Çelik Köseoğlu, Berk Evren Abbasoğlu

CS223 - Group 2

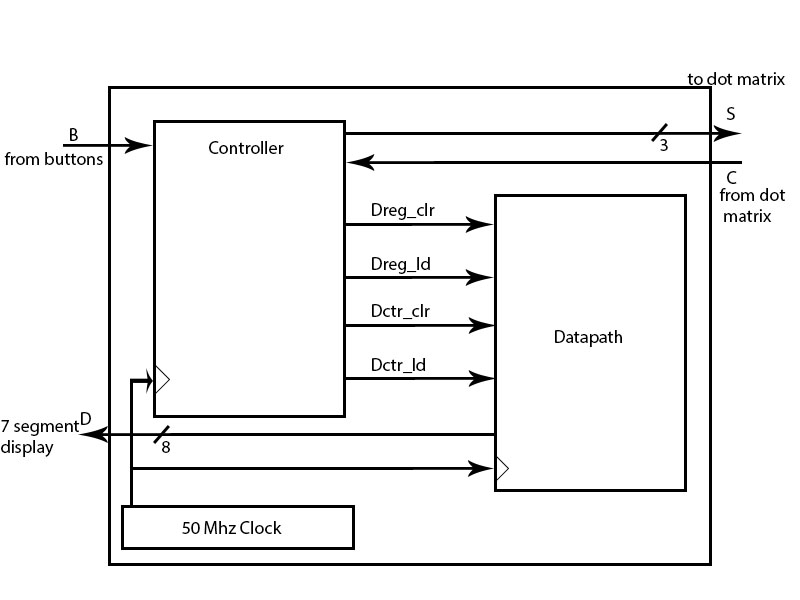
Section 2

1 December 2015

Rocky Journey - Project Progress Report

In the first week, we started reading the data sheet for the Beti Board’s 8x8 LED Matrix because our project requires us to use at least 15 of them at the same time. While trying to figure out how we are going to use the 8x8 LED Matrix, we saw that we can’t control 64 LEDs with the 16 outputs provided in the Beti Board. It got worse than that. It turned out, LED Matrix 64x3 LEDs inside (red, green & blue for each point) and controlling these 192 LEDs by using the 16 outputs on the Basys Board is not possible. Fortunately, we learnt that there are shift registers built underneath the LED Matrix while we were trying to reverse engineer the board because the schematics for the beti board were very hard to understand and some of the pin numbers in the schematic were wrong. The schematic might be for an older Beti Board. Trying to figure them out ourselves by hard wiring some components was easier.

Then we continued by learning how shift registers work. It turns out that it is possible to control an infinite number of outputs by using only 3 inputs with shift registers. However, a faster clock is needed to control more and more outputs with an acceptable speed.  
 The other problem we faced was the shift registers did not work with 3.3 volts. We solved that problem by using buffers. We got 5V input from the Beti Board’s USB input, then plugged it into the Vcc of the buffer.  
 We’ll be starting to write Verilog Code this week to control the LEDs.

Given above is the draft for our project’s High Level State Machine diagram. Provided it is still in development, we will explain it in detail. The B input comes from the buttons on the Basys Board, which gives the direction the ship will move, up or down. S is the output that causes the ship to move in the provided direction. C input comes from the dot matrix, telling whether the ship has crashed or moves on. Our controller sends load signals to the datapath, which in return shows the obstacles on the ship’s track. It can also reset it when starting a new game. When the ship successfully passes through a hole, player’s score is incremented by one (also reset to 0 at the start). This is displayed on the 7 segment display of the Basys board. We use a 50 Mhz clock throughout.

HLSM Diagram