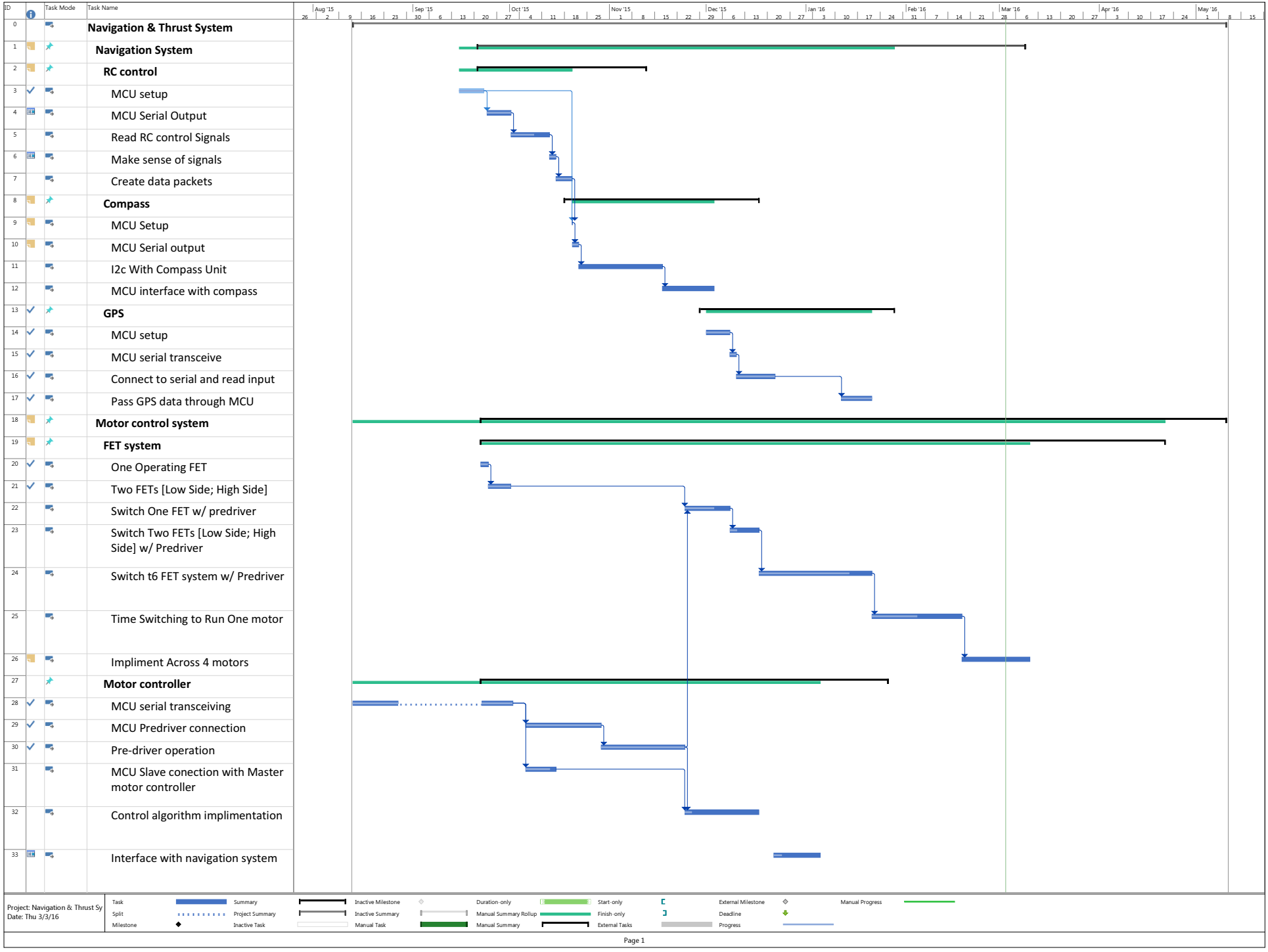
Michael S. Barnes: At this point the preliminary designs of the MOSFET circuit to interface the brushless direct current motor to the A4960 predriver has been completed. The next step is to verify the design, both by calculating the worst case power consumptions of the circuit as well as connecting the circuit to the predriver system Dan has designed to read any faults it may have. After the design is verified construction of four permanent circuits will conclude my work on the MOSFET circuit design for the Navigation and Thrust System.

