**Autonomous Payload Delivery Challenge**

**Design Team:**

Rami Brikho

Mohammad Iqbal

David Trinh

Bao Phan

Jack Kennedy

Kyle Rodrigues

**Submitted to:**

John Kennedy and Lal Tummala

**Design Co. Ltd, San Diego, CA**

**Sponsored by:**



**Department of Electrical and Computer Engineering**

**I. Abstract**

As a team, Project Mercury is looking to design an autonomous vehicle that can navigate in an open space and deliver a payload (golf ball) to three different locations. The three locations will be provided as GPS coordinates specified as longitude and latitude. Each location will have a transmitting beacon that will be sending out a low-frequency signal. The ultimate goal for our vehicle is to be able to deliver a golf ball to each GPS location and return to the base within the allotted two minutes.

**II. Problem Definition**

Design an autonomous vehicle that can locate radiating beacons in an open, generally flat, field. There will be obstacles placed throughout the field that our vehicle will need to avoid. Given only GPS coordinates, our vehicle must navigate through obstacles, locate the beacons in the field and drop a golf ball directly on top of it. The entire vehicle must fit within a 20” x 20” x 20” cube. We are given $750 for parts to accomplish our task. The vehicle must also have a clearly labeled and functioning emergency stop switch.

**Functionality:**

1. Must be self-contained and autonomous.
2. Able to navigate through an open area.
3. Able to avoid obstacles.
4. Able to locate three beacons given only GPS coordinates.
5. Able to drop a golf ball on each beacon.

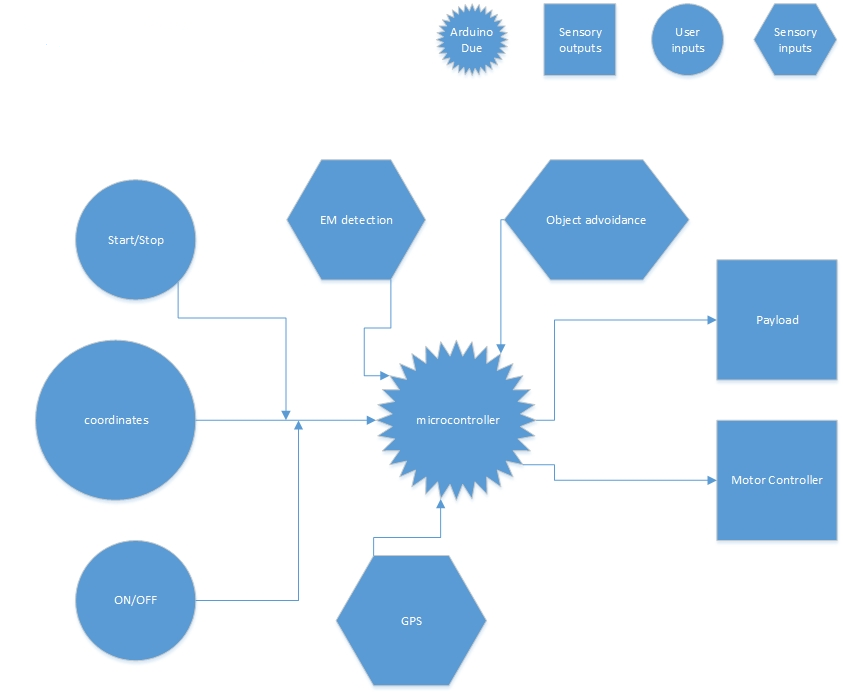
**Conceptual Design:**

The main structure will be the chassis of the Traxxas Stampede (Model # 36054) remote-control car. The heart of our design is the Arm Cortex microcontroller, which takes data from a GPS module and magnetometer to control the steering and speed of the vehicle. The magnetometer points the vehicle in the right direction, while the GPS module gets the vehicle to within 10 feet of the inputted coordinate. A ferrite rod antenna is then used to search for a low-frequency signal that is being transmitted by the beacon. Ultrasonic sensors are placed in the front, and on the sides of the vehicle to locate obstacles and help avoid them. The payload delivery system will consist of a PVC tube and servo motors to drop each golf ball individually.

