



RTOS PROJECT

Kanin McGuire



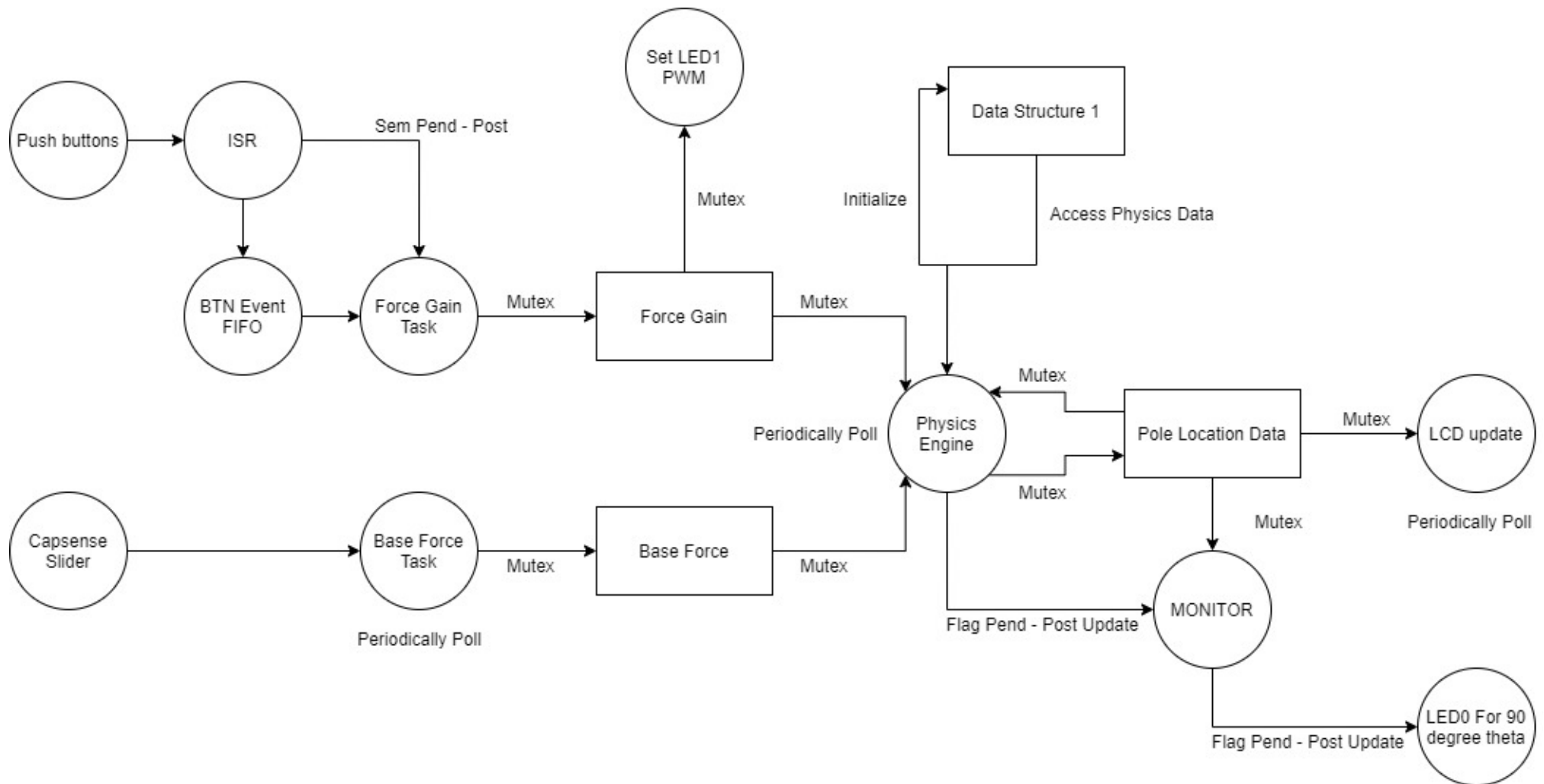
APRIL 2, 2021

Contents

Block Diagram	2
Week 1: Planning	3
Time estimates.....	3
Unit Test Plans:	4
Project Stands	5
Effort Summary	5
Scope Items.....	6
Risk Register	8
Week 2:	9

GitHub: https://github.com/KaninMcGuire/RTOS_FINAL_PROJECT

Block Diagram



Week 1: Planning

Time estimates

Estimate

Read project description: 20 minutes

Task Diagram: 1 hour

Unit test Plan: 1 hour

Project Stands Statement: 15 minutes

Effort numbers: 30 min

Scope Items: 1 hour

Risk: 10 minutes

Actual

10 minutes

44 minutes

1:10 hours

2 minutes

5 minutes

15 minutes

14 minutes

Unit Test Plans:

Cuts

Cuts at force gain Task: Variable updated with push buttons (closed loop)(gain only updated here)

Cut at base force Task: Variable updated with slider(force only updated here) (closed)

Cut for LED1 PWM: Single variable number and operation(just LED PWM) (closed)

Cut for Data structure 1: Data structure Witten to Once then just referenced(closed)

Cut for Monitor: Monitor and LED0 closed, jumped to through a flag event(closed)

Cut for LCD update: LCD can be tested on its own. (closed)

Cut around Physics engine: Physics data can be updated and used in a single location(closed)

