Tommy Ramirez
4/22/22
ECEN 3753

Project HarkonnenPong Week 6

Unit Testing

Platform Motion

- Test if platform doesn't move when no force is applied and velocity is 0 (Pass)
- Test if platform maintains nonzero velocity when no force is applied (Pass)
- Test if platform increases velocity when force is applied in the same direction of the velocity (Pass)
- Test if platform decreases velocity when force is applied in the opposite direction of the velocity (Pass)
- Test if low and high force has difference on accelerations (Pass)

Holtzman Mass Motion

- Test maintained x velocity of Holtzman Mass (Pass)
- Test constant downward acceleration of Holtzman Mass (Pass)

Holtzman Mass Out of Bounds Check

- Test LED flag set when Holtzman Mass crashes (Pass)
- Test if Holtzman Mass is replaced after exiting the range (Pass)
- Test if Holtzman Mass is replaced for a number of times based on a variable (Pass)

Functional Testing

1. At reset observe that the left LED is blinking at the slowest rate. (Pass)
2. Place your finger on the touch slider and slide your finger to the middle right and observe the platform accelerating rightward. (Pass)
    a. Release your finger and observe the platform maintaining velocity. (Pass)
3. Slide your finger to the far right before it hits the border and observe the platform accelerating faster rightward. (Pass)
    a. Release your finger and observe the platform maintaining velocity. (Pass)
4. Slide your finger to the far left before it hits the border and observe the platform decelerating before accelerating leftward. (Pass)
    a. Release your finger and observe the platform maintaining velocity. (Pass)
5. Press Btn1 and observe the Holtzman Mass count and the y-position of the holtzman mass reset to the top of the screen. (Pass)
6. After pressing Btn1 several times, observe that the button no longer has any effect. (Pass)
7. Move the platform away from the falling Holtzman Mass and observe it activates the game over screen and the right LED blinks at 1 Hz. (Pass)
8. Move the platform in the Holtzman Mass' path and observe it bounce off the platform. (Pass)

    a. Observe after repeated bounces that the peak of the parabola is getting lower. (Fail)
    b. After several bounces, observe that when the peak is vertically close to the platform that the mass phases through the platform. (Fail)
9. Press Btn0 when the Holtzman Mass is near the platform but hasn't hit it yet and observe that upon bouncing, the peak of the parabola is higher than before. (Fail)
10. Use the slider to match the right and left LED's pwm while the right LED is blinking and observe that the platform will collide with the holtzman mass. (Fail)
11. Bounce the holtzman mass outside of the canyon and observe the same thing happen as what happens when btn1 is pressed. (Fail)

Functionality Deliverables and Usability Summary

This week I fully implemented most of the initial variables into my function. I also got the bounce function working for the platform and HM mass on the edges of the canyon. I had trouble trying to implement the bounce function due to the capsense only working for one sensor, but after the problem was identified, the bounce code was a relatively simple task. Most of the function tests now pass leaving only the collision and LED tasks for implementation. Lastly I got the HM to bounce off the canyons and platform, but the shield boost still requires implementation.

Summary effort & estimate numbers

I have completed **56.67%** of my currently-scoped, estimated work time (68 actually spent /120hr total estimate) with **87.08%** of the initially-estimated work. (104.5 estimated for the items I have completed, of 120hr total estimate). For the work that has been completed, I took **0.651x** (68/104.5) as much time as I estimated.

No scope changes have been made this week, My latest scope is still my original scope (120 hrs), but the estimates include partial completion contributions.

In-scope work items

Completed before this week:

- Project Reading and Task Diagram First Draft creation (est 10 hrs) (actually 8 hrs)
- Learn How to Manipulate LCD basic drawings (est 20 hrs) (actually 1 hr)
- Lose Condition LED Control (est 1 hr) (actually 0.5 hr)
- Configure slider to control LED PWM (est 2 hrs)(actually 3.5 hrs)
- Create Tasks and Test on Segger (est 2 hrs)(actually 2 hrs)
- Slider Testing/Fixing (est 6 hrs)(actually 2 hr)
- Task Diagram Revision for clarification and necessary optimization (est 3 hrs) (actually 2 hrs)
- Laser Implementing (est 1 hr)(actually 1 hr)
- Task Integration Testing (est 10 hrs)(actually 4 hrs)

Completed this week:

- Platform Motion (est 1 hr)(actually 9 hrs)

This took longer than expected. The complexity of achieving fluid motion combined with issues in the capsense reader resulted in hours of troubleshooting and adjustments. I had expected the slower acceleration to be much simpler as well but the slider is difficult with the inner sensors, so that made confirming that the platform is accelerating at a lower rate difficult.

- Platform Bounce (est 2 hrs)(actually 4 hour)

This task was also underestimated in time consumption. The border that the platform bounces off of has been created before this week. Then the platform was able to stick to the wall after hitting it, which was not difficult to program. Then the difficult task programming the bouncing was completed after a couple hours. That last part was more time consuming due to the complexity of the bounce conditions. There was an issue with the HM mass losing its motion in the x direction, but that edge case was taken care of. The testing for the final part did not take long after the code was operational.

- Basic Platform/HM collision (est 2 hrs)(actually 1.5 hrs so far)

The basic collision bounce without the shield boost has been implemented into the physics task. Once the physics and bouncing functions were implemented and advice on the kinetic energy modifications occurring using math.h was received, the programming and testing only took a short amount of time.

Significant Partial Completion for estimate numbers:

- HM Physics Unit (est 10 hrs)(10.5 hrs so far)(95% progress complete)

Completed unit tests. Just need shield boost bouncing conditions.

- Task Unit Testing (est 10 hrs)(5 hrs so far)(90% progress complete)

All of the unit tests created last week now pass but more could be created for the right led predictions.

- Motion/Position Physics Task (est 10hrs)(7 hrs so far)(95% progress complete)

Physics unit tests passed. Needs implementation of boost collision physics and ensure that kinetic energy is lost when the HM mass hits the platform.

- Platform/HM collision with bounce (est 5 hrs)(3 hours so far)(90% completion)

With the physics function added, the bounce collision can be added through the acceleration parameter and a few new lines of code. Default values of collision physics have also been set. implemented. Default values of collision physics have also been set.

- Qualification Check/Debugging (est 15 hrs)(4 hours so far)(80% progress complete)

11/16 tests have passed. Many of these tests will pass once the platform is tied to the collision and boost physics which is close to completion.

Not Completed yet:

- Commenting and Code Cleanup (est 5 hrs)
- Platform Interception Prediction LED implementation (est 5 hrs)

Risks

Item			Risk (P*	Recognize	Mitigated/ Resolv	ROA	How
I will too long to figure out the LCD screen	2	5	10	11-Mar-22	Resolved	R	Got help and clarification on the basics from a TA
PWM is too slow to change brightness	8	3	24	15-Mar-22	Mitigated	M	Changed goal from changing brightness to blinking rate
Changes in PWM can't be recognized/decoded	3	13	39	17-Mar-22	Mitigated	M	Differences have been made very noticable
My slider is sticking after some time operating	1	1	1	11-Mar-22	Resolved	R	New includes seemingly made the problem disappear
Program Freezing	20	5	100	2-Apr-22	Mitigated	M	Restart demo and that fixes the issue most of the time
			0				
			0				
			0				
			0