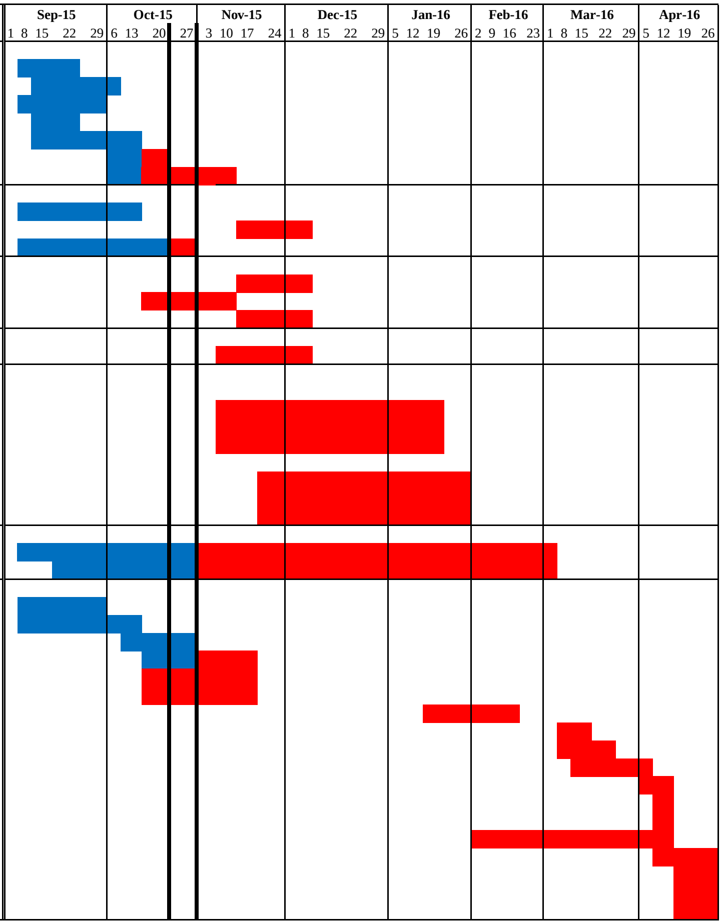
Task Name	Group Member	Finish by Date/Due
Individual Behavior		
Research Kilobot Sensors	Jared	September 28, 2015
Research Kilobot Communication Protocol	Jared	October 12, 2015
Research Q-bot Image Processing	Ryan/Greg	October 5, 2015
Research Q-bot Sensors	Ryan/Greg	September 28, 2015
Reseach Q-bot Communication Protocol	Ryan/Greg	October 19, 2015
Reseach E-puck Sensors	Brittany	October 26, 2015
Research E-puck Communication Protocol	Brittany	
Individual Communication		
Research/Test Kilobot - Kilobot	Jared	October 19, 2015
Research/Test E-puck - E-puck	Brittany	December 14, 2015
Research/Test Qbot - Qbot	Ryan/Greg	November 2, 2015
Integrated Communication		
Test Kilobot - E-puck	Jared/Brittany	December 14, 2015
Test Kilobot - Qbot	Jared/Ryan/Greg	November 16, 2015
Test E-puck - Qbot	Brittany/Ryan/Greg	December 14, 2015
Algorithm Design		
Design Linear Based Model	All	December 14, 2015
Integrated Behavior		
Formation Control Behavior		
Localization	All	January 25, 2016
Point Convergence	All	January 25, 2016
Leader Follower	All	January 25, 2016
Flocking Behavior		
Neighbor Repulsion	All	February 1, 2016
Enpoint Attraction	All	February 1, 2016
Heading	All	February 1, 2016
Testing		
Software Implementation	All	March 7, 2016
Hardware Implementation	All	March 7, 2016
Deliverables		
Project Proposal - Oral Presentation	All	October 1, 2015
Project Proposal - Document	All	October 15, 2015
Webpage Release	All	October 28, 2015
Fall Progress Presentation	All	November 19, 2015
Fall Performance Evaluation	All	November 19, 2015
Fall Performance Review	All	Decemeber 3, 2015
Spring Progress Presentation	All	Feburary 18, 2016
Student Expo Abstract	All	March 18, 2016
Progject Demostration	All	March 24, 2016
Final Presentation	All	April 7, 2016
Student Expo Poster Printing Deadline	All	April 11, 2016
Student Expo Poster Setup	All	April 12, 2016
Sudent Expo	All	April 14, 2016
Final Report (Draft)	All	April 14, 2016
Final Report	All	April 28, 2016
Final Web Page	All	April 28, 2016
Advisory Board Poster Printing Deadline	All	April 28, 2016
Advisory Board Poster Presentation	All	April 29, 2016



🟦 Complete 🟥 In progress

Cooperative Control of Heterogeneous Mobile Robots Network Update

October 28th, 2015

Jared successfully used an Atmega128 board to send a message to the Kilobot.

The kilobot was running a simple program that made the Kilobot's onboard RGB turns from red to green if a message was received, and then turns back to red after a set amount of time has passed without receiving a new message.

The Atmega128 was setup so that PB0 was connected in series to a 100 Ω resistor and an infrared LED, then connected to a common ground. The Atmega128's clock was changed from the default 1 MHz to 8 MHz as the Kilobot's onboard Atmega328p runs at 8 MHz.

The Atmega128 was running the code that Jared wrote, which was continuously sending a message of three integer values (100, 0, and 0). Although three eight bit values can only be sent at one time, a fourth value is needed and is used as a checksum. The checksum is simply the summation of the three values to be sent and 128. After the checksum is created, all four values are manipulated further to prepare them to be send.

After the values are manipulated, it begins to send out these values. To start a message, PB0 is set high for 0.75 microseconds, and then returned to low. At this point PB0 stays low for 92.25 microseconds. Now the message the specific message begins to be sent bit by bit. Each bit is sent out 8 times, with each bit requiring 13.875 microseconds each. To send all four bytes takes a total of 444 microseconds. To calculate the period of a message, the duration of the start signal added to the duration of the low state, and the time it takes to send the three bytes, which yields a period of 537 microseconds or a frequency of 1862 Hz.

For the next lab period Jared would like to check if the kilobot is receiving the correct values that the microcontroller is trying to send. The process can be done by connecting the Kilobot to serial cable and printing the message values to a hyper terminal.

In recent weeks progress has been made with the E-puck. The E-puck was first programmed to make each of LEDs (Light Emitting Diodes) blink on and off. The next step was to make the E-puck move forward. This step took about two and half weeks to find an example and execute the example. The E-puck can now move forward at a constant speed until the user shuts the E-puck off. The next steps include object avoidance algorithms, and communication with the Kilobot.

In the last week, Greg and Ryan researched different localization equations and created a new Simulink model. Ryan and Greg connected the GetImage block to the GetDepth block so that the Kinect will read the distance to the desired object. The distance and position of the desired object is used in the localization equations that were coded in a MATLAB function. The distances were read and were correct to the real-life distances.