

Final Statement on Project Standing

The project is now finished. In the short time between the week five submission and now the configuration implementation was finished and all work tasks are now completed. One of the highlights of this project for me is the timer for blackouts. It does not show up much in gameplay because the user will usually lose before the LED blinks multiple times but I like my implementation of a software timer toggling a flag variable that signals whether the user can drive again or not.

I have completed 100% of the scoped work, (34/34hr) in 91% of the initial time estimate (31/34).

I got ahead of schedule by changing my configuration settings. I planned for the user to supply a config file and my code would take that in and parse it but I decided not to implement that because it did not add any value to the gameplay itself and parsing the data is tricky. Loading in a file is not a large challenge but correctly parsing it and storing the inputs was too much of a headache after attempting for an hour or so. Still the game is playable and mildly fun.

Scoped Work

Work Item	Description	Estimated Time	Status
Project Planning	Create an initial task diagram and documents for week1. Have an idea of project design.	2 hrs	Complete
Data Structure Setup	Setup initial data structures for rocket, message queue, angle setpoint, and throttle setpoint.	2 hrs	Complete
Writing Unit Tests	Write initial unit tests. Should fail until further development. Challenge in porting in data needed between cutpoints.	3 hrs	Complete
ITC and Shared Resource Setup	Establish the structures needed between tasks, such as semaphores, timers, mutexes, and the tasks themselves.	1 hr	Complete
Angle Task	Develop code for angle task to pend on semaphore from button ISR and write to angle setpoint.	0.5 hrs	Complete

Throttle Task	Develop code for throttle task to pend on semaphore from timer and write to throttle setpoint.	0.5 hrs	Complete
LED/PWM Math	Develop code to drive LED's based on PWM calculations. PWM code itself is created in a separate work item.	0.5 hrs	Complete
Rocket Design	Practice with the micrium graphics library to design how the rocket will look and what are the meaningful points.	1 hr	Complete
Display Task	Develop code for display task to take data from rocket data structure and display the rocket. Challenge is to display the rotation of the rocket graphic, here the math is done to move the vertices of the rocket.	5 hrs	Complete
Physics Task	Develop code for physics task. This is the bulk of the project. Includes programming kinematic equations to computing the acceleration, thrust, fuel, position, and rotation of logic. Will also be responsible for knowing if a win or loss has occurred.	8 hrs	Complete
PWM config	Create the routines necessary to program configurable PWMs using software timers. Needed by LED task.	2.5 hrs	Complete
Configurability Implementation	Takes config either through changing settings in project code.	4 hrs	Complete
Blackout Calculations	Figure out max acceleration and how to respond to blackout.	1 hr	Complete
Debug	Built in time to debug. After all previous work items complete it is expected that the project is not fully functional without substantial debug work.	3 hrs	Complete

Completed this Week

Configurability Implementation:

The original intention for this work item was to write code that would load in an external file containing a configuration struct. It was since decided that the loading and, more specifically, the parsing of a file go beyond the scope of the project. Instead there is a struct which holds the configuration data and it can be modified from the code itself to set different parameters.