Akash Badshah

6.115 Final Project Proposal

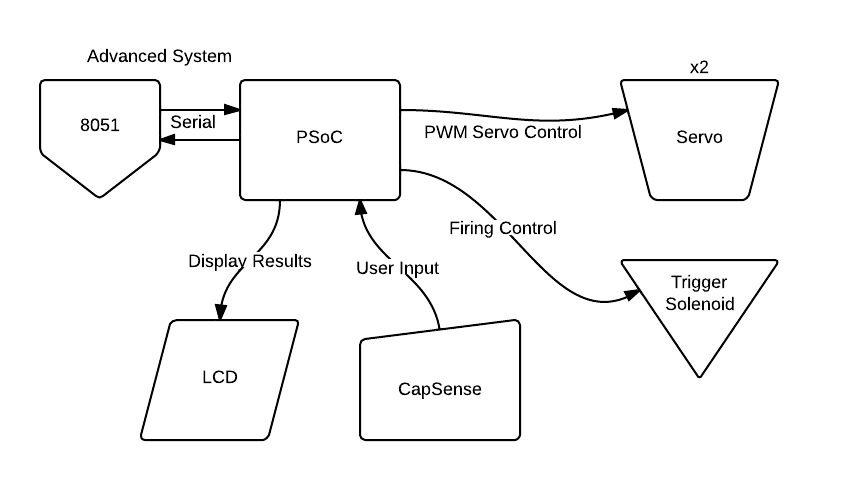
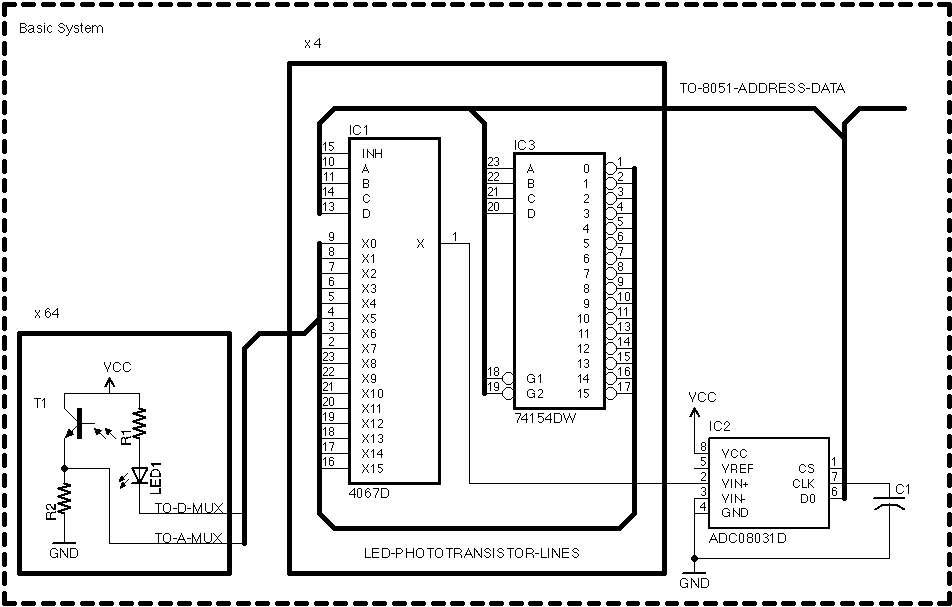
**Background & Introduction**

For my 6.115 Final Project I would like to make an electronic target board and projectile launcher. Games like darts are a lot of fun, but require a special kind of projectile that implants itself in the board in order to be scored. Instead of requiring such equipment, I would like to design and build an electric target that records the impact coordinates of an arbitrary projectile and provides a score back to the user.

Furthermore, I would like to build an electronically controlled slingshot which both fires a projectile according to some digital signal and can be rotated in 2 axes to aim the projectile with some input controls. Finally, I would like to add a feedback loop into the system, so the launcher can use the results of a previous launch to correct itself and find the center of the target itself. Ideally, this system would be able to calibrate against arbitrary target placement or conditions and find the bull’s eye within a few shots.

I think this would be a very interesting project simply because it would be a fun game to be able to play and would be a great way to demonstrate my knowledge from 6.115. The electronic target board would require usage of different sensing systems we learned about (ie SpinDude) and consideration of timing in order to make sure that the projectile is measured at the right time. The projectile launcher would require usage of motor knowledge and user input. Finally the feedback system would close the loop and essentially make an autonomous input/output device.

**Hardware Description**

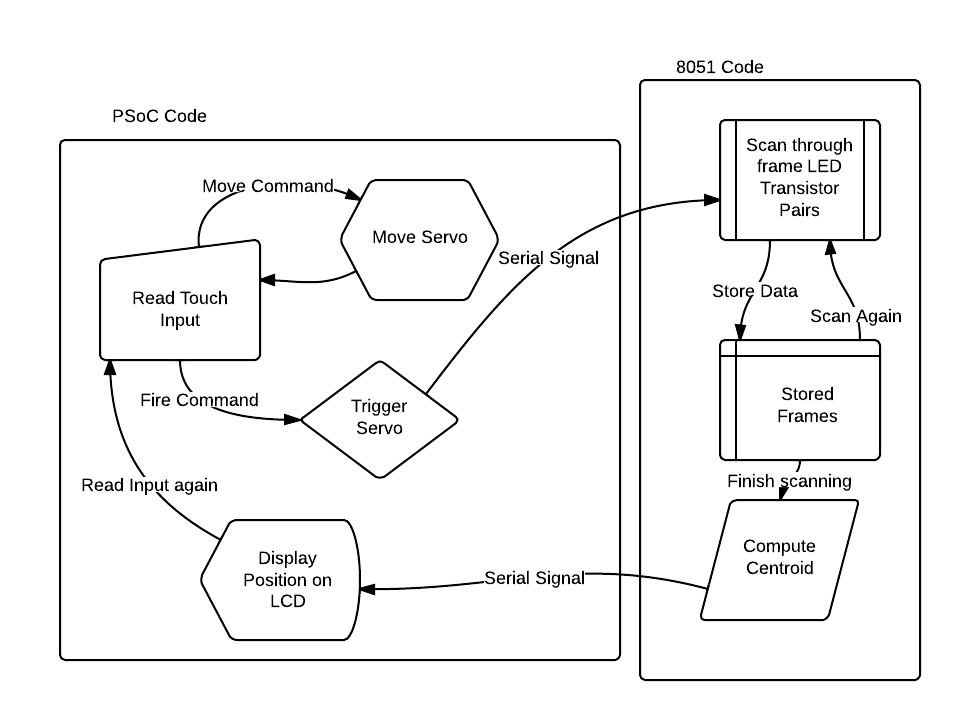


The hardware system required for this target / launcher would use the r31jp to control the electronic target board, and the PSoC for the launcher and control system.

The target would essentially be a frame on top of some backboard with the bull’s eye drawn or etched onto it. This frame would consist of red LEDs on two edges and phototransistors on the other two edges. When it came time to locate the projectile, it would scan through the LEDs and transistors and see which x,y coordinates were being occluded by the projectile. This is the same kind of system that was used in the SpinDude. This information would be sent via serial to the PSoC for processing by the launcher.

The PSoC controlled launcher system would utilize key features of the PSoC for controls and display to the user. The information recorded by the electronic target will be displayed on the PSoC LCD screen, and the position of the launcher will be controlled by the CapSense touch pad located on the PSoC evaluation kit. The launcher itself will consist of a platform mounted to 2 servo motors oriented to allow for rotation in the xy and yz planes, allowing the nozzle of the launcher to be pointed at different places on the xz oriented target. These two hobby servos will be controlled by PWM signals generated from the PSoC. The launcher itself will be a slingshot that is pulled back and latched onto a relay. When firing the pull relay will be retracted and the elastic released, thus propelling the projectile forwards. Another relay will sit in front of the projectile to keep it in place before launch. Finally, the whole slingshot will shoot through a tube of pvc to help collimate the path of the projectile.