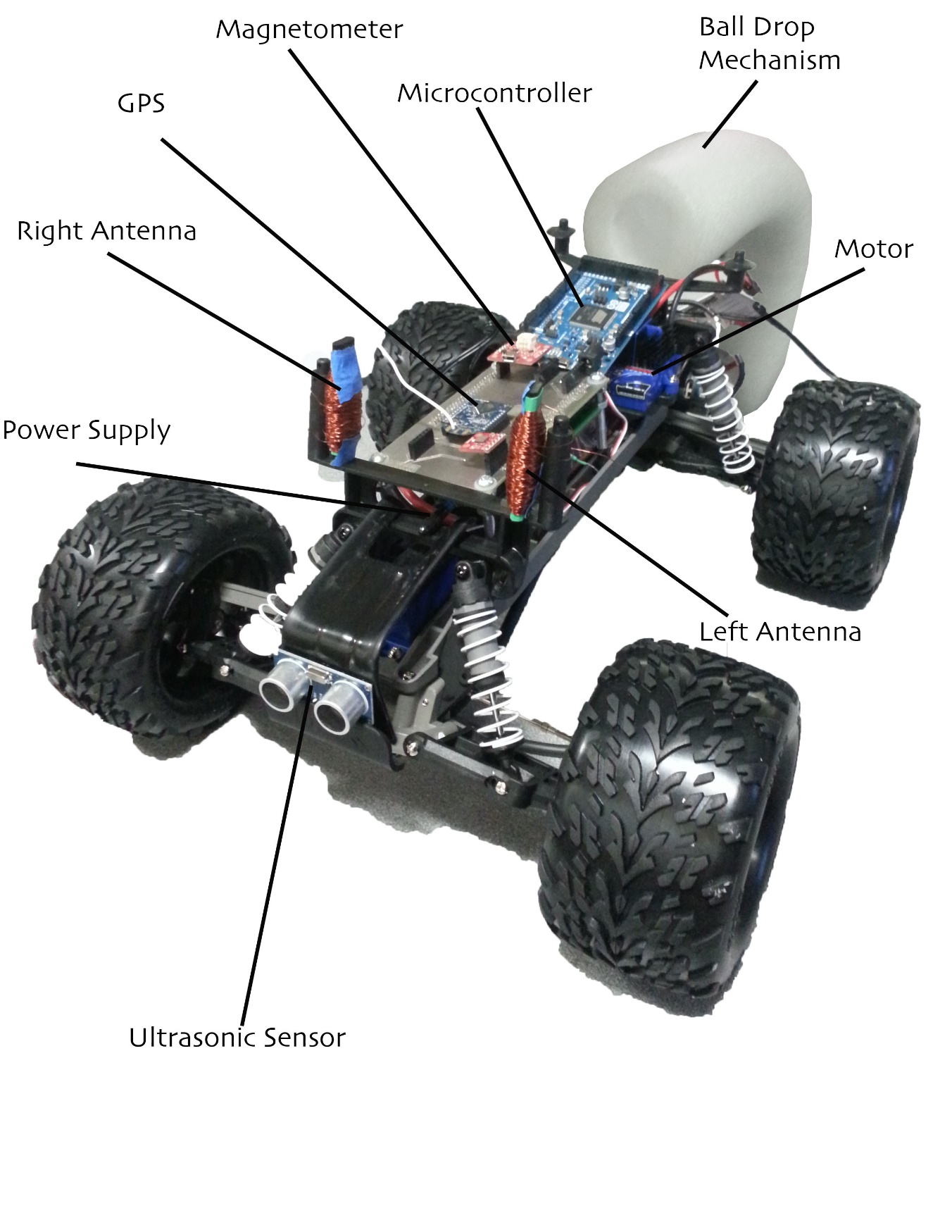
**III. System Design**

**1. Block Diagram:**

**2. Functional Diagram:**



**3. Mock-Up Illustration:**

****

**4. Background Research:**

GPS

The 3 payload drop-points will be given to us via GPS coordinates. So our project will require a GPS unit, which can tell the robot where in the world it currently is. The particular GPS module we are using for this project is built around MTK3339 v3 chipset, which can track up to 22 satellites on 66 channels.

Due to the nature of our competition, up-to-date positional information is crucial for proper vehicle navigation, so speed is a serious concern. Our module is capable of up to 10 location updates a second with a measured sensitivity of -165 dB with its built-in Aerial. Power draw for our GPS module is super low, at only 20 mA during navigation, power usage is not a concern.

Two features, which distinguish this GPS module, are the external aerial functionality and data logging capability. For the maximum aerial sensitivity, the GPS unit has a coaxial RF connector (SMA), which allows for strap-on and strap-off of an external aerial. The module will automatically detect the active aerial and switch over. Since there is a built-in microcontroller and empty flash memory, with a single “Start Logging” command delivered to the module from the microcontroller, the GPS will begin to internally log data. This is important because this

processing of data is done externally from the microcontroller, and all the microcontroller has to do is request the data from the GPS module, this will significantly improve performance and add to the stability of the system. The GPS is interfaced via serial connection to the Microcontroller.

The GPS (Global Positioning System) is to give accurate coordinates based on current location with an accuracy threshold of approximately 3.5 meters. The extended SMA connected Aerial brings the threshold down to 1.8 meters. The complete vehicle and payload mission are defined and programmed into the vehicle prior to the day of the competition. The particular GPS coordinates of the drop-off points will be given to us by John Kennedy the day of the competition. This information will be passed through as arguments to the main method of the top-level program, which will execute the day of the competition.

Microcontroller and Software

Among the various microcontrollers available on the market, we decided to employ the Arm Cortex as the central processing unit (CPU) of Project Mercury. Employing the Arduino software, a proven platform created specifically for projects such as Project Mercury, provides a vast support base, which caters to Do-It-Yourselfers and students alike. There is no need to download and setup a sophisticated Integrated Development Environment (IDE) such as Keil or IAR Systems or set environmental variables for command line interfacing. The Arm Cortex is programmed via an easy to use executable program called “*Arduino*” which checks the syntax and uploads files called “*Sketches*” into an Arduino Board via a communication port you choose on your computer. The Arduino Due is based on a 32-bit Arm Cortex-M3 architecture clocked at 84 MHz, giving us ample processing power and access to a plethora of libraries exclusively written for the Arm architecture on top of the Arduino Platform. The Due also has an upgraded 512 KB of flash storage for code, giving us ample space for the sophisticated programming required for a project such as this. The Due also has a Direct Memory Access (DMA) controller for, as its name implies, direct memory access to save processing power. The Arduino Due also has over 50 pins, for interfacing to external devices and sensors.

Because the Arduino Development environment gives us a comfortable level of abstraction based off the C++ programming language, which itself is a powerful performance orientated language, with heavy borrowing from the Process Hardware development language and an extensive supported code base, we saw no better option than to go with the Arduino Due. Most C++ constructs carry into the Arduino environment flawlessly; concepts like inheritance, polymorphism and object abstraction are just some examples.

Time is a serious consideration for our modus operandi so we, unofficially, decided to attack the software side of this project from a modular prospective; developing and testing units for this project in parts while working in parallel which will considerably help in our effort to meet project milestones. Arduino is an on-going open source project, and we are excited to contribute to this development.

Antenna

For this project we have to build a standalone ferrite rod antenna. This antenna has to be tuned to pick up an 80 kHz AM signal by adjusting the number of windings and put in parallel with a (calculated) capacitor. Next, the ferrite rod has to have circuitry to connect to the Arduino Due. This circuitry consists of an amplifier and precision peak rectifier along with a voltage follower. We need the amplifier because we need to be able to pick up the signal from a reasonable distance. The precision peak rectifier will not only keep the positive cycle of a sine wave but hold the peak voltage value within a capacitor because of the voltage follower. This value is then put through an A to D converter inside the Arduino to be able to set a flag as to when to drop the payload.

Motor/ESC and Servo

The motor is easily controlled by the Electric Speed Controller (ESC) with PWM. The PWM is also applied to the servo which makes programing the steering and speed of the chassis simple. The ESC controls the speed of the motor by allowing certain amount of voltage to the motor depending on what kind of duty cycle was sent using PWM to the ESC. The ESC also controls the motor to rotate backwards by sending a certain duty cycle with PWM. This concept is of certain duty cycle to control the motor also applies to a servo. The Arduino has a library called Servo which makes sending certain duty cycles very simple.

HC-SR04 Ultrasonic Sensor

HC-SRO4 consists of ultrasonic transmitter, receiver, and control circuits. When triggered it sends out a series of 40 KHz ultrasonic pulses and receives echo from an object. The distance between the unit and the object, is calculated by measuring the traveling time of sound and the output is a TTL pulse. This will be used for object avoidance. The HC-SR04 is able to sense objects at a range of 2cm - 400cm with a detection angle of 15 degrees.

**5. Testing and Verification:**

Motor

Test the speed of the motor by connecting it to its maximum voltage allowed and control the speed of the motor by using PWM via ESC. Use the oscilloscope to see what duty cycle is needed to make the motor spin full speed in one direction and the opposite direction. To finalize the test, find what duty cycle is needed to stop the motor from spinning.

Servo

Test the maximum steering angle the wheels can go to the left and to the right by sending a certain PWM from the Arduino Due to the Servo. Use the oscilloscope to verify the duty cycle

needed to make the wheels turn left and to turn right. Also find the duty cycle needed to have the wheels centered.

GPS

Test the accuracy of the GPS module by taking it to the field and standing in one spot without moving for 3 minutes to collect data. In order to verify the GPS module is accurate, stand at two other different locations on the field for another 3 minutes each. The second test is to determine how fast the GPS module is updated when it is in motion from one side of the field to the other. Finally, plot the data into Matlab via Scatter Plot.

Antenna

Test the Antenna with the oscilloscope while shorting the function generator set at 80KHz with a certain Vpp to see how well built the Antenna is. There are three tests that need to be conducted to verify the Antenna is ready. The first test is to insure the signal detected from the Antenna is not noisy. The second test is to insure that the Antenna is receiving the maximum voltage at 80KHz. Finally, the third test is to insure that the peak detector can envelop the entire signal.

Ultrasonic

Test the Ultrasonic Sensor by placing an object 1/2/3 feet away from the sensor and determine the accuracy of the sensor by using Serial Monitor. This test will be stationary and will be moved from 1 ft to 3 ft in increments of .5 ft. The second test is to determine how fast the sensor can

detect the distance if the sensor is moving from 3ft to 1ft.

Magnetometer

Test the Magnetometer by taking it to the field and read the values from the Serial Monitor. Stand facing North and rotate the Magnetometer clockwise and determine if it can detect that it is now facing East. Repeat the steps until the Magnetometer is facing North once again. This test can be conducted going counter-clockwise. Final test is how fast the Magnetometer can update by rotating it.

Power Management

Test the Power Management being supplied from the 8.4V battery included with the Traxxas Stampede. Using voltage regulators and resistors, we can send different voltages to different parts of the robot. Test these different voltages with a DMM to insure that the proper voltage is being supplied.

Payload Delivery

Test the Payload Delivery by making sure that the mechanics used to deliver work properly. Determine if the Payload Delivery system responds immediately when sent a command.

Arduino Due

Test the power of the ARM Cortex by separating each of the categories above, simulate and test each individually. Slowly begin to combine each functionality of the robot together and make sure the ARM Cortex can handle all the different functionality of the robot without much delay in the system:

**6. Benchmarks:**

Each component of project mercury would need to meet specific design requirements shown below:

Motor

The ability to drive straight, in curves, and stop from the pulse width modulations coming from the Arm Cortex processor.

Servo motor (1)

The ability to turn the wheels at select angles determined by the Arm Cortex while the motor are being driven.

Servo motor (2)

Turn payload rotor mechanism from input from the Arm cortex.

GPS

Give accurate coordinates based off current location within a 10 foot radius

Antenna

Pick up an 80 kHz signal from 5feet away from the beacon and send the relative signal amplitude into an A to D converter of the Arm Cortex.

Magnetometer

Send accurate coordinates with respect to a compass to the Arm Cortex within 10% accuracy

Payload mechanism

The ability to drop 3 golf balls independently by turning servo motor (2)

Microcontroller

The Arm Cortex processor must accurately interpret information from the GPS, Antenna, and Magnetometer. Also, the processor must drive the motors appropriately and drop the payload accordingly.

**7. Constraints:**

Economics – This project is quite affordable. Radio controlled (RC) cars are plentiful with many varieties. Parts for sensor and the microcontroller are inexpensive and fairly easy to build. The autonomous payload can be used for dangerous situations not suitable for human. The price of life is greater than the small budget for the robot. The robot can also be used to improve home life luxury.

Environmental and social factors – The robot will be navigating through an open soccer field. Since the tires used are off-road tires, it can easily navigate through grass, small holes and sprinklers. The brain of the RC car is the Arduino Due. The Due must communicate with GPS, antenna, magnetometer, ultrasonic sensor and motor drive all at once. The user must input location for the GPS and the robot will move accordingly to the sensors until payload is delivered.

Standards and Legislations – In the United States, regulatory responsibility for the radio spectrum is divided between the Federal Communications Commission (FCC) and the National Telecommunications and Information Administration (NTIA). Currently, only frequency bands between 9 kHz and 275 GHz have been allocated. Since we are working with 80 kHz, we are within law range.

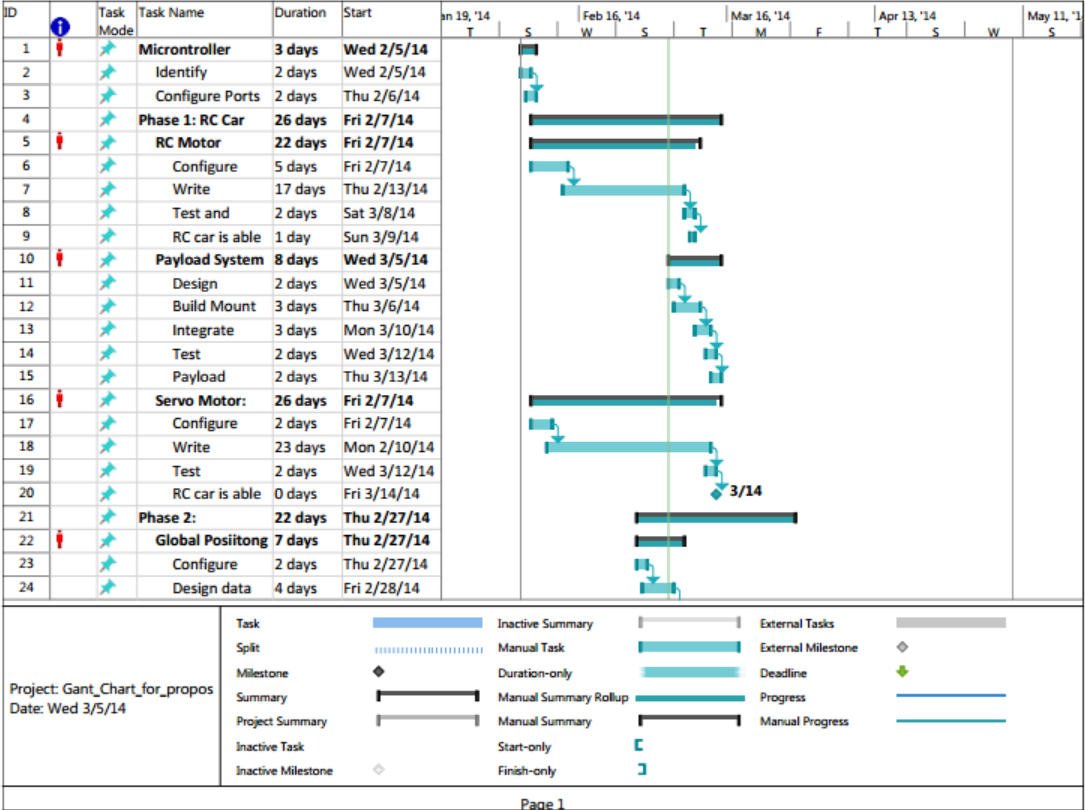
Maintainability – The robot can be easily maintained because parts are cheap and plentiful. Sensors are easily replaceable if damaged. The RC chassis body is made for outdoors/off-road, so it is made of durable material. Voltage/current should be regulated due to Arduino Due ability to only receive 5V max.

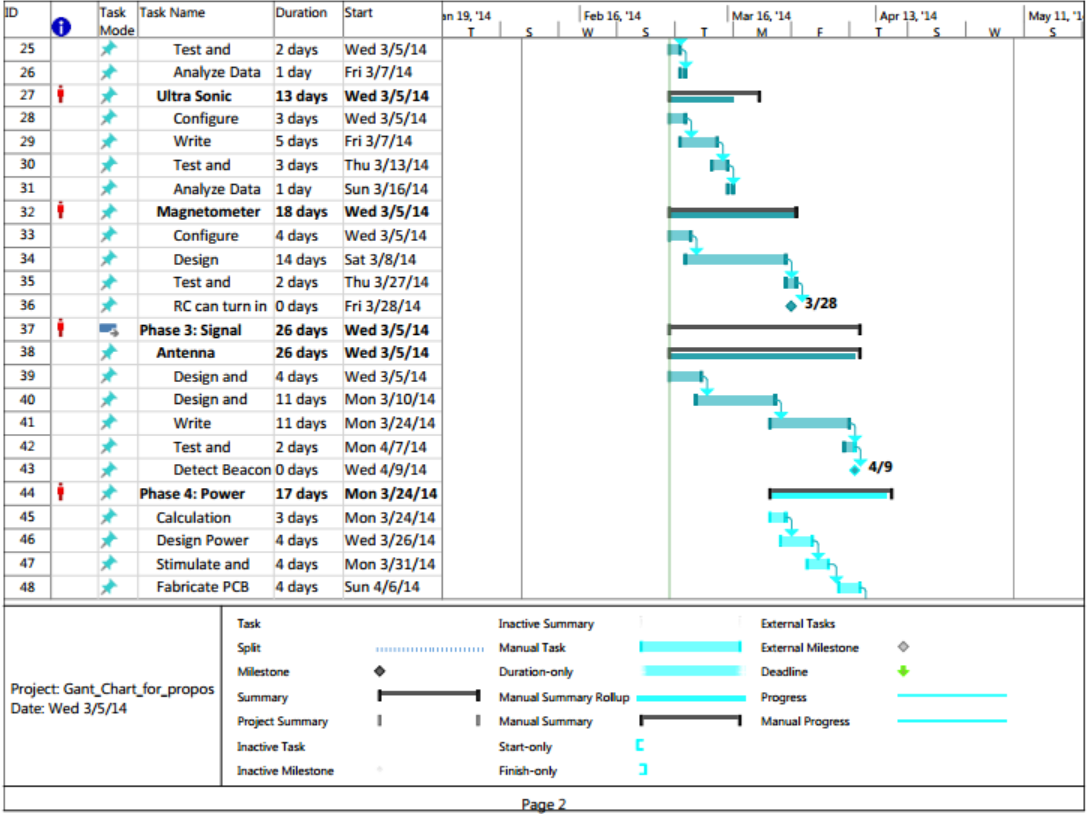
Serviceability – There are many online information about Arduino boards. The Arduino Cookbook and Make an Arduino-Controlled Robot also offer basic information for the Arduino board. The Arduino is made for everyone. Non-programmers can easily learn the Arduino language.

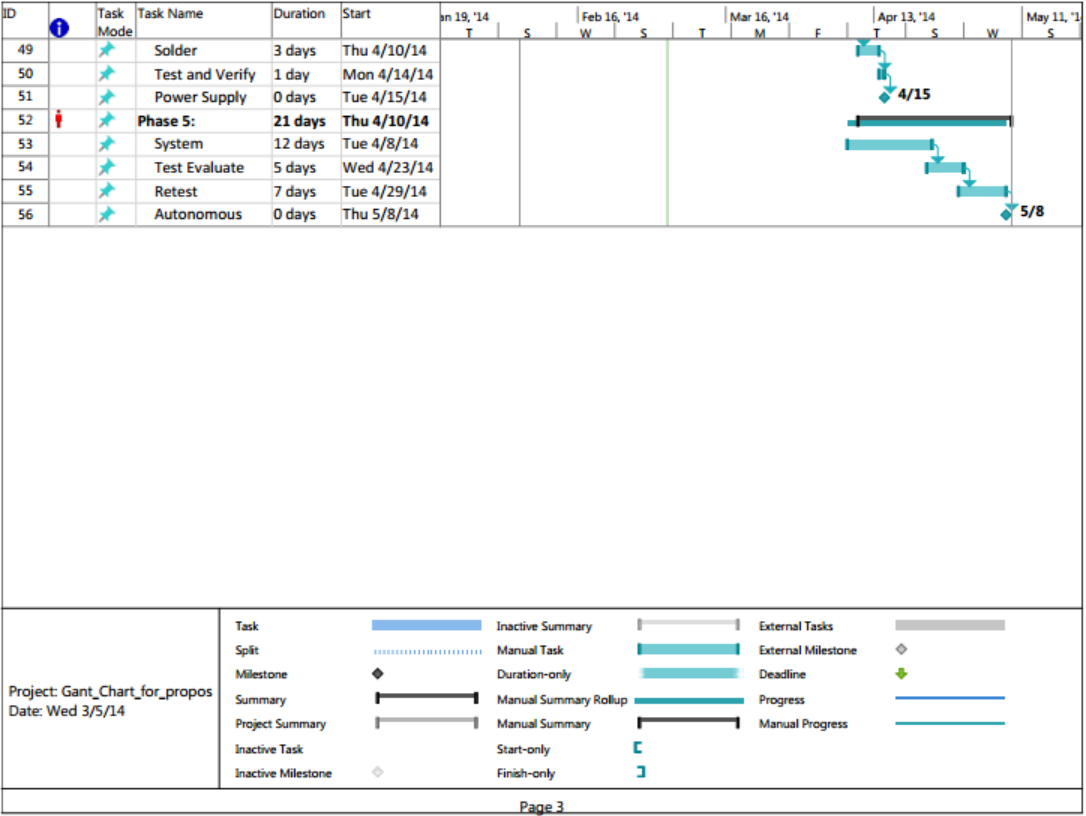
Manufacturability – Manufacturing will be simple once coding algorithm are written. Parts are easy to order, thus keeping manufacture price low.

