Digital Security Lock System

Using VGA to display coordinated pixels

By Chris Barr & Jett Guerrero

Introduction

This project is a security lock that requires a password code to unlock a system. The locking system has a digital lock display to indicate the current condition, locked or unlocked. When the system locks or unlocks, a physical actuator opens or closes. The password code is resettable if one decides to change it.  
This project will involve a 7-Segement Display, 5 onboard Switches and a single pushbutton, and VGA output. This will be a comprehensive guide that will help lead any beginner level programmer to completing this project.

Note: This guide will be using Windows 10 to guide the user.

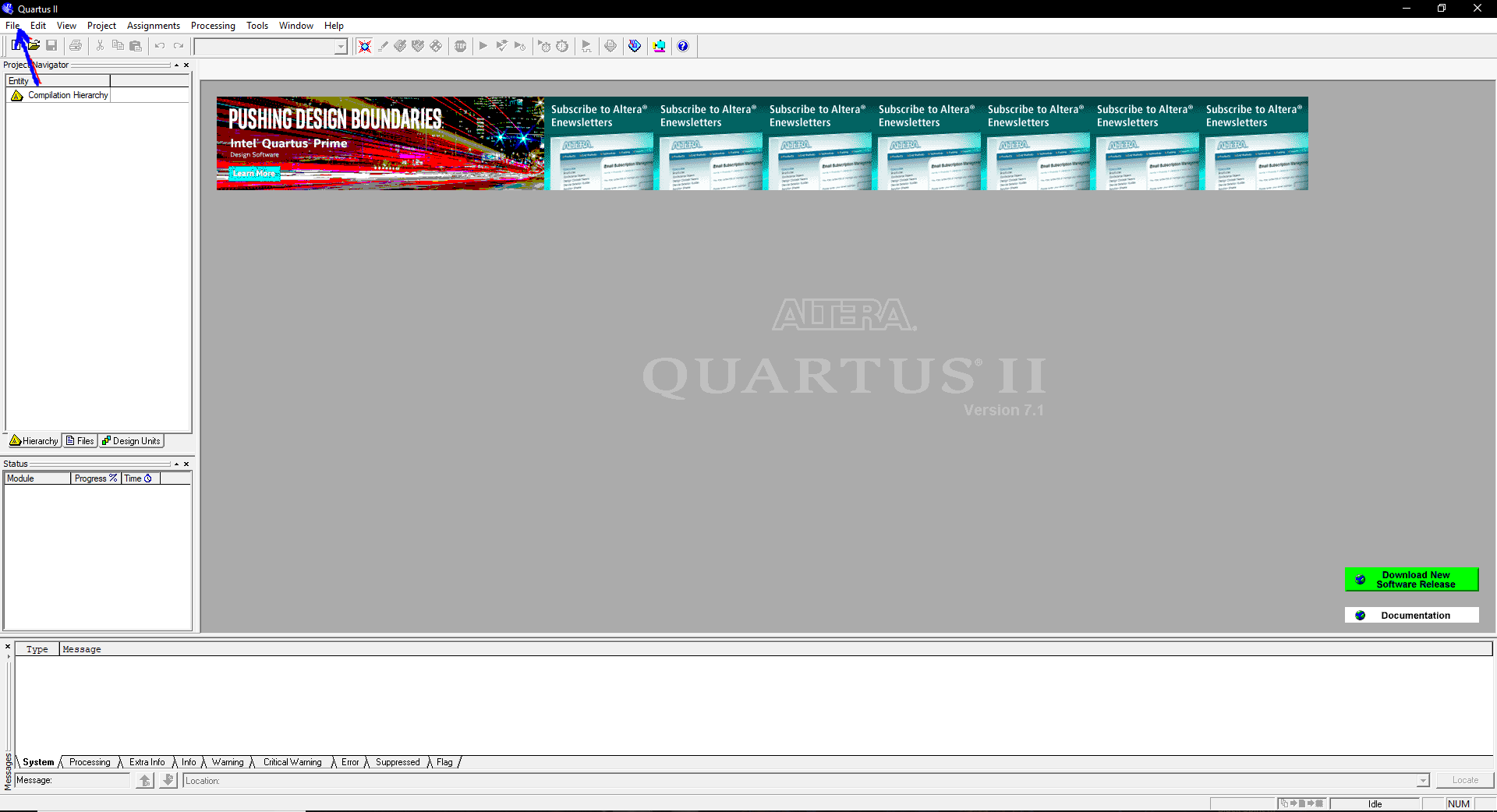
Software/Hardware Used  
\* Quartus II 7.1  
\* Cyclone II (EP2C35F672C6) DE2 Board  
\* USB Blaster (x64)