**Software Description**

**** The software required to run this system consists of a module running on the 8051 and one running on the PSoC. The 8051 module will wait for a signal across serial from the PSoC indicating that the projectile has fired. From that point, the 8051 will start scanning the frame to get x,y coordinates as quickly as physics will allow. After collecting each frame, it will compute the centroid of the data to give a coordinate for the object blocking light in that frame. Any coordinate which suggests no occlusion will be discarded, and those with a position will be averaged to give a final output x and y coordinate. Once computed, these numbers will be sent over serial to the PSoC.

The PSoC will run a software module which both displays data from the game and take input from the user. It will read the values on the CapSense pad to see if the user is providing input. Depending on which buttons are being depressed, the PSoC will change the position of the servos connected by changing the duty cycle on the PWM waves output to the two servos. It will also reflect these changes on the LCD screen. After a certain button press from the user, the system will fire the projectile and send a signal to the 8051 to begin measurement. It then waits for the results of this firing and will display those results on the screen. If in auto-correct mode, the PSoC will then use this target-hit data and automatically correct the positioning of the servos to account for its error. It will continue to fire and correct until it consistently hits the target in the desired location. In this case user input will not control the position of the servos but the ideal hit point on the target.

**Project Scope / Management**

In order to make the project manageable, the different pieces of this project will be broken into 3 categories: basic, advanced, and extreme.

The basic system will simply consist of a frame which reads the value when a projectile is fired, and the display the position onto the PSoC LCD. Essentially this will not include a digitally controlled projectile launcher, and will assume that the user has some other way of launching projectiles which can be recorded. This will still be somewhat interesting because users could use this to play a game like darts with arbitrary objects. I’ll also allow the user to pick from a variety of target configurations from the PSoC so it can keep track of scores between players.

The advanced system will consist of the entire basic plus the digitally controlled projectile launcher. Specifically this will add the servo-controlled slingshot and the input controls so that users can play the game against their friends. With this system, people will have a full end to end to play games.

Finally the extreme system will include the feedback loop and automation, so that the launcher shoots a projectile and then corrects itself afterwards. This is largely a software addition, but might prove challenging because it will require some calibration step for the system to find the target and further consideration of how best to adjust the trajectory according to results from a previous shot.

If I end up having a lot of extra time, I can also explore a more complex launching mechanism, by building something like a coil gun to drive a projectile at higher speeds and entirely with electronics.

**Special Component Needs**

In order to build these systems I will need the hardware required to build the frame as well as the servos and relays for the launcher.

Specifically the frame will require 64 LEDs and 64 phototransistors of the same sort used in the SpinDude project. Similarly, I will use Analog Multiplexers and Digital Multiplexers to switch between these components as I read them. Finally I will need an ADC to read the values on each of the phototransistors. This part of the project will highly resemble the circuitry within SpinDude and used to operate it (sans the motor control components). The frame itself will be built using wood or Plexiglas as the backboard, engraved via laser and then another layer of wood or Plexiglas to protect the circuitry from projectiles. I may end up printing PCBs to house the phototransistors and leds.

The launching mechanism will require 2 hobby servos that can be controlled by PWM. These are easily found on a website like Sparkfun. I will also use metal or plastic parts cut by a waterjet or laser cutter to provide mounts for the different pieces. Finally, I will need two relays, some elastic, and a pvc tube to actually produce the launching mechanism. All of this will be interfaced through the PSoC board, which also already has the LCD screen and CapSense inputs.

**Timetable**

Week of April 14: I will design the circuitry for the frame, build test examples on the breadboard and order the PCBs which I will need.

Week of April 21: Solder the PCBs for the frame and finish building the frame with wood and Plexiglas. I will also spend a significant amount of time on the software required to make the frame work. I should be finished with the Basic system by this point.

Week of April 28: I will build the 2 servo mounted projectile system and wire up the circuitry necessary for launching projectiles.

Week of May 5: I will finish the code and input / output system required for the Advanced system by this point, completing the launcher and software required to run it.

Week of May 12: I will spend this week working on feedback controls and leveraging data from the target to guid the launcher. The Extreme system should be completed during this week.