**IV. Project Plan**

**Gantt chart:**

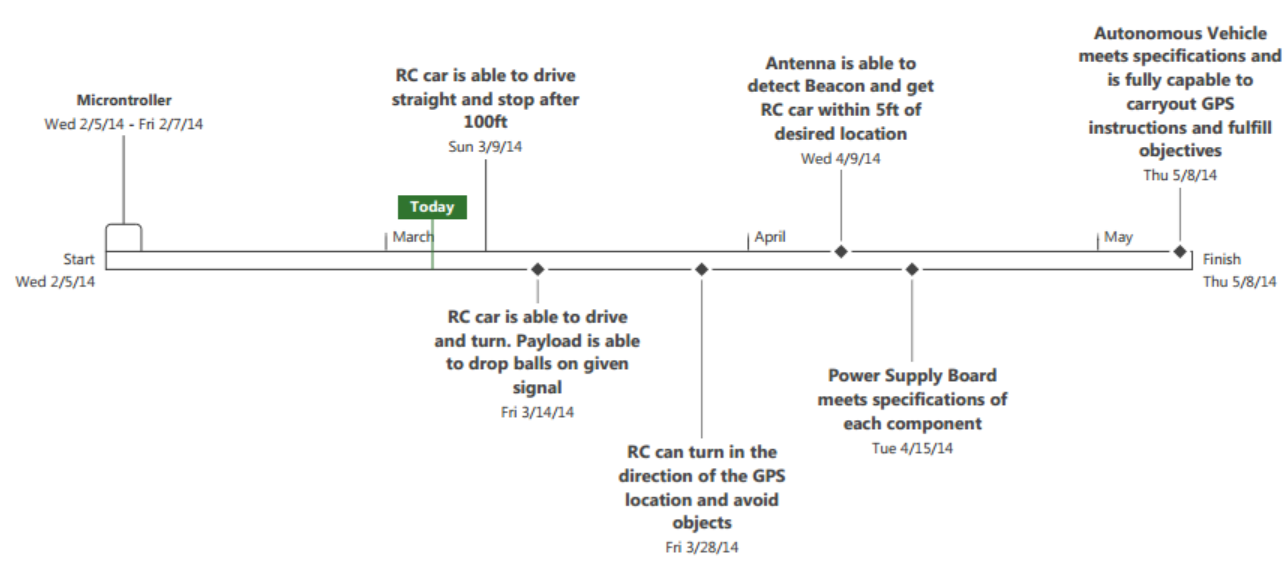
****

****

****

**Project Plan**

The project is split into five phases with specific resources allocated to each phase. The first phase will be to design the circuitry and algorithm that will control the RC motor and the Servo motors for the steering and the payload system. The second phase is navigation that will involve GPS, a Magnetometer, and an ultrasonic sensor to avoid objects. The third is to design an antenna that will allow the vehicle to accurately drop the payload in the vicinity of the radiating beacon. The fourth is to design a power system that will allow each component to operate using a battery. Once these prior phases have been completed, tested, and evaluated, phase five will integrate each phase so the autonomous vehicle is fully functional and the desired objectives have been met.

**Timeline:**

**Milestones:**

**3/14/2014**

1. Autonomous Vehicle is able to drive straight for 100 feet and turn. Also, the payload system is able to drop three payloads given a signal from a push button.

**3/28/2014**

1. GPS, Magnetometer, and Ultrasonic Sensor can interface with microcontroller to control the RC and Servo motor so that the autonomous vehicle is in the correct direction of a GPS location and is able to avoid objects while proceeding to that location.

**4/9/2014**

1. Electromagnetic Sensor is able to detect an 80KHz signal being transmitted 5 ft away from a beacon. It has to be fully integrated into the Arm Cortex and can signal the payload to drop its package.

