Unit testing:

Cutting point-After button task reads from the fifo:

1. Test that a fifo read from an empty fifo returns false.-Fail
2. Test that a fifo read from a fifo returns the correct value.-Fail
3. Test that the rail gun charge gets calculated correctly.-Fail

Cutting point-After physics task updates values periodically.

1. If capsense force is not equal to 0, check if the horizontal position of the platform is correct.-Pass
2. For satchels, check that the proper number of satchels are in the air at all times.-Pass
3. Check that the vertical position of the satchels was updated correctly.-Pass
4. If something collides with a wall, check that its velocity changed sign.-Pass
5. If the shield was activated during the physics update, check if any satchels in range were properly destroyed.-Fail
6. When a satchel reaches the ground, check that the satchel’s x-position is on target. Pass

Cutting point-after the display/LED task updates periodically

1. Check that the left LED turns on and off at the correct duty cycle.-Fail
2. After the castle evacuation time expires, check the left LED is constantly on.-Fail
3. Check that the Pulse width for the right LED is relatively equal to the current force magnitude.-Fail

Functional Tests:

1. Press the left/right side of the CAPSENSE slider and check if the platform moves to the left/right.
2. The outer quarters should cause the right LED to stay constantly lit while the inner quarters should have the LED blink on and off evenly.
3. Pressing two sections of the CAPSENSE slider on opposite sides combines their effective force, potentially canceling each other out.
4. If BTN1 is pressed, a force field should appear on the display and destroy any satchels in range.
5. Pressing and holding BTN0 then releasing should fire a projectile with speed increasing as you hold the button, maxing out after 5 seconds.
6. If the generator does not have enough power for the shield or the railgun nothing should happen.
7. Hitting the Castle wall should break a piece of the foundation.
8. After foundation 3 hits the left LED should start blinking.
9. Either hitting the castle 5 times or 2 minutes after the evacuation sequence begins should take the user to a victory screen.
10. The platform hitting a satchel, a wall too fast, or running out of shots when the castle is in tact should take the user to a game over screen.

Project Summary:

This week, I predicted which of my unit tests would pass/fail, created functional tests, and implemented the CAPSENSE slider and platform into my program, as well as game over conditions for my game.

Summary Effort/Estimate:

I have 44.4% of my current work (15 estimated hours out of 45 total) in 30% of the budgeted time (15 hours spent out of 45 hour estimate) For the work that has been completed, I took 0.9x (13.5 actual hours/15 estimated hours) as much time as I estimated.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Task | Estimated Effort | %Estimate | Actual Effort | Status |
| Task Diagram | 3 hours | 6.66 | 2.75 hours | Complete |
| Unit Testing | 8 hours | 17.77 | 2.5 hours | In progress |
| **Platform Phys** | **5 hours** | **11.11** | **5 hours** | **Complete** |
| Satchel Phys | 5 hours | 11.11 | 4.75 hours | Complete |
| Railgun Phys | 5 hours | 11.11 |  | Not started |
| Display Task | 12 hours | 26.66 | 4 hours | In progress |
| Button Task | 5 hours | 11.11 | 0.5 hours | In progress |
| **CAPSENSE Task** | **2 hours** | **4.44** | **1 hour** | **Complete** |
| Total Complete | 15 hours | 44.4 | 13.5 hours |  |

* Platform Physics
  + A good portion of this implementation was quick, since the code for updating the platform’s physics was similar to updating the satchel’s position. The places where I struggled the most was getting the platform to bounce correctly, since I didn’t initially realize that I needed to check that the edge of the platform was hitting the wall as opposed to the center of the platform that gets updated by the physics task. Checking to see if the platform force was applied correctly also took a bit of time, which would be easier if I had my LEDs set up already.
* Capsense Task
  + This task was also not that hard to implement, since most of my capsense code from the previous labs were easily carried over into the project. The main difference was changing the enumeration for the capsense position, and creating a handful of new edge cases for when multiple positions are pressed at the same time. If I have some extra time near the end of the project, I think it would be interesting to add more positions to the slider so the force LED has more options when doing PWM.

Work time for week 3

Mon. 2 hours platform physics, 1 hour platform display

Tues. 2 hours platform physics, 1 hour display

Weds. 1 hour platform physics, 1 hour unit testing