Display Info

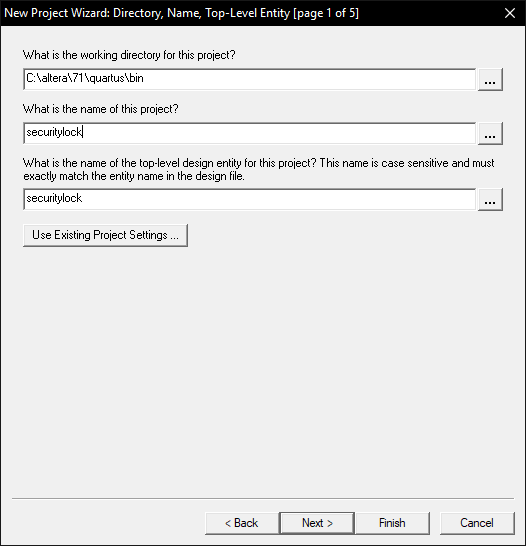
The password is to be entered using four switches on the DE2 board. The password values (0 or 1) is displayed on the seven-segment display. After the password code is entered, the right most push button (KEY0) must be pressed to unlock the system. If password is valid, the digital lock opens and turns green on the monitor. If invalid, the lock remains red and locked. If one wishes to change the password, one must first enter the valid password then switch on the left most switch (SW17) to enter a new password. In this mode, digital lock turns yellow.

Section 1: Quartus 7.1

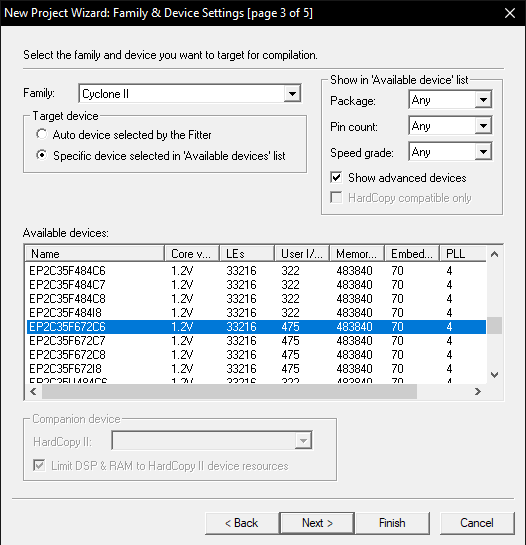
1. When Quartus 7.1 is opened, you will get this pop-up. Click on the **File** tab on the top-left of the screen, then hit **New Project Wizard…**.



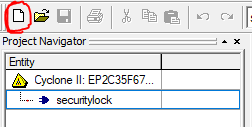
2. A smaller window will pop-up and tell you about the Project Wizard. Go ahead and click **Next**. Afterwards, you’ll be asked to find a directory for your project as well as a name. Click **Next** when completed as shown below.



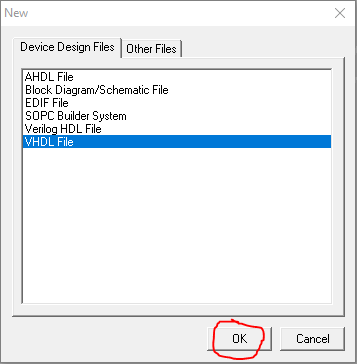
3. Page 2 of 5 will ask you to select any additional files you want to include into your project, we’re going to skip that and click **Next**. Here’s the important part, make sure you set your Family to **Cyclone II** and your Available Devices to **EP2C35F672C6**. Otherwise, check your boards manual to see which version you have of the board and select that under Available Devices.



4. Nothing more needs to be completed, so go ahead and hit that **Finish** button on the bottom right of that small window. Afterwards, you should see the name of your project listed under the Entity column as shown below. Now, **left-click the icon** shown from the picture circled in red.