Unit Tests

1. Test force different gain numbers with PWM of LED1 see variation in modulation.
2. Test writing to the LCD Display
 - a. Does it correctly draw your figure?
 - b. Using a locational loop in different positions does it update periodically and correctly.
3. Physics engine - Test Data Structure 1 information is accurate.
 - a. Check correct initialization.
 - b. Check erroneous pole position outside of zone.
 - c. Check unused data components are zero.
4. Slider input Check max and min force limits
5. Push Button Activation
 - a. Check lower and upper limits.
 - b. Check push button activation interrupts.
6. Write angles below 90 degrees and ensure LED not lit.
 - a. Write above 90 and observe led light up.
 - b. Write below 90 degrees and see that led stays lit.
7. Check the proper direction change happens in base force structure.
8. Create an array of states for the physics and pole location and view the effects of physics through the LCD.
 - a. This test will show physics and LCD is correctly implemented.
9. Test pole locations updating with physics (does angle update)
10. Check pole tip stops at bottom of screen (physics knows what a ground is.)

Unit Test	Status
1	Not Run
2	Not Run
3	Not Run
4	Not Run
5	Not Run
6	Not Run
7	Not Run
8	Not Run
9	Not Run
10	Not Run

Project Stands

This week I did project planning, planned out unit tests, created my rough task diagram.

Effort Summary

I have completed 11.3% of my currently scoped, estimated work (2.66 / 23.5hr) in 62% of the initially estimated time. (4.25 of 23.5hr). My best guess of my say/do ratio is 62%, so to unbiased my estimates after this class, I may want to multiply my estimates by 1.61 (100%/62%)

No scope changes to report at the current time.

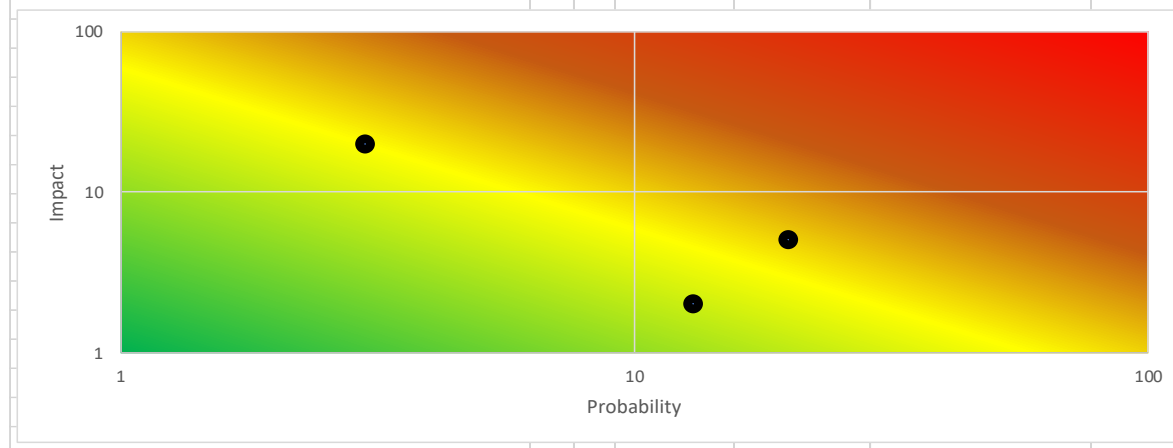
Scope Items

Work Items	Estimated Time	Actual Time	Status	Summary Statements
Task Diagram	1 Hour	50 min	Complete	Task Diagram consist of all tasks that I considered adding and implementing. More tasks may still be added/removed as the project progresses. An uncertainty I had was with the Data structure and whether it was also intended to hold force gain, base force, and pole location data.
Unit Test Plan	1 Hour	1:10	Complete	I have overall the minimum 10-unit tests though some of these tests may be split and considered their own as well as some may even be merged. Unit test plans are always a difficulty for me to conceptualize before I have any base code going in the area. Usually, I must structure my unit tests around my code though it is intended to be written first.
Overall Planning Document	2.25 Hours		Complete	The overall document contains all my estimates and plans. I plan on updating this document weekly. I plan on making every week a chapter that way progression and past information is not lost.
Create base template off previous code	1 hour			
Create Task Structures and create Global variables	1 hour			
Build Display driver	1 Hour			
Create Physics logic without inputs	4 hours			
Implement PWM with LED	1 hour			
Implement Push buttons	1 Hour			
Implement Slider	1 Hour			
Implement LED1	45 minutes			
Implement Unit Tests	3 hours			

Build Mutex, Semaphore, Event Flags in appropriate tasks	1 hour			
Connect Push Buttons to Physics	30 minutes			
Connect Slider to Physics	30 minutes			
Connect Physics with LCD Display	30 min			
Final Functionality and Testing	3 hours			
Total	23.5 Hours			

Risk Register

Item			Risk (P*)	Recognize	Mitigated/ Resolve	ROAM	How
Task Diagram being incorrect	20	5	100	2-Apr-21		O	Planning and Implementation often don't align perfect
My gecko is not working	3	20	60	2-Apr-21		O	Test it. If mine not working, contact instructor
Include issues	13	2	26	2-Apr-21		O	Including files during programming
			0				
			0				
			0				
			0				
			0				
			0				



Week 2: