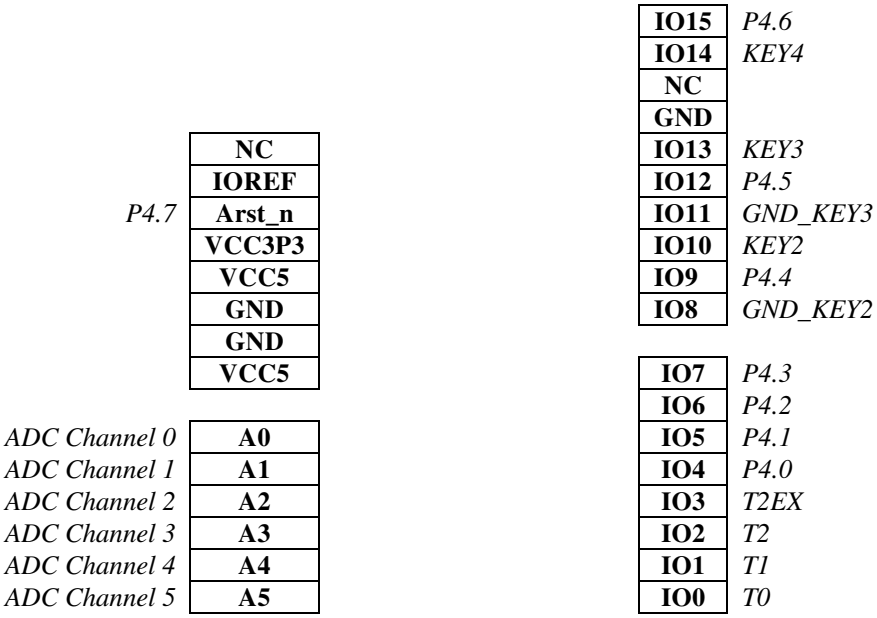
Some SFRs used by the loader and debugger are not listed above. Their addresses and functions can be found in the files "MAX_8052.h" if programming in 'C', or MODMAX10 if programming in assembly.

The ADC must be reset by setting bit 7 of ADC_C. After a brief delay, enable the ADC by resetting bit 7 of ADC_C. Bits 0 to 2 of ADC_C select the channel to read.

[2] KEY2 to KEY4 are assigned to pins in the arduino compatible connector. External push buttons need to be added between KEY4 and GND, KEY3 and GND_KEY3, and KEY2 and GND_KEY2. Writing zero to KEY2 sets GND_KEY to ground. Similarly, writing zero to KEY3 sets GND_KEY3 to ground.

Pin Assignments

All the standard 8052 I/O pins are assigned to the expansion headers of the Altera DE10-Lite board. By default all the port pins (P0 to P4) are configured as inputs. To configure any of the pins of a port as an output write 1 to the corresponding bit in the PxMOD register described above. For example to make P0.0 and P0.7 outputs and leave P0.1 to P0.6 as inputs, write 81H to P0MOD. These are the pin assignments:



JP2

P0.0	**1**	**2**	*P0.1*
P0.2	**3**	**4**	*P0.3*
P0.4	**5**	**6**	*P0.5*
P0.6	**7**	**8**	*P0.7*
P1.0	**9**	**10**	*P1.1*
5V	**11**	**12**	*GND*
P1.2	**13**	**14**	*TXD*
P1.4	**15**	**16**	*RXD*
P1.6	**17**	**18**	*P1.7*
P2.0	**19**	**20**	*P2.1*
P2.2	**21**	**22**	*P2.3*
P2.4	**23**	**24**	*P2.5*
P2.6	**25**	**26**	*P2.7*
P3.0	**27**	**28**	*P3.1*
3.3V	**29**	**30**	*GND*
P3.2	**31**	**32**	*P3.3*
P3.4	**33**	**34**	*P3.5*
P3.6	**35**	**36**	*P3.7*
INT0	**37**	**38**	*INT1*
P1.3	**39**	**40**	*P1.5*