

1 - Executive Summary

The purpose of this project was to fabricate an apparatus which controls an external vehicle's every movement. The apparatus, here on referred to as the Cockpit, transmitted information via a wireless transceiver to an external vehicle, here on referred to as the RC Car. The Cockpit provided driving direction controls as well as regulate the speed and acceleration of the RC Car.

The RC Car consisted of a store bought radio controlled vehicle equipped with a middle grade analog camera fastened to the front and streamed real time video to the Cockpit display screen. The driver used the video feed to navigate the RC Car through the surrounding terrain. The RC Car will also had a MINIMU-9 sensor module that contained an accelerometer, magnetometer, and gyroscope. The information that was gathered from this sensor was then sent back to the cockpit in order to give the relevant information about the movement of the vehicle. This information in conjunction with the information gathered from the steering wheel and the pedal was then used to determine how the three actuators that were attached under the drivers seat should move.

Using the video information seen on the Cockpit's display monitor the user controlled the vehicle with the gas pedal, steering wheel, and buttons to maneuver the RC Car. The pedal and the steering where were spring loaded to ensure that the driver experienced slight resistance while operating the simulator. The pedal and steering wheel also contained potentiometers to that were realized by the RC Car and the car acted accordingly.

2 - Project Description

2.1: Motivation and Goals

When taking on a project such as this, a clear set of motivations as to why the project is being pursued should be outlined. Some primary motivations for the project were to learn about working in a team environment with collaborative constructive brainstorming, to solidify the knowledge attained of electrical engineering, to learn more about mechanical and computer engineering, to get involved in an in depth challenging project, and to create an interactive realistic experience for the user of the product.

When analyzing different project ideas, it became readily apparent that a project such as this would provide more than adequate design to understand a variety of interesting topics. This is important because the primary goal of this project is to learn about relevant technologies in the electrical engineering field. Some of the topics that will be understood by the completion of this project are software and hardware interfacing, precision control of analog devices by way of digital output from an MCU, analog and digital RF communications and interfacing, and data packing and transportation protocols.

2.2: Objectives

RC Ghost Rider was a realistic simulator that actualizes motion from an RC Car. The user was able to use the vehicle with very little extra knowledge other than very basic understanding about vehicle operation, simple fundamentals of batteries, and the safety protocol of operation. The cockpit accommodated people of varying sizes and weights. Safety protocol were established and accommodated, as safety was the greatest concern in all product usage. It was dynamic, and adjust based on intense and thorough testing procedures.

The RC Car was created to be wirelessly controlled, and thus, had the longest lasting battery life possible (within reason). Also, the RC Car was able to travel as far away from the Cockpit as possible, with little interference. This distance was limited (or even correspond) by the distance of the camera's transmission. Also, the RC car was durable, as to be able withstand the obstacles that it encounters. The RC car had a functioning camera, also with extensive battery life. A clear and detailed dynamic image must be displayed in the weakest delay of real time as possible. Wide angle of vision was displayed to give a greater awareness of surrounding area to the operator.

The Cockpit mwas safe, and achieved the most realistic simulations possible without our budget. It tilted based on forward and backward acceleration, and left and right based on turning. Also, vibrations from the RC car were wireless transferred to the cockpit to simulate realistic obstacle encounters such as

bumps and dips. All motion took place in the smallest delay of real time as possible.

2.3: Requirements and Specifications

Technically speaking, the vehicle needed to be small. That being said, it could not be too small as it will need to house a variety of pieces in order to function as intended. It needed to house one accelerometer, one gyroscope, a DC drive motor, a DC servo motor, two batteries, many small DC voltage regulators at 5 volts, an MCU, a transceiver and a transmitter. The sensors needed to be small, accurate and relatively inexpensive.

The accelerometer needed to measure at least 2g in order to sufficiently complete it's task, and thus was not as inexpensive as desired. It will also needed to be at least a two axis accelerometer (preferably three) that has a digital output mode. In addition it had to have a sampling frequency of at least a few hundred samples per second. To fit in the amount of space allotted to entire packaging of the device needed to be less than one square centimeter. The voltage that will be used by it was 5 volts.

The DC motor had a max spin speed of at least a few thousand RPM's. It took in a variable DC voltage as its driving force. It was small enough to comfortably house on the back of the vehicle, light enough as to not over encumber the vehicle, and strong enough to move the vehicle at an acceptable speed. The variable DC voltage range was within the range of 0-9V.

The batteries needed to have a voltage output of at least 9V each (preferably 12V) in order to properly power the DC motor that moved moving the car. In addition it needed to be at least 1000mA/H in order to give the experimenter a reasonable amount of time to test the vehicles operation. It was fairly small, and had approximately the same proportions as a toy RC car battery.

Three voltage regulators needed to be used in order to properly power the MCU and the DC motor. The MCU needed a stable power supply of between 2 and 5 volts (depending on the MCU of choice). The next regulator was responsible for the variable voltage of the DC motor. This regulator was able to fluctuate from 0-9V and be controlled by a serial digital output from the MCU that will give it a signal to drive the motor. The last voltage regulator controlled the behavior of the DC servo motor and was able to fluctuate from 0 to 5 volts to decide whether the vehicle will turn and to what extent.

The MCU on the vehicle needed to be a small unit that used a very small amount of power. It needed to have an operating frequency of over 10 MHz and power consumption under load of less than 2W. It was used to receive information from the components and relay this information to the transceiver to be sent to the

cockpit. In order to send the required information of at least 20 packets of 20 bytes each, the transceiver needed to relay information at a rate of over 1kBps in order to have a channel in which to move all the information that needed to be moved, at a rate that will be observed as real time to a person that is identifying the information coming in. Ideally the information moved much faster and thus the transceiver in question transmitted at a rate of greater than 5kBps. The range of the transceiver was sufficient to test the remotely controlled device and thus was over 50 feet indoors and over 150 feet outdoors.

The transmitter needed to operate on a frequency that is free to use, as to not violate any laws instituted by the FCC. The possible frequencies were 900MHz and 2.4GHz. Regardless of what transmission frequency is used it was necessary to implement a receiver on the vehicle that corresponds to the chosen frequency. The transmitter that was received must be able to send an analog video signal, and thus will need to occupy a bandwidth of at least 50MHz. In addition the transmitter had a range that is higher than the range of the transceiver that was used.

The Cockpit needed to have a comfortable chair equipped with a restraint harness, an adjustable steering column, adjustable foot pedals, a mounted flat screen TV, (that was used for the incoming video transmission) and a receiver. All of this was mounted to the bottom of the chair onto a metal platform. On the bottom of the chair, there were mounting brackets for three linear actuators, and it was decided that a shock absorbing central pivot point would not be used, because of the expense necessary to allow for it, as well as the fact that it was unnecessary. The actuators moved up and down at a rate greater than one and a half inches per second under a load of at minimum 350 pounds per actuator. Using these standards, it can be said that the apparatus will move enough to offer an enjoyable simulation experience. The comfortable chair was a seat from a Chevy 1500. It was equipped with a five point harness in order to ensure the safety of the rider. The seat was in relatively good condition without any rust or structural damage. The harness was fastened to a metal mounting bracket that will be fastened to the back of the seat.

The adjustable steering column sat at approximately the same level as one would assume it to be when driving a vehicle. It had a pivot joint in order to allow it to pitch up and down as a normal steering column would. The control system for the steering column consisted of a wheel retraction mechanism (which will bring the wheel back to center) as well as a potentiometer that will measure the difference in the wheels spin, relative to the zero position. This potentiometer needed to be very sensitive in order to properly account for the minuscule changes in the system. In addition the wheel retraction mechanism needed to be fast, and slightly over-damped. This way when the user was to let go of the wheel, it will return to center for a more realistic user experience.

The adjustable foot pedals needed to be mounted to the platform that extends from the chair. They needed to be at a distance away from the user, such that it would feel akin to riding in a vehicle. In addition the pedals needed potentiometers in order to judge the movement of the pedals. That being said, the pedals also needed to be spring-loaded. The purpose of this is to allow the pedals to spring back to their upright position after they have been depressed. Since the position of the seat was static, the pedals will also need to be placed in the proper location on the floorboard to allow for various sized users to interact with the apparatus.

The linear actuators were controlled by PWM to DC output networks that would receive its control input from the Cockpit's MCU. This voltage regular needed to be able to sync a maximum of 20 amps of current at a voltage of at least 24 volts. There needed to be one voltage regulator per actuator The MCU delivered a pulse width modulated voltage signal with a duty cycle between 0% and 100% that needed to then be translated to a reasonable voltage by the regulator, to the actuators. Because there were three actuators there needed to be one PWM to DC output network per actuator.

All of the readings were to be taken in by the MCU located on the cockpit and converted into usable information. The readings were read one at a time, and three readings needed to be stored in memory as an array. These three readings were to be averaged to calculate the average value of the voltage and eliminate possible noise. The receiver that was used by the cockpit should likewise operate on the same frequency that is picked for the transmitter on the vehicle. This receiver was able to accept the input and output the information to the display unit that will be located on the cockpit.

3 - Research Related to Project

3.1: Similar Products

When designing project RC Ghost Rider, two similar products were used as the basis of the cockpit design. They were the *Blue Tiger Motion Simulator* and the *Force Dynamics 401 Racing Simulator*. While no specific part of RC Ghost Rider was specifically mirrored or mimicked, some design features from each of the aforementioned products were used as a starting point during the cockpit design portion. These products have been thoroughly tested for efficiency and safety; however, given the lack of research resources and development time for this project in comparison to these products, it was more than necessary to understand, compare & contrast, and dissect these products to determine which attributes would be best suited to the RC Ghost Rider project. When using these products as benchmarks and references, extreme care was taken to ensure that RC Ghost Rider was a distinct one off design but it undoubtedly shared similar qualities. The products listed below are strictly racing and flight game simulators, whereas RC Ghost Rider afforded users the opportunity to control of an external remote controlled vehicle. The largest difference in this project versus similar products is that it simulated the movements of another physically moving object, where the other two are virtual reality simulators.

Blue Tiger Motion Simulator

The Blue Tiger Motion Simulator seems to be the cream of the crop when it comes to prefabricated simulators. It appears to have the most aesthetically pleasing appearance and uses the highest quality parts. Its three monitor visual display unit incorporates state of the art monitors with high resolution and refresh rate (120 Hz). The material used to construct the simulator is comprised of high grade steel, fiberglass, and plastic. The body is concept design with jet fighters and formula cars as the inspiration.

The base of the simulator houses the power supply, control electronics, actuators, and motors. The body is the part of the simulator that moves and is attached to the top of the base and actuators. There is nylon material between the base and body to protect the actuators from dust and debris. The base also uses outriggers which can be bolted on for added stability.

This simulator uses an unknown amount and type of actuators to control the body. The actuators are located directly under the center of the seat with a rubber and plastic casing around them providing both a base and also to keep the actuator assembly hidden. It uses a combination of fast and slow frequency of movements proportional to the action in the games to convince the rider that they are experiencing both the dynamic forces of acceleration and braking. The actuators also move the body into the same physical position that one would

expect to be in if they were inclining and declining on a hill or slope. There was also care taken to provide a variety of special effects such as engine vibrations, road surface texture, and minute and frequent bumps in the road. This is achieved by the actuator control circuit just as the RC Ghost Rider is. The Blue Tiger's maximum weight payload that the actuators can properly function concerning both the rider and accessories is 350 lbs. The actuators can also provide is positive or negative 20 degrees of independent pitch (tilting the body left and right) and positive or negative 20 degrees of independent roll (inclining and declining the body forwards and backwards), and an 8 inch sway (moving the seat forwards and back on the lateral plane). The RC Ghost Rider was not be capable of this movement given that the front two actuator were stationary and placed in a vertical position with bracing used to ensure no movement laterally of the actuators was possible.

The body is comprised of lightweight fiberglass, where-as the RC Ghost Rider was fabricated using steel and aluminum. The Blue Tiger is a comprised of two pieces, a one-piece chassis around the seat and footwell, and a one-piece front cowling, monitor enclosure, and dashboard. There are two prefabricated bodies available and there is also an option for the buyer to help design a custom one off version.

One glaring difference between this product and RC Ghost Rider is that there are various components that can be added to the simulator to enhance the user's experience. Blue Tiger offers a variety of seat, steering wheels, and joysticks available to consumers. Project RC Ghost Rider offered no upgradeable components or modes. This was due to the predetermined budget and also that RC Ghost Rider was not a consumer grade product with no immediate plans to recreate and redistribute for financial gain. Another is that three monitors can be integrated, equally distributing the image into thirds, with each screen displaying its corresponding portion. This option is designed so the driver can easily view the video game. The Blue Tiger is designed to accommodate up to three 42 inch flat screen televisions or monitors. Another is that the Blue Tiger Motion Simulator is both a racing and flight simulator. The company is currently working to make PC and internet games compatible with the simulator. The Blue Tiger offers either 2 or 3 pedal assemblies with the latter including a clutch pedal. This was not needed in RC Ghost Rider due to the fact that it was controlling an electric remote controlled vehicle with no transmission. Also, The power consumption of the Blue Tiger is much higher than RC Ghost Rider, with a 700 watt maximum load drain and a 200 watt no load drain.

There are also many similarities between these two products though. One is that both were completely electric to create a quiet and clean user experience. Both could be operated from standard wall sockets. Both simulators offered a 3 point harness system to keep riders placed firmly, comfortably, and safely in the drivers seat. It should also be noted that RC Ghost rider's 15 degree pitch and

roll standard was based on the Blue Tiger.

Force Dynamics 401 Racing Simulator

The Force Dynamics 401 Racing Simulator was more closely related to the RC Ghost Rider. The 401 implements a three actuator system with a series of ball and hinge joints used to connect them to the body of the simulator and recreate the forces and movements felt by the car in the video game. Just as the Blue Tiger, this product uses the highest grade components available to create a realistic driving experience. It is however, another video game simulator not a mechanism used to remotely control an external vehicle.

The 401 Racing Simulator provides four-axis motion that utilizes pitch, roll, yaw, and heave. The 401 has the most unique design with regards to the motion of the body. All movement is based off the driver's shoulders as being the relative center of mass. The roll axis is located around the shoulder level on the driver's plane of symmetry. This plane of symmetry cannot be precisely located given the varying sizes of the simulator operator. The pitch axis is located somewhere around 4 inches forwards of the driver's shoulders again, but perpendicular to the driver's estimated plane of symmetry. The yaw axis is based off the vertical plane and uses the driver's symmetry, it is approximately 20 inches forward of the rider's shoulders. For RC Ghost Rider the center of mass was assumed to be around the driver's waist/hip area.

The one glaring draw back to the 401 when compared to the other two simulators is the maximum load the actuators can handle to efficiently operate. The maximum passenger weight is 280 lbs and the maximum passenger weight for full performance is only 220 lbs. Project RC Ghost Rider was designed to comfortably handle a maximum full performance load of 300 lbs.

The linear thrust actuators can provide is positive or negative 30 degrees of independent pitch and positive or negative 30 degrees of independent roll, positive or negative 90 degree of yaw, and 8 inches of sway just as with the Blue Tiger. These are very good statistics considering RC Ghost Rider boasts dynamic performance values of positive or negative 20 degrees of independent pitch and positive or negative 30 degrees of independent roll. RC Ghost Rider lacked in the area of sway with its measuring at a constant 0 degrees of sway due to the actuator only operating in an up down motion with no dynamic movement about the Z axis.

Another unique design aspect of the 401 is how the images and video is displayed to the display screen. The screen consists of a powder coated aluminum sheet surrounded by a steel frame. It is a neutral gray color to enhance contrast with brighter ambient light. The Blue Tiger and RC Ghost Rider both use mass produced monitors to display video, the 401 uses a projection system. This feature separates the 401 from the other two simulators and also shows another

level of sophistication and uniqueness in the 401's design. This ensures that the video display unit is free from all problems associated with vibrations, but adds a dimension of new problems in that the projection lens must be at the perfect angle at all times and that nothing can obstruct its projection path.

The power supply of the 401 is compatible with a standard wall socket, but in an added design enhancement it can be used in not only North American wall sockets, but in many European wall sockets as well. It is equipped with an isolation transformer and can be used with common non-polarized outlets found throughout the world. There are many safety precautions taken in the 401, one being that it contains a relay with self-checking contacts, as well as software monitoring contacts. There is also a separate manual reset function to make sure that if any malfunctions or anomalous behavior occurs it can be manually shut off by someone other than the driver. Located on the body of the 401 there is another emergency stop switch within close range of the driver. The seat belt and door both have monitored locks that place the machine in pause mode when they are not in the locked position. The RC Ghost Rider has also taken extensive measures to ensure driver and simulator safety by also placing a power on/off switch within arm's length of the driver, as well as the type of harness that was used to secure the driver in cockpit seat.

The body of the 401 is closer in material used to the material used for the construction of the RC Ghost Rider than the Blue Tiger. The strut tubes are comprised of 4140 steel. The frame is fabricated from low-carbon steel and a textured polyester powder coating used to combat erosion. There are also various aluminum parts used to construct other components on the body. The electrical components are mounted in a separate structure fabricated from Galvalume. This structure contains a series of fans to keep circuitry cool. There is a 25 Amp single pole circuit breaker that is used to protect the circuit in case it is connected to an unfused power source or a power surge is experienced. The RC Ghost Rider housed all of its cockpit electrical circuits in a housing not directly mounted to the cockpit to ensure that there was no noise or vibration interference caused by the actuators felt upon the circuitry. Just as with the Blue Tiger, the 401 comes available with a 2 or 3 pedal system and a transmission shifter. The RC Ghost Rider was again only available in one model and it has a 1 pedal system. The RC car that was controlled did not contain a transmission so no clutch pedal is required.

The 401 adds a dimension to its simulator that neither of the other simulators does, audio system. The 401 utilizes a brand name five-way speaker system with two being located on the side and one on top of the screen, and two located behind the driver's head. A sub-woofer located in the base just below the rider.

3.2: Relevant Technologies - RC Car

3.2.1: Battery and Power

In order to adequately power the various components that will be used on the vehicle, it was necessary to have two 12V battery Packs. The primary battery pack was a 12V Nickel Metal Hydride 3800mAh battery pack. That battery is used to power the forward motion of the vehicle, as well as all of the electronic components on the vehicle. The other battery that was used was a 12V 1800mAh Lithium Ion pack. That battery is used to power the negative voltage on the operation amplifier, and the reverse motion of the vehicle. The dimensions of the batteries are 9.6" X 1.5" X .5" for the Nickel Metal Hydride battery and 4" X 3.2" X .5". The batteries are wired in series and the common node for the circuit is the interconnect of the batteries, such that the the Ni-MH battery's anode shows 12V, while the cathode of the Li Battery shows -12V relative to the center interconnect. Because of the capacity of the Ni-MH battery, it can be used for the motion as well as the component power.

In addition to the battery power, it was also necessary to properly regulate the voltage that was applied to various components. Two LM7805 voltage regulators were used to supply the proper voltage to the components on the PCB. One was used to power the MCU, while the other was used to power both the Xbee communication module, and the Shift register. To dampen the effect of the oscillator it was necessary to tie 10uf capacitors across the 5V output of the regulator for the MCU to both ground and 12V. This reduced the bounce that was caused by the oscillator.

3.2.2: RC Car and Accessories

The actually accessories of the vehicle needed to be picked in order for the project to begin to come into fruition. The skeleton that was used for the RC car was a mid 1990's RC race car frame. It measured in at 22"x 14" and weighed about two pounds. The vehicle was equipped with independent suspension on the front two wheels, as well as a small coiled suspension that both back wheels utilized. The vehicle was acquired for free from Paul's uncle.

The servo motor that comes with the car was exchanged with a Tower Hobbies PWM - DC servo motor. This servo motor was selected because of its input voltage, price, speed and total angular movement. The voltage range of operation is 4.8V to 6V. This voltage coincides nicely with the output voltage from the LM7805. In addition the price point of \$0 was rather pleasant, as it was donated to the group. Likewise the speed of motor is a surprising 438 degrees per second, which works out to a speed of 71 RPM. This speed of motion is more than adequate for the purposes of controlling the vehicle, considering that

the motor has a maximum angular displacement of 40 degrees. In addition, a separate drive motor was selected, because the speed of the vehicle with its current motor (not moveable) is unacceptable. For that purpose a GWS RS-777 DC motor has been selected. This motor operates at 7.2V and thus a circuit was devised to properly control it. Under no load the current draw of the motor is 700mA and the speed of the motor is 16,000 RPM. The motor has a stall torque of 836 g/cm with a stall current of 21.2A. For the purposes of this project, these values should never be necessary, and to ensure this point does not get reached a 15A fuse will be placed on the motor drive voltage input.

3.2.3: Processing Platform

When searching for a micro-controller to be used on the vehicle, there were many factors that were taken into account. Among those factors was the unit's price, size, power consumption, processing abilities, and analog to digital inputs. When weighing all of the potential prospects for this type of application, it was decided that the Atmega328p would be the best fit for this particular situation. When concerned with the price, the development board version of the chip seemed to fit well within the constructs of the design specifications. The processor met both the size constraints that have been put in place, as well as the speed constraint. The competitors could not match the ease of programming. That being said, there are some drawbacks to the selection of this version of the Atmega328p but to the project's specifications, these drawbacks should not be an issue.

Based on the scope of the project at hand, the main functions of this micro-controller unit were to take voltage readings from the digital components and relay that information to a transceiver that will move the information from the vehicle to the cockpit, as well as driving the vehicle. For this functionality the Atmega328p worked swimmingly, as it only needed to take in a few hundred readings per second, and relay the information it takes in. Even though it only had 1024B of RAM, this shouldn't be a problem, as most all operations took less than 128B, none of the information needed to be stored and was easily overwritten and disregarded.

With the extremely low power consumption of the device, the mobility of the vehicle was not dramatically affected by its power usage (compared to the vehicle's motor, servos, etc). When operating at 1 million instructions per second (MIPS Active) the device uses approximately 220uA. With an input voltage of 5V volts this means that the power consumption of the device under a considerable load would be about 1.2 mW. Considering the specifications of the parts the car will be using and their power consumption, this small number is practically negligible. The mention of the MIPS Active rating was a very high approximation of the actual power consumption of the micro-controller, because the design specs allowed the controller to use significantly less power, based on the fact that it was performing far less calculations than this rating is intended for.

The size of the parts in question was also another determining factor in what type of MCU (micro-controlling Unit) will be used for the design. It was very important that the unit is small and could be mounted relatively easily to a remote controlled vehicle. Taking these specifications into consideration, there was yet again another reason to use the Atmega328p. With the size of the components and the varying packaging arrangements available, there was no problem getting a solution that met the requirements for space in question.

This micro-controller's input pins were used to take the information from the active/passive circuit components used to measure the vehicles disposition. Because of the fast clock rate (16 MHz) there was no problem reading all of this information. In addition to reading the information the MCU was responsible for making calculations on the data that was sampled in order to verify the legitimacy of the received signal. This was done by taking multiple samples of one digital input and averaging it over time. For instance in taking 45 samples from the accelerometer each second, the programming that needed to take in three samples at a time into an array and use the average of the three samples, then after computing the average of these samples, they needed to store the last average in a variable so that it can be compared to the new average, in order to get the change in how gravity is acting on the part.

3.2.4: Sensors

There are various sensors that needed to be implemented and retrofitted onto the RC car. Among those sensors was the MINIMU-9 which contained an accelerometer, a gyroscope and a magnetometer. These sensors took in information and relayed that information to the MCU so that it could be used by the cockpit to determine how to adjust the cockpit accordingly. The main reason behind the selection of the MINIMU-9 chip was the fact that the chip contained the two main parts that were necessary to calculate the disposition of the vehicle. Having both of these parts in one easy to use I2C package allowed for a simplistic interface between the microcontroller and the sensor.

3.3: Relevant Technologies - Cockpit

3.3.1: Power

The Cockpit apparatus will be supplied power using two DuraComm RM-5024 power supplies. These can operate using a wall socket connection (120V, 60Hz). RM-5024 can supply 23-29 volts DC at 50A, so 100A can be supplied using a parallel connection in using two. Approximately 100A is required. In the worst case scenario approximately 90A will be drawn from the power supply. This is a switching power supply, so efficiency in power delivery takes place. Also, this quality provides that the power supply is small, and light-weight. A 12V supply is not a sufficient supply for the cockpit operation, as a 12V supply would not

reliably deliver 12V to the actuators when required. If the average operation of a 12V supply is 12V, then it will almost never be able to supply 12V to an actuator (as some voltage is dissipated by the source follower in the PWM to DC output stage) limiting extension speed. It should also be noted that this is a non-isolated power-source, so supplying negative voltage is not an option.

All of the cockpit will be supplied power using this power supply, aside from the visual interface and the -5.1V DC power supply. However, not all components powered by the power supply require 24V to be powered, so LM-78XX series voltage regulators will be used between the power supply connections to the devices that require DC power. Note that all schematics assume a 24V DC power will be supplied; However, voltage fluctuations are bound to happen. Therefore, an LM-7824 will be used on all 24V connections where another voltage regulator is not used.

The DuraComm RM 5024 is typically used in communications, and is typically valued new at \$737.33. However, two of these power supplies have been obtained, and are available for use. Commonly these are used on rack mounts, making it simple to mount to any housing apparatus. It is 3.3" deep, 5.95" wide, 5.5" tall, and weighs 6.5lbs. Due to such specifications, transportation of the apparatus is made easier.

The display and 900MHz receiver are supplied power indirectly from a wall socket, as it was previously designed to do so. There is no reason to bypass this specification and power the display from the power supply, as it would likely cause an issue in maintaining the reliability of the display. Also, it would cause unnecessary current drain from the power supply. Subsequently, the power supply, the -5.1V power supply, the display, and the display's receiver are powered using a power strip (obviously connected to a wall socket) mounted in the housing made for the circuitry.

Due to the price of high current DC power supplies, it should be noted that all designs were made around the basis that only positive voltage will be supplied to components and devices that require high current. Thus, adjustments have been made to accommodate requirements for actuator retraction.

3.3.2: Cockpit Processing Platform

To choose a processing platform one must consider cost, reliability, performance, and ease of use. Many different processing platforms were considered for this part project, such as the TI Piccolo board (which was much too expensive, though the specs were better), and some 16-bit processors such as the Arduino and the MSP 430 (both of which are not powerful enough to process the information required for the project). Ultimately, it was decided that a 32-bit processor which was affordable and open source would be the best option. The

Netduino, currently, is the best processing platform balancing all of those things. The Netduino is available at \$34.95. This is a major benefit, as many 32-bit electronics platforms are more expensive. Comparable platforms range from \$70 to \$120. All accessories necessary for programming the platform are also included. This product is available for purchase on-line consistently.

The micro-controller used in the operations of the cockpit unit will be the Netduino. This is an Atmel 32-bit micro-controller that operates at 48MHz. The code storage is 128 KB, and consists of 60 KB of RAM. The processing speed will be sufficient to perform the tasks required; the 128KB of code storage is certainly more than sufficient to store the code necessary for the RC Ghost Rider cockpit; the 60KB of RAM is much more than necessary, as the most of the stored elements of the code will be replaced very shortly after execution.

The I/O features include 20 analog and digital I/O pins. 14 of which are digital I/O pins, and 6 are analog inputs which can also act as digital I/O pins. All 20 pins are General Purpose I/O. The pins have a variety of specific uses; however, not all will be used for the purposes of this project. Pins 5, 6, and 9 will be utilized for their PWM capabilities; 6 analog input pins will be utilized as intended, and 4 other pins will be utilized for GPIO utility. This was a major reason to choose this board, as the abundance of I/O pins is necessary to meet the goals and specifications designated.

The input required is 7.5-12.0 VDC, or USB powered. The output voltage is 5 VDC and 3.3 VDC regulated. The analog reference is 2.6 – 3.3 VDC, but only required when using analog features. Max current is 8mA per pin (Digital pins 2, 3, 7: 16mA per pin; analog pins 0-3: 2mA per pin; micro-controller max current 200mA). Digital I/O pins are 3.3 V, but 5 V tolerant. Default pulse width modulation frequency is 10 kHz, but it is most probable that 1 kHz will be utilized. The operating temperature is within the range of 0 - 70 °C (32 - 158 °F). These specs are standard for electronic devices comparable to the Netduino, but are necessary to perform actions such as PWM and controls. It should be noted that the electrical design of this project should not be taken lightly, as nearly 100 times the current tolerance is may be used to power the linear actuators. This issue would not have been avoidable had any other processing platform been utilized.

Design files and source are easily available, as the Netduino is an open source electronics platform. This is a major benefit to using this particular platform, as the code and files are easily and immediately available, and completely free. The support for the product is the community that uses it. Hobbyists make use of open source platforms such as the Netduino and the Arduino to perform a multitude of tasks which include electromechanical designs like this one, so it is optimal for use in this project.

3.3.3: Actuators

The most vital and essential components of the cockpit, and possibly the project are the actuators. The actuators that were chosen recreate the movement of the RC car were linear thrust actuators. These are screw driven actuators and were chosen for their simplicity and strength. They don't require a large cumbersome compressed air tanks like pneumatic actuators and they don't use messy and expensive fluid where- as hydraulic actuators do. These screw driven actuators operate under the principle of the simple screw machine and do not implement a vast amount of moving parts. This greatly decreases the chances of the internal actuator components malfunctioning. This feature ties into cost efficiency theme of the project.

The actuators will be controlled by the cockpit design circuitry and will use the information sent from the potentiometers to accurately recreate the motion of the RC car. The actuators feature a built in wiring for potentiometers, this is extremely beneficial given the nature of what these actuators will be required to do. The RC car will send information regarding terrain, turning, and gradients. This in conjunction to the visual display should yield an enjoyable and realistic driving experience.

The actuators that were used in this project are three, 12 Volt thrust linear actuators manufactured by Servo City. The operating voltage range of the actuators is 6 to 12 volts DC. As mentioned before these actuators are powered by electricity. This feature was desirable given the nature of the project and behavior of linear actuators. The behavior of these actuators is that they experience identical behavior when extending and retracting. By simply changing the polarity of the voltage applied to each actuator they will extend and retract accordingly. This aspect made the programming and fabrication of the control circuitry of the actuators relatively simple and attractive to use for this project. This property of electric linear thrust actuators made them an obvious choice when compared to the pneumatic and hydraulic varieties.

The actuators are composed of aluminum frames with high grade plastic gear housing which should translate into high durability and low risk of corrosion. The extension shaft is 1.125 inches in diameter and is also made of high grade aluminum. The combination of quality grade materials and material thickness will be more than adequate to yield a dependable product with great response. There is a 0.5 inch diameter mounting hole located on the bottom of the gear housing, this will be connected to the circular cockpit base for each of the three actuators. Each one will be connected to the Cockpit via a Heim joint at both the top and bottom of the actuator.

Each actuator is capable of efficiently moving a maximum load of 450 lbs while generating repeatable and dependable results. This aspect of the actuators made them the ideal choice to be used in the three actuator design for this project.

is implementing. Some of the driver and cockpit load will be dispersed across the center ball joint located under the seat, so there is not foreseeable instance or situation where each actuator will have to move the maximum load by itself. The heaviest load all three are expected operate under is 450 lbs combined. Since combined they will normally be working within a range of 20% to 35% of their maximum load, this should be a relatively moderate work environment and should lengthen the life of the actuators.

They also boast a 6 inch stroke length. This length was chosen to make certain that the cockpit feels a pitch or roll of plus or minus 20 degrees. Though they are capable of 6 inches, due to programming and a desire to keep them from operating at any maximum specification they will only be implemented to reach a length of 5 inches. This keeps them from operating under too much stress, putting too much stress on the cockpit and driver, and also to ensure that the cockpit can reach maximum pitch or roll in under two seconds. Since all motion is designed to occur are the central joint, the extension had to be tailored so that neither the actuator nor joints ever rub or come in contact with the frame.

Another crucial aspect of the actuators is the speed at which extension and retraction occur. They operate at 2.9 inches per second under no load and 1.89 inches under maximum load (450 lbs). These specifications were vital in choosing the actuators since the response must be quick enough that the driver experiences a realistic simulation without any lag. The actuators must extend or retract to an exact position and then change position again in a fraction of a second hundreds of times during any given simulation. This implies that the response time must be fast enough that it can keep stride with the vast amount of continuous information being sent from the RC car. This is achievable given that the actuators will typically extend and retract at a range between 2 and 2.6 inches per second. Therefore in less than two seconds the actuators will be able to reach the maximum extension length that the programming will allow them to reach.

The actuators have a full retraction length of 17.72 inches and full extension length of 23.72 inches. When the cockpit is off the actuators will be at the full retraction length so they don't feel the static force of the cockpit when it's not in operation. When the cockpit is on, but not in use they will remain at a stable length of 20 inches until the RC car is in motion. This ensures that when the cockpit is on that each actuator receives a steady supply of voltage until the potentiometers send information. This is an essential feature to better protect the actuators and as a measure to make sure that there will be no need to replace any of them.

3.3.4: Body Composition and Joints

The cockpit base and floorboard was mainly constructed using two materials: 6000 series Aluminum and 5000 Galvanized Steel. These materials were chosen due to their ease of acquisition, strength, durability, weight, and the minimal amount of work it took to weld each material respectively. The other materials used that are not classified as either cockpit base or floorboard are the actuator bracing, which was fabricated using 2 inch Schedule 40 Galvanized Pipe with a two inch inner radius to fit over the actuator casing. Another was the 1.5 inch Galvanized Unistrut that was MIG welded to the Galvanized Pipe and was bolted to the 1.5 inch by .375 inch Galvanized Bar steel. The bar steel was bolted to both the Unistrut and the floorboard to hold the actuators in place.

The cockpit base is comprised of 6000 series Diamond Plate Sheet Aluminum (4 feet by 5 feet) and a plastic pallet of the same dimensions. The plastic pallet was selected due to its strength, ability to withstand extended periods of unspecified amounts of force, and its height (6 inches). The diamond plate was attached to the pallet by applying generous amounts of industrial grade quick drying epoxy and a series of four self-tapping screws on each side of the base (16 screws) total once the epoxy dried. Once this was completed six 90 degree .25 inch thick aluminum tabs were TIG welded to the top of the cockpit base which were used to securely connect the base of the actuators.

The cockpit floorboard was fabricated using 17 inches by 60 inches sheet of 6000 series Diamond Plate Sheet Aluminum and 5000 Series Galvanized Steel I beam. The steel I beam was used to create the frame or “skeleton” and was MIG welded in such a way that it created a rectangle with a width of 17 inches and a length of 60 inches. The diamond plate aluminum was then placed on top and fasted to the frame by a series of .375 inch thick steel self tapping screws placed 6 inches from one another along the perimeter of the floorboard.

One reason the steel I beam was chosen to fabricate the cockpit floorboard was the fact that it's easier to cut and weld a straight line. This created a more secure feel when the cockpit was in operation and also created a secure and safe insertion point for the actuators to connect to. Also through considerable research it was found that since all portions of the cockpit frame were straight portions of material they didn't require there to be any bending or shaping of the tubing, thus saving copious amounts of time. There was also considerable lateral, shear, and moment forces placed on the framing while the cockpit was in operation due to the shift in center of gravity (of the operator in conjunction with the platform), the Steel I beam was far more resilient to warping, bending, and stretching.

The pedal was constructed of the same 6000 series Diamond Plate Aluminum. It was 3 inches wide by 6 inches long and had the top two corners cut and rounded

so that if the operator was wearing sandals or barefoot they wouldn't cut the bottom of their foot. The pedal was mounted on Galvanized Steel Square Tubing that was 3 inches long by 3 inches wide with a wall thickness of .125 inches. The pedal was connected to a self closing steel hinge joint using three steel self tapping screws, and the hinge joint was connected to the steel square tubing spacer using 2 of the aforementioned steel screws. This square tubing was used to elevate the pedal to a more comfortable position and also to lift the pedal so that when a Knob was glued to the hinge joint it would rotate the potentiometer the would control the RC Car's acceleration and brake. The steel square tubing was connected to the floorboard using 2 of the same screws as mentioned before. There was a 6 inch steel bolt that ran through the bottom of the floorboard and was used to limit the range that the pedal could move.

To securely attach the top of the actuators to the cockpit floorboard a heim joint, also known as a rod end bearing joint, was used (which is shown above). This joint type was bought not fabricated and it was made of an unspecified class of Galvanized Steel. This particular joint is an extremely strong and durable joint used in a wide variety of applications, but most often used as control rods, steering links, and tie rods in vehicles. Heim joints are precision articulating joints consisting of 3 parts. The first is a ball swivel which allows the joint to move in any direction on the three axes with almost 340 degree range. This portion of the joint is a hollow circle which allows a bolt or many other connecting devices to pass through and attach to the heim joint. This portion of the joint created a secure connection of the actuator and floorboard frame underneath the seat. Three of these joints were utilized in the project and connected to the frame under the seat and attached to the floorboard I beam skeleton by attaching to a steel bolt with a shaft length of 4 inches that was MIG welded to the frame. The next is the housing, a circular casing that the ball swivel is encompassed by and its function is to hold the ball swivel in place and is the rigid part of the heim joint. The housing required some lubrication be applied to it approximately once an hour depending on the amount of use to be sure the joint didn't seize up during operation which would have put additional lateral force on the joint, the framing, and the actuator. The final part of the joint is the threaded shaft. This is connected to the casing and acts as the anchoring part of the joint. The thread can be either left or right hand threaded and allowed it to be screwed or bolted into various materials (mostly metals). The threading was cut off and the bottom of the heim joint was MIG welded directly on top of the actuator piston head. This did not compromise the joints integrity in any way and gave us the most secure and structurally safe attachment method. And given that the bolt and the joint are made of the same material, there was no grinding or shearing of any material due to the joint and the bolt having the same density and hardness.

This joint easily allowed each actuator to move in the desired directions without fear of rubbing, forced over/under extension, and provided a solid connection

that avoided slipping. The base of the Cockpit which touches the floor was considerably “wider” than the floorboard, underneath where the front two actuators will be mounted. This meant that since each actuator was placed in an exactly vertical positioning, that there was a vast amount of material on any which side of the front two actuator so that if there was a design change or amendment there was plenty of room. This also eliminated the required dynamic movements that would have been placed on the actuators had they been placed on any sort of angle.

To ensure that the Galvanized Steel used to construct the actuator’s bracing system is connected safe and securely a Metal Inert Gas (MIG) weld was used to create every joint on the frame and to create the base. MIG welding, also known as Gas Metal ARC welding is a process developed specifically for welding steel and other non-ferrous metal .This method of welding securely connected the Unistrut and Steel tubing together without heating the steel too much causing the end pieces to loose structural integrity and become brittle.

One advantage of MIG welding is that it allowed metal to be welded much more quickly and efficiently than other more traditional methods. This made it ideal for welding softer metals such as again, steel. Another advantage of MIG welding is that it produced long, continuous welds much faster than traditional methods. This specific welding process help cut down on the time spent fabricating the actuator bracing so we were afforded more time to work on the software aspect of RC GhostRider. The shielding gas that was used during the welding process helped to protect the welding ARC and also produces a clean weld with minimal splatter.

There were very few drawbacks that may be encountered when performing this type of weld. The equipment for this is considerably complex. This means that when outsourcing this portion of the project the welder that was selected had to be extremely well versed in this form of welding. Given that this type of welding is somewhat of a specialty, the fabrication could have cost a little more than was previously expected. Another disadvantage is that since there was a need for an inert gas shield, this weld could not be performed in an open area where wind would have been a problem destroying the gas shield, so the backyard was not an option.

The welding process that was used for all of the Aluminum on the cockpit base and floorboard was a Tungsten Inert Gas (TIG) weld. TIG weld is used exclusively for Aluminum materials only as it is a very high heat welding discipline and would melt many of the other non ferrous types of metal. TIG welding is another form of ARC welding as described in the MIG welding portion and it uses a non consumable tungsten electrode to produce the weld. The weld is protected from the atmospheric contamination by a shielding gas. The shielding gas that was used for our TIG welds was a mixture of 75 percent Argon

25 percent Nitrogen. This produced a very clean and secure weld, but the mixture is also known as “self cleaning”. This means that while the weld was structurally as strong as possible, it also burnt off specks or flakes of the Aluminum that when cooled could have become sharp debris that cut group members when transporting the cockpit or users if it got in their shoes or clothes. This mixture of Argon and Nitrogen was a little more expensive, but when along with the theme of user safety first.

3.3.5: Display

For the cockpits display a Video Display Unit (VDU) with reasonably high specifications was selected. The video feed from the RC car was displayed on the cockpit via a 22 inch Samsung television. This television boasts a flat screen with a 1080p resolution Liquid Crystal Display (LCD). This screen had a very high quality and displayed the live video feed from the vehicle with sharp, vibrant, clear images. It had a 5 millisecond maximum response time meaning that as long as the transceivers and moving data were an adequate rate the monitor did not experience any sort of lag or distortions. It had a 60 Hertz standard refresh rate which is among the industry standard so that the driver assured all cockpit components were of the highest quality. The Standby Power Consumption (the amount of power the television drew while it was plugged in but not in use) was 290 mW, while the Operating Power Consumption was 47 W. These values were not important however since the monitor was plugged in directly to the wall to minimize the amount of voltage and current the power supply needed to provide the cockpit. The monitor received its display information from transceivers that were connected to it and the RC cars digital camera. The digital camera being used had a reasonably high resolution so it captured information with enough detail that there wasn't any portion of the video that the monitor displayed as hazy, blurry or pixelated. The transceiver streamed the footage fast enough and the televisions specifications were high enough, that the user experienced little to no lag or delay with the images. The display also had to be prompt and crisp to stay in synchronization with the movement of the actuators. The LCD type television also has a horizontal and vertical viewing angle (over 170 degrees) such that no matter the user's position in the seat, a clean and crisp picture was always be viewable. This is one reason an LCD display was chosen because since the monitor was permanently secured in the cockpit it was able to be viewed at different angles given the various heights of the drivers. Each driver was easily be able to see the screen and view the footage captured by the vehicle.

3.4 RF Technologies and Camera

Two sets of RF transmitters/receiver sets were implemented, one for video transmission/reception, and another for data. It should be noted that both the video communication, and the transceivers used must have corresponding transmission distances. Thus, the XBee 1mW will be used for data exchange,

and a wireless surveillance camera with receiver set will be used for video transmission. Both operate at 2.4GHz, and have an estimated maximum range of 150ft-300ft (depending on obstacles). It is very important that not only do these signals have corresponding ranges, but also that they have a similar SNR at a distance from the cockpit. These details will be further discussed later, in the *RF Discussion* section.

Two XBee will be utilized for communication between the cockpit and the RC car. The antennas available are PCB or wire antenna, thus the wire antenna will be used. The transmit power of this transceiver is 1mW, with a receiver sensitivity of -92dBm. The features include adjustable power, 3.3V CMOS UART, API or AT commands, 7 10-bit ADC Inputs, and 8 digital I/O. The supply voltage required is 2.8V-3.4V, with a transmit current of 45mA and a receive current of 50mA. It costs about \$30, well within the price range allocated in the budget.

KY-2.4GR01+C-203A is a camera/receiver set that functions in 2.4GHz frequency, with a transmission power of 10mW. The miniature color camera is of suitable dimensions to be mounted on the RC Car. The camera has a built in transmitter, built-in rechargeable Li-Battery that operates for 3 hours continuously, and includes a built in microphone, which will also be implemented for use with the display. The camera has a horizontal resolution of 380, a viewing angle of 62 degrees, a bandwidth of 18MHz, and a consumption current of 80mAh. The receiver has 4 channels, is supplied by an AC/DC adaptor (12V, 500mA), and a consumption current of .3W. The receiver has RCA outputs, which can very easily be utilized with the display. With a value of \$32.99, all visual and audio goals are satisfied for use in the RC ghost rider.

3.5 Ergonomics

The cockpit was designed to be as ergonomic as possible. Since the cockpit was simulating a driving experience it was decided to use a seat from a truck, a 1997 Chevrolet Silverado driver's seat specifically. This seat was designed and tested vigorously by Chevrolet before being used in their cars with comfort and safety in mind. Also by using the driver seat from a truck the rider should be in a comfortable upright seating position so that when the actuators thrust there wasn't any discomfort felt in the upper or lower back. The Silverado seat had an added luxury of being able to increase or decrease lower lumbar support with the turn of a knob to generate a more comfortable and stable feel. Given that the floor board was mounted right beneath the seat, the 7 inch thickness provided enough distance between the bottom of the user and the top of the floorboard. This property again adds to the ergonomic feel of the cockpit. The bottom padding thickness of the seat also helped ensure that the user's back side does not feel the metal ball joint located under the center of the user's seat.

The seat also had a 4 point harness connected to the back of the seat. (The

fifth point of the harness was removed because in some motions the cockpit experienced it was more than 'uncomfortable' for men) The 4 point harness was chosen for its comfort and safety since many drivers complained about the traditional single over the shoulder belt, saying it was uncomfortable and cut into their neck. It also allowed children to operate the vehicle without the advent of a booster seat. This served to safely hold the rider in place and gave them an added sense of security and stability while the cockpit was in operation. The fastener of the harness was more reliable than a traditional belt since they are designed for racing vehicles and made of higher grade materials.

The floorboard of the cockpit is where the pedal was be mounted. It was mounted in a stationary position, but given then height and distance it was from the front of the seat, it was in the optimum position. This means that every rider that operated the cockpit felt it was in the perfect position for them to where the didn't have to stretch or bend their leg to reach it. This meant the each rider was in a comfortable and more importantly safe position when operating the cockpit which dramatically lowered the risk of injury due being seated uncomfortably. The angle of the pedal when in operation also allowed riders ankles to never be too outstretched when applying the gas again, diminishing the risk of injury from operating RC Ghost Rider.

The cockpit display was be firmly mounted. The television came with a prefabricated mounting bracket that allowed for it to be pitched 135 degrees up and down. The television was mounted securely using 3 bolts with locking nuts holding the mounting bracket snugly attached to the column that was welded to the floorboard. Small sharp vibrations or disturbances of the screen would case added strain to the driver's eyes causing fatigue, dizziness, or disorientation while operating. Another precaution that was taken since the display would remain stationary was to utilize a high quality monitor so that any driver, regardless of height was able to easily see the screen. That is why that specific LCD monitor was chosen. It was a viewing angle of 170 degrees and any driver able to reach the pedals and steering wheel will be able to view all images with ease.

The steering wheel will be mounted horizontally to the vertical structure supporting the television. It was be approximately 18 inches in length and will be able to move forwards and backwards just like the pedal platform. The steering wheel will also utilize a spring-loaded pin or a bolt and cotter pint design. By designing the steering column capable of moving forward and backward the driver was able to easily operate the cockpit at comfortable arm position. This alleviated any chance of the driver experiencing pain in their wrists, arms, shoulders, or neck while in operation. The actual steering wheel was a Sega Genesis gaming steering wheel. Once again extensive amounts of man hours were put into the research, design, and testing of this product to make sure that it

was ergonomically satisfactory. The steering wheel was slender with a relatively narrow diameter to accommodate drivers of larger girth. The steering wheel was also connected with a spring so that the driver felt the expected resistance that they would if they were actually driving a vehicle. There was no air bag to worry about which would have taken up precious time to remove had an actual vehicle's steering wheel been used. The steering wheel came with a prefabricated housing and great feature that allowed it to be tilted up or down and the locked into place. The spring that again came prefabricated was contained inside the plastic housing so that the riders could not displace or dislodge the spring causing it to be inoperable. The spring also provided the resistance that driver's have come to expect when operating any type of vehicle. Which in turn created a more realistic driving simulation. It also reset the steering wheel back to the center or starting position which in turn caused the RC car's wheels to return to their respective starting or centered position so that there was a baseline reading. The final use for the spring was to protect the driver from injury. If the steering wheel would have been capable of uninhibited movement when the cockpit was in operation. then controlling the RC car would be nearly impossible. It also somewhat prevented warranted or unwarranted quick jerking movements. These movements, when amplified by the ever changing pitch and roll of the cockpit would have put added and unneeded strain on the driver and ultimately resulted in an injury.

3.6: PCB

Two PCB boards were used in the RC Ghost Rider: one was for the Cockpit, and another was for the RC Car. These were both two layer boards, and cost \$61.10 for each board from PCBFabExpress.com. Eagle software was used to digitally construct the PCB board, and them then they were sent to the designers about 10 days after being submitted to PCBFabExpress.

4 – Project Hardware and Software Design Details

4.1: RC Car

4.1.1: RC Car Hardware Design Overview

Concerning the electrical design of the car, it was first necessary to identify the basic structure of the system in question. The MCU needed to accept input from the accelerometer, gyroscope, and the transceiver. That input needed to be interpreted and either used to control the vehicle or sent to the cockpit so that it could be processed at the cockpit's MCU. The control input was used to control the car by way of its DC servo motor for turning or DC drive motor for moving. In addition there was another subsystem of the car that was used to send data, however this system did not utilize the MCU because the video transmission would have put too much stress on the processor. The following block diagram depicts the basic electrical subsystem structure that was used on the vehicle.

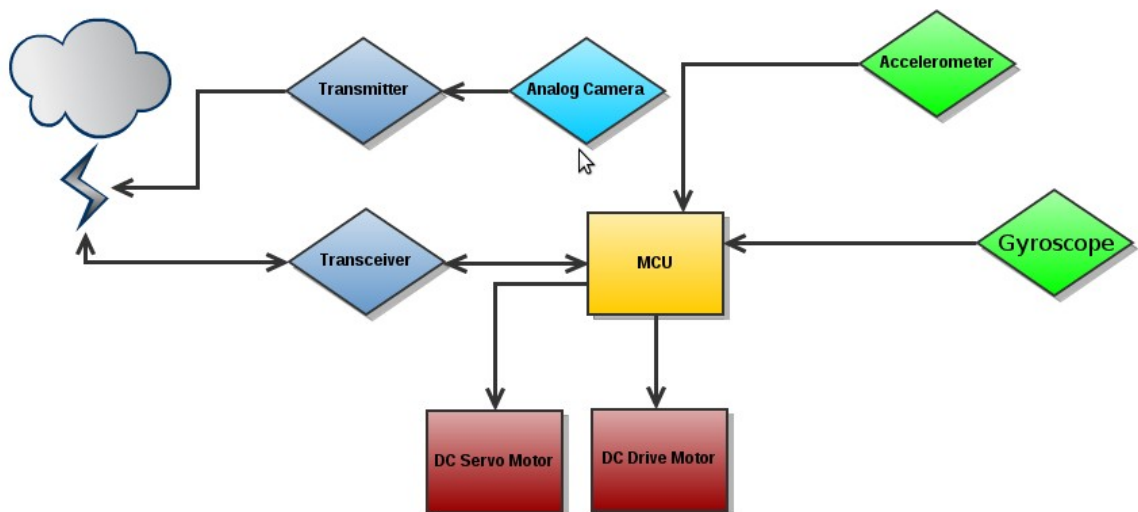


Figure 4.1: Electrical Overview

To be more specific, coming from the battery there will need to be three voltage regulators. One of the voltage regulators was a LM7805 that was used to properly power the MCU. The second voltage regulator was once again a LM7805. The final voltage regulator was more of a motor controller rather than a voltage regulator. This motor controller was designed for this specific purpose and can be viewed more in depth in the hardware design section.

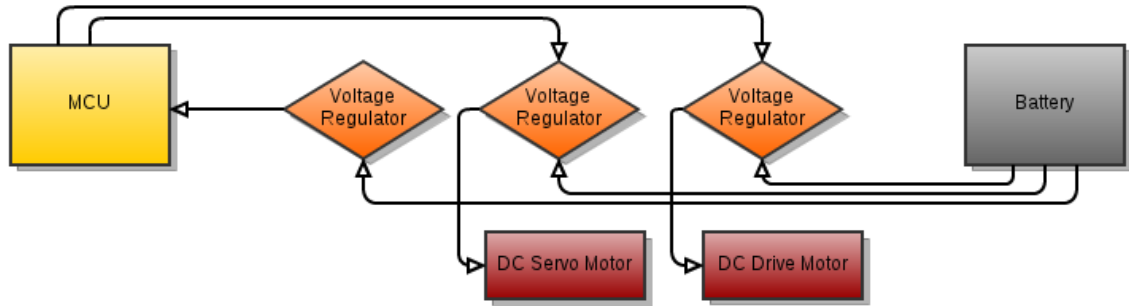


Figure 4.2: Power System Overview

In addition to this, the accelerometer and the gyroscope also needed to be interfaced with the battery and the MCU. In order to interface the sensors with the battery it was necessary to use a voltage regulator in between to supply the proper voltage. The power system for the MCU,transceiver, gyroscope, accelerometer, and two DC motors will be two batteries wired together in order to achieve a positive and negative voltage. This interfacing of the sensors is relatively trivial and can be accomplished in the manner outlined by the following block diagram.

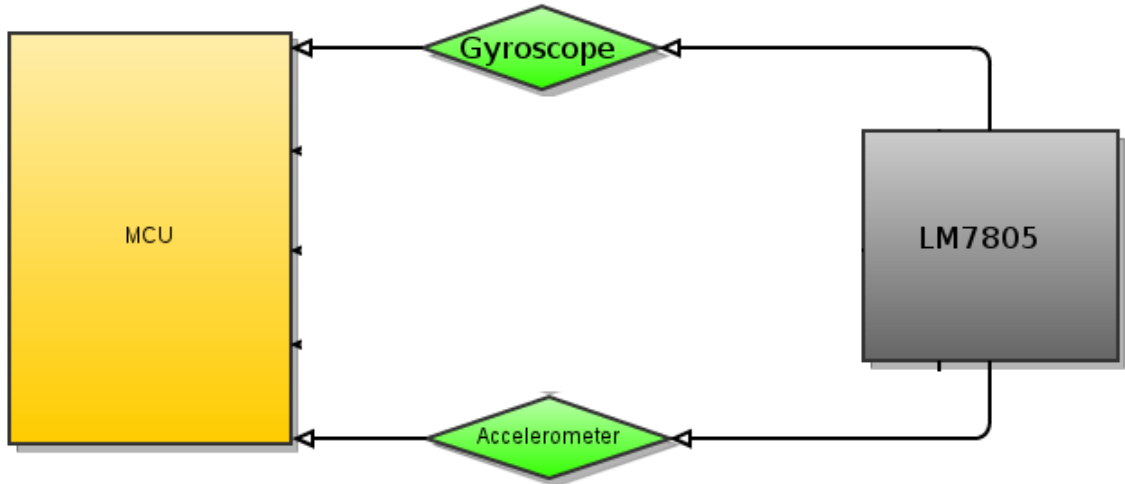


Figure 4.3: Sensor to MCU interfacing

Following this basic structural design, the power that was used for the RF communication between the camera and the cockpit needed a separate battery. This was so because of the massive amount of current draw, and the fact that it will vary in time because of the amplitude modulated signal that is being transmitted. This design was instituted in order to minimize possible anomalous behavior of the system due to various current draws. The following block diagram illustrated how the RF components were integrated

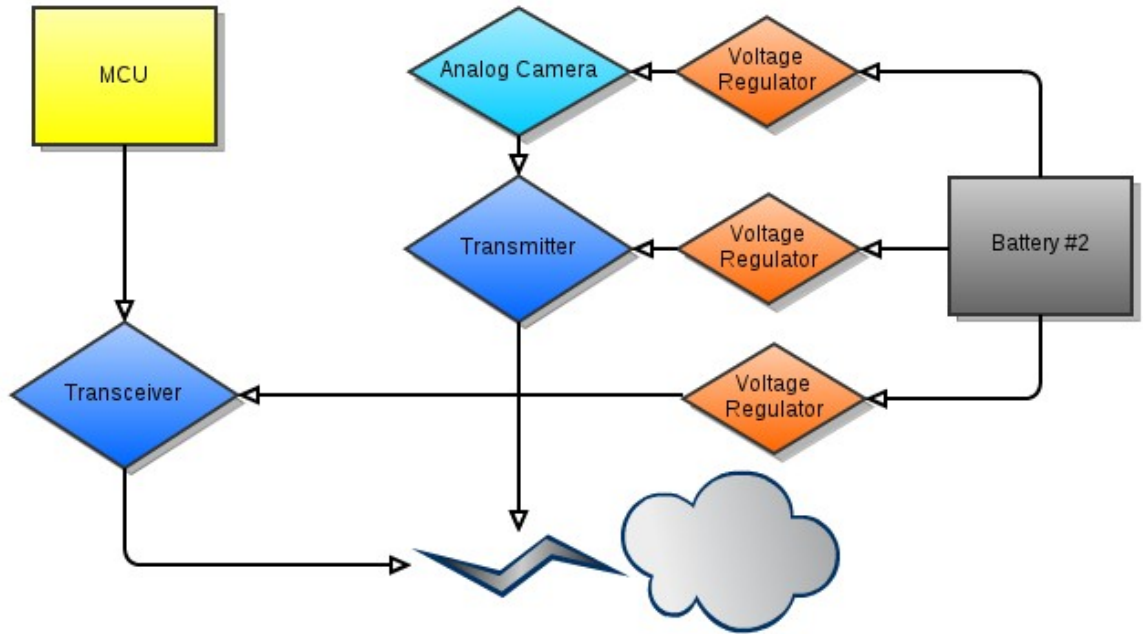


Figure 4.4: Transmission Subsystem

4.1.2: RC Car Software Design Overview

In regards to the structure of the programming the MCU needed to collect the data from the gyroscope and the accelerometer. Following this collection it was necessary to analyze the data and determine whether any of the information is significantly different enough to change. If the information in question was significant enough to change, the output buffer was then populated with the new information that was received from the peripherals. When either the output buffer has been changed, or the information was not significant enough to change the output buffer, the collection of transmission data then began. This data was then compared to the old data to determine whether or not it is significantly different. In the case that the information received was important, control changes were made before the data located within the output buffer was transmitted. Otherwise the control change step can be skipped and the data was immediately transmitted. The last step of the programming was to update and destroy the necessary information. The following flowchart depicts the basic program flow for the vehicle.

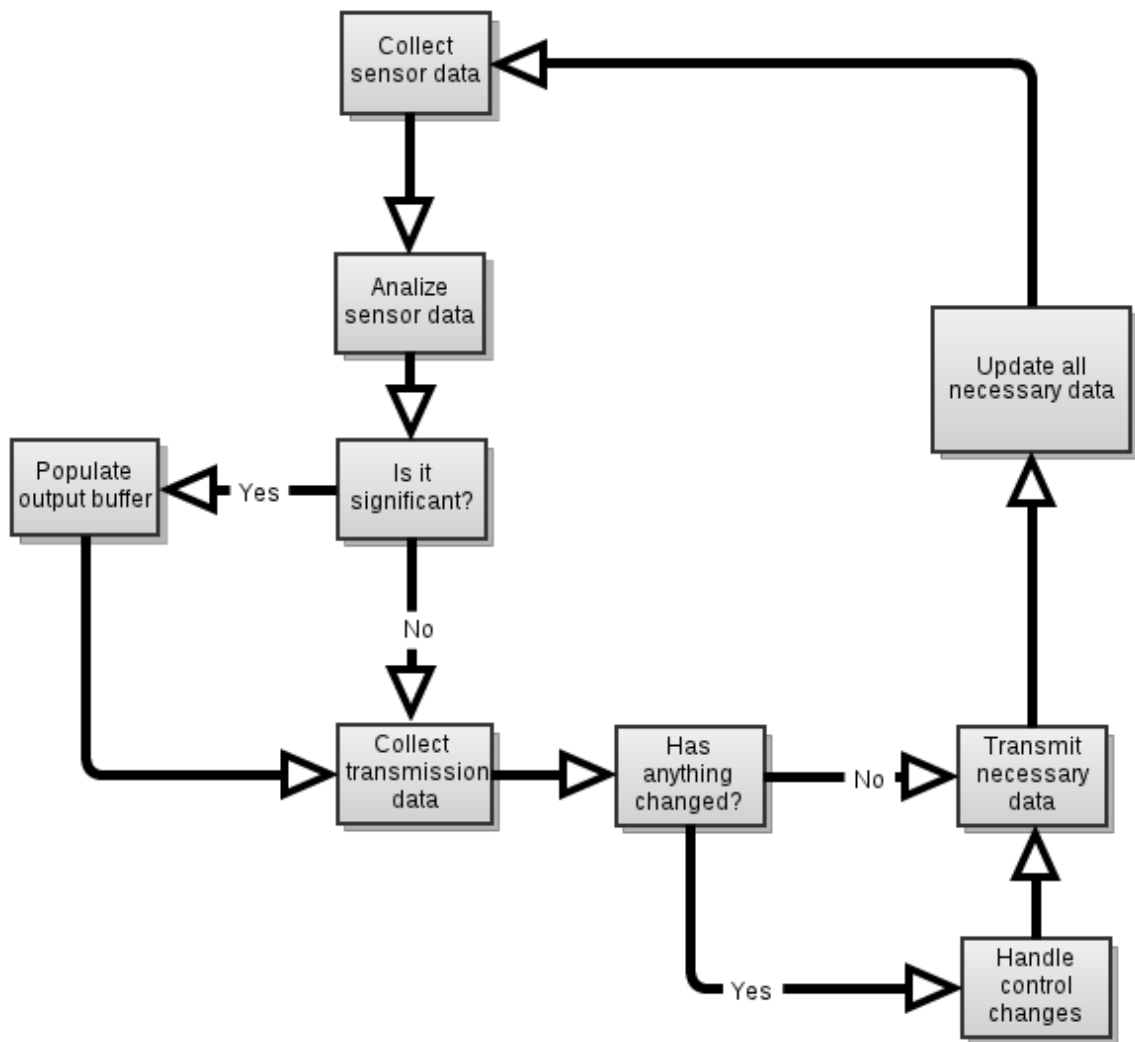


Figure 4.5: Program Flow Overview

4.2 Cockpit

4.2.1: Cockpit Hardware Design Overview

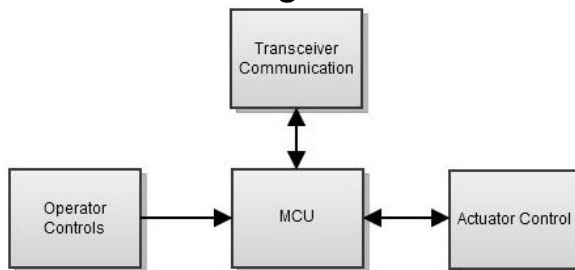


Figure 4.6: Block Diagram of MCU Data Exchange

The MCU receives information systematically from the Operator Controls, the Transceiver, and the Actuator Control. Separately, the MCU transmits information to the transceiver, and the actuator controls. The Operator Controls, Transceiver Communication, and Actuator Controls blocks are all summarized below, displaying their connections to the pins on the Netduino. Note that pins with suffix “D” are digital pins, and pins with suffix “A” are analog pins.

Flow Chart Color Key: Purple - Sub-Circuit, Green – Netduino Input/Output Pin, Blue - Circuit Input/Output, Red - Hardware Input/Output, Yellow - Decision, Orange - Process.

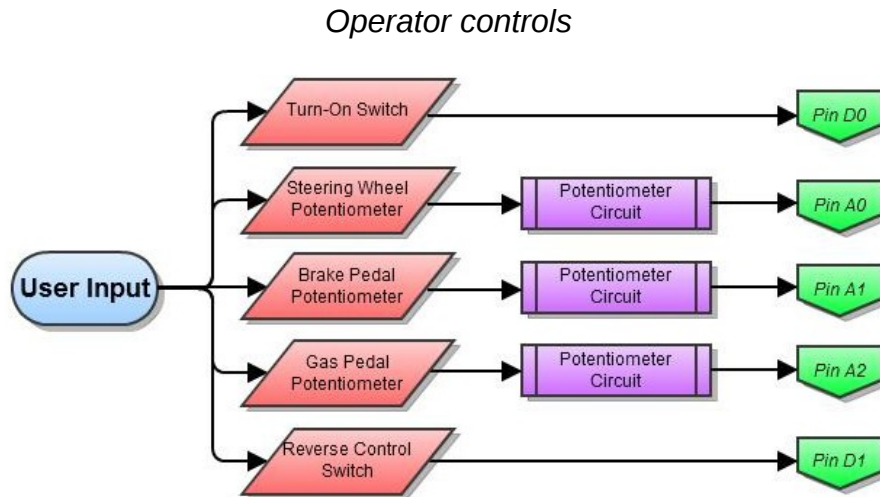


Figure 4.7: Operator Controls Block

User input operates all controls, which include the turn on switch, the steering wheel, the brake pedal, the gas pedal, and the reverse switch. The turn-on and reverse control are both connected directly to pins D0 and D1 (respectively). However, the steering wheel, brake pedal, and gas pedal potentiometers are each connected using a potentiometer circuit due to their continuous variation. Their analog inputs to the Netduino at analog pins A0, A1, and A2, respectively.

Transceiver Communication

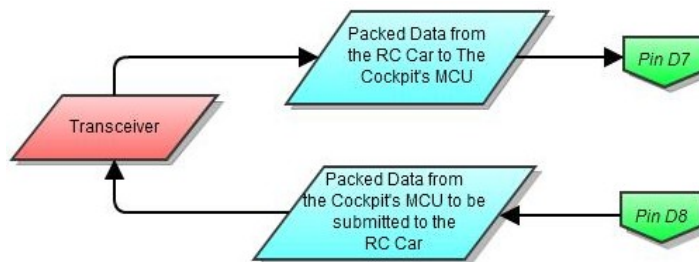


Figure 4.8: Transceiver Communication Block

The transceiver is, very simply, connected to pins D7 and D8. D7 will receive the packed data from the RC Car, and D8 will transmit packed data from the Netduino to the transceiver, which is to transmit data to the RC Car. 5 bytes of serial data are to be transmitted to D7, and 5 bytes of serial data are to be transmitted from D8. These “packed bytes” are described in the RF sections.

Actuator Control

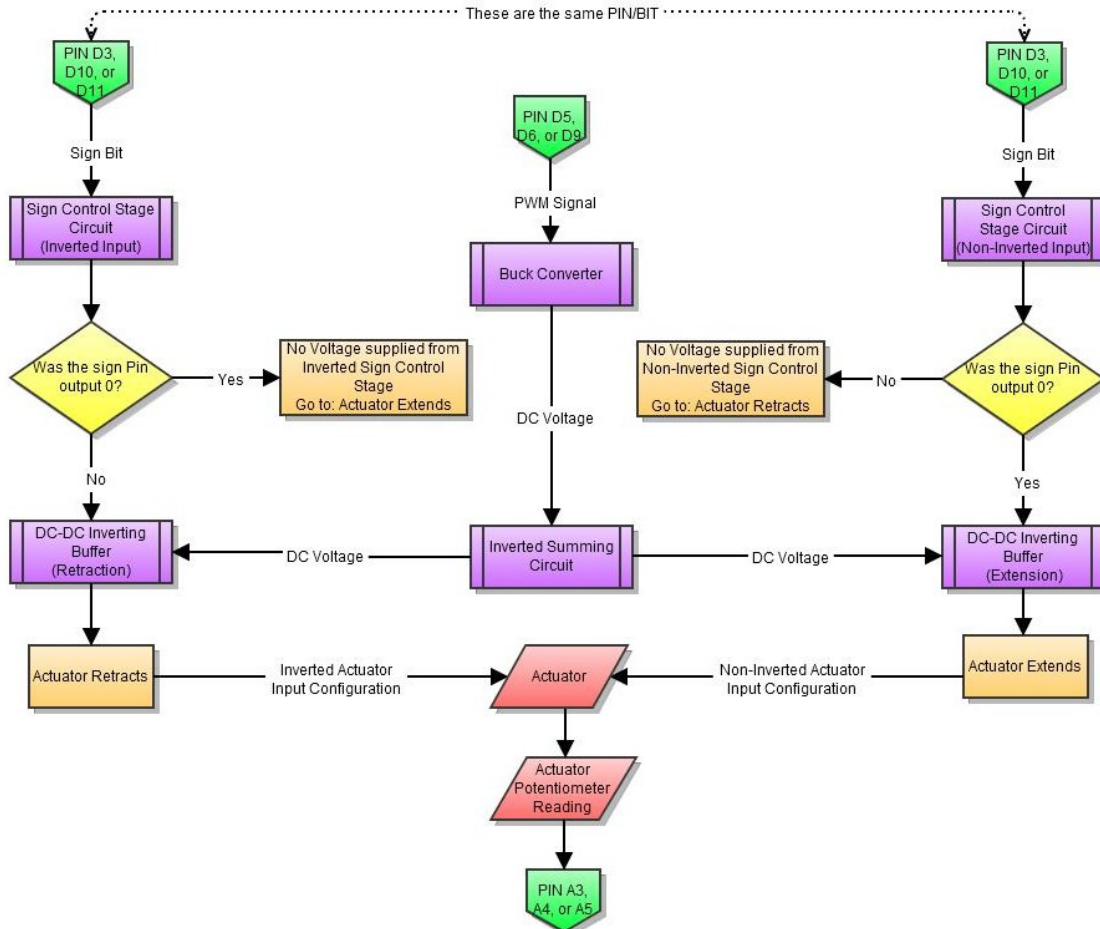


Figure 4.9: Actuator Control Block

The Actuator Control circuitry is the bulk of the cockpit circuitry, using a buck converter and inverted summing circuit for each actuator, and two sign control stage circuits for each actuator. A Sign Bit from Pin D3 and PWM signal from D5 will correspond to the circuit input to Actuator 1, (Sign Bit) D10 and (PWM) D6 will correspond to Actuator 2, and (Sign Bit) D11 and (PWM) D9 will correspond to Actuator 3. The sign pin will choose which circuit is used to activate retraction or extension, 0 corresponding to extension, or 1(3.3V) corresponding to Retraction. If the appropriate value of the pin is not the one necessary to turn on the Sign Control Stage, no voltage will be submitted to the actuator from the DC-

DC step-up buffer. The buck converter receives the PWM signal, and converts it to a DC Voltage with some negligible ripple current and voltage, and then the signal is added to be input to 2 DC-DC Step-up inverting buffers, but only the signal with the “Sign Control” stage activated will turn on. Detailed descriptions of these blocks are further enhanced in the Hardware Design Summary section.

4.2.2: Cockpit Software Design Overview

In figure 4.10 is a block diagram describing the programming operation of the Netduino. The I/O connections are as described in 4.2.1.

The turn on switch initializes the *Synchronization Function*, which determines whether or not the RC Car’s transceiver and the Cockpit are able to transmit packed information between each other. The turn on switch also acts as a “turn off” switch once the program is initialized, thus whenever it is pressed, the program will end, and the cockpit will stop all motion, and move back to base level. If the *Synchronization Function* determines that data packets cannot be exchanged, then the program ends. Then, the main loop begins, and ends execution under particular conditions to be determined in testing, along with when the “turn off” is switched.

Once synchronization has taken place, the cockpit will weigh the operator, as discussed in 5.3.2, using the *Operator Weight Function*. Data from the operator controls will be taken in, and converted into percent displacements, and submitted to the *Pack and Transmit Function*. The output of such action is then submitted to the transceiver pin to control the car. Then, data received from the transceiver will be received and unpacked using the *Unpack Function*, which outputs voltage readings from the car’s potentiometers and the voltage reading across the motor.

The values output from the *Unpack Function*, the steering wheel reading, and the actuator potentiometer reading will be converted to their actualized displacements and velocities by their respective *Potentiometer/Speedometer Voltage to Displacement/Speed Functions*. The actualised values are then changed into their average velocities and accelerations over sample times and are input into the *Final Actuator Output Function*. In the *Final Actuator Output Function*, a voltage will be chosen to apply to the actuators, and a duration of that voltage, given that it is safe. If it is not safe, the voltage will be reduced. Within the *Final Actuator Output Function*, retraction voltages will be chosen using the *Reaction Voltage Function*, which also output reaction velocity using operator weight, actuator velocities and actuator accelerations. Then, the values will be output to the PWM output function, which will output the digital output for either extension or retraction, and the PWM signal for the given desired voltage output.

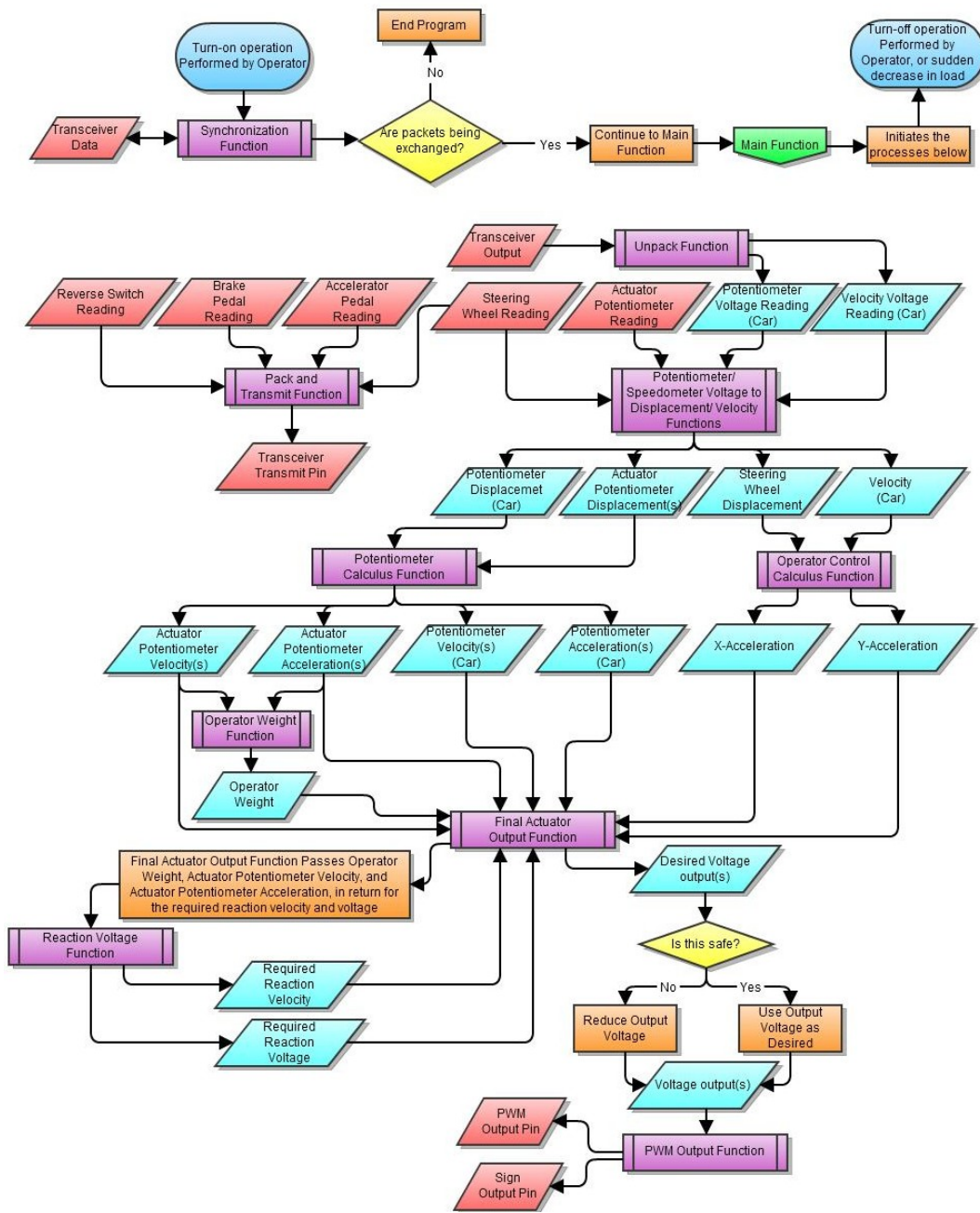


Figure 4.10: Block Diagram of Cockpit Program Operations

Flow Chart Color Key: Red - Hardware Input/Output, Blue - Functional Input/Output to be used for another function, Purple - Predefined Function, Green - Output that initializes, Orange - Process, Yellow - Decision

5 – Design Summary of Hardware and Software

5.1: RF Discussion

General environmental issues in RF communication revolve heavily around noise, as well as other uncontrollable concerns like heat, weather, signal obstruction, ect. However, weather is not of tremendous concern (Who is going to drive an RC car in the rain? Or possibly more-so, who is going to operate a simulator in the rain?), heat should not be a problem, so long as both the RC Car and Cockpit are circulated/insulated properly. It is impossible to eliminate noise, or avoid obstruction of the signal; however, the specifications of the XBee, and the Camera and its corresponding receiver were prebuilt to help reduce these issues. Thus, it is at some point the responsibility of the operator to use the RC Ghost rider within the range specifications listed.

Using only one transceiver set to establish all RF communication was insufficient in practice (for the particular set of processors used in this project). No transceiver that transmits data, video, and audio feed would have had the proper range of communication, price point, and would have cause too much delay in execution of commands set by the MPU's. Therefore, two separate sets of RF communication devices were utilized for the purpose of this project. To design either of these devices would be impractical, as it would be more expensive, and more time consuming while sacrificing power efficiency, range, size, and range. Thus, a black-box approach was utilized.

Camera-to-Display communications were of priority over XBee data transmission, as it would have been impossible to observe obstacles while driving the vehicle. Although operator safety is the most dominant concern, preservation of RC Ghost Rider shortly follows. It was the responsibility of the operator to not use the RC Ghost Rider if a visual feed cannot be established.

Synchronization must have taken place to ensure that communication between the XBee devices is effective. Each XBee must have be awaiting the moment when it is to submit data, the data it is to receive, and they also knew exactly how much data they are about to receive. This was to be accounted for in the process of its integration with its respective processing platform. Given that synchronization was not able to take place due to an issue with blocked transmission, significant RC car to Cockpit displacement, or lack of battery power to the RC car, a connection could not be established and control operations cannot executed.

Data Communication implemented between the RC car and the cockpit took place by making packets of information to be transmitted and received by their respective XBee's. The XBee received data, and transmitted it in a serial fashion.

The RC car transferred 15 bytes of allocated packed data, and the Cockpit transmitted 15 bytes of allocated packed data as well. The RC car transmitted 3 bytes for the digital readings that were read from the gyroscope, 3 bytes corresponding to the digital readings from the accelerometer, and 3 bytes for a voltage reading for the battery power of the power source operating the controls of the RC car. The Cockpit submitted 3 bytes for the voltage reading from the steering wheel potentiometer, 3 bytes for the voltage reading from the gas pedal, 3 bytes for a voltage reading from the brake pedal voltage reading, 3 bytes for the reverse switch, and 3 bytes for the on/off switch. 3 bytes was more than sufficient for each voltage reading, as then the voltage reading could be represented by 512 unique values.

5.2: RC Car Design

5.2.1: Structural Modification

Because the RC car that was modified was not exactly what was needed, structural modifications needed to take place in order to properly fasten all necessary components. Two aluminum L-brackets were mounted to the front of the vehicle to properly hold the turning servo in place. In addition some metal components from an erector set were used to create a mounting bracket on the front of the vehicle in which the camera would mount to. The mounting situation for the PCB and the batteries proved to be somewhat more problematic. A dowl rod needed to be cut into two 1.5" pieces and mounted to the frame of the vehicle to act as the two front legs for the PCB mounting table. On the rear of the vehicle a piece of 1x4 was cut in order to have a small mounting for the back of the PCB table. Two holes were drilled into a piece of 6" x 9" x .25" plexiglass, that would be used as a mounting point for the PCB. The PCB was then mounted underneath the glass by way of the two holes that were drilled. The driver battery was placed on top of the PCB at the very rear of the vehicle. Velcro was installed on the battery to ensure that it would not move unintentionally. Likewise the reverse battery was mounted underneath the PCB because that was about the only space that was left for it to be placed. Some of the old screws that were on the vehicle were stripped or falling out and thus it was also necessary to place new 6/32 screws in their place.

5.2.2: Hardware Design

The hardware design for the vehicle was somewhat complicated when it came to the DC drive motor. In order to adequately control the vehicle there needed to be a system devised in which to control the DC drive motor as well as the DC servo motor. In order to explain the circuit diagram, it will first be necessary to mention that an eight bit serial to parallel converter was used. This part was used to con-

control the circuit corresponding to the DC drive motor, and a resistive network was created in order to have a binary equivalence circuit (homemade D to A). The part is produced by Texas Instruments and the part number is 74LV8153N. It is more than sufficient enough for the purposes of this design as it can move 24k-bps. Since the DC drive motor is a 7.2V motor it was necessary to create a circuit that has an output of 7.2V in order to get the most out of the motor. A diagram of the circuit is as follows.

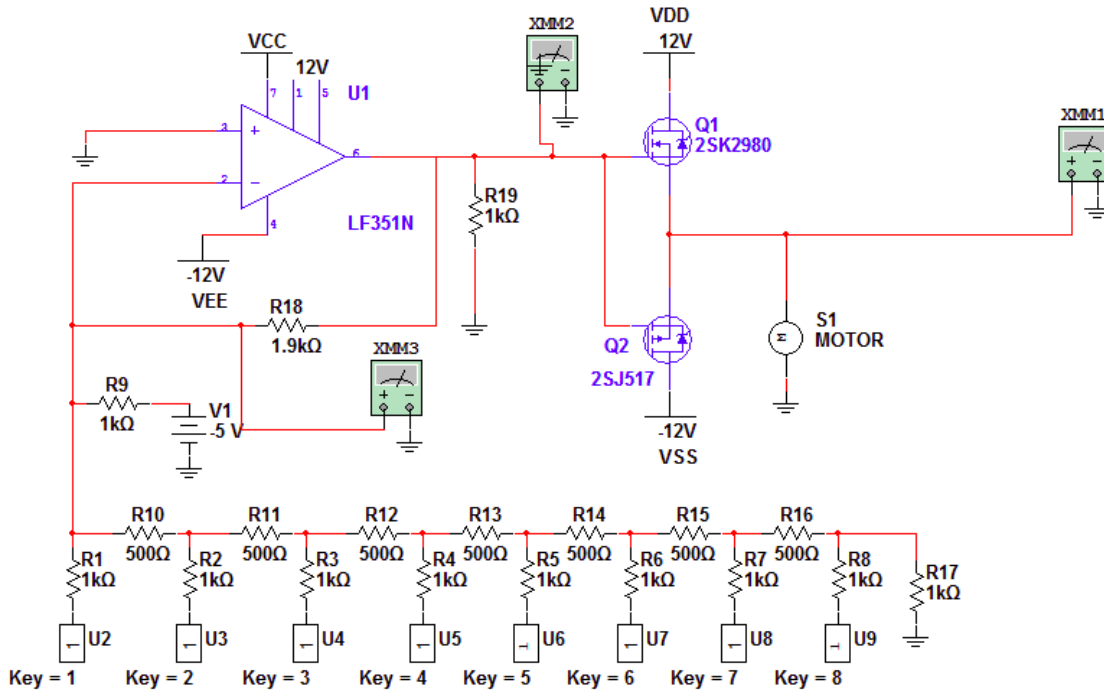


Figure 5.1: DC Drive motor controller schematic

The input to the circuit is a parallel binary signal that came from the MCU and was realized via a serial to parallel converter in order to get this information from the MCU. This binary signal was then weighted as a binary signal and a negative DC offset is applied. This offset allows for the realization of both the positive and negative values. The output was then multiplied by 1.9 in order to get the highest possible output of the circuit while still remaining in the relatively linear region of the power MOSFET's. This voltage was used to bias the power MOSFET's and a push pull output stage allows for a varying applied voltage to the DC servo motor. The final varying voltage range for the above circuit is -7.6V to 7.2V. This will allow the vehicle to operate both in forward and in reverse. In order to supply the negative DC offset voltage the eighth bit of the parallel output was used as the negative offset for the op-amp. This allowed for the circuit to zero out at both "00000000" and "11111111". Even though this design only allowed for 128 various voltage outputs, this amount of values was more than sufficient for the control purposes of the motor. The most significant bit was used to determine whether or not the output voltage is to be negative. In the case where

the input is "10000000" the maximum positive value of 7.2 volts is realized. Performing the one's complement on the seven bit number will give the current representation for the voltage value relative to its maximum. Likewise in the case when the input was "01111111" the maximum negative output is realized. Once again this binary representation can be manipulated and the voltage reading across the motor will change accordingly.

For the aforementioned circuit the 12V and -12V inputs was supplied by using the two supply batteries connected in parallel. The ground node that was used for the circuit was the connection point of the two 12 volt batteries. In addition the 12V power supply needed to be used to power all of the voltage regulators needed in the circuit, as well as the gyroscope and the accelerometer. There needed to be multiple LM7805's to control the 5V inputs on accelerometer, transceiver, transmitter and gyroscope.

The DC servo motor control was relatively trivial, especially compared to the control circuitry for the drive motor. One of the LM7805's with the 5V output needed to be connected as the power source for the motor. In addition one of the GPIO's from the Armega328p needed to be configured in software to use a PWM signal with a 50Hz output. In order to control the motor the active high needed to be one from between 1ms and 2ms. Any value less than 1.5ms would have caused the servo to spin counter-clockwise, while any value longer will cause the motor to spin clockwise.

5.2.3: Processing Platform: Structure and Duty

When the software was being designed, it was necessary to account for the collection, transmission and acquisition of data, it was very important to understand the complexity of such a system. Likewise one must have understood the steps needed to accomplish such a task. The first order of business was to layout the components needed in order to complete such a task. To begin it was necessary to have location in memory at which the data was to be stored. In this case, there needed to be an array of length three for each of the sensory readings as well as each of the motors (both drive and servo). In addition there need to be a three by three array for the accelerometers values to hold the information pertaining to the changes along each axis. It was also necessary to have two more locations in memory for each of the aforementioned arrays. (and three for the accelerometer) One of these locations stored the computed average value and the other stored the old computed average value. Also, there needed to be two more arrays that held the information for the control of the vehicle. These arrays had two elements, the first being the newest received value, and the second being the old value. The final memory elements were a buffer that is used to send data to the cockpit via the transceiver, as well as a buffer that will receive input from the cockpit.

In addition to having these memory locations, there also needed to be a set of functions that are to be called along the way in order to break this seemingly daunting task into small, accomplish-able tasks. There needed to be a function called “Collect_Data”. Collect data read the sensor values, as well as computed the average. It will also compare the new average data that has been collected to the last average data. It returned a Boolean that let the user know if the information was different enough from the old data to cause something to happen

In order for the “Collect_Data” function to accurately measure the voltages across any sensor it was necessary to take multiple consecutive readings of said voltages. In this case, there needed to be three consecutive voltage readings. The readings across any one part should not be directly consecutive in time. Consider the case where potentiometers (example case) were being used. The corresponding readings would have been named PFR,PFL,PBL,PBR and the accelerometer readings were named AY,AX,AZ. The readings would have taken place in that order three times.

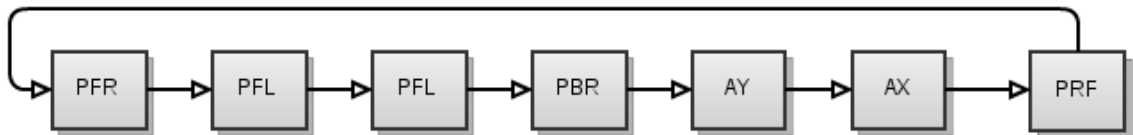


Figure 5.2: Receiving order of information

After the readings have been taken and the new information has been stored in the memory locations, it would have been necessary to analyze the data. This analysis consisted of taking the average of the new three readings in each section. These new averages were to be compared against each of the old averages to see if there was a difference in magnitude greater than 4%. In the case that any one value has that difference, all of the new averages were to be used to populate the output buffer. In the case that the values are all below the threshold, there was to be no change to the output buffer, and the previous values would not be altered. The follow block diagram illustrates the basic flow of the function.

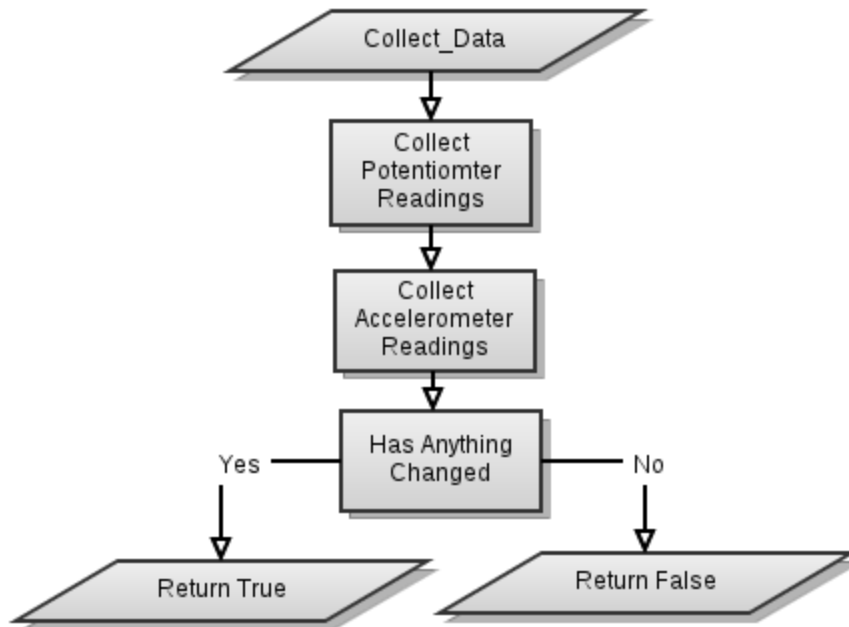


Figure 5.3: Data Collection Function

The next important piece of code that was to be used in the programming for the MCU on the vehicle will be the “Receive_TX” function. This function was relatively simple and was used to receive the information that was being sent from the cockpit. The information that came in contained the current control values for the DC drive motor as well as the DC servo motor. These values were compared to the old values that were received from the prior transmission. In the case that nothing had changed, the “Receive_TX” function will return a false Boolean value to the main program. This signified that there has been no change and the program then handed control over to the “Send_TX” function. Otherwise in the case that something had changed regarding the control of the vehicle, the program flow was to be passed to a function that will be called “Control_Car” and the values within memory that contain the information pertaining to the control of the car were to be updated.

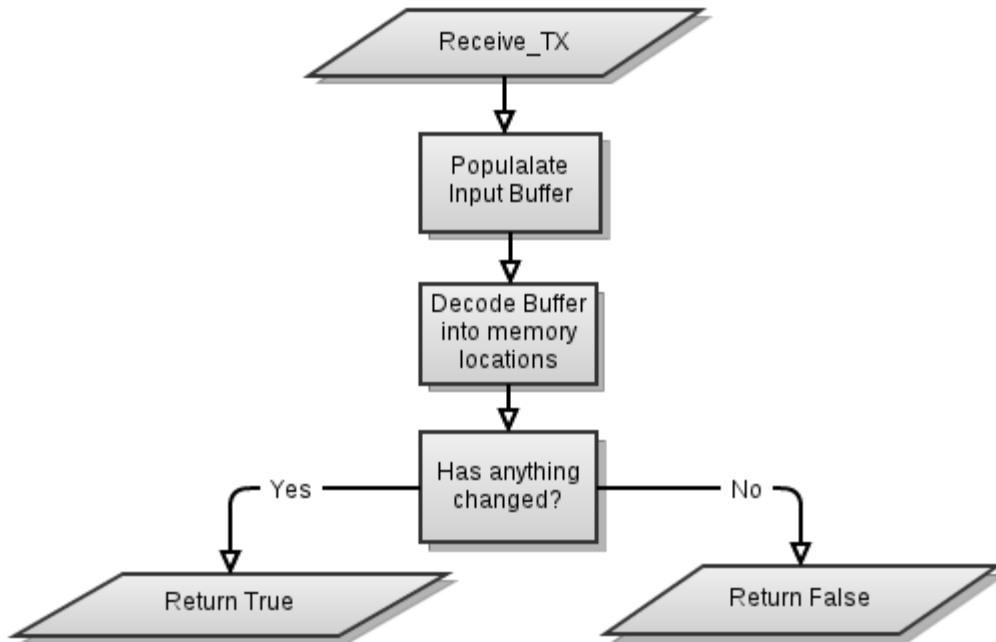


Figure 5.4: Receive Information Function

The function “Control_Car” was responsible for handling all changed values. If this segment of code was being executed, there were differences in the control values for one, or more of the vehicles controlling components. The values that were held in memory that contain the new drive information were to be written to the digital output ports that pertain to each of the motors. At this point the program control would then be passed to the “Update” function.

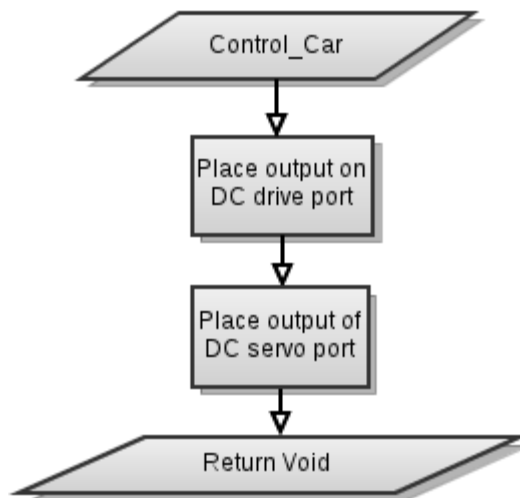


Figure 5.5: Vehicle Control Function

The function “Send_TX” was used to actually send the necessary information to the cockpit. When this function was called all the values that had been stored into the output buffer were sent, by giving the transceiver the input it expects to actually send the data. After this function had been successfully executed the “Update” function will be called.

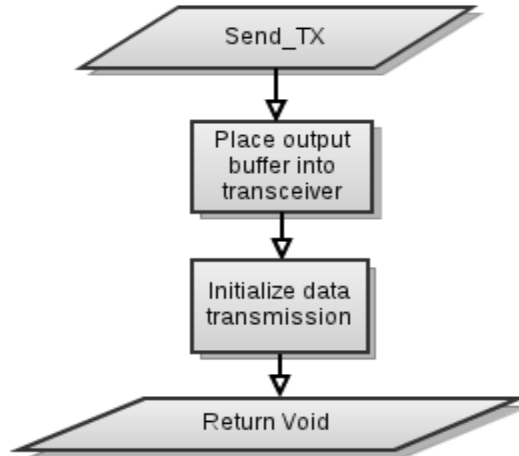


Figure 5.6: Send Information Function

The “Update” function was the final step of the programming before everything started over. This function was to be used to move all of the readings that were in the “new” spots in memory to the places that contain the “old” information. In addition the output buffer as well as the input buffer were purged of all information and stored with zero's. This was done so the “Cockpit” was able to easily assess the situation where nothing has changed.

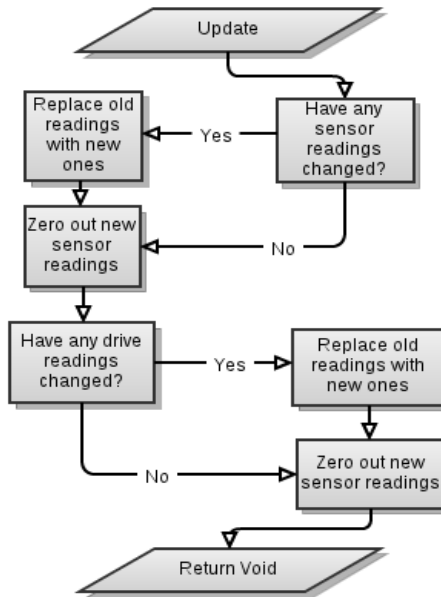


Figure 5.7: Update Information Function

5.3: Cockpit Design

In the design of the cockpit, there are recurring relations pertaining to the motion of the cockpit. The geometry of the apparatus requires that not only is the behavior known, but it also requires that it is regulated. Capital letters denote constants. Let H be the height of the central ball and socket joint relative to the bottom of the apparatus, W be the horizontal distance of the bottom of any actuator from the central ball and socket, R be the distance of the top of any actuator from the origin in which rotation takes place, θ_1 be the angle made between the top of the actuator to the to the central ball and socket (relative to being set completely horizontally), θ_2 be the angle made by the bottom of an actuator to the top of an actuator (relative to the normal axis from the ground plane), L be the fully retracted length of the actuator, and ℓ be the length of extension by the actuator. Then, the system can be modeled as follows:

$$(5.1) \quad \begin{aligned} W &= (L + \ell) \sin(\theta_2) + R \cos(\theta_1) \\ H &= (L + \ell) \cos(\theta_2) - R \cos(\theta_2) \end{aligned}$$

Which implies, by setting both equations equal to θ_2 (due to θ_2 as an element between 0 and 90 degrees):

$$(5.2) \quad \chi = \frac{1}{2R} [(L + \ell)^2 - H^2 - W^2 - R^2] = H \sin(\theta_1) - W \cos(\theta_1)$$

Using χ for simplicity, and reuse. Thus, θ_1 can be found in relation to χ (Due to θ_1 being a positive or negative number in the 1st or 4th quadrant):

$$(5.3) \quad \theta_1 = \pm \arccos \left(\frac{\sqrt{H^4 + H^2 W^2 - H^2 \chi^2} - W \chi}{H^2 + W^2} \right)$$

Note that positive sign will be used when the top of the actuator is above the ball and socket joint, and negative if the case would be in the contrary. In addition, taking the derivative relative to time it can be found that:

$$(5.4) \quad \frac{d\theta_1}{dt} = \pm \frac{d\chi}{dt} \left(\frac{H^2 \chi + W \sqrt{H^4 + H^2 W^2 - H^2 \chi^2}}{(H^2 + W^2) \sqrt{H^4 + H^2 W^2 - H^2 \chi^2}} \right) \left[1 - \left(\frac{\sqrt{H^4 + H^2 W^2 - H^2 \chi^2} - W \chi}{H^2 + W^2} \right)^2 \right]^{\frac{1}{2}}$$

Where

$$\frac{d\chi}{dt} = \frac{d\ell}{dt} \left(\frac{L + \ell}{R} \right)$$

$d\ell/dt$ is the rate of extension of the actuator, $d\theta_1/dt$ is the rate of change of the angle made by the platform. The negative case takes place in the cases when the top of the actuator is below the ball and socket joint, in the retraction phase and when the top of the actuator is above the ball and socket joint in the extension phase.

For programming purposes, it is necessary (5.2) to solve for ℓ in terms of θ_1 :

(5.5)

$$\ell = \sqrt{2R[H \sin(\theta_1) - W \cos(\theta_1)] + H^2 + W^2 + R^2} - L$$

Then, in differentiation:

$$(5.6) \quad \frac{d\ell}{dt} = \frac{d\theta_1}{dt} [RH \cos(\theta_1) + RW \sin(\theta_1)] \left(2R[H \sin(\theta_1) - W \cos(\theta_1)] + H^2 + W^2 + R^2 \right)^{-\frac{1}{2}}$$

It is extremely important to understand that any actuator will extend at a rate slower than one which is retracting in this design for two dominant reasons: an actuator fighting against gravity will have a harder time in extension than one which is working with gravity, and the geometry of the project provides that the actuator with the lowest angle made by the actuators and the axis normal to the ground plane will have a greater component of acceleration in that direction. Potentiometer readings from the reference on the actuators can be used to find the instantaneous value of the extension, denoted as ℓ .

It should also be noted, that θ_2 can be found in terms of θ_1 as follows:

$$(5.7) \quad \theta_2 = \arctan \left(\frac{H + R \sin(\theta_1)}{W - R \cos(\theta_1)} \right)$$

And thus,

$$(5.8) \quad \frac{d\theta_2}{dt} = -\frac{d\theta_1}{dt} \left(\frac{R(H \sin(\theta_1) + R - W \cos(\theta_1))}{H^2 + 2HR \sin(\theta_1) + R^2 - 2RW \cos(\theta_1) + W^2} \right)$$

5.3.1: Body and Physical Design

For the physical design of the cockpit simplicity, functionality, and safety were the goals. Given the limited amount of time to complete the design and fabrication of the cockpit a no frills design was needed to ensure that the milestones dates were met. The cockpit was however designed to keep the driver in the most comfortable and ergonomic position possible when operating the simulator.

When designing the framework simplicity was again the goal. The steel I beam used to fabricate it couldn't be bent or shaped into any sort of curvature. That means that the framing was comprised of only straight segments of steel I beam welded to other straight segments of I beam ensuring a proper weld and connection. The base consists of aluminum diamond plate material secured on top of a plastic pallet. The base acted as a platform to help some user's up into the cockpit seat. The base's primary function was to raise the entire cockpit height enough so that while the cockpit is in motion the floor board will never come in contact with the ground. To make sure of this the platform had a height of 5 inches. This allowed for the platform to reinforce the actuators as well as help stabilize the cockpit during movement. Attached to the bottom of the platform was a series of rubber stoppers. The stoppers were used to prevent the whole cockpit assembly from sliding and shifting across the floor. The base of the platform is also where the bottom of the three actuators connected. The figure below shows the actuator place relative to the base of the cockpit.



Figure 5.8: Cockpit Base Top View

The base was constructed of a plastic pallet of 4 feet by 5 feet with a piece of aluminum placed on top with the same dimensions. The base was cut and shaped from a piece of sheet metal with a thickness of .120 inches, equal to that of the aluminum used on the floorboard. Inside the base there were a series of trusses and supports that will be used to support the weight of the entire weight of the cockpit and driver. This was one of the reasons why the plastic pallet was chosen since it had that feature. This portion had a combination of aluminum and plastic to stabilize the cockpit while it is in motion and exerting a barrage of forces on the base.

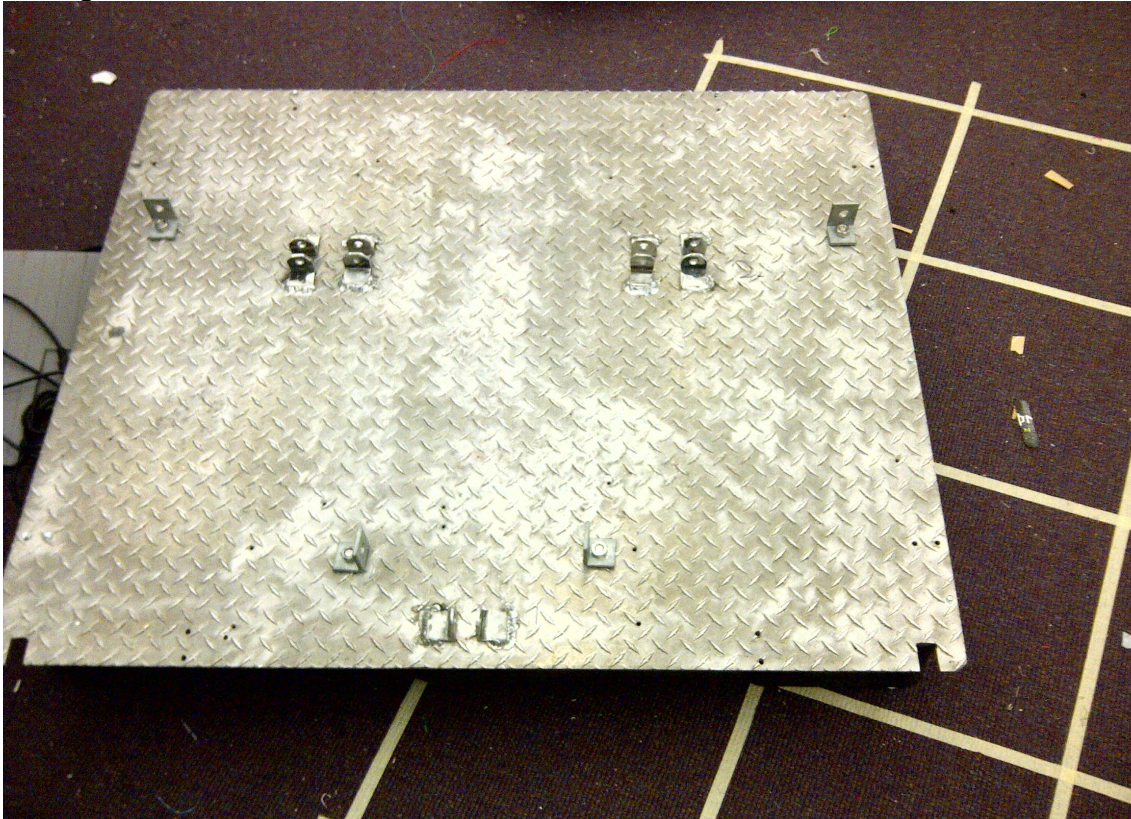
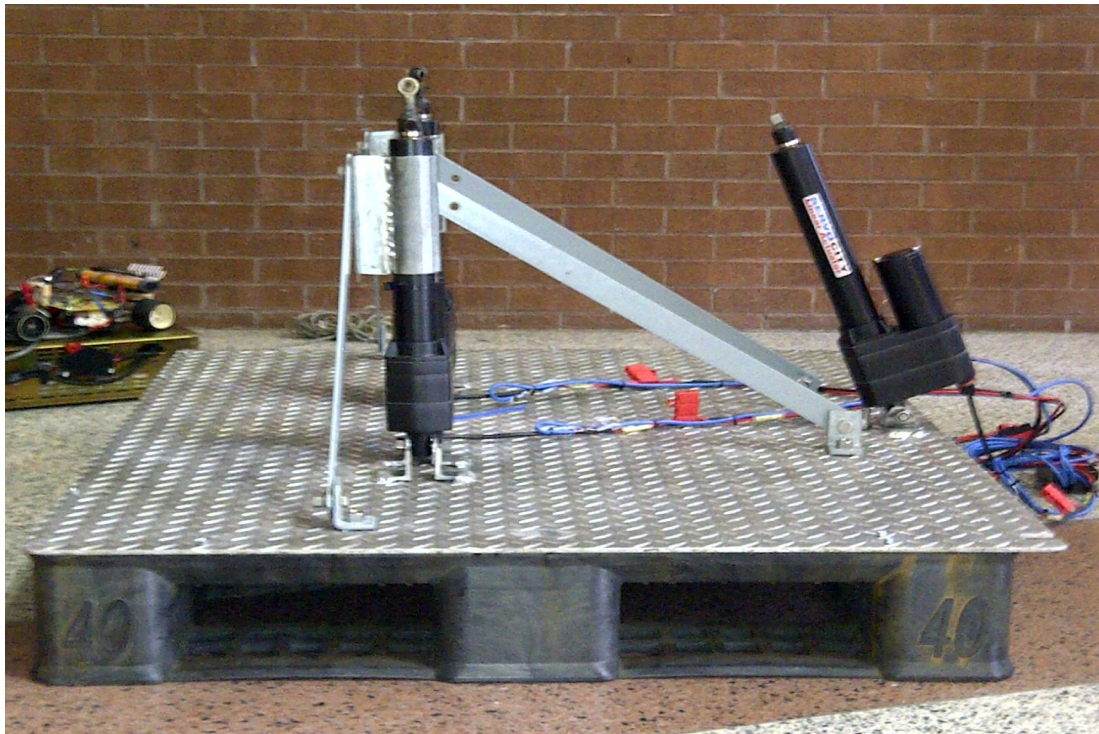


Figure 5.8: Cockpit Base Top View

The actuators connected both the base and the floorboard apparatus. With the loss of the central oversized ball joint, it was necessary to place the actuators completely vertical. This will allow for less stress and force on any one single actuator at any given time, and allow one to act inversely from another in most cases (turning, grade change, etc.). With the addition of the heim joints used on the top of the actuators all of the articulation that was encountered was felt on the joints. This is why heavy grade material heim joints were selected and that the bolt that extends out on each side of the floorboard was used.

The actuators acted in one of three manners. The first is that the two front actuators were asked to move proportionally inverse from one another in many

applications. Thus even though there are three actuators, they acted as though there are only two in the dynamic side to side movement. Since the RC car was a solid one piece axle, the distance one side moves, the other side must move an equal distance in the opposite direction. To achieve this one actuator thrusts while the other retracts to the same length, but the retracting actuator must move at a speed approximately 95% of the extending actuator. This aptly simulated a vehicle turning. This will be simulated in various turning situations, grade changes, and in certain terrain. While the front two actuators are working inversely proportional to one another the back actuator will more than likely be stationary and by virtue of the heim joint act as a “spit in a barbeque” or the center lateral rotating point. Utilizing the cockpit design the user with experience the same change in degree about their center of mass while only two of the actuators are actually extending or retracting. Below is figure 5.9 and shows the Cockpit Base Side View.



5.9 Cockpit Base Side View

The second manner the actuators worked in is when both the left and right front actuators are worked together performing identical movements while the back actuator is working inversely proportional. Again even though there were three actuators, the movement performed was only be 2 dimensional (front to back with no side to side lateral movement). This was mainly happen when the RC car experiences an inclining or declining plane. To achieve a real life simulation or recreation the front actuators received precisely the same amount of voltage

applied at the exact same time for the same duration.

The final manner in which the actuators were asked to work is when all three actuators were in motion providing front to back and side to side movement simultaneously. This was the most challenging movement to mimic. The three RC car potentiometers were sending information at the same time so each actuator will be working independently or one another.

The spacing of the actuators is set at the above mentioned dimensions and would provide a maximum angle in the most extreme case of extension vs. retraction of the actuators (one of the front actuators fully extended [retracted], the rear actuator fully extended [retracted], and the remaining front actuator fully retracted [extended]) of 4 inches (safely) was required for planar movement relative to the angle made by the actuators. At a maximum, 6 inches in the back, relative to the ground plane, will be required to account for movement of the top of the seat while in use. At maximum, 6 inches will be required in the front, relative to the ground plane, to account for movement of the extended platform. The maximum extension speed of the actuators is 1.89 inches per second at maximum load capacity. Thus, there is either a clockwise or counter clockwise rotation caused by the extreme case of an actuator extending, and another retracting.

The design of the actuator connections were figures with the prone length being 20 inches, not the fully retracted length of 17.72 inches. This placed a constant or static load on the actuators while they are non-operational. This was a desirable feature of the cockpit design because it will help ensure a long actuator life. To achieve this the cockpit control circuitry was designed to always send a constant base voltage to each actuator while it is non operational, but still on.

At each corner of the floorboard was a heim joint to which the actuators mount. The jointing acted as the insertion point of the actuators to both the cockpit framing and the base. These joints will allow the actuators to extend and retract with having the motion causing the actuators to feel the full lateral force. They have many of the same attributes of a ball joint with the enhancement of added strength. They will be fastened to the cockpit frame and base using the threaded end section of the heim. The bearing inside the joint allowed for the actuators to connect to the joint using the prefabricated mounting holes at the top of the rod and the bottom of the housing. The side view of the seat, ball joint, actuators, and cockpit base is as shown below.



Figure 5.10: Central Cockpit Side View

The seat of the cockpit will be the driver's seat from a 1997 Chevrolet Silverado. This seat was chosen due to its comfort and design. In this design the driver's leg extend forward and away from the body as in a sports car, not down as someone driving a truck would experience, so a sports car seat was a feasible option. The seat has a bottom cushion thickness of 7 inches when the driver is not sitting in it. This thickness provided sufficient backside support. The seat will remain in a fixed position once welded to the cockpit frame. This measure is taken to guarantee that the operator is safely attached to the cockpit.

The harness is an aftermarket modified racing harness. It has two straps which connect to the end of the "T" structure and was pulled down over the user's shoulders. The bottom portion of the harness fastened directly to the frame under the front edge of the seat. When fastened together properly they connect in the navel region. This type of seat belt was chosen due to its increased restraining power and to make it a more universal fit. This type would also provide a more accommodating fit if a child or person of smaller stature was operating the cockpit since there would be a series of three belts to adjust as opposed to the traditional one. A standard seat belt would require an area where the unused would be housed and also a complete redesign of the cockpit frame so the above the riders left should there would be a spot for the belt to mount. The floor mounting bracket on the seat is made of a high grade steel and will salvaged to mount the seat above the central ball and socket joint..

The floorboard measured 60 inches in length from where it is welded behind the seat to the end where the driver's feet will be. It will be fabricated using a single 60 inch length of aluminum sheet metal and two 60 by .375 inch I beams. Then the two portions of I beams will form a set of parallel lines that measure 17 inches from end farthest edge of each. The square tubing will be fastened so that

the cut line faces in. This means that there will be 15.75 inches in between both pieces of tubing and that is where the portions of sheet metal will be welded. Once these sections are welded together they will form a hollow rectangular box. Given the dimensions of the floor board "box", the end that is welded to the triangular bracing should be of the same height and wall thickness assuring an air tight seal. This will help keep the floorboard and bracing keep a very tight connection over extended periods of the simulator's thrusting and moving. Starting 20 inches from the weld connecting the floorboard to the triangular bracing a series of holes will be drilled. Each series of holes will be 4 inches in distance from the previous set of holes. The holes will be on the top of the floorboard 1 inch from the the center of the hole to the outer edge. The radius of the holes are both .125 inch. These holes will allow the pedal platform assembly to move closer or farther away from the driver operating the cockpit. The top view of the floorboard is show below with the pedal platform assembly shown and the tree mount cut out.

The pedal platform also contained, pedals. A gas pedal was mounted to the platform and when pressed will correlate to acceleration of the RC car. The pedal consist of a pedal from a 1992 Nissan 240 SX and will be oriented with the gas pedal on the right side of the brake pedal. The gas pedal measures 3 inches wide and 6 inches tall.

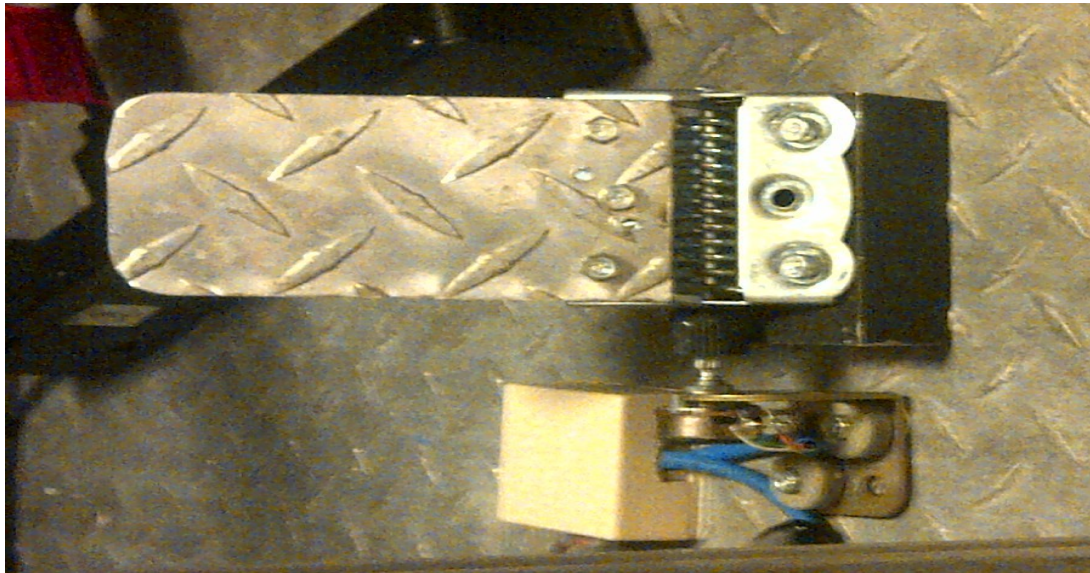


Figure 5.11: Pedal Top View

The pedals connect to the the platform by a hinge joint. The hinge joint was screwed to the platform and glued to each pedal with an industrial strength quick drying epoxy. The reason quick drying epoxy is used is over a screw or bolt fastening is because exposed bolts increase the possibility of a driver injuring their foot when pressing the pedal. The epoxy is quick drying so that if any must be reapplied the cockpit won't be nonoperational for a long period of time, or any, while the epoxy is drying. The hinge joint was connected to the to the pedas at

the bottom of each and will let the pedals be fully pressed to the platform yet eliminate lateral movements. The movement of the pedals mirror the movement experienced when operating an automotive vehicle.

In between each pedal and the angled portion of the platform was a spring. The spring was fastened to the bottom of pedal and to the edge opposite of the hinge joint. The springs were fastened to the bottom of the pedal by using a core out drill bit going about .375 inches deep. The core out drill bit hole will be the same diameter of the spring so that the top of the spring fits inside of the drilled hole. The spring and pedal connection will also be glued using the same quick drying epoxy as before. The bottom of the spring will connect to the platform by drilling the smallest possible hole in the platform and threading about 20% of the prone length of the spring though. Given the design of a spring by threading it through the platform there is effectively no chance of the spring ever backing out of the hole and if it does it will take no time to fix the problem. The spring-loaded pedals will right themselves and extend to a predetermined distance when each pedal is not being used just as in an automobile. The spring will also generate a resistance that mimics those of automobiles and the farther the pedals are pressed the more resistance that is felt by the driver just as expected. Potentiometers will be located under the pedals and will send data to the RC will be used to control the acceleration of it.

The pedals operated within a range of 45 degrees. This measurement in addition to the angle of the cockpit means that a fully pressed pedal will still feel a 30 degree angle and when prone will be on a 75 degree angle (both relative to 0 degrees/the floor). This element was designed to be as close to actual operating ranges of automotive pedal as possible, but given the need for design simplicity and efficiency and cost effectiveness some dissimilarities are present. They are however vastly ergonomic with the pedals operating in a manner closely mirrored to the foot movement performed while it stretches when pressing the pedal decreasing stress on the on the foot and ankle joint. The angling of the platform and pedals also put the driver's feet in a comfortable position which was necessary given the constant motion of the cockpit. The side view of the floorboard and pedal is shown in earlier figures.

Extending 90 degrees up from the floor board and approximately 18 inches from either end is where the television tree mount and corresponding steering wheel portion is located. The television tree mount will once again be created using 6000 series Aluminum square tubing. This is the only place where the square tubing was used. The television tree mount will have a side length of 3 inches. Square tubing will extend 19 inches up from the floor board upon which shaped sheet metal will be welded to form a rectangular cuboid of 21"x 20" x 4" dimensions. This is where the television display will be housed.

The height of the television mount was measured so that television will be positioned with the center of the viewing screen located at the eye level of the expected “average” driver. This design allowed the drivers to be in a comfortable viewing position in most cases. This reduced stress on the neck of the driver and not cause soreness or stiffness in the driver’s neck. On the right side of the mount is where the power switch was located. The switch will be comprised of a household light switch and acted as user controlled power switch within reach of the driver. This will allow a driver or spectator to power off the simulator if any mechanical problems occur or when the simulator is not in use.

Down the television tree is where the steering column will be placed. The steering column was mounted horizontally inside of the the 3 inch wide television tree square tubing creating a 90 degree angle. The steering wheel consists of a Sega Genesis Gaming controller shown below in figure 5.12.



Figure 5.12: Steering Wheel Front View

There was a hole drilled through the television tree congruent in height and area to those in the steering column. This allowed the steering column to be positioned securely in place once the driver has moved it to a comfortable length. To securely position the steering column a nut and bolt was used.

5.3.2 Cockpit Hardware Design

Located in **Figure 5.15** is the circuit that was used to convert the PWM signal to a DC value between 0 and 10.5V. It can be broken into three stages: To the left of M2 is a Buck Converter with smoothed output, and starting with M2 and ending with M5 is DC to DC step-up output, and the stage following M5 changes the direction of the actuator by inverting the signal using a relay system triggered by the sign control pin which is output to the motor. The relationship between the duty cycle (D) and the output voltage (V_{od}) when $0 \leq D \leq 0.14$ and $0 \leq V_{od} \leq 10.5$ (V) was found to be given by the regression $V_{od} = 69.184D - 2.6199$ (V), with a statistical R value of $r=0.998$.

This range is sufficient, as 0V is not enough to turn on the actuators (and if it is not, no duty cycle can be applied if desired), and are tolerant to approximately 1V above 12V, though not recommended. Simulations were done using MultiSim 11.0. Note that the N-MOS transistors used are IRFP250N, which are in the linear region $\sim 2V \leq V_{Gs} \leq 4V$, with an $R_{DS} = 0.075\Omega$ and is tolerant to 30A and 200V.

The same MOSFETs are used for the Cockpit electrical design, for cost reasons (discount on quantity), for accessibility (if one is to burn out, it is easily replaced by the bank of them possessed), for compatibility, and for ease of reference. The desired output voltage takes place in approximately 150ms (when outputting 12.07V), and can discharge in less than 100ms (more than sufficient for the tasks of this project). Also, this circuit is designed to work for a PWM output frequency anywhere between 1kHz and 10kHz. This range is necessary, as 10kHz is the maximum (and default) output frequency of the PWM, but less may be used because tasks to be performed by the Netduino may be too abundant to perform this task effectively.

This circuit is rated for 1kHz, as it is the frequency used based on the previous facts. The sign pin used operates as a 3.3V(1/True) or 0V signal(0/False), 1 assigning for extension (positive to positive, ground to ground), and the other (0) assigning for retraction (positive to ground, ground to positive). To switch between these, the sign pins will be utilized. 0 will be used to activate the extension circuit, and 1 will be used to activate the retraction circuit. It is necessary to perform this operation in this way, as the power supply available is unipolar.

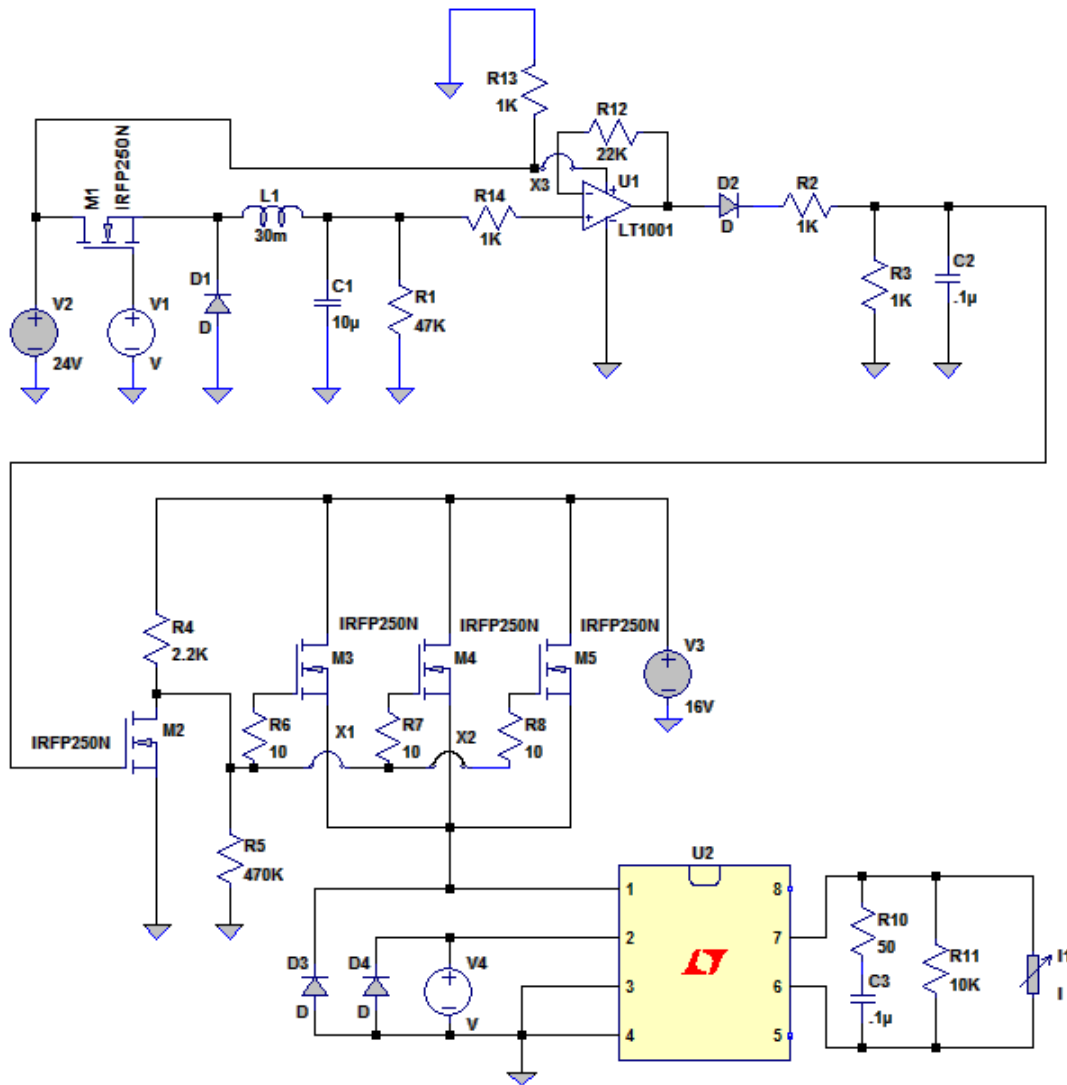


Figure 5.15: PWM to DC Voltage to Actuator Output Phase

1.) *Buck Converter*

A Buck Converter is ideal for this scenario, as the power applied to the actuators must have a quick reaction. It is not sufficient to use a standard Boost converter, as the charge time is too extensive, and the required transistors for such operation are not tolerant enough to the specifications for the actuators. Buck converter stage found in **Figure 5.16**.

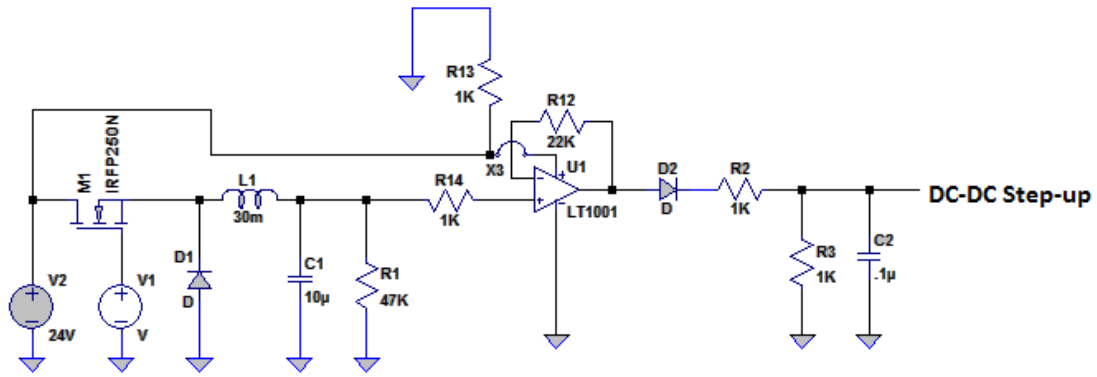


Figure 5.16: Buck Converter Circuit

The Buck Converter is of a standard design, using a 24V input voltage, 30mH inductor, a 10uF capacitor, a max 0.7V drop diode, a 47K Ω resistor (required for the situation when the output voltage is not high enough to turn on the diode on the DC-DC output stage input, and in discharge while not in use), with an enhanced mode power NMOS for switching. V1 in this simulation picture represents the output of the PWM pins of the MCU. The maximum current applied by the MCU is approximately 1mA, well within the proper range specified. The inductor used will be PLA10AN3030R4R2, rated for 400mA, 300V while in common mode with a leakage of 2.7 Ω . The max voltage applied at any given time to this inductor is approximately 5V, and the maximum current across it is approximately 225mA (well within range of its specifications). It is essential to have a 1A fuse between the power supply and the drain of Q3; if there was a situation in which the active components were to fail in this circuit, the diode can fail to short, causing damage to the circuit, or in the situation that the MOSFET is to fail, a temporary short will take place, applying too much voltage to the MPU. The intended maximum output voltage is 2.205V (to be explained in the inverting buffer section). This circuit achieves such at 22% duty cycle. Through simulation inspection, a sufficient linear regression was found between .04 and .22 duty cycle (D), ranging between approximately 0 and 4.205V (V_{buck}), with a value $r = .995$: $V_{buck} = 8.967D + 0.3148$

The requirement for 4.205V is within the requirements of the “DC/DC” increase. 2.1-4.2V was found to be needed to step up the voltage at Q1. The sign control stage output must be activated for this circuit to produce an output of any significant voltage. The Diode ensures the voltage from this portion of the output

circuit doesn't leak into the Buck Converter. The power NMOS must be in the linear region to achieve a variable voltage based on an input. This NMOS had to be tolerant to 20A, 24V, and had to have a low R_{DS} .

The Op-amp used is a LM324N, as it has use in uni-polar configuration while also being a QUAD Op-amp, thus requiring only one on the PCB. The LM7812 will be used, connected between the power supply, ground, and turn on terminal of the OP-Amp, so that voltage spikes in operation of the motor will never disrupt operations, or damage active components. The maximum current delivered by the operational amplifier is approximately 1mA, and the maximum voltage applied to Q1 is 4.4V. In no particular scenario does the current delivered (to one circuit) by the -5.1V power supply exceed 4.5mA, and thus in the worst scenario, will never exceed even 13.5mA for all three circuits.

2.) DC-DC Step up

An inverting buffer (common source) at Q1 then follows, giving gain to the DC voltage at the gate. The output from the drain is then received by Q3, a source follower, so that the output resistance is not affected by operation of the actuator. This source follower must have a 20A Fuse at the power source, as it is recommended while using the actuators. This is the point in which an actuator will most directly draw current from the power source, so this is essential in operation. That being stated, it is of great importance that Q3 be mounted to a heat sink. The output of the source is then followed by a 20A tolerant diode (.7V drop), making the max voltage applied to the actuators 12.07V. The output of the DC/DC stage also had a sufficiently linear regression for the input voltage between 0.574V and 2.205V (V_{buck}), and the output voltage between 1.83mV and 12.07V (V_{od}), with a value $r = .9987$: $V_{od} = 7.684V_{buck} - 5.002(V)$. If, for any reason, the buck converter output is to exceed 2.194V, the input of the actuator will not exceed 12.9V, as the MOSFET is in saturation at that point. The circuit can be found in **Figure 5.17**.

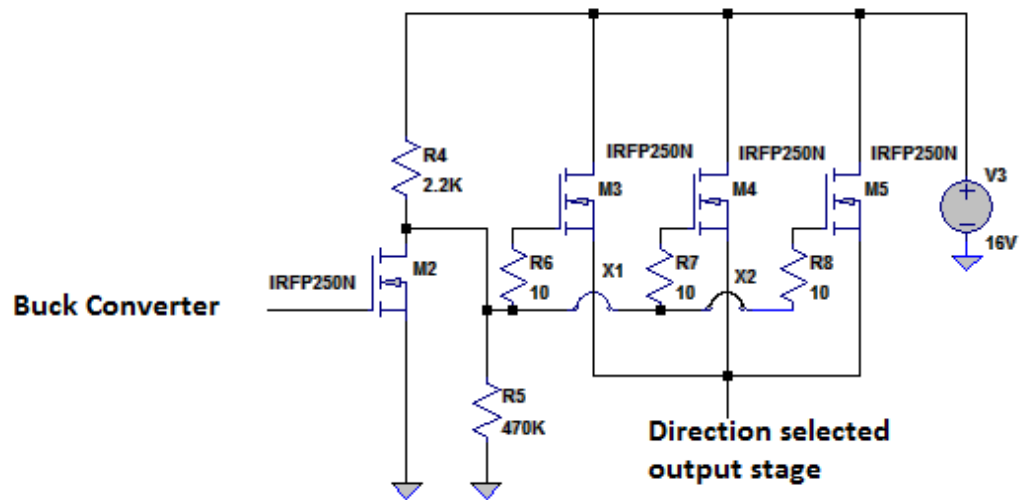


Figure 5.17: DC-DC Step up

3.) *Direction Selected Output Stage*

The DC-DC step-up circuit outputs the Direction selected output stage with a snubber circuit across the load. U2 represents a DPDT relay circuit used to change the direction of the actuators while also aiding them in stopping. The snubber circuit was necessary to help eliminate noise from the actuators, as they damaged output stage MOSFETs and provided back-feed to the processor. The diodes were necessary to help prevent back-feed through the processor in the switching process. V4 represents the sign bit output from the Netduino. In **Figure 18** the Direction Selected Output stage can be found.

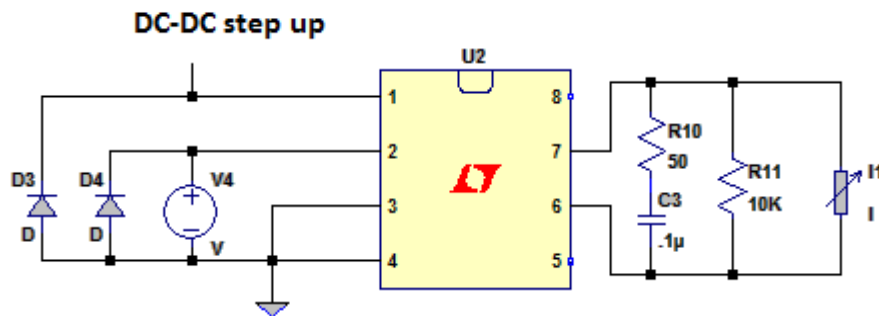


Figure 5.18: Direction Selected Output Stage

B.) *Netduino Power*

The Netduino is regulated by an LM7812 (12V voltage regulator mounted to a heat sink), to protect it from voltage fluctuation. It is very common in DC motor

operation that the power source is likely fluctuations in Voltage output. This can cause the Netduino to reset itself, or in the worst case scenario, be destroyed. Though this is not likely an immediate concern to the operator, it is essential for the RC Ghost Rider to not destroy itself.

D.) Transceiver Power

The XBee® operates at 3.3V DC to maximize the distance in which the hardware from the Cockpit can communicate with the RC Car. That being said, The XBee requires approximately 45mA in transmission, and 50mA in reception phase, with a power-down current of less than 10uA at 25 degrees Celsius. To accommodate the requirements to supply the XBee with 3.4V, an LM7805 will be used with a simple voltage divider between the voltage regulator output and the XBee supply port.

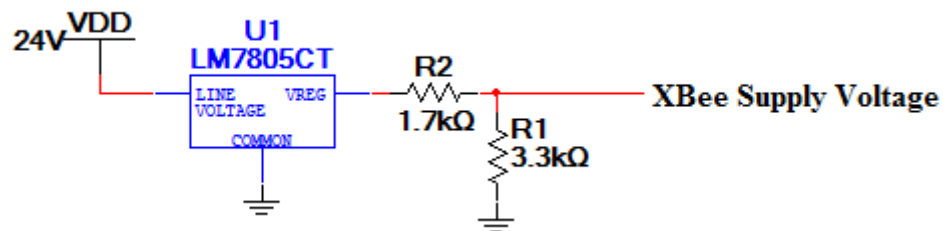


Figure 5.19: Transceiver Power Connection

E.) Power Delivery

It is clearly evident that all the parts chosen for the hardware design require no more than 24V (with exception to the -5.1V DC power supply); However, current delivery is quite essential in this scenario. The PWM to DC output stage while in connection to the actuators, and while rapidly changing between extension and retraction stages (stepping swiftly between 12V DC and -12V DC) received no more than 75A from the power supply in simulation. This situation causes a drain from the power supply at slightly more than 22A delivered to each actuator, noted as a much more than worst case scenario for this design, as the 20A fuse at the placement of the Drain of the voltage follower is present. Also, note that it the leakage of the motor was assumed to be abnormally low for these power distribution simulations (0.6 Ohms).

That being said, the XBee requires no more than 50mA, and the potentiometer circuits each drew less than 5mA. Assume, for the purpose of this discussion on power delivery, that the Netduino is delivered its maximum current of 200mA, which, due to testing in programming and careful construction, will not happen unless a manufacturing error takes place. This would bring the total current delivered by the power supply to 75.255A. This is well within the range of the specifications stated in the "Relevant Technologies" section, and thus, has further verified the selection of the power supply.

5.3.3: Cockpit – Processor Platform Programming: Structure and Duty

The Netduino connected at analog pin 0 to the steering wheel, analog pin 1 to the brakes (through D/A), and analog pin 2 to the accelerator pedal. Digital Pin 0 will be used for the turn-on operation. Pin 1 will be utilized for the reverse switch. Digital pin 7 is to receive the signal from the transceiver that is transmitted from the RC car to detect vibrations. Digital pin 8 will transmit a data signal to the transceiver so the cockpit can control the car's motion. Pins 5, 6, and 9 will be used to transmit the signal to the buck converter, then to the DC to DC converter which controls the actuators, as they are PWM output terminals which can be used for D/A conversion. Pins 3, 10, and 11 (from now on referred to as "sign pins") will be used for switching between positive and negative D/A conversion networks. Analog pins 3, 4, and 5 will be used to for potentiometer readings from the actuators.

The program is to initiated by a button found in the cockpit. If a person is not in the seat, then no signal will be sent to the actuators. Given an appropriate weight is detected; this value will be stored for later use in the program. If a person in conjunction with the weight of the apparatus is deemed too heavy for operation, no signal will be sent to the actuators. This will be discussed again later in this section.

The main loop begins following the reception of the above mentioned value, and the program will stop once a person pushes the off button, which is found on the steering wheel, thus ending the program. This, mind you, is the same button that initiates the main loop.

Flow Chart Color Key: Red - Hardware Input/Output, Blue - Functional Input/Output to be used for another function, Purple - Predefined Function, Green - output that initializes, Orange - Process, Yellow - Decision

Synchronization Function:

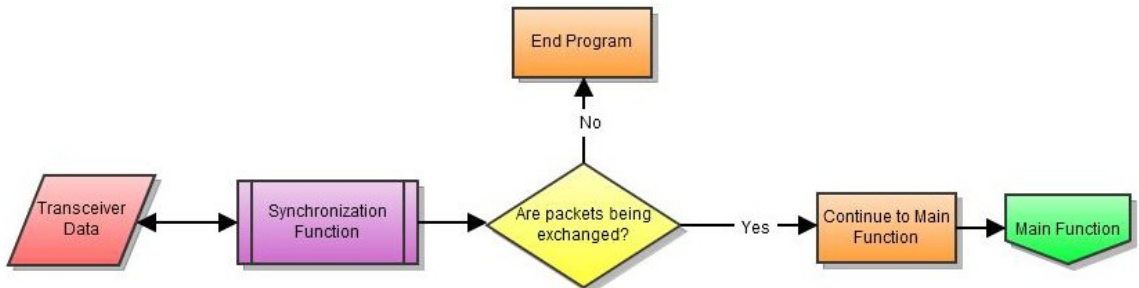


Figure 5.20: Block diagram of Synchronization function

This function begins the operation of the car from the cockpit as it will detect that the car is on, and then synchronize so that the data signal will be unpacked properly.

Unpack Function(s): The signal received from the car at pin 7 will be decoded in software to separate into four separate values for each of the four potentiometers from the car and the speedometer.

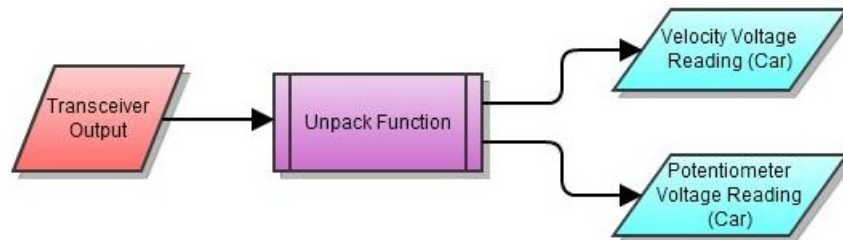


Figure 5.21: Block Diagram of Unpack Function

Potentiometer/Speedometer Voltage to Displacement/Speed Conversion Functions:

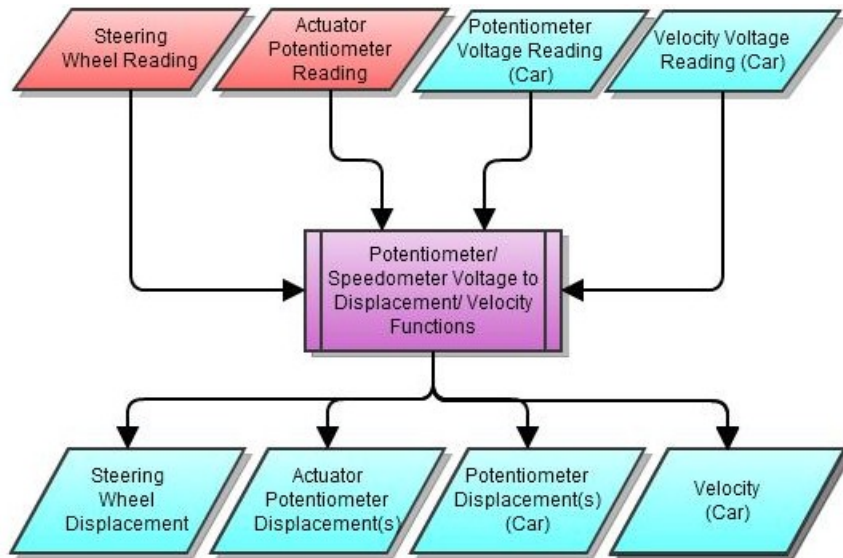
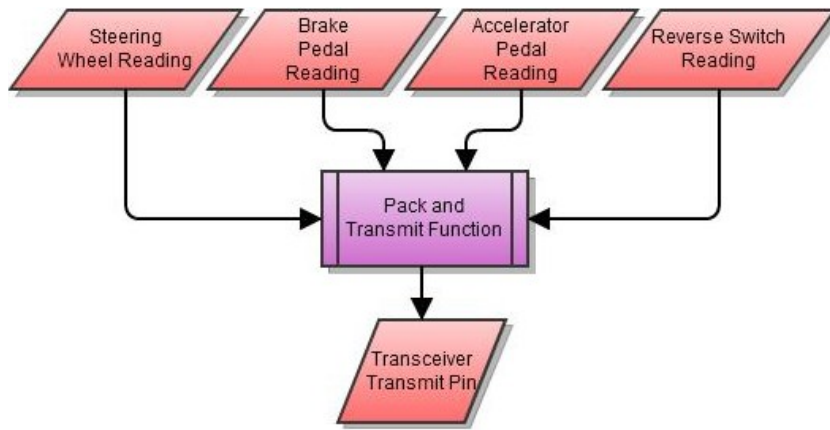


Figure 5.22: Block Diagram of Potentiometer/Speedometer Voltage to Displacement/Speed Conversion Functions

This function takes in one of the voltage values from the Unpack Function, the steering wheel, the accelerator pedal, and the brake pedal and convert it back to the displacement of the potentiometer, and the measured speed from the speedometer. Then, displacement and speed will be scaled depending on the proportions of the dimensions of the cockpit, to the dimensions of the car. This function will also convert the potentiometer readings from the actuators to

displacements. The voltage read is transmitted as a value between 2.7V and 3.2V, as these are within the specs of the analog input of the Netduino. The value of the potentiometer reading the steering displacement will note that anything less than 2.95V is a negative displacement, any value greater than such as positive, and any value nearly equal to is exactly equal to 2.95V as 0 displacement.

Pack and Transmit Function: The signal from the steering wheel, brakes, and accelerator pedal (here after deemed operator controls) is taken to, so that they can be transmitted to control the car using pin 8. The steering wheel signal is interpreted to be passed later by the *Operator Control Calculus Function*



. **Figure 5.23: Block Diagram of Pack and Transmit Function**

Operator Control Calculus Function(s):

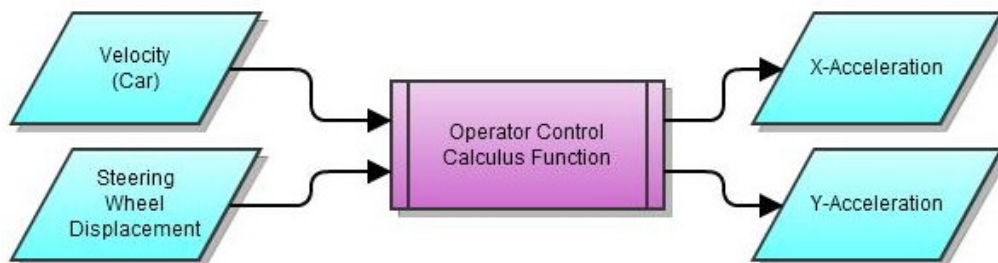


Figure 5.24: Block Diagram of Operator Control Calculus Function

Two functions is created to pass three variables, which are the values submitted from the steering wheel and two which are the velocities of the car from the *Speedometer Voltage to Displacement Conversion Function* sampled at a time spacing of t_s . Each of these functions will output the net acceleration applied in two dimensions. Let x denote the function pertaining to the dimension left and right of the car, where left is negative and right is positive. Let y denote the function pertaining to the dimension relative to the front and back of the car,

where toward the front is positive, and toward the back is negative. Let s be the percent of application applied to the steering wheel, where s when negative is left, and when positive is right (s is a decimal value). If $s \neq 0$, let the resulting radius from turning be denoted by $r_c = r_{\min} / s$, where r_{\min} is the smallest turning radius possibly made by the RC car. For the “ y ” function: Let v_{c1} and v_{c2} be the sampled velocities, respective to sample order. Then the acceleration of the car over the duration of sampling can be found as follows:

$$a_{cy} = \frac{v_{c2} - v_{c1}}{t_s}$$

For the “ x ” function: if $s = 0$, then the acceleration $a_{cx} = 0$.

$$\text{If } s \neq 0, \text{ then } a_{cx} = -\frac{v_{c2}^2}{r_c}.$$

Note that the direction is determined by the sign. If the car is turning left, the car’s motion from the top is counter-clockwise, thus the feeling is rightward. The contrary would be true, given the car is turning right.

Potentiometer Calculus Function:

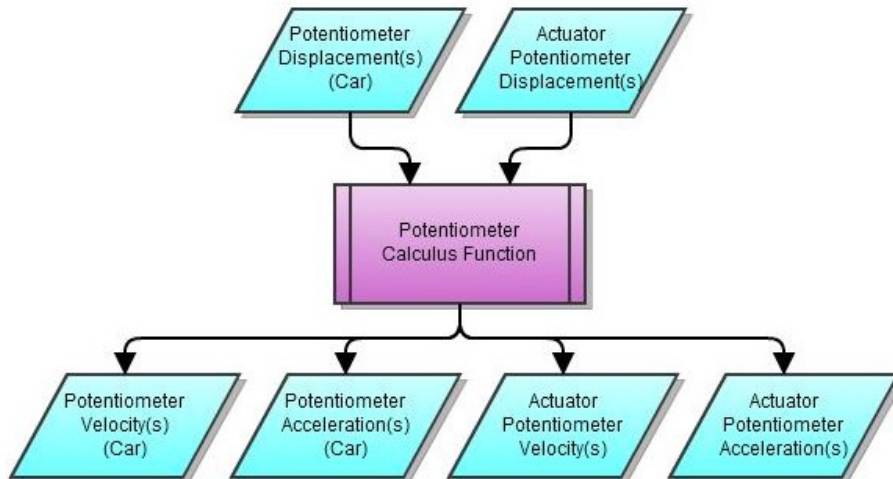


Figure 5.25: Block diagram of Potentiometer Calculus Function

This function passes one value from the *Potentiometer Voltage to Displacement Conversion Function* to determine the amount of force to be applied in the vertical direction by the actuators. Three samples from the signal are to be taken at a time, corresponding to three separate, evenly timed, values measured from the potentiometers, as at least three samples are required to sample to find acceleration based on displacement. These values is d_n, d_{n+1} and d_{n+2} , while t is the time between measured samples. Thus, the average velocity between d_n and d_{n+1} is v_n , and the average velocity between d_{n+1} and d_{n+2} is v_n , and the average acceleration over the range of the three samples is noted as a_n . Thus the

following is the mathematical framework of this particular function is:

$$v_n = \frac{d_{n+1} - d_n}{t} \quad \text{and} \quad a_n = \frac{v_{n+1} - v_n}{2t}$$

This function will return both v_n and a_n .

Operator Weight Function:

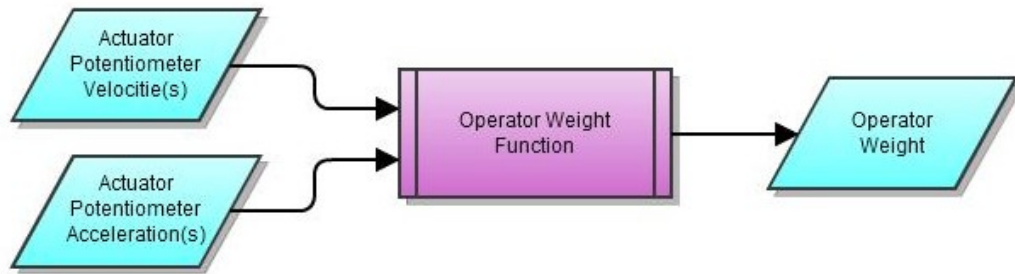


Figure 5.26: Block Diagram of Operator Weight Function

This function will determine the operators weight based on the *Potentiometer Calculus Function*. Upon turning on the cockpit, and once synchronization has taken place, the rates is noted by the MCU in which the force applied on the actuator will determine the weight of a person. If the actuators move slower at a set voltage, relative to software testing results, the person is heavier. The contrary would also be true.

Final Actuator Output Function:

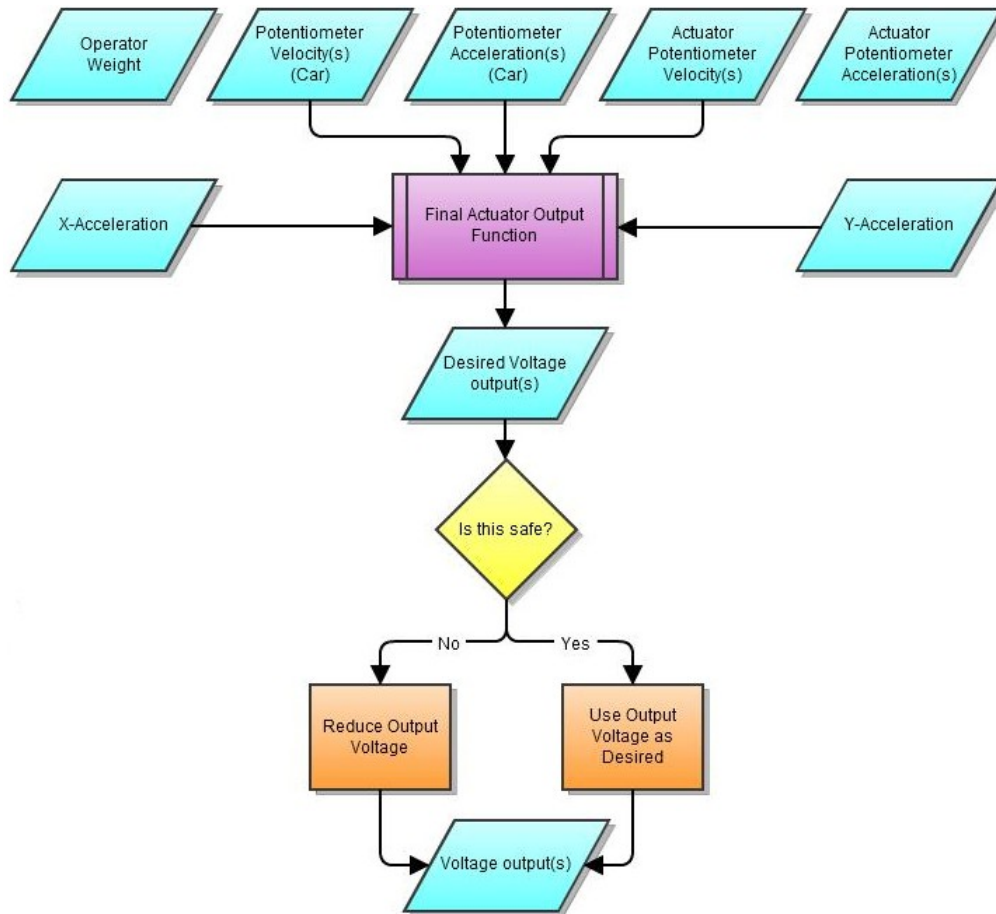


Figure 5.27: Block Diagram of Final Actuator Output Function

A function is created to pass the following variables: the weight of the person with the apparatus, the values taken from the actuator potentiometer reference, the values taken from the *Operator Control Calculation Functions*, and one *Potentiometer Force Functions*. This function will determine if the applied force from the actuators is safe for the vehicle, and most importantly, safe for the operator. This is set by a threshold, to be determined in construction of the vehicle. Then, the function will apply the input values to determine how much voltage is to be output to an individual actuator. The output of *Final Actuator Output Function* is to be modulated, and output through pin 5, 6, or 10, based on the corresponding actuator. Given that the actuators extend a total of 6 inches, 5 inches of motion is dedicated to the output of a_{tot} , where the other inch will, roughly be dedicated to the potentiometer motion on the front actuators. When the car is at rest and synchronized to the cockpit, the cockpit is lifted 3 inches in the front, 3 inches in the back, and will retract and extend in the range of d_r , the maximum range of motion allowed from the center point (or “base” level) of the actuator. This value, d_r , is allowed 2.5 inches swing in both the upward and

downward direction. These voltages are to be applied for a time-duration so that those actuators is adjusted to the desired height. Mind that these displacements are to be kept for the next loop-through of the function, and these values is adjusted to account for changes in acceleration. The mathematical framework of this function is as follows: Let a_{tot} be the total force, also accounting for direction (in radians). Let $a_{x\max}$ be the maximum acceleration as calculated in the x dimension, and $a_{y\max}$ be the maximum acceleration as calculated in the y dimension (both with margin of error fully considered). Mind that $a_{y\max}$ is the stopping acceleration, calculated by $a_{y\max} = v_{\max} / t_{stop}$, with v_{\max} as the maximum speed of the RC car and t_{stop} as the time required to bring the car to a full stop from maximum velocity. These are scaled for ease of calculation, and to make a more enjoyable, rather than a necessarily more “realistic” experience, as turning intensity would be greater in more frequent circumstances. Then, let $a_x = a_{cx} / a_{x\max}$ and $a_y = a_{cy} / a_{y\max}$, and let the angle of the acceleration be θ .

If $a_x = 0$, then

$$a_{tot} = \begin{cases} |a_y|, \theta = \frac{\pi}{2} & \text{If } a_y \geq 0 \\ |a_y|, \theta = \frac{3\pi}{2} & \text{If } a_y < 0 \end{cases}$$

If $a_x \neq 0$, then the total acceleration $a_{tot} = \sqrt{a_x^2 + a_y^2}$ at an angle of

$$\theta = \begin{cases} \arctan \left| \frac{a_y}{a_x} \right|, & \text{If } a_x > 0 \text{ and } a_y \geq 0 \\ -\arctan \left| \frac{a_y}{a_x} \right| + \pi, & \text{If } a_x < 0 \text{ and } a_y \geq 0 \\ \arctan \left| \frac{a_y}{a_x} \right| + 2\pi, & \text{If } a_x > 0 \text{ and } a_y < 0 \\ -\arctan \left| \frac{a_y}{a_x} \right| + 2\pi, & \text{If } a_x < 0 \text{ and } a_y < 0 \end{cases}$$

Let V_{motion} be the (variable) greatest scaled voltage submitted from the Netduino to the actuators for acceleration transmission to the cockpit to allow for comfort of the operator, applied as a ratio of the current acceleration against the largest acceleration in a two-dimensional direction. This particular value is found in the ongoing testing phase. Then, let V_{react} be the voltage required to react to make a change in angle that is the negative of V_{motion} . This can be found by multiplying a constant (to be verified and found in testing and construction) to the result of Equation 5.7. If $0 < \theta < \pi/2$, then V_{motion} is noted to be assigned to the pin corresponding to the rear actuator, and V_{react} is noted to be assigned to the front-right actuator. If $\pi/2 < \theta < \pi$, then V_{motion} is noted to be assigned to the pin

corresponding to the rear actuator and V_{react} is noted to be assigned to the pin corresponding to the front-left actuator. If $\pi < \theta < 3\pi/2$, then V_{motion} is noted to be assigned to the pin corresponding to the front-right actuator and V_{react} is noted to be assigned to the pin corresponding to the rear actuator. If $3\pi/2 < \theta < 2\pi$, then V_{motion} is noted to be assigned to the pin corresponding to the front-left actuator and V_{react} is noted to be assigned to the pin corresponding to the rear actuator. If $\theta = 0$, then V_{motion} is applied to the pin corresponding to the front-left actuator and V_{react} is applied to the pin corresponding to the front-right actuator, while the rear actuator is set to the “base level”. If $\theta = \pi$, then V_{motion} is applied to the pin corresponding to the front-right actuator and V_{react} is applied to the pin corresponding to the front-left actuator, while the rear actuator is set to the “base level”. If $\theta = 3\pi/2$, then V_{motion} is applied to pins corresponding to the rear actuator and V_{react} is applied to the pin corresponding to both of the front actuators. If $\theta = 5\pi/2$, then V_{motion} is applied to pins corresponding to both front actuators and V_{react} is applied to the pin corresponding to the rear actuator.

Then, the values from the *Potentiometer Calculus Function* is applied using the values received from the potentiometer readings from the car. The voltage, V_{Pot} , corresponding to the velocity, v_{n+1} , is applied for the sampled duration t . $V_{Pot} = \alpha v_{n+1}$ is the relationship between V_{Pot} and v_{n+1} . The constant, α , is experimentally found and verified in the building and testing procedures. The displacement of the front actuator is stored for the next loop through, as this displacement is $d = v_{n+1}t$. This displacement is restricted by the 5 inch total range of motion discussed earlier for potentiometer control of the cockpit. Thus the swing of motion is 6 inches on the varying actuator axes in both directions. The greatest acceleration is used from the potentiometers, and applied to front actuators, V_{Pot} at the pin corresponding to the side with the greatest acceleration and $V_{Potreact}$ (from equations 5.4 and 5.6) at the pin corresponding to the side with lesser acceleration. In the event that all potentiometers are at full extension (for example, if one was to propel the RC car off a ramp) the only voltage to be applied to the actuators would be the voltage required to keep the actuators at the level of extension they were in prior to the time in which the RC car was momentarily a projectile. This voltage is further determined by the weight of the person.

Once the acceleration in any planar direction is decreased, or decreasing, the cockpit will steadily make its way back to base position. Once the acceleration in any one planar direction is zero, the cockpit should already be placed back at base level.

Reaction voltage:

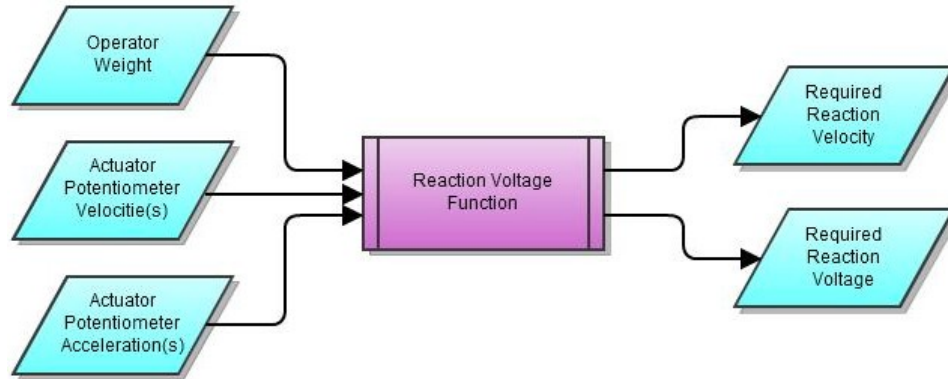


Figure 5.28: Block Diagram of Reaction Voltage Function

The reaction voltage, $V_{Potreact}$, found by scaling equations 5.4 and 5.6, is calculated by the processor using the values from the *Potentiometer Calculus Function* after passing the values calculated from extending actuators to solve for $d\ell/dt$ (the velocity function output). The value of the change of angle of the retracting actuator(s) must be the additive inverse of the change of angle of the extending actuator(s). This is absolutely necessary, as not all people have the same center of gravity or weight. Safety is of the greatest concern, and so is stability in this project.

PWM Output Function:

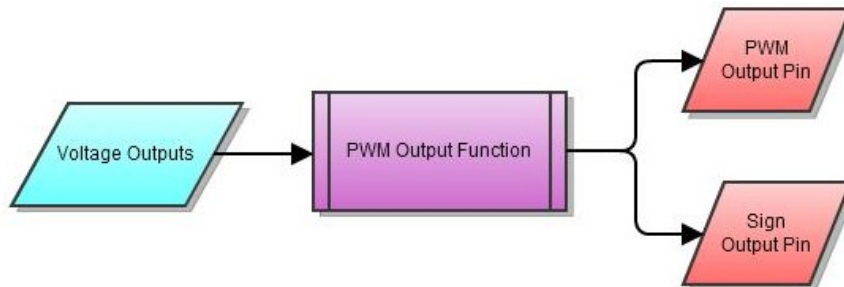


Figure 5.29: Block Diagram of PWM Output Function

The PWM is adjusted through programming and output from the designated PWM pins. For the duty cycle, D , and the desired output voltage, v_o , $D = (v_o + 11.46) / 69.184$ in decimal form (Based on simulation, not necessarily true to actualization of the circuit). The sign of the voltage is transmitted as the output of the sign pins to choose which D/A conversion network to use. In the event of a nearly immediate sign change in the direction of the actuator, the PWM signal must be delayed for approximately 0.08 of a second, to discharge the previous signal. If this step is not taken, motion is jerky, and the actuator will likely draw current succeeding 20A, breaking a fuse in operation.

6 – Prototype Construction and Coding

6.1) Structural Modification: Process

In order to adequately secure the various components to the vehicle, proper precautions needed to be taken. The linear motion actuators needed to be mounted to both the top and the bottom in order to ensure their stability. For this purpose a set of small hinge joints was used in order to account for the various motions the actuator will encounter. The part of the actuator from which the wires extend needed to be at the top of the shock in order to ensure that the wires involved were not tangled unnecessarily. In addition the accelerometer needed to be securely fastened to the frame of the vehicle. This was a relatively simple accomplishment as the accelerometer was directly mounted to the PCB board. Four small screws were used to fasten the PCB board to the frame of the vehicle to ensure stability.

In addition the MOSFET's that are to power to motor also needed to be securely fastened. A heat sink for each of the MOSFET's was mounted to the vehicle's chassis by way of four small screws. The MOSFET's was then to be mounted to the heat sink by way of the single screw located on them. The batteries that were used for the cars motion system were also need to be securely mounted. As stated before a Velcro attachment system was to be used to attach the new batteries. The one battery was placed into the cavity beneath the PCB and sealed in place with a small piece of Velcro, while the other battery was velcroed to the top of the plexiglass on the car. Likewise the transceiver, transmitter and camera were also securely mounted.

The transceiver was soldered onto the PCB board that was mounted to the back center of the vehicle, and thus needed to be very securely mounted. The transmitter needed to be mounted closely to the front of the vehicle, this time on the erector set attachment. A small stand was made on the vehicle in order to accommodate the transmitter. This stand consisted of a small piece of erector set with Velcro on the top that was screwed into place to secure the transmitter. There will need to be a flat part on top of the erector set to allow for the antenna connected to the transmitter to stick out of the top. As previously stated, the top of the erector set platform also housed the camera that was used to get the optical data. The camera had also been mounted onto the vehicle by means of Velcro.

6.2) Programming RC Car Processor Platform: Process

In order to program the Atmega328p, there needed to be an environment in which to program in. For the purposes outlined in this project, it has been decided that the "C" programming language was to be used rather than "Assembly

Language”. This had been decided because of the amount of time and effort that would have been necessary to be spent debugging assembly code, which does not outweigh the subtle performance drawbacks of writing in “C”. In order to get started with the programming process, the Arduino 1.0 development environment was installed onto the development machine. The next step in the programming process was to get comfortable programming in the new environment. Once this comfort was established, the programming for the vehicle began.

In order to adequately test various parts of the structure of the program, it was necessary to complete the subroutines or functions in a specific order. Doing so allowed for the proper testing of each routine before another routine was written, saving time in the debugging process. This led to great confidence that the overall program would function as intended when the code writing had finished. Because it was necessary to use the values that were read in from the input pins on the board, the collection of data was the first routine that was written. When this routine had been written it was tested as the software testing portion of the paper dictates. After the acquisition of data has been handled, the routine that decides the importance of the data acquired needed to be coded. This portion of programming must have been done after the program had received data, because in order to test the data for validity, there must first be data to be tested. Likewise, the manipulation of the output buffer happened after the check for significance of the data in question.

The next step in the process was where the MCU received the data from the transceiver. In order to write the software portion necessary for this, there also needed to be software for the cockpit's MCU that will transmit data, unless a dummy signal was to be routed directly in through the transceivers input/output ports. Using a computer and an XBEE explorer board, it was necessary to route a dummy signal to the MCU on the vehicle in order to test the acquisition of the data in question. After it had been established that the data was being received, another function was created to allow for the loop back of information from the MCU to itself. This needed to be written after data had been received because in order to test the data to see if anything has changed, there needed to be data in the first place. The next portion of the program that was written was that in which the MCU actually transmits the data to the cockpit. At this time it was necessary to implement this function. This portion was written after the portion about collection data from the sensors, because in order to send off the data, there must first be data to be sent. In order to create this program given the set of functions that needed to be written, it was easy to see that a simple flowchart could depict what portions of code needed to be written in what order. The two separate flow trees showed that both trees could be simultaneously accomplished without the other tree.



Figure 6.1: Order of program creation #1

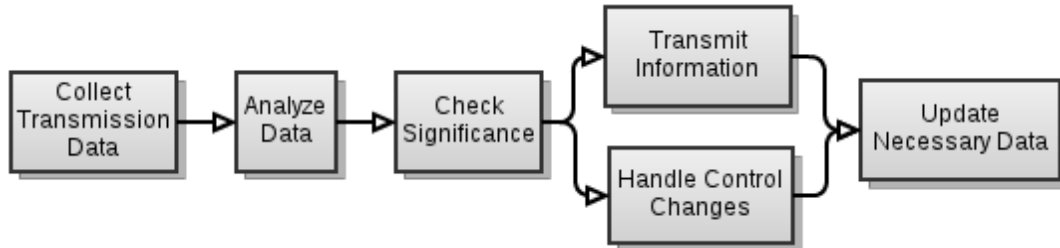


Figure 6.2: Order of program creation #2

6.3 Construction of Cockpit

The first step in the construction was to buy all of the prefabricated components of the cockpit which include: three actuators, driver's seat, 3 point harness, steering wheel, pedals, heim joints, actuators and television. It was paramount that the actuators be purchased as early as possible to ensure there is adequate time to test and analyze the movements of each independently. During the period of time the actuators were being tested the first portions of metal material were bought. These portions were used to create the cockpit base and the floorboard. The steel I beam used to fabricate it couldn't be bent or shaped into any sort of curvature. That means that the framing was comprised of only straight segments of steel I beam welded to other straight segments of I beam ensuring a proper weld and connection. The base consists of aluminum diamond plate material secured on top of a plastic pallet. The steel was MIG welded when being connected to other steel components and the aluminum was TIG welded when being connected to other aluminum material.

The base acted as a platform to help some user's up into the cockpit seat. The base's primary function was to raise the entire cockpit height enough so that while the cockpit is in motion the floor board will never come in contact with the ground. To make sure of this the platform had a height of 5 inches. This allowed for the platform to reinforce the actuators as well as help stabilize the cockpit during movement. Attached to the bottom of the platform was a series of rubber stoppers. The stoppers were used to prevent the whole cockpit assembly from sliding and shifting across the floor. The base of the platform is also where the bottom of the three actuators connected. The figure below shows the actuator place relative to the base of the cockpit.

The cockpit base and floorboard was mainly constructed using two materials: 6000 series Aluminum and 5000 Galvanized Steel. These materials were chosen

due to their ease of acquisition, strength, durability, weight, and the minimal amount of work it took to weld each material respectively. The other materials used that are not classified as either cockpit base or floorboard are the actuator bracing, which was fabricated using 2 inch Schedule 40 Galvanized Pipe with a two inch inner radius to fit over the actuator casing. Another was the 1.5 inch Galvanized Unistrut that was MIG welded to the Galvanized Pipe and was bolted to the 1.5 inch by .375 inch Galvanized Bar steel. The bar steel was bolted to both the Unistrut and the floorboard to hold the actuators in place.

The cockpit base is comprised of 6000 series Diamond Plate Sheet Aluminum (4 feet by 5 feet) and a plastic pallet of the same dimensions. The plastic pallet was selected due to its strength, ability to withstand extended periods of unspecified amounts of force, and its height (6 inches). The diamond plate was attached to the pallet by applying generous amounts of industrial grade quick drying epoxy and a series of four self-tapping screws on each side of the base (16 screws) total once the epoxy dried. Once this was completed six 90 degree .25 inch thick aluminum tabs were TIG welded to the top of the cockpit base which were used to securely connect the base of the actuators.

The cockpit floorboard was fabricated using 17 inches by 60 inches sheet of 6000 series Diamond Plate Sheet Aluminum and 5000 Series Galvanized Steel I beam. The steel I beam was used to create the frame or "skeleton" and was MIG welded in such a way that it created a rectangle with a width of 17 inches and a length of 60 inches. The diamond plate aluminum was then placed on top and fasted to the frame by a series of .375 inch thick steel self tapping screws placed 6 inches from one another along the perimeter of the floorboard.

One reason the steel I beam was chosen to fabricate the cockpit floorboard was the fact that its easier to cut and weld a straight line. This created a more secure feel when the cockpit was in operation and also created a secure and safe insertion point for the actuators to connect to. Also through considerable research it was found that since all portions of the cockpit frame were straight portions of material they didn't require there to be any bending or shaping of the tubing, thus saving copious amounts of time. There was also considerable lateral, shear, and moment forces placed on the framing while the cockpit was in operation due to the shift in center of gravity (of the operator in conjunction with the platform), the Steel I beam was far more resilient to warping, bending, and stretching.

The pedal was constructed of the same 6000 series Diamond Plate Aluminum. It was 3 inches wide by 6 inches long and had the top two corners cut and rounded so that if the operator was wearing sandals or barefoot they wouldn't cut the bottom of their foot. The pedal was mounted on Galvanized Steel Square Tubing that was 3 inches long by 3 inches wide with a wall thickness of .125 inches. The

pedal was connected to a self closing steel hinge joint using three steel self tapping screws, and the hinge joint was connected to the steel square tubing spacer using 2 of the aforementioned steel screws. This square tubing was used to elevate the pedal to a more comfortable position and also to lift the pedal so that when a Knob was glued to the hinge joint it would rotate the potentiometer the would control the RC Car's acceleration and brake. The steel square tubing was connected to the floorboard using 2 of the same screws as mentioned before. There was a 6 inch steel bolt that ran through the bottom of the floorboard and was used to limit the range that the pedal could move.

To securely attach the top of the actuators to the cockpit floorboard a heim joint, also known as a rod end bearing joint, was be used (which is shown above). This joint type was bought not be fabricated and it was made of an unspecified class of Galvanized Steel. This particular joint is an extremely strong and durable joint used in a wide variety of applications, but most often used as control rods, steering links, and tie rods in vehicles. Heim joints are precision articulating joints consisting of 3 parts. The first is a ball swivel which allows the joint to move in any direction on the three axes with almost 340 degree range. This portion of the joint is a hollow circle which allows a bolt or many other connecting devices to pass though and attach to the heim joint. This portion of the joint created a secure connection of the actuator and floorboard frame underneath the seat. Three of these joints were utilized in the project and connected to the frame under the seat and attached to the floorboard I beam skeleton by attaching to a steel bolt with a shaft length of 4 inches that was MIG welded to the frame.

The next is the housing, a circular casing that the ball swivel is encompassed by and its function is to hold the ball swivel in place and is the rigid part of the heim joint. The housing required some lubrication be applied to it approximately once an hour depending on the amount of use to be sure the joint didn't seize up during operation which would have put additional lateral force on the joint, the framing, and the actuator .The final part of the joint is the threaded shaft. This is connected to the casing and acts as the anchoring part of the joint. The thread can be either left or right hand threaded and allowed it to be screwed or bolted into various materials (mostly metals). The threading was cut off and the bottom of the heim joint was MIG welded directly on top the of actuator piston head. This did not compromise the joints integrity in any way and gave us the most secure and structurally safe attachment method. And given that the bolt and the joint are made of the same material, the was not grinding or shearing of any material due to a the joint and the bolt having the same density and hardness.

This joint easily allowed each actuator to move in the desired directions without fear of rubbing, forced over/under extension, and provided a solid connection that avoided slipping. The base of the Cockpit which touches the floor was considerably "wider" than the floorboard, underneath where the front two

actuators will be mounted. This meant that since each actuator was placed in an exactly vertical positioning, that there was a vast amount of material on any which side of the front two actuator so that if there was a design change or amendment there was plenty of room. This also eliminated the required dynamic movements that would have been placed on the actuators had they been placed on any sort of angle.

To ensure that the Galvanized Steel used to construct the actuator's bracing system is connected safe and securely a Metal Inert Gas (MIG) weld was used to create every joint on the frame and to create the base. MIG welding, also known as Gas Metal ARC welding is a process developed specifically for welding steel and other non-ferrous metal .This method of welding securely connected the Unistrut and Steel tubing together without heating the steel too much causing the end pieces to loose structural integrity and become brittle.

One advantage of MIG welding is that it allowed metal to be welded much more quickly and efficiently than other more traditional methods. This made it ideal for welding softer metals such as again, steel. Another advantage of MIG welding is that it produced long, continuous welds much faster than traditional methods. This specific welding process help cut down on the time spent fabricating the actuator bracing so we were afforded more time to work on the software aspect of RC GhostRider. The shielding gas that was used during the welding process helped to protect the welding ARC and also produces a clean weld with minimal splatter.

There were very few drawbacks that may be encountered when performing this type of weld. The equipment for this is considerably complex. This means that when outsourcing this portion of the project the welder that was selected had to be extremely well versed in this form of welding. Given that this type of welding is somewhat of a specialty, the fabrication could have cost a little more than was previously expected. Another disadvantage is that since there was a need for an inert gas shield, this weld could not be performed in an open area where wind would have been a problem destroying the gas shield, so the backyard was not an option.

The welding process that was used for all of the Aluminum on the cockpit base and floorboard was a Tungsten Inert Gas (TIG) weld. TIG weld is used exclusively for Aluminum materials only as it is a very high heat welding discipline and would melt many of the other non ferrous types of metal. TIG welding is another form of ARC welding as described in the MIG welding portion and it uses a non consumable tungsten electrode to produce the weld. The weld is protected from the atmospheric contamination by a shielding gas. The shielding gas that was used for our TIG welds was a mixture of 75 percent Argon 25 percent Nitrogen. This produced a very clean and secure weld, but the mixture is also known as "self cleaning". This means that while the weld was

structurally as strong as possible, it also burnt off specks or flakes of the Aluminum that when cooled could have become sharp debris that cut group members when transporting the cockpit or users if it got in their shoes or clothes. This mixture of Argon and Nitrogen was a little more expensive, but when along with the theme of user safety first.

6.4: Programming Cockpit Processor Platform and Circuitry Construction: Process

It should be noted that testing in this project is an ongoing procedure, as testing is necessary to ensure safety and proper operation without destroying the apparatus. Programming the platform was begun by ensuring that imaginary input values are accepted as predicted, and assure they are converted properly using the given functions. Following such action, inputs were placed for operator controls, and output values from the converting functions were found. Voltage values and processor function outputs were documented until acceptable results are found. Then, pulse width modulation outputs were tested with the PWM to DC circuitry, to ensure that the proper DC voltages are output.

Though hardware testing on the non-moving circuitry would be considered physical construction, the circuitry is heavily dependent on the processing platform's performance. Circuitry were tested using the collected components and required components. First, the buck converter was constructed and tested for a desirable range of voltages, corresponding to the simulated design. Once a desirable range of voltages can be found on the output of the buck converter, a regression was formed to suit the response of an output voltage due to the duty cycle. This can vary, as other stages may require that this output vary differently in a real-world environment.

The sign-selector circuit along with the DC to DC step up function was tested simultaneously, as resistor values on the drain of the sign selector circuit can produce undesirable results, as it is a common-source buffer. A regression was formed based on output voltages versus input voltages (where the buck converter is to be connected) once adequate results are achieved. Then, a reverse in polarity test will take place to determine that second-circuit inversion of the voltage applied to the actuators will work as predicted. In this "inversion testing" process, things such as current applied to the unused circuit was tested. Given that the prior takes place, the circuit was connected in its entirety, and verification will take place to ensure that interfacing all three circuits yields desirable results. Finally, a duty cycle to final output regression was constructed to be used in programming. Discharge times of the circuit will also need to be documented.

The process of programming the Netduino was heavily dependent on hardware testing. Once the prior steps are taken, the complete circuit was applied to an actuator to determine the actuator's response to varying loads. In this process, it

can be determined how displacement, velocity, and acceleration are effected by active loads, which will lead to a significantly safer operation. Documenting such things will determine the total force applied by the actuators, which was used in computing the weight of a person. The proportional change in weight will vary actuator extension, and a regression was established. Also, this will determine specific factors about live load that effect lateral forces on the actuator, to help deem variables which can cause unsafe operation. A regression establishing a connection between weight and velocity of response in both extension and retraction will need to be documented. Potentiometer readings are essential in such action, and were received by the MPU and processed to find the velocities using the functions established in the *Cockpit Processor Platform: Structure and Duty* section.

The prior step was applied to programming, adjusting factors for the response voltage for retraction vs. the extension of actuators. All were tested, to ensure that changes in angle are cohesive allowing that smooth pivoting around a central point are correlated. This was tested with two actuators first, bottom portion affixed to the lower platform of the prototyped cockpit without the seating apparatus affixed, with a flat surface (such as a wooden board) balanced on a pivotal axis attached to the top. Then, varying loads were added, and results were documented using potentiometer readings. Then, virtually the same procedure was performed using all 3 actuators. Adjustments were made in programming until acceptable results are found.

Interfacing the previous steps will allow that the structural construction of the cockpit can take place, and then the previous tests were performed on the fully constructed prototype. Varying tests described in the testing section will take place at this phase. Then, RF interfacing will take place to wirelessly connect the cockpit to the car, and prioritization testing will take place to determine how long and how frequently command execution is to take place. This part of programming will likely take place until completion of the project.

After meeting a significant milestone of the prioritization phase, examination of forces applied to the car were tested based on motion will take place. Using output data from the USB port of the Netduino, without load or connection to the actuators, response due to X-Y acceleration was documented. Once deemed desirable, the same will take place, using no applied load, to document actuator response to such values. Then, once desirable response is achieved, a load may be applied, and the same procedure will take place.

Once a desirable acceleration response, potentiometer readings from the car were examined independently of acceleration response in the same manner, first using no load and only USB readings, then using no load and actuator response, and finally using a load. Once independent potentiometer readings are adjusted

and verified, prioritization phase will take place again, and potentiometer readings were integrated with acceleration calculations to give realistic response. Given that there is enough time, the same procedure was done using an accelerometer to detect the pitch and roll (relative to the force of gravity) exerted on the car.

7 - Project Prototype Testing

7.1: Hardware Test Environment

Testing the hardware was a very important phase of the project. There needed to be multiple configurations in which the hardware on the vehicle was tested in order to ensure the proper functionality of the hardware when the software was interfaced with it, as well as later when the cockpit was interfaced. The first and most basic set of information was the testing of each component by itself and collection of the data needed to preform later tests.

The potentiometers needed to be tested on both extremes of their operation. With a constant voltage applied they were then extended to full angular rotation in one direction and then to full angular rotation the other direction. The information concerning their whereabouts was logged everywhere in between in order to adequately understand the readings that were to be expected with various distances of angular rotation. After understanding the potentiometers the accelerometer and the gyroscope needed to be understood. The accelerometer needed to be tested similarly to the potentiometers, however since the accelerometer changed based on orientation rather than a difference in length, there will need to be a system devised in which to orient the accelerometer and take readings. Because the accelerometer readings will not matter beyond a grade of 20 degrees, a simple platform that could be rotated about the accelerometer worked just fine. The accelerometer was mounted to this platform, and rotated in order to receive various output readings. When the accelerometer was to be attached to the vehicle, it was necessary for it to be oriented in such a way that the force acting on it due to gravity was only acting on one axis, in our case this was the Z axis.

The transceiver was mounted to the top of the vehicle at a high point. This was done in order to reduce possible noise that could be introduced by the various electrical components that will be used to operate the vehicle. In addition it was to be wired up to the MCU in order to send and receive data. It was tested by hard wiring a set of data into the input line and verifying that the output of the transceiver is that which was wired into it. The DC servo motor that was used to control the vehicle when in motion needed to be securely mounted to the vehicle so that when activated the wheels could pivot in order to allow for radial motion when the vehicle was in motion. This motor will be tested by using a variable PWM GPIO and applying various pulse width signals across the servo. For the

servo to be functioning correctly, it should needed move when the various signals were applied. The DC drive motor needed to be mounted to the rear axle in order to allow the vehicle to be put into motion. This motor was also connected to the variable DC power supply for testing purposes.

After the vehicle is assembled in this manner it was necessary to test the components with the MCU. Concerning this phase in the hardware test environment, the readings from the accelerometer were taken. These readings were measured relative to the compression and depression of the side of the vehicle. These results closely coincided with those assessed previously concerning the accelerometer, and the approximate potentiometer readings should be able to be deduced based on this information. Just as before the servo motor and the DC drive motor were also tested with the MCU. At this point the circuitry concerning the voltage regulation of the drive motor and the servo motor was in place and the MCU was able to control both motors.

In addition to testing all of these pieces and parts the transmitter and the receiver for the video feed also needed to be tested. The analog video feed was connected into the transmitter and the receiver on the other was connected to a display unit. To ensure that this hardware was functioning as is to be expected, there was seen to be a video feed that coincided with the position of the analog camera on the display unit. In order to get this to work properly, the receiver needed to be calibrated by means of a knob located on the side. After some fine tuning it was easy to see the image that was being transmitted by the vehicle, and thus it was stated that this part was working properly.

After the rigorous testing of the hardware for the vehicle it was then necessary to run through similar tests concerning the hardware that will be used for the cockpit. The voltage regulators that were designed for this purpose were rigorously tested to ensure the proper functionality. In order to verify their functionality there was a PWM signal applied via a GPIO from the processing platform on the cockpit. The signal that was used had a varying duty cycle. This duty cycle needed to be changed and the output of the regulator needed to be validated with the hypothetical values that were calculated. After the validation of said regulators, it was necessary to test the actuators that were used on the cockpit. The actuators were tested with and without the voltage regulators, and thus it was easier to test the actuators after the voltage regulators have been tested. In order to test the actuators, in the first case they should be wired directly to a DC voltage supply. The speed and disposition of the actuators was noted and measured and coincided with the manufacturers specifications. A variable DC supply was used to change the speed of the actuators under various voltages. It was seen at this point that the speed of the actuator did not vary as much with respect to speed as previously stated. It is worth noting that the only thing linear about the actuators is the direction of their motion. The actuators did not have a linear change in

speed and the actuators also did not have any sort of linear displacement other than that which was based on time pulses. Likewise after this testing had been accomplished for the actuator in the case where the voltage regulator is not tied into the system, it was also need to be tested in the case of the regulator. Here the actuator was the load that was applied across the voltage regulator that was designed for this case. The voltage regulator was controlled with a PWM signal as it has been in the past and the behavior of the actuator was compared to that of the DC supply. At this point it was seen that there was no real difference between the results obtained from the supply and the results obtained from the regulator.

After the testing of the hardware components for this subsystem, the potentiometers that were used on the cockpit needed to be tested. The cockpit used standard 1k radial potentiometers. These potentiometers needed to be tested by changing the degree of rotation of the dial upon the potentiometer. The voltage readings needed to be taken in order to get an accurate idea of what range of rotation the potentiometers allowed for, as well as where the potentiometers were most sensitive to allow for the best possible location of the potentiometer. The potentiometer needed to be tested with a voltage applied across the at their most and least compressed positions of the pedal, which caused a radial change in motion of the potentiometer. They needed to be thoroughly tested for the ranges in between, as potentiometers are known for having certain parts that are more susceptible to the rotation than other parts. After the potentiometers have been tested and verified it was then necessary to mount them to their desired locations and test the range of values that will be used. For the first radial potentiometer it needed to be mounted at the base of the steering column to the rotating shaft that is connected to the steering wheel. Likewise the potentiometer that was attached to the pedal needed to be fastened in a similar way to the base of the pedal to realize the desired angular rotation.

Just as the potentiometer was previously tested, it needed to be tested again. This time the testing was not be done by turning the dial, but by actually rotating the steering wheel. The values that are received from the maximum clockwise and counter clockwise rotations were adjusted so that the center point of the turning lies at the center point of the resting steering wheel. Once this had been accomplished the gas pedal needed to be manipulated in the same fashion. The pedal needed to be mounted where the potentiometer on the pedal would rotate based on the pressing of the pedal. The readings were logged at the maximum and minimum depression of the pedal as well as many points in between. This process should take place on all necessary potentiometers.

After the pedals had been in place it was be necessary to mount the seat. At this point the actuators were mounted and the hardware design of the seat can be tested. The software should caused the seat to move about the center point and

the seat at any given point in time ideally should not be displaced along the Z axis. The software varied the movement speeds as well as positions and measurements were taken of the center pistons displacement. After verifying that the seat was functioning as is to be expected, the display unit was mounted to the bracket located above the steering column. The display mounted appropriately and was very secure. The display position was chosen based on an ergonomics relationship of where it should have been in relation to the rider, and the riders field of view.

After each of the hardware components had been tested individually it was important to test them as a system. This system for testing the hardware needed to be put into place after the software has been verified so that the idea of potentially breaking pieces of the hardware is not an issue. At this time the actuators were connected to the MPU and the movement of the actuators will be closely analyzed.

7.2: Hardware Specific Testing

In order to adequately test the hardware on the vehicle, there needed to be a system of testing procedures devised on each subsystem, as well as each individual component, due to subtle changes in component specifications. To get an accurate metric of all the outliers the potentiometers, accelerometer, battery, DC drive motor, DC servo motors, voltage regulators, camera and transceiver needed to all be thoroughly tested.

The voltage regulator that will be used to control the DC drive motor of the vehicle was a binary eight bit input for its values. This input was changed for each of the 256 possible values and the output voltage readings were seen and a small map of the output was used to control the actual changes in speed of the vehicle. For a number of intermediate values the output voltage was zero. The readings for each intermediate point were compared with the results obtained from the simulation and were surprisingly close to the values that were to be expected by the simulation. The next voltage regulator was the one that actually powers the MCU. It is necessary to note that a capacitor needed to be placed across the 12V and the 5V as well as across the 5V and the GND. This was done to alleviate noise in the system that was created by the crystal oscillator. The oscillator made enough noise to cause the MCU to power off and then back on obscurely. As can be seen from section 4.1, this circuit has an output voltage of 5V. The three voltage regulators that will be used for the transceiver, transmitter, and analog camera were all black boxed, and thus will just need to be connected to a power source and verified.

The measurement of the potentiometers was very important. Now that the power supply for the potentiometers had been accounted for, these components were tested. This test was somewhat rigorous, as a thorough understanding of the re-

strictions of this part was very important. For each of the four individual components, the voltage across the potentiometer at the output current node was measured. The maximum difference in position of the potentiometer needed to be divided by 20, as there will need to be many measurements between the max and minimum output reading. For each of these measurements the output voltage then need to be analyzed with respect to the position of the potentiometer. In addition these measurements were very important because the potentiometers difference in position was sending a measurement to the Cockpit, so each set of measurements will need to be specific to each individual potentiometer.

Next on the docket is the measurement that was read from the accelerometer. The accelerometer in question has three axes, and therefore each axis will need rigorous testing. The accelerometer needed to be temporarily mounted to a platform whose angle can be adjusted. The device needed to test through the extremes. In other words, the device was position in such a way that the -Z axis will receive all of the force of gravity. The device was then turned 30 degrees at a time about the X axis until it is once again resting at the -Z origin. The same process was repeated about the Y axis. After this test the device was placed with -Y as the origin and rotated about the Z axis for completeness. All of this information will then be used to formulate a table, so that the disposition of the vehicle can be determined by voltage output of the accelerometer.

The DC servo motor that was used to control the turning of the vehicle needed to be tested next. As per their nature, they move a certain amount depending on the PWM signal that is applied. A voltage regulator will need to be used in order to control this part, as well as a PWM signal. To get an accurate understanding of the part, the off center angle of the motors spindle will need to be measured 20 times. These measurements began at the minimum possible input PWM signal and end at the maximum possible PWM signal. There center point was the middle point of rotation, not in PWM signal, but in actual position, which in our case ended up at 70 degrees. Ideally they will be very close, however our case was off by approximately 20 degrees.

The next order of business was to test the MCU. It was very important that the MCU could properly interface will all of the parts in question. The first thing to do with the MCU was to become acquainted with all of the possible inputs and outputs. This was accomplished by using the board to supply outputs and measuring said outputs. After becoming familiar with the ways in which the board outputs information it was necessary to learn the idiosyncrasies with doing this. In addition this process needed to be repeated for inputs to the board, as well as repeating of this process when the MCU was located on the PCB rather than being located on the DEV board. It became readily apparent that the MCU located on the DEV board behaved quite differently. After becoming familiar with the board it was necessary to wire up the transceiver.

In order to adequately test the transceiver there needed to be a sending and receiving board, which was known as the XBEE explorer board. After properly wiring up the boards to host the transceivers a sequence of binary digits were sent from one board to another to ensure the proper functionality of the transceiver. In the case of the analog transmitter/receiver that was used to transmit the data from the camera to the an off-vehicle display, it was easier to test them synonymously and receive an output as soon as communication was properly established. The transmitter was wired up to the analog video camera, so that it would transmit its data over the air. The receiver was wired to a display unit. After tweaking the broadcasting frequencies a signal was acquired at the display unit, allowing for the transmission of an analog video signal over the air.

The DC drive motor was tested next. In order to test this, the voltage regulator discussed above will be used. The RPM of the motor was analyzed as a function of the voltage being supplied to the motor by the voltage regulator. This information needed to be taken after the vehicle is assembled because the stress put on the motor by the weight of the vehicle will drastically impact the analysis. After the vehicle is assembled and running it was necessary to take another assessment. This assessment was once again be based on the RPM of the motor in order to get an understanding of the upper and lower bounds of what this motors capabilities are.

Another sequence of test procedures was needed to be run on each component of the Cockpit. This included testing voltage regulators, actuators, potentiometers, the MCU and the transceivers. The voltage regulators that were used in this case were more heavy duty than those previously tested as they had been designed to sync 20 amps of surge current. Because of the amperage that needed to be synced through these devices it was very important to be careful when testing them. As an initial test, a simple resistive system was used in order to verify the regulators function as intended. A PWM signal of 100% duty cycle was used to begin with. Because of the design of the regulators, at this percentage of duty cycle the output of the system was zero volts. The duty cycle was then changed between 0% and 99% and the various output voltages were analyzed and the regulator was seen to be working correctly. After this test, a single actuator was connected to the regulator. The voltage was changed and based on the load applied by the actuators, a graph can be formed to show the linearity of the system. There needed to be multiple relationships taken into account that show the movement of the actuators over time relative to the voltage applied to them as well as the load applied. These relationships were used later on in the process of programming the actuators.

The potentiometers that were used for the steering wheel as well as the gas pedal was then tested. The potentiometer used for the steering wheel was a dial

in which the variable resistance depends on the angular change of the device. This was tested by starting at the lowest extreme. The potentiometer was followed by a resistor to ground and then set at its lowest position. A constant voltage was applied to the other end of the potentiometer and it was then rotated about its axis and the voltage on the node where the current leaves the potentiometer were measured. There were then to be twenty measurements taken at increments of 12 degrees along the turning of the potentiometer. This information was used later to determine the total amount the steering column was turned, based on the change in voltage across the potentiometer.

After the steering column had been assembled and the radial potentiometer is mounted to the base of the column, the above measurements were taken again. The angle at which the potentiometer needed to be adjusted so that the center part of the steering wheel lay on where it is to be expected. There were at least twenty measurements taken from the steering wheel. Ten of those measurements took place on the counter clockwise rotation, while the other ten took place on the clockwise rotation. The information needed to be stored and the voltage reading that represents the center point were taken into account as zero.

Just as the potentiometer for the steering column needed to be calibrated after it has been mounted, so did the potentiometer that are located on the gas pedal. After the pedal was mounted with the potentiometer in place, the pedal needed to be adjusted from the highest point in which there is no depression applied to them, to the lowest point where they are compressed as far as the user could push them.

In addition it was also necessary to verify that the spring loaded pedal returns to the position of least compression. This simple task was accomplished by reading the potentiometer then compressing the pedal. The pedal then was released and the potentiometer read the zero position again. The final output voltage matches the initial output voltage and thus it was seen that it was in proper function. Now the display unit was mounted to the brackets that have been designed to hold it. This unit was securely mounted with no possibility of falling. This needed to be further tested when the cockpit was completely assembled.

After getting familiar with the MCU located on the cockpit, measurements and conversions were necessary to account for inputs in a way that was usable. As the MCU had already been tested sufficiently in the software portion, the hardware portion was for analyzing the control of the seat. It was very important that the maximum angles in all possible situations were assessed. The most important components to test while the cockpit was in motion were the displacement of the actuators as well as the displacement of the center of the seat with respect to its expected center. With the current design it was very important that

the actuator displacement causes a negligible change in the displacement of the center position, in order to keep the center of mass equidistant from all of the actuators.

7.3: Software Test Environment

In order to adequately test the software, it was necessary to create a hardware test environment that would allow for managed testing of the software. Going through the programming process for the vehicle step by step with altered hardware would allow for this. In the beginning stages of the programming process it was necessary to collect various data from the analog input pins. To test this portion of the software it was essential to isolate these pins and to ensure they receive a structured input. In order to do this there was a variable voltage DC power supply that will be tied to the input pins. The following diagram illustrates how this environment was set up for the initial testing purposes.

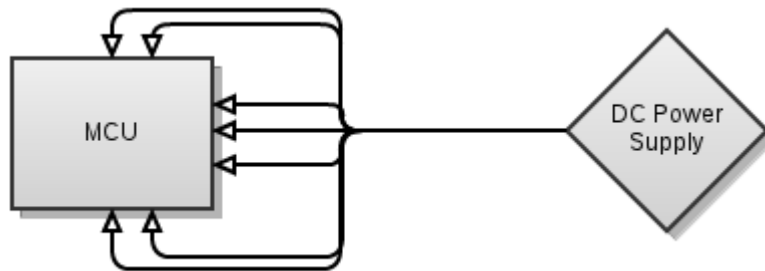


Figure 7.3.1: Input voltage software test environment

The same DC supply was used for all of the inputs. This environment allowed for rigorous testing of the sensory collection data. The analysis of the data collected needed to be tested after the collection of data. That being said the same structure that had been described above was used to test this section of code. Next on the chopping block was the collection of data from the transceiver. This testing environment was quite different from that which was written about above. It was necessary to interface the MCU and the transceiver with another MCU and transceiver in order to adequately test the collection of data. This testing environment consisted of two MCU's, two transceivers, and an over the air connection between them both. The following diagram illustrates the environment.



Figure 7.3.2: Data Transmission Test Environment

This specific test environment came in very handy because it doubles as an environment for the second MCU that was used to control the cockpit. This environment was used to not only test the software that deals with data coming in, but also to test the transmission of data. Another test environment that needed to be used in order to thoroughly understand how the MCU and the software is operating was the output of the device. This environment needed to consist of the MCU, two eight bit serial to parallel integrated circuits, two resistors, and sixteen LED's. Two of the digital outputs of the MCU were attached to the serial to parallel converters. The LED's were then linked up to the eight outputs of each of the converters. The LED's were then be wired into a resistor that was then grounded. This system allowed for adequate testing of the output by changing the value to be written at each of the two pins.

In addition to testing the programming for the MCU on the vehicle, it was also necessary to test the MCU on the cockpit. The MCU on the cockpit received input from a button that the user will press to initiate a program. Thereafter it checked potentiometer readings from the actuators, analyze the data received from the potentiometers, receive data from the vehicle, process this information, issue voltage commands to the actuators, make necessary data changes, and if the button was still active it will continue these operations indefinitely.

The environment that allowed for the testing of the button was relatively simple, and was the first testing environment that was created for the cockpit. One of the analog input pins had the button wired to it, and then to a voltage. This allowed the user to test the software interface with the hardware in the event that the button was pressed.

After it is ensured that the button was working properly, it was now necessary to move on to the reading of the potentiometer values from the actuators. This testing environment involved receiving readings from the potentiometers. These readings were the voltage values across the potentiometers. In order to test these adequately it was necessary to wire the MCU to a variable DC power supply. This testing environment was the same process outlined above concerning the MCU on the vehicle, only this time the Netduino will be used instead of the Atmega328p. As previously stated, the testing environment mimiced that of the MCU on the vehicle. Concerning the data that will be received from the MCU on vehicle, the transmission of data needed to be tested next. Once again the process for testing this data was as stated in the previous section and the transmission testing could be accomplished simultaneously.

When testing PWM signals that were used at the output to control the actuators the use of an oscilloscope was necessary. Unlike the analog to digital converter

that was used on the vehicle, this signal was used to determine the voltage output based on the duty cycle of the waveform.

It was necessary to take readings of the output and measure the duty cycle in order to conclude that the software was functioning as intended. After these measurements were taken it was then necessary to connect the MCU to the regulator that was used to drive the actuators. The voltage outputs were collected as a function of duty cycle and time and a representation of what the voltages were to drive the actuators can be assessed. Assuming that the voltage readings that were gathered coincide with what was to be expected, an actuator can then be connected to the system

7.4: Software Specific Testing

In order to adequately test the MCU on the vehicle and the associated software, the test environment was implemented. The first order of business was to incorporate the DC supply that was specified in section 7.3. The variable DC supply was connected and gave a voltage reading to the analog input of 3 volts. This 3 volt input was taken in by the software and the values in memory were changed to 3 volts (i.e. the respective value of the A to D). After it had been seen that the values that are being taken in by the software were what is to be expected, the 3 volts must vary by $\pm 1\%$ until the value that was stored in memory changed to the new value. As stated previously this would take place at both plus and minus 4%. At the point in which the values change, the stored memory value was approximately 2.9 for the minus 4% and the 3.1 for the case in which the voltage has gone up 4%. It was checked that all of the locations in memory appropriately change based on the percentage change in the voltage at the input pins. Assuming these cases function as intended (which they did), the voltages varied between 0 volts and 3.3 volts and the places in memory where the information was stored were seen to be changing value. The places where the voltages change were logged in order to further investigate the behavior of the software. Likewise, all of the inputs were disconnected and each individual input port was now tested one at a time. When the values from this test were that which was to be expected, it was said with relative confidence that the software is functioning as intended.

After this information was validated it was necessary to manipulate the output buffer to ensure that the information to be sent was what is to be expected. As the information was read, so shall it be sent. Concerning the buffer the information was stored as first the potentiometer in the front right (PFR) followed by the potentiometer in the front left (PFL) followed by the back two (PBR,PBL). The buffer then contained the accelerometers readings from the Y,X and Z axis. (in that order) After validating this buffer by checking it for each of the changes listed in the above paragraph, the software was said to be functional.

The next environment where testing was vitally important was the case in which the transceivers were used for input to the MCU and output to the MCU on the vehicle. In this testing environment it was necessary to fill the output buffer with information that was meaningful to the programming. While the information will, in fact, be “dummy” information it was in the format in which it is to be expected. This packet of information was then sent from the transceiver holding the information about the potentiometer sensors, as well as the information pertaining to the accelerometers current disposition. This information was sent via the transceiver to the MCU located at the cockpit. After the information was interpolated at the cockpit and determined to be correct, nine more sets of different dummy information were sent in order to ensure the correctness and stability of the connection. Likewise the MCU at the cockpit now needed to send information to the MCU on the vehicle. There was then ten different sets of information sent to the vehicle to once again ensure the reliability and stability of the system.

After the testing for the transceiver's had been completed, it was then necessary to link the testing of the transceivers with the testing of the collection of the input. The input would then be collected from the variable DC power supply and placed into the buffer, at which point it would be transmitted to the MCU that is to be located on the cockpit. The DC supply was individually connected to pins and their values changed in order to verify the validity of the output. In addition all the pins were connected to the supply in another test instance to see that all the information does in fact change. The time delay that is noticed from the collection to the reception of data was negligible.

The following test were now need to be performed in order to test the control signals that were used on the DC motors. As outlined in the test environment section, the output pins were wired to a serial to parallel converter. The parallel outputs were connected to LED's and resisted to ground. This allowed the user to put an output on a specific pin and verify its value by the illuminated LED's. A set of dummy information was passed to the output for the first cases. The values between 0 and 256 were placed on the output pin and the user verified that the pins are illuminated correctly. Upon validation of this fact the user then attempted to use the values that were sent from the MCU on the cockpit. The software then replaced the dummy input with the input that is taken from the cockpit. Once again a large variety of values was tested to ensure that the circuit was functioning as intended.

Following the testing of the software on the vehicle, it was pertinent to test the software on the cockpit. As stated in the testing environment section the first and simplest task was to attach a button to one of the input pins that was tied to a DC voltage. The program then started when the button was depressed and a voltage was applied to the pin. In order to test this, the user needed to press the button multiple times and ensure that when it is depressed the first time the program be-

gins, and when it was pressed again the program ended. After the user was confident in the functionality of the button there needed to be tests performed on the actuator input. The readings from the potentiometers on the actuators varied rather significantly, and thus there needed to be a rigorous testing over broad voltage ranges. As stated the potentiometers were all wired up to the analog input pins on the Netduino. In order to ensure the functionality the DC voltage was varied from 0-12V. The program then read the values and placed them in memory. Therefore this portion of the program was considered to be valid if the values were properly stored in memory. Following the testing of the input readings from the actuators, it was then necessary to test the transceiver. The steps for this testing needed to mirror those above concerning the MCU on the vehicle. After confidence was ensured in both the potentiometer readings coming in, and the data coming from the transceiver, it was evident that the actuator control programming needed to be tested. The circuit that had been designed to convert the PWM signal to an output voltage now needed to be used in this section on three different output ports.

Under normal circumstances the information from the potentiometers was actually the information that was sent to the cockpit. However because this was the testing phase it was necessary to send mock information to the cockpit for, as stated, testing purposes. The MCU on the vehicle was connected to a computer and the information for the potentiometers was manually input. This allowed for changes to the actuators based on user input, and thus allow for specific testing. Extremes, such as the front potentiometers completely depressed and the rear completely compressed were needed to be taken, as those conditions were vital to proper implementation. This information was sent through the software on the MCU of the cockpit and the output was as desired. The information sent from the vehicle to the cockpit in this phase contained all extreme cases, as well as may more normal cases.

After testing the potentiometers it was necessary to test the movement of the car. It was important for the software that controls the vehicles movement to work as anticipated. In this case the MCU on the cockpit was connected to the computer and mock information was sent to the vehicle. The information that was sent was based on the information that would be read from the radial potentiometer that was housed within the steering wheel on the cockpit. This mock information was sent to the vehicle and the DC servo motor that actuates the wheels on the front of the vehicle pivoted as anticipated. The measurements of the pivoting of these wheels was the same angular offset in both directions relative to the center point, and that was verified based on the programming.

Since it had been ensured that the vehicle could properly turn, it was now necessary to test the DC drive motor. At this point, the necessary information would be sent from the cockpit to the vehicle by way of a mock up program. The processor

on the vehicle would need to receive this information and interpret it as intended by converting the information

7.5: Stress Testing

After performing the basic structural and software testing, it was necessary, and also vital, to perform a set of stress tests on the system. These tests included stressing both the cockpit and the vehicle. Concerning the stress test for the vehicle, it was pertinent to control the vehicle as erratically as possible. Since the car had been tested in a controlled environment where the inputs were precise and expected, a more real world situation needed to be tested. The car needed to be put in a situation where it would start from a stop and accelerate as fast as possible. After this test the car was once again start from a stop and accelerate as fast as possible, only this time the vehicle turned. The vehicle then accelerated and decelerated as much as possible, as well as turn erratically in order to test a real world situation. All the while the movements were seen to be proper and they coincided with the information that was being transmitted concerning the disposition of the vehicle. There was however one hiccup with the control of the vehicle. It was seen that since the algorithm used to synchronize the vehicle with the cockpit was created by college students, in the case when the vehicle lost connection with the cockpit, the vehicle would drive erratically. Other than this issue, the vehicle worked fine for the most part. After it was said with confidence that the vehicle behaves as intended (for the most part) with unexpected user input it was necessary to then stress test the software and hardware that controls the cockpit.

The stress testing of the cockpit was a very rigorous time consuming process, as there would be people sitting in it and the safety of the occupant is of the highest priority. The cockpit first began with no load on the seat, and the input that was given to the cockpit was erratic. The information that was collected from the vehicle was then streamed to the cockpit and the movements of the cockpit were seen to somewhat mimic the movements of the car. Because of the erratic readings from the accelerometer, it was very hard to discern the turning of the vehicle properly. After this basic test had been performed and it has been noted that the cockpit was functioning correctly under tempered input, riders of various sizes were used to test the cockpit. Because of the diverse ranges in size of the team members and their friends, it was relatively easy to get a good understanding of how the actuators move the cockpit with occupants on it. It was seen that regardless of how large or small the individual on the cockpit was, there was not a clear distinction in differences in speed of the actuators. Because of the hefty nature of the actuators, the programming aspect concerning the riders weight became negligible.

Another factor to take into account was the stability of the display. The display was fastened into place using lock washers in order to ensure that it would not be

loosened during use. The bolts on the stand were fastened securely before the following test. The cockpit had no load on the seat and it was then taken through the tests previously outlined with the various sized riders. After the test it was ensured that the display was securely mounted. The information that was previously taken from the vehicles erratic behavior was now looped into the cockpit and the cockpit ran for over 30 minutes. After the 30 minutes were completed the display mounting was checked to ensure that it was still securely fastened. The next test should be a very long day where many users operate the cockpit for extended periods of time.

The actuator control for the cockpit now needed to be temporarily disabled in order to allow for the control of the vehicle by way of the cockpit. The potentiometer readings from the pedal and the steering wheel were transmitted to the vehicle and the vehicle was controlled by these readings. The speed of data transport was analyzed and assessed and the transmission were averaging 19.2 handshakes a second. This number gave the latency between transmissions as 52ms which was surprisingly better than expected. The next stage of testing was the stage that tested the real time simulation capabilities of the design as it pertains to the cockpit. The vehicle was then synchronized with the cockpit and the information that is sent from the vehicle controlled the cockpit. The information that was used to control the vehicle was then sent via a connection with a computer rather than directly from the cockpit in this testing phase. The motion of the cockpit coincided nicely with that which was to be expected.

Since the last test succeeded, it was now necessary to test the end-state in which an occupant of the cockpit actually controls the vehicle and experiences the feedback provided by the actuators. In this state the user sat within the cockpit and controlled the vehicle. The feedback received from the vehicle was experienced in a very timely manner. In this testing phase there needed to be a wide range in weight of the riders in order to test the cockpit under various load conditions. After the variety of riders had completed their thorough testing of the cockpit and the information pertaining to each rider has been analyzed, The information that was received was once again surprising and the actuators more than did what they were expected to. At this point in the testing it was seen that RC Ghost rider was a success!

8 – Administrative Content

8.1 Milestone Discussion

To ensure this project was completed in a timely manner and to keep all members of the group informed of important upcoming goal dates a “Milestones” chart was much needed. The first grouping of dates correspond with ordering, testing, and building the vehicle portion of the project. The first thing that was accomplished was to finish the project write up by December 2nd, this allowed for sufficient proof reading and binding time. The next was to submit the final edited copy by 12:00 pm on December 5th. The next was ordering all parts for the RC car and acquiring the RC car. The transceivers and cockpits MCU were also ordered at this time. On January 30th the testing of all parts began with necessary information being logged to ensure components worked as expected. February 5th was when the project started to take form, that is when the RC car began being assembled. The final date in the staging process was February 15th when the actuators and other cockpit components were ordered. Below is the first grouping of dates important to making sure that the hardware was ordered and tested sufficiently before being integrated into the Cockpit and RC car.

- 12-2-11 – Complete the documentation

- 12-5-11 – Submit the final copy of the documentation

- 1-1-12 – Order all necessary parts for the car, as well as the transceivers and the cockpit's MCU
- 1-30-12 – Test all of the parts and log the info

- 2-5-12 – Begin assembling and testing our design

- 2-12-12 – Order the actuators and the various cockpit components

The next grouping of dates was geared towards fabrication of the cockpit and cockpit circuitry. On February 18th the car's circuitry was finalized sent to be fabricated on a PCB. A week after that, the cockpit actuators were tested in a controlled environment to ensure that they would preform as expected under various loads. On the 28th of February the design circuitry for the cockpit was tested under different loads making sure that they yielded the expected/simulated outputs. Given that all circuitry performs as expected, on March 5th after extensive testing, the design circuitry for the cockpit was also finalized. On March 25th all testing on the singular components of the cockpit was completed, compiled, and analyzed so that an end-state circuit board for the cockpit could be fabricated.

- 2-18-12 – Confirm the layout of the car's circuitry, and get the PCB fabrication started.

- 2-25-12 – Test all of the cockpit components and log the necessary information
- 2-28-12 – Test the design circuitry for the cockpit
- 3-15-12 – Finalize the design circuitry for the cockpit
- 3-25-12 – Create an end-state circuit board for the cockpit

The final grouping of milestones was primarily to ensure that the cockpit and RC car properly interfaced and worked well in conjunction with each other. On April 1st interfacing the cockpit and RC car commenced. Given that all components acted as designed and interface with one another properly on April 6th, after two days of testing, there was only be fine tuning needed to ensure the cockpit always acted as expected. During this portion, run time behavior was be logged and anomalous behavior was be identified. During the week of April 8th said behavior was attempted to be eliminated.

- 4-1-12 – Properly interface the cockpit with the vehicle
- 4-2-12 – Fine tune the cockpit to operate as intended
- 4-3-12 – Log run time statistics and search for anomalous behavior
- 4-6-12 – Track down and eliminate said behavior
- 4-8-12 – Add finishing aesthetic touches

8.2 Budget and Finance Discussion

The budget was split into two portions, the RC Car and the Cockpit. The budget for this project turned out to be somewhat of a hassle to stay within as is the case with virtually any project. It was crucial to stay within the milestones time line to make sure this project got done in a timely manner. It was important to keep to the budget plan as close as possible so that there was no negligent spending which could inhibit buying all of the parts needed to complete the project. Another concern was destroying or damaging parts such that replacement components must be bought. This could have again impeded the acquisition of parts needed to complete the project.

The portion of the budget concerning the RC car was considerably easier to stay within compared to the cockpit. The RC car will not require a large amount of outsourced work to be done on it, while most of the cockpits construction was done by a professional. The most expensive portion of the RC car was the PCB board fabrication which has an allotment of \$100.00. The budget was created with the belief that the RC car would be built, but given recent developments will now be a store bought manufactured model. That meant that the money intended for: 1 DC motor, 2 batteries, 4 wheels, 2 turning servos, and 1 chassis is equal which were all going to be implemented to build a one of a kind RC car will now be used to purchased prefabricated model. The total cost of the above listed parts is equal to \$241.00 and it is expected that the RC chosen to buy will not be anywhere close to that value. This means that there is now more money

available in the budget that can be used to buy higher quality components for the project or to buy back up ones with the notion that some will be damaged beyond repair and need replacement.

This project also required one accelerometer expected to cost \$5.00. The accelerometer was be mounted to the RC car and send information to the cockpit that is necessary for the actuator movement. Some of the unused money in the budget went to buying the best accelerometer available. This may exceed the \$5.00 stipend, but thanks to possible over estimations that was a viable situation.

The information from the potentiometers and accelerometer was sent to the cockpit by a transceiver. The transceiver in the budget was of mediocre quality and cost \$45.00. With the loss of a group member means this portion will now be black boxed so a set of previously made transceivers will be used and came with a price tag of \$90.00. The total amount for transceivers in both the cockpit and RC car portions of the budget was \$90.00 (\$45.00 for each). This did not create skewing of the budget however.

Another component of the RC car parts list is the camera. This project required an analog camera of moderately high specifications. The camera haf a high resolution since the television monitor in the cockpit will be displaying the video captured by the camera and if there is low quality or grainy images it will dilute the simulation experience. The also had a high a high transmission rate. If the camera doesn't then the driver will experience a lag in the video feed which would make the RC car harder to operate due to the video to not matching the actuator movement to be out of synchronization, which is the main focus of the project. The camera stipend is \$20.The RC car budget is as shown below in figure 8.1.

RC Car

<u>Parts</u>	<u>Quantity</u>	<u>Price</u>	<u>Total</u>
Accelerometer	1	\$5.00	\$5.00
Potentiometers	4	\$4.00	\$16.00
Turning Servo	1	\$13.00	\$13.00
DC Motor	1	\$40.00	\$40.00
Battery	2	\$30.00	\$60.00
Voltage Regulators	2	\$2.00	\$4.00
RC Car	1	Free	Free

MCU	3	\$6.00	\$18.00
Transceiver	1	\$45.00	\$45.00
PCB Fabrication	1	\$100.00	\$100.00
Misc. Parts	1	\$50.00	\$50.00
Camera/Transmitter	1	\$80.00	\$80.00
		<u>Subtotal</u>	<u>\$427.00</u>

Figure 8.1: RC Car Budget

There is a section of the budget titled miscellaneous parts and \$100.00 is set aside for it. This was put in when the RC car was going to be built not store bought. Therefore from over estimations there is \$125.00 or so that was used for the project in any manner desired. The RC car portion of the budget is shown below to easier see that parts needed, quantity, price, and subtotal cost.

The cockpit portion of the project was where most of the worries lye with the budget. This was due to components necessary to complete the project design. Given the expensive nature of all the components and the limited budget the project must be completed on, there was no room to replace and part on the cockpit price list. In this section there is also some discrepancies with the parts needed in the list and the actual parts needed. This is again due to the budget being based on a previous cockpit design.

The budget calls for 3 heim joints. It was previously thought that there would be five actuators that would move the cockpit, but the current design called for only three. That means there only needed to be three of the same specifications, two per actuator and one over sized ball joint that s located under the cockpit seat. The three joints that are implemented in the current design are now heim joints as well. A quality heim joint made of 5000 Series Galavnized Steel cost anywhere between \$50.00. The means that the three heim joints will cost a total of between \$100.00.

In the budget it states there is \$1,000.00 that will be used to buy four linear actuators each costing \$225.00 each. The current design only uses three actuators, so they are of higher power and this drove the cost up. The Servo City actuators that will now be used have a hefty price tag of \$399.99, this means that the total cost of the actuators is \$1,200.00. This is the only known under estimation in the budget and means somehow \$200.00 had to be freed up to afford the new stronger actuators.

In this the biggest problem with the budget is that a slight oversight was made and there is no allotment for the material that was needed to build the cockpit

frame or any to pay the entity that cuts/welds/fabricates the cockpit. It is estimated to cost upwards of \$1,200.00 and it would be impossible to build this project as described given the current budget. Amendments were made so a new updated budget was submitted to Workforce Central Florida in the hopes that they will give more funding than previously thought necessary to help complete the project. But to no avail, we didn't get a new budget. Below is the cockpit subtotal of the budget and the total price of the project. The cockpit budget and total cost is shown below.

<u>Cockpit</u>				
	<u>Parts</u>	<u>Quantity</u>	<u>Price</u>	<u>Total</u>
Seat		1	\$45.00	\$45.00
Pedals		2	\$25.00	\$50.00
Display		1	\$100.00	\$100.00
Potentiometers		6	\$8.00	\$48.00
Steering Wheel		1	\$30.00	\$30.00
Heim Joints		2	\$25.00	\$50.00
Hardware		1	\$10.00	\$10.00
Linear Actuators		3	\$400.00	\$1,200.00
Transceiver		1	\$45.00	\$45.00
Materials/Fabrication		1	\$1,100.00	\$1,100.00
MCU		1	\$80.00	\$80.00
Misc. Parts		1	\$100.00	\$100.00
Misc. Parts		1	\$100.00	\$100.00
			<u>Subtotal</u>	<u>\$2,858.00</u>
			<u>Total</u>	<u>\$3,2850.00</u>

Figure 8.2: Cockpit and Total Cost

8.3 Safety Discussion

A.) User Safety

In the interest of protecting irreplaceable assets, user safety is of primary concern in design of the RC Ghost Rider. Certain protocol was strictly adhered to in all testing and leisure use of the apparatus. The rotational velocity of the cockpit is adjusted by the MCU to ensure that no excess force will be applied on the operator, so that injury to the user by impaction due to the apparatus, or so that the user is not propelled out of the cockpit through thorough testing. The suspension

from the ground on the cockpit will not be excessive, thus if the operator is to exit the cockpit, possible injury is minimal. Electrical components were placed distant from the apparatus to ensure no accidental electrical discharge is to affect the operator, as it is a high current application. All wire is to be properly shielded to protect the user from electrical discharge. The electrical housing for the cockpit will be locked, so that the user cannot access it inadvertently. A three-point harness is implemented in order to adequately restrain the user under stressful operating conditions. Persons under the age of 18 operating will provided with a helmet to wear.

It is the responsibility of the user to not wear loose fitting clothes or jewelry. Pregnant persons, people with high blood pressure, back problems, or are otherwise not medically clear to undergo heightened stressful situations are advised to not operate, and to do so take on all risks and liabilities associated with injury. Users under the influence of drugs or alcohol are strictly prohibited from use. Users are responsible for fastening the seatbelt, and persons under the age of 18 must wear a helmet in operation. People weighing more than are strictly prohibited 400 lbs, or less than 75 lbs risk injury. A person cannot comfortably fit in the seat, or the three point harness should not ride the vehicle, as they will risk injury. Persons over the height limit should not operate the vehicle, and persons under the height requirement probably can't operate the vehicle anyway. Observers must stand a safe distance of 10 feet from the cockpit apparatus.

B.) Construction Safety

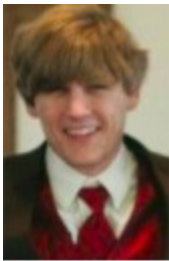
When working with high current applications, fuses were used. Fire safety utilities will be on hand during all testing phases. Complete care and standard protocol in handling electrical components will be executed. The touching of high current components and wire will be strictly prohibited when voltage is applied. No operator sat upon the actuators until extensive documented testing takes place, as discussed in the stress testing section (7.5).

9 Project Operation

Be sure to read 8.3 before operating RC Ghost Rider. Be sure that at least 2 people other than the user are present, one to ensure proper operation of cockpit circuitry, and the other to ensure safety of the RC car. Ensure that assembly is completed properly by a professional, trained for special use by the RC Ghost Rider team. It is encouraged that one of the people present while use is such professional. Make sure the area of operation is open and safe for use, free of obstacles which are living things.

To use, first turn on the RC car and its camera. Then, turn on the display found on the cockpit, and its receiver. If the receiver is not calibrated, calibrate it using the knob found on the receiver. Once the previous steps have taken place, turn on the Cockpit using the switch found on the PCB. The user must not touch any buttons, switches, the steering wheel, the accelerator pedal, or any sensors until calibration is finished(noted by the ceasing of motion of the actuators). Check to see if the RC car is synchronized with the cockpit by corresponding turns of the steering wheel with motions of the front wheels of the RC car. If synchronization has taken place, RC Ghost rider is ready to enjoy! Do not allow RC car to exceed the range of 150 ft, or synchronization will be lost.

10 Team Information



James Russick is a senior at the University of Central Florida. He plans to graduate in May 2012 with a Bachelor's of Science in Electrical Engineering. After graduation he plans to continue to expand his business, Epixam LLC.



Paul Shimei II is a senior at the University of Central Florida. He plans to graduate in May 2012 with a Bachelor's of Science in Electrical Engineering. After graduation he plans to work for Bionetics Corporation in Cape Canaveral.



Adolph Arieux is a senior at the University of Central Florida. He plans to graduate in May 2012 with a Bachelor's of Science in Electrical Engineering. After graduation he plans to work for United Space Alliance in Cape Canaveral.

10.1 Acknowledgements

The authors wish to acknowledge the assistance and support of Richard Barrett of ITT Exelis, Work Force Central Florida, Michael Russick, Christina Ecker, Gretchen Russick, Nicole Briggs, Michael Siravo, The Arieux Family, Bill Bobagins, Trevor Murdoch, Bill Wells, and Paul N. Shimei for the continued support and assistance in the process of establishing a working prototype. Jeff Silvestri for performing the Cockpit Fabrication and Materials needed to create the cockpit base and platform and also working in a timely manner with little time's notice.

11 Final Product

Enclosed in this last section are the final images of the project. These images include but are not limited to the components that were vital to the operation of the final RC Ghost Rider.

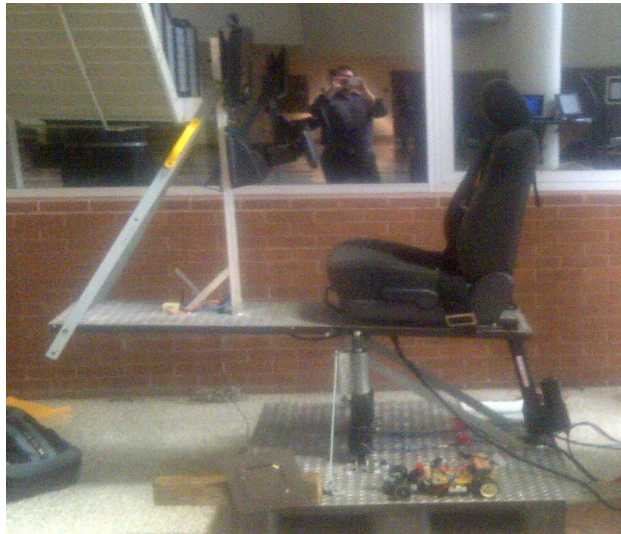


Figure 11.1: The Cockpit side view

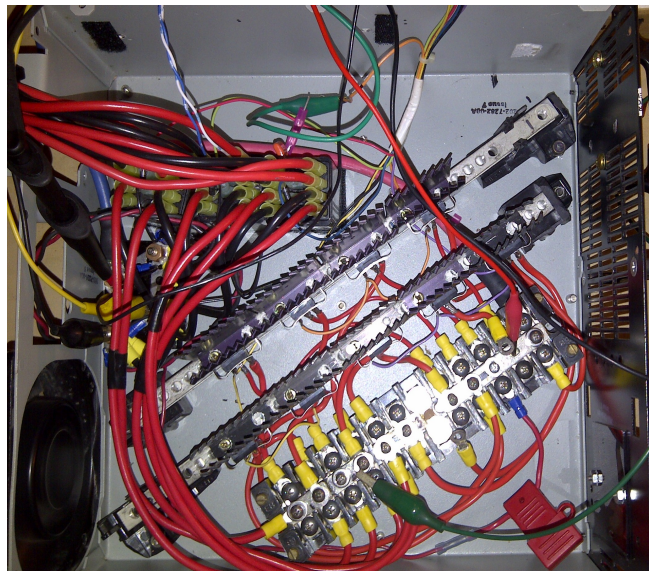


Figure 11.2: The Power Box

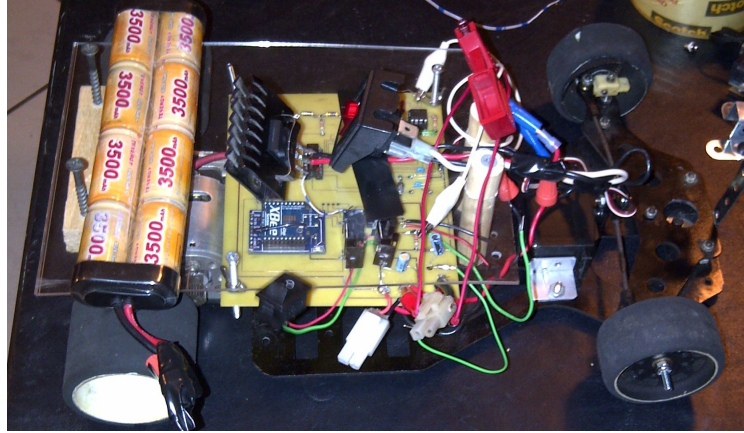


Figure 11.3: The Car

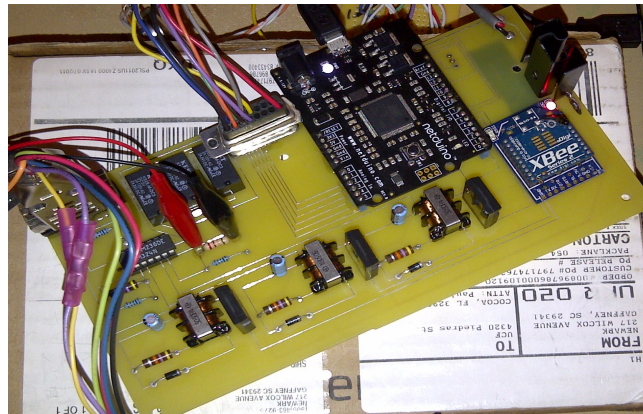


Figure 11.4: The Cockpit PCB

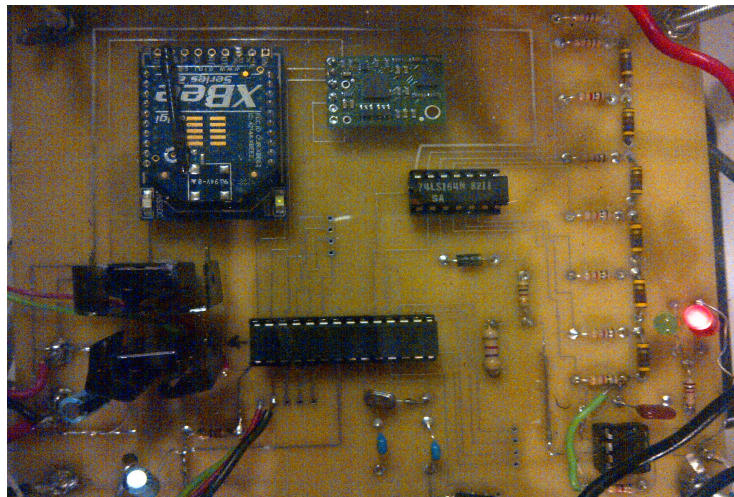


Figure 11.5: The Car PCB