**4/15/2014**

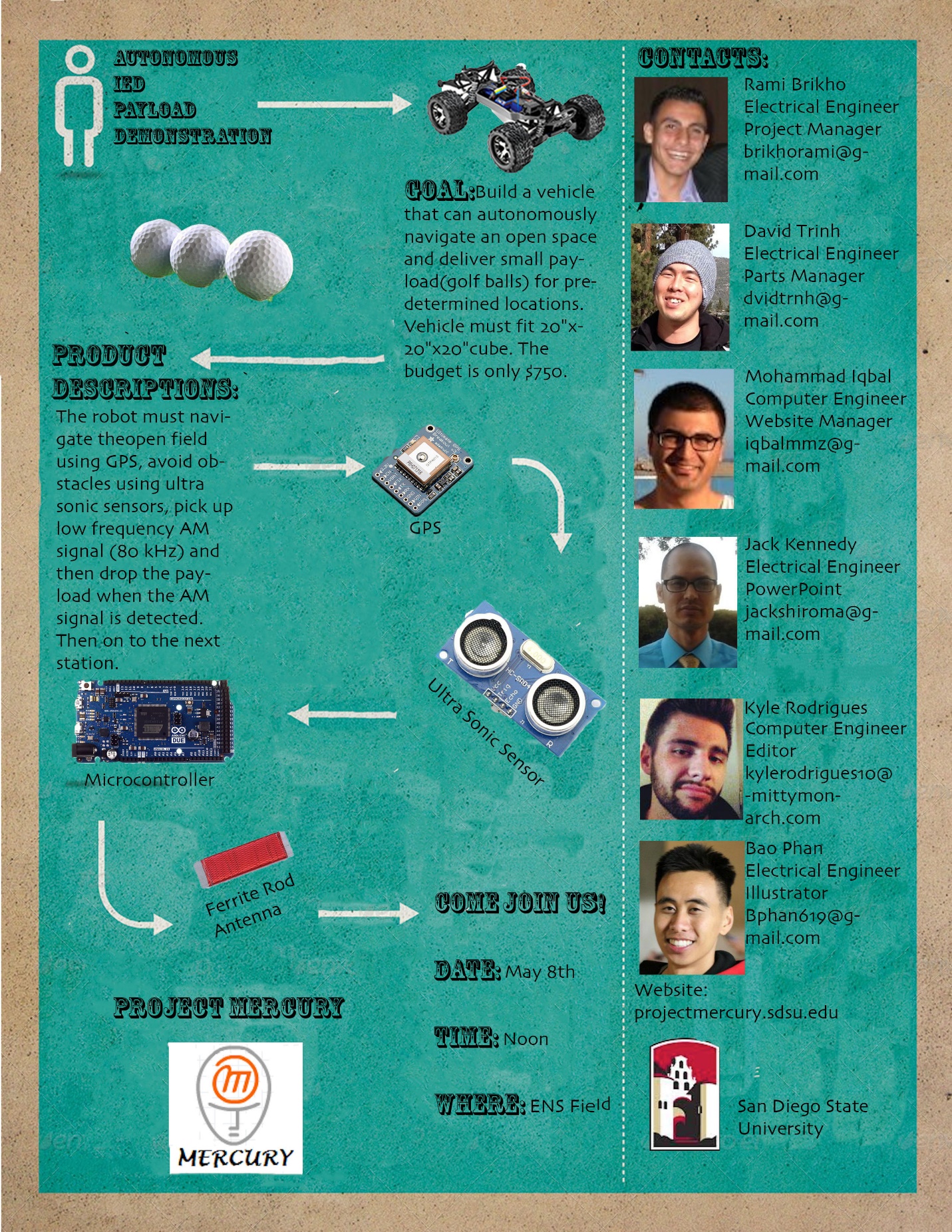
1. Power supply meets output specifications of +5V for analog components, Sensors, and the Electrical Speed Controller, +6V for Servo motors, +7V for the Arm Cortex, and +8.4V for the RC motors.

**5/8/2014**

1. Autonomous Vehicle is able to pin point GPS location, drive to the desired target, avoid objects, detect radiating beacons, and drop a payload on desired targets within 5ft.

**Proposed Budget:**

|  |  |  |
| --- | --- | --- |
| **Item** | **Description** | **Cost** |
| Power Supply | Barrel Jack with Batteries | 20 USD |
| Chassis | Prefabricated RC Vehicle with NiMH Battery | 250 USD |
| Microcontroller | Arduino: 1 Due, 1 Uno | 60 USD |
| Payload Mechanism | Servo motor with PVC | 30 USD |
| GPS Module | GPS Module with Aerial | 55 USD |
| Ultrasonic Sensor | No intelligence | 30 USD |
| Magnetometer | Compass | 55 USD |
| Circuitry | Analog Circuitry | 30 USD |
| Aerial | Full custom | 35 USD |
| Miscellaneous | Potential replacements | 185 USD |
|  |  | - |
| Total |  | 750 USD |

**Promotional Flyer: **