Task:	complete/not yet complete	Estemated Time To Complete	Time Spent So far	Bold items are larger items with subtasks below them. Their estemated time to complete is a sum of the time to complete each subtask
Read and digest project document	complete	15 minutes	30 minutes	
Create Task Diagram	complete	15 minutes	2 hours	
create task list	complete	15 minutes	1 hour	It took me quite a while to really wrap my head around the project and be able to break it into smaller tasks
create config data data structure	complete	5 minutes	30 minutes	I created this structure as well as a function which loads the default values into this struct. I had some issues deciding which file to include the data structure in
create menu at start of game to allow users to set config data	**not yet complete**	**1 hour total**	**1.5 hours**	
take input from touchscreen	complete	50 minutes	1.5 hours	I did not end up implementing the menu, but I was able to take input from the screen. I had the user press on the screen to begin the game
randomly generate maze	**complete**	**3.25 hours total**	**8 hours**	
Be able to make mazes on the LCD display	complete	30 minutes	3 hours	This took much longer than expected to complete as I ran into several errors along the way. I actually thought my board was broken for a while because my LCD screen was showing a screen which seemed to be a broken screen. After playing around for quite some time with the configuration and initialization of the LCD I as able to fix the issue.
use random number generator	complete	30 minutes	30 minutes	This did not take me very long as I have quite a bit of experience using the RNG on the STM
random number for each possible wall position to determine if wall is there and if there is a hole there	complete	1 hour	1.5 hours	I was able to accomplish this without running into any major bugs. I was not sure how to use this random number, but decided that I would do the number modulo 10 and then divide the number by 10 and this was a much more managable way to use this very large random number
2d matrix of cells	complete	30 minutes	2 hour	This took much longer than expected as I was running into an error where my threads were unable to access the memory addresses that was storing the data from my pointer to the array. I was able to solve this issue by dynamically allocating memory for this array
implement waypoints	complete	30 minutes	1 hour	I implemented waypoints by making a function to draw the outline of a circle and then using the distance formula as I did with the holes
make sure no holes or walls where the first waypoint is	complete	15 minutes	15 minutes	I put the ball in the upper left corner and made it so that walls and holes did not spawn here
hard code a maze for when config maze size is 0x0	complete	30 minutes	0	
test for game ending conditions	**not yet complete**	**3 hours total**	**2.75 hours**	
be able to detect tilt of more than 90degrees	complete	30 minutes	30 minutes	
Implement hard edge	complete	1 hour	1 hour	I was able to draw borders around the maze and then stop the ball if it made it to these borders. I knew how to calculate where the borders were, so if the ball reached where the border was I just moved it back 1 pixel.
detect wheather drone is above hole	complete	30 minutes	30 minutes	I was able to use the distance formula and then if the distance was less than or equal to the radius of the whole the game ends
detect if too much time has passed	complete	30 minutes	30 minutes	I used the system timer to subtract time from value set in configuration for the time to complete the game.
calculate game score	not yet complete	30 minutes	15	I did not end up printing a score when the game was complete, but I did display the time left for the game to complete as of when the game was won
implement user botton disruptor	**complete**	**1 hour 20 minutes**	**3.33 hours**	
Implement button ISR	complete	20 minutes	1 hour	This did not take long as I was able to essentially copy what I did on previous labs. I also used a function returnPressed() which returns true if the button is pressed and false if the button is not pressed
Implement "Drone Update Mutex"	completed	20 minutes	2 hours	the inter task communication was somewhat simple
Add appropriate functionality in the drone thread	complete	40 minutes	20 minutes	This was also fairly simple. I only needed to use a couple if statements
Implement LED PWM to correlate with fullness of energy store (green) and time to charge energy store to minimum activation energy	**not yet complete**	**1 hour 45 mintues**	**2.25 hours**	
Create global variable to keep track of energy	complete	15 minutes	15 minutes	
Implement logic to correctly keep track of the drone's energy	complete	30 minutes	1 hour	
Figure out how to make the LED's PWM and calibrate it and have it reflect the energy left in the tank	not yet complete	1 hour	1 hour	I was able to make the LEDs PWM but I was not able to make it calibrate with the fuel left in the tank
Physics	**complete**	**3.25 hours total**	**12 hours**	
read gyro	complete	15 minutes	30 minutes	This was not too hard as we have used the gyro frequently in our previous labs
Approximate rotation angle	complete	1 hour	1 hour	I was able to calculate the corrected gyro velocity using the following formula: correctedGyroVelocityY = (17.5/1000) * gyroVal; //deg/sec //17.5 is sensitivity
compute drone velocity: old v + acceleration	complete	30 minutes	1 hour	I first implemented physics based on only the current angular velocity of the board. I then was able to implement physics which accounted for past velocities.
Make it so when drone hits a wall, it loses all its momentum in that direction	complete	30 minutes	7 hours	I did not think I would be able to complete this. In the end I realized that my generateMaze function was incorrectly assigning values to the start and end of the wall, which was making it so that I could not detect the distance away from the drone that the ball was
indicate to user how tipped the board is	complete	30 minutes	1.5 hours	I printed out the x and y values for velocities.
make it so when the drone hits the boarder of the maze, it loses its momentum in that direction	Complete	30 minutes	1 hour	This was not too difficult to accomplish. I used a bunch of "if statements"

TOTAL		12.75 hours	33.83 hours	265%