Evan J. Dinelli: The majority of the GPS subsystem has been completed. The ATmega1284 can successfully communicate with the Adafruit Ultimate GPS via serial connection by writing commands and receiving raw GPS data. The system looks for GGA type NMEA sentences and once found, parses the latitude and longitude. The GPS coordinates are formatted in the degree decimal minute format. Results have been verified to accurately show the position of the GPS unit. Future work for the navigation system includes using I2C communication to read and process compass bearing. Current work is with the RC Control Unit. System design work has been completed and a 1 ms timer has successfully been set up. The next step is to trigger an interrupt on the rising edge of an RC pulse.

Dan R. Van de Water: I have been able to successfully finish the diagnostic function for the atmega644A to commune with the predriver. Display of the diagnostic data is observed through a serial connection on the computer monitor. The code on the Atmega644A has been rewritten to allow for interrupt operation. The code also has operation of a watchdog timer to maintain functionality in case a critical error occurs that causes the microcontroller to stop responding. I was able to implement an interrupt driven variable PWM signal to be sent to the predriver. A menu system to control the slave microcontroller has been set up such that I can receive command characters from the computer and change if the slave is to continually collect diagnostic data, what messages to transmit to the predriver, and the duty cycle of the PWM signal. Elementary Operation of the T100 thruster was achieved with significantly distorted diagnostic messages. These errors were corrected and accounted for to allow consistent observation of the diagnostic data. I accidentally burned a predriver when trying to determine the source of a recurring fault. The second predriver I worked with was determined to be faulty and was replaced. The third predriver functions fine, with the same errors I was observing before I damaged the first predriver. Suspected ground noise is the cause of the errors collected from the predriver. This currently under investigation. I also have a set up to test the motor's force capability constructed by Nick Schmidt. This will allow for collection of force data to determine the speed the thruster is going at different PWM cycles, which will eventually be used to create the closed loop control of the motor systems. I starting working on getting the Master Microcontroller operational for SPI communication. I have been able to successfully send and transmit messages as well as program the slave microcontroller to initialize into master mode after being placed in slave mode. This occurs after a specific number of characters which I hard code into the predriver. I will be working on connecting the master and slave together and obtaining a functional communication setup to transmit and receive information back to the master microcontroller.


ID
Task Mode
Task Name
Aug '15
Sep '15
Oct '15
Nov '15
Dec '15
Jan '16
Feb '16
Mar '16
Apr '16
May '16
0 Navigation & Thrust System
1 Navigation System
2 RC control
3 MCU setup
4 MCU Serial Output
5 Read RC control Signals
6 Make sense of signals
7 Create data packets
8 Compass
9 MCU Setup
10 MCU Serial output
11 I2c With Compass Unit
12 MCU interface with compass
13 GPS
14 MCU setup
15 MCU serial transceive
16 Connect to serial and read input
17 Pass GPS data through MCU
18 Motor control system
19 FET system
20 One Operating FET
21 Two FETs [Low Side; High Side]
22 Switch One FET w/ predriver
23 Switch Two FETs [Low Side; High Side] w/ Predriver
24 Switch t6 FET system w/ Predriver
25 Time Switching to Run One motor
26 Impliment Across 4 motors
27 Motor controller
28 MCU serial transceiving
29 MCU Predriver connection
30 Pre-driver operation
31 MCU Slave conection with Master motor controller
32 Control algorithm implimentation
33 Interface with navigation system
Project: Navigation & Thrust Sy
Date: Thu 3/3/16
Task
Split
Milestone
Summary
Project Summary
Inactive Task
Inactive Milestone
Inactive Summary
Manual Task
Duration-only
Manual Summary Rollup
Manual Summary
Start-only
Finish-only
External Tasks
External Milestone
Deadline
Progress
Manual Progress
Page 1