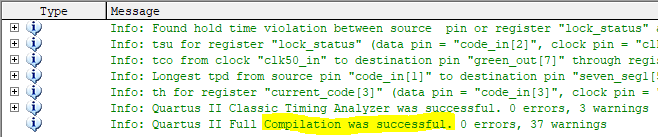


5. Highlight **VHDL File** by left-clicking it and then clicking **OK** at the bottom of that window.

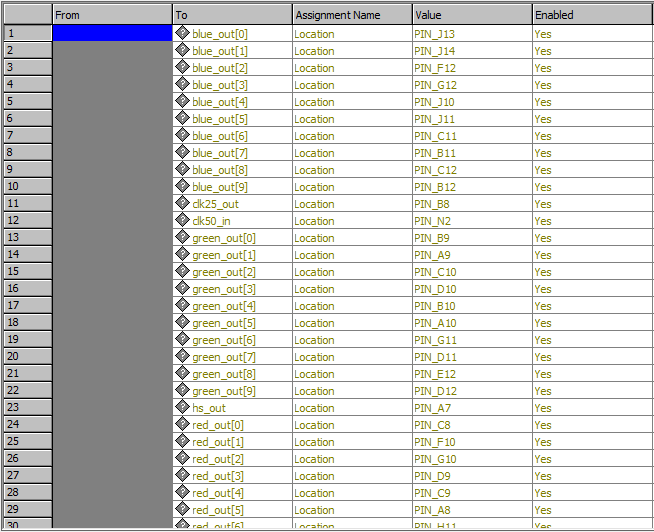


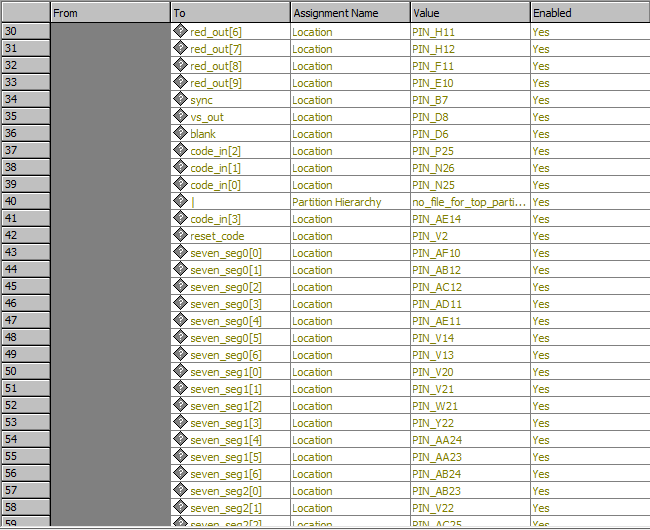
6. You’ll be introduced to a blank notepad that can be filled with VHDL code. Go ahead and **paste the code** at the very bottom of this document.

7. Compile the code by clicking the icon ►, or through the tabs Processing > Start Compilation, or CTRL+L. Make sure the code was compiled successfully as shown below.



8. Now, we will need to pin-map the inputs, outputs, and signals to the components on the DE2 board. To do that, go to **Assignments > Assignment Editor**, or CTRL + SHIFT + A. In the image below, set all spots to these specifications:

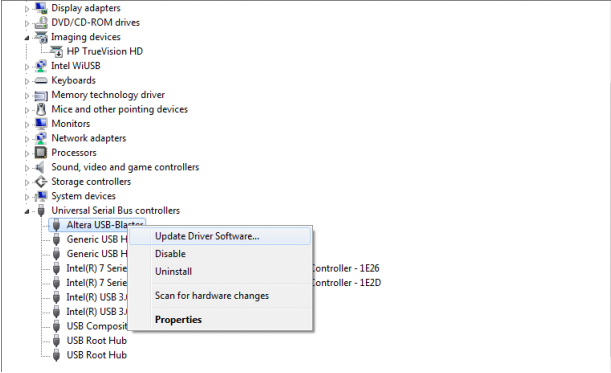




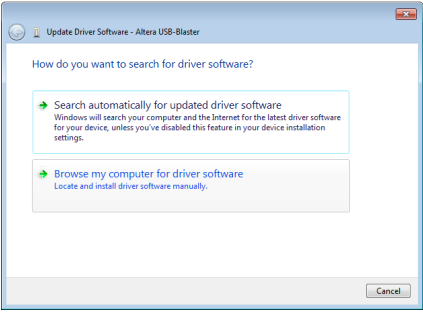


9. Once you’ve set these pins. Go ahead and recompile.

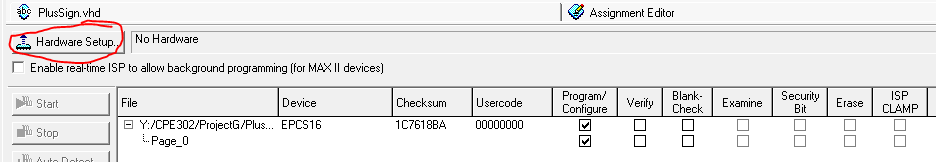
10. Great, now we have to program the board with our newly created code and pin-map locations. Before we do that, we must make sure our board is updated and in the system. If the board is already updated onto the computer, skip this step and go to step 12.   
To update our board, we need to go to **Device Manager**. We can do that by hitting the window key and searching for Device Manager. Once open, you should get a window like this.



11. As shown, make sure you click **Update Driver Software…**. Then a window will pop up and ask you to search automatically or to browse your computer. Click **Browse my computer for driver software**. From there, you can navigate to your driver update by going from **Local Disk (C:) > altera > 7.1 > quartus > drivers > usb-blaster > x64**.



12. Going back to Quartus II 7.1, go to **Tools > Programmer**. Go ahead and click the **Hardware Setup…**.



13. Hit the drop-down menu **Currently selected hardware:** and select the **USB-Blaster**.

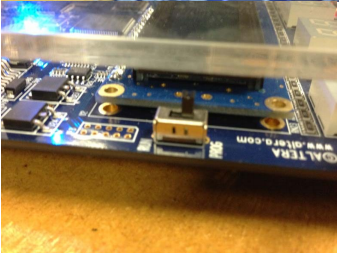
14. Afterwards, you want to make sure that your mode is in **Active Serial Programming**.



15. Make sure your **Program/Configure** has the two boxes check-marked underneath their column:



16. Now, set your board to **Program** and hit **Start**



17. Let the board program for roughly 30 seconds till the blue LED that says “load” disappears or till you get a message back on Quartus saying the programming was a success.

18. Switch the board to **Run** mode instead of Program and restart the board by turning it off and back on. If the VGA cord is plugged into the monitor, you should see the image projected onto the monitor screen like this:



The picture of the left indicates that the lock is open, while the picture on the right indicates that the lock is closed.

19. To use this project, you must familiarize yourself with the button layout:

