Miami Redblade: A GPS-aided Autonomous Lawnmower

Technical Report

ION Autonomous Lawnmower Competition
Datyon, Ohio
June 5th – June 8th, 2008

Miami University

School of Engineering and Applied Science

Student Team Members
Daniel Anderson, Electrical and Computer Engineering Department
Kyle Green, Electrical and Computer Engineering Department
Xiaolei Mao, Electrical and Computer Engineering Department
James Reynolds, Mechanical and Manufacturing Engineering Department

Advisors
Dr. Jade Morton, Electrical and Computer Engineering Department
Dr. James McCollum, Electrical and Computer Engineering Department

Executive Summary

The following report details the technical aspects of the Redblade, Miami University's fourth generation autonomous lawnmower. The Redblade was created for entrance in the Institute of Navigation's 5th Annual Autonomous Lawnmower Competition by a team of students majoring in electrical, computer, and mechanical engineering at Miami University. The report details (1) team organization, (2) lawnmower design, (3) electronic design, (4) software strategy, (5) system integration, (6) the safety system, and (7) the overall system specifications. This report discusses each aforementioned system in detail, along with providing analysis and conclusions.

Table of Contents

1 Team Organization

Miami University has participated in the ION Autonomous Lawnmower Competition for four years. This year, two of the students who competed last year have returned for this year's competition. The Redblade team consists of four students in total. There are two subgroups with one having a focus on the electrical/computer parts of the project and the other subgroup having a focus on the mechanical portion of the project. Within the electrical/computer subgroup, different students focus on different portions of the project. For example, the students with more of an electrical engineering background focus on the non-programming parts and the students with more of a computer science or computer engineering background focused more on the embedded systems or the control algorithm.

2 Lawnmower Design

The mechanical chassis has been completely redesigned and fabricated from the ground up. The wheels have been placed in the middle of the chassis which enable the mower to complete precise turns about the center axis where the mower blade is located. Also, the profile has been slimmed down significantly from past competitions. A CAD drawing of the design is shown below in *figure 2.1*

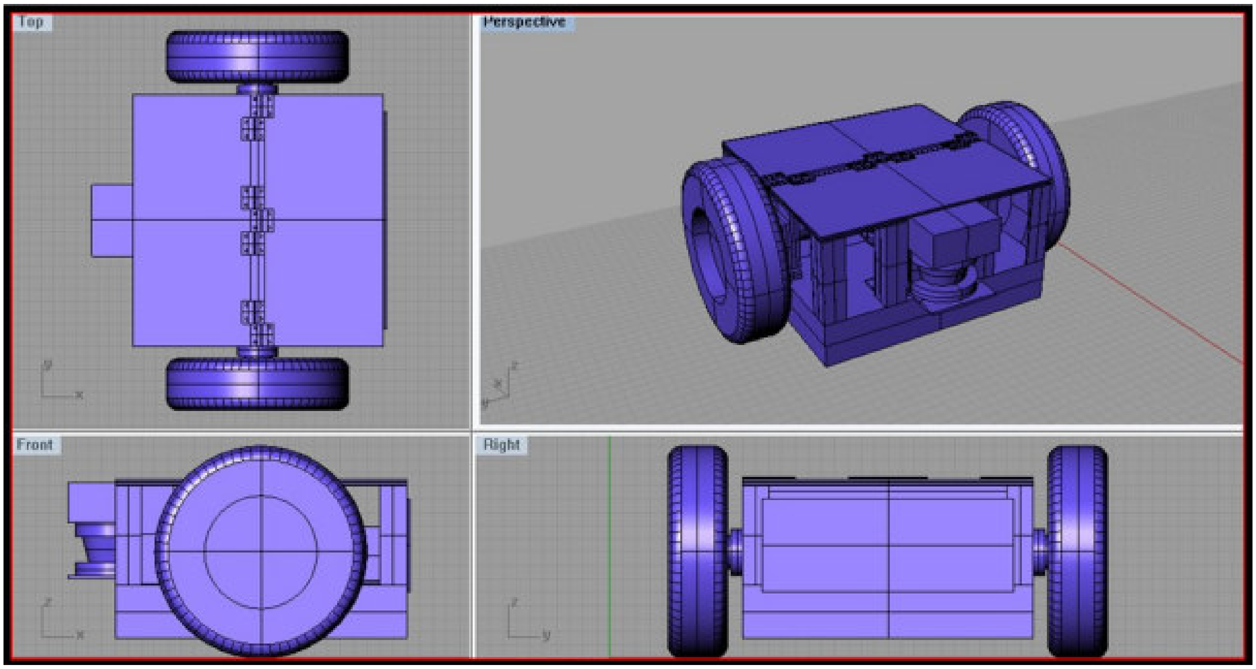

Figure 2.1 Square Chassis Design

In *figure 2.1* the supporting caster wheels are not shown but they are mounted to the front and back of the mower to provide forward and backward stability. The supporting casters keep in constant contact with the ground due to springs holding them down. If there were no springs, the mower would be apt to rock back and forth due to the unique design of having the main wheels in the middle of the mower.

The final design is shown below in *figure 2.2* and *figure 2.3*

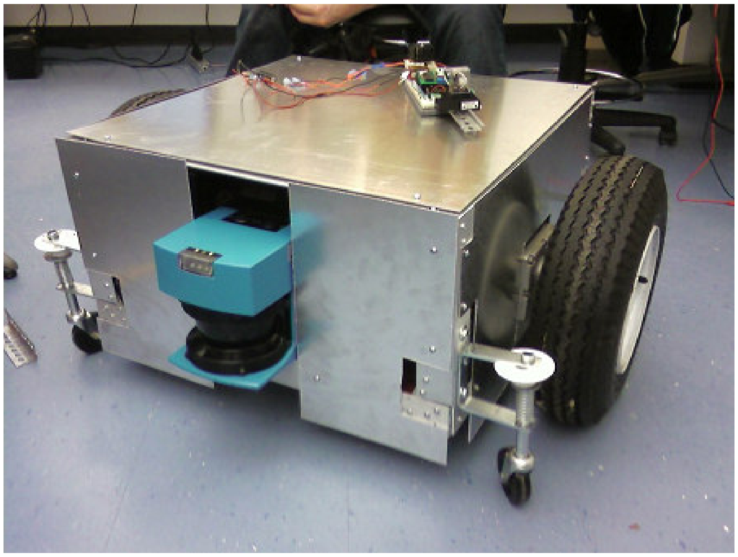

Figure 2.2 Front View of Chassis

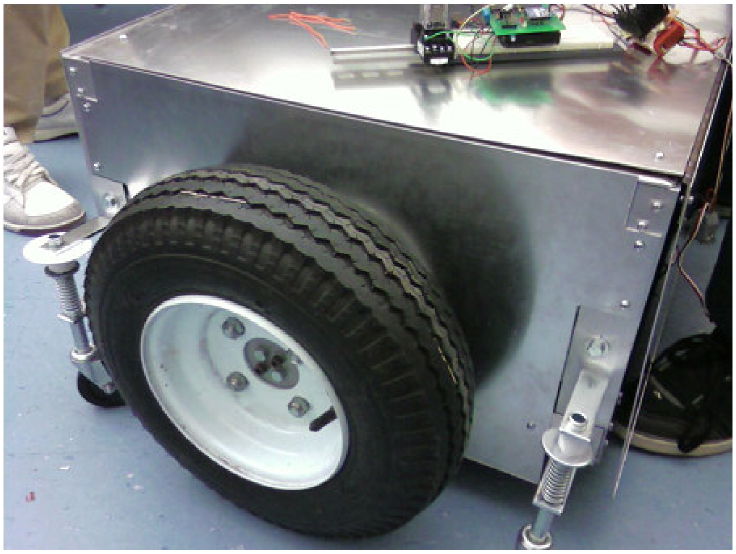

Figure 2.3 Side View of Chassis

Also, one last important consideration for the final design was the overall weight of the chassis. Last year, testing with the mower was cumbersome because transporting around a heavy and bulky chassis was difficult. That is why this year all of the metal used is Aluminum.

3 Electronic Design

3.1 Sensing Components

The sensing system is comprised of four main components: a Topcon Real Time Kinematic (RTK) GPS system, an Inertial Measurement Unit (IMU), a SICK LIDAR, and wheel encoders. The RTK GPS provides position information with the 1 sigma positioning accuracy less than 1 cm over large baselines at up to 10Hz. The IMU provides heading information as well as aiding in positioning information. The LIDAR is used purely for obstacle avoidance. The wheel encoders are used in conjunction with a Proportional Integral Derivative (PID) controller to ensure the wheels are turning at the rate we want them to.

In last year's competition, our team used a custom Differential GPS (DGPS) system which was considerably less expensive. However, last year the refresh rate of the DGPS system was only 1Hz and with our new dynamic path planning algorithm, we really needed a faster GPS system. The new RTK GPS system is capable of faster refresh rates (up to 10Hz), and it is more accurate than our previous system which had a 1 sigma accuracy of less than 3cm.

This year, for heading, we are using an IMU as opposed to the electronic compass that we used last year. We did this because while the electronic compass worked very well last year, it required calibration almost every time we used it. Also, it is not as accurate as an IMU for heading information.

For obstacle avoidance, we are using a SICK LMS 200 Laser Finder which provides 180 degree visibility about a vertical axis and a 30m range. However, one problem with the LIDAR unit is that sometimes if it is sunny out, the unit can be report back false "hits". We were able to successfully solve this problem in the way we store where the mower thinks obstacles are. In order for the mower to think there is an obstacle in front of it, there has to be more than one "hit" at that spot.

The wheel encoders on the *Miami Redblade* use US Digital E7MS quadrature optical encoders (see *Figure 3.1.1*) which are installed inside of the motors. Each encoder has two different signal channels which have phases that are 90 degrees apart. Each time the optical sensor detects a change, a pulse is sent on one of the signal channels, and then a second pulse is sent 90 degrees offset from the first pulse. With this two channel configuration, detecting whether the wheel is moving forward or backwards becomes possible. Because the number of pulses there are per revolution of the wheel is known, dead-reckoning can be used to figure out how far the mower has gone as well as using the encoders with a PID control loop. In order to make both of the wheels turn at the exact same speed, an encoder module from RobotEQ was purchased which installs directly into the existing RobotEQ DC controller. The encoder module decodes the pulse train coming from the quadrature optical encoders and it will increment or decrement a counter register in the RobotEQ DC controller depending on if the wheel is going forward or in reverse. *Figure 3.1.2* displays the quadrature encoder output waveform and encoder setup.

Figure 3.1.1 US Digital E7MS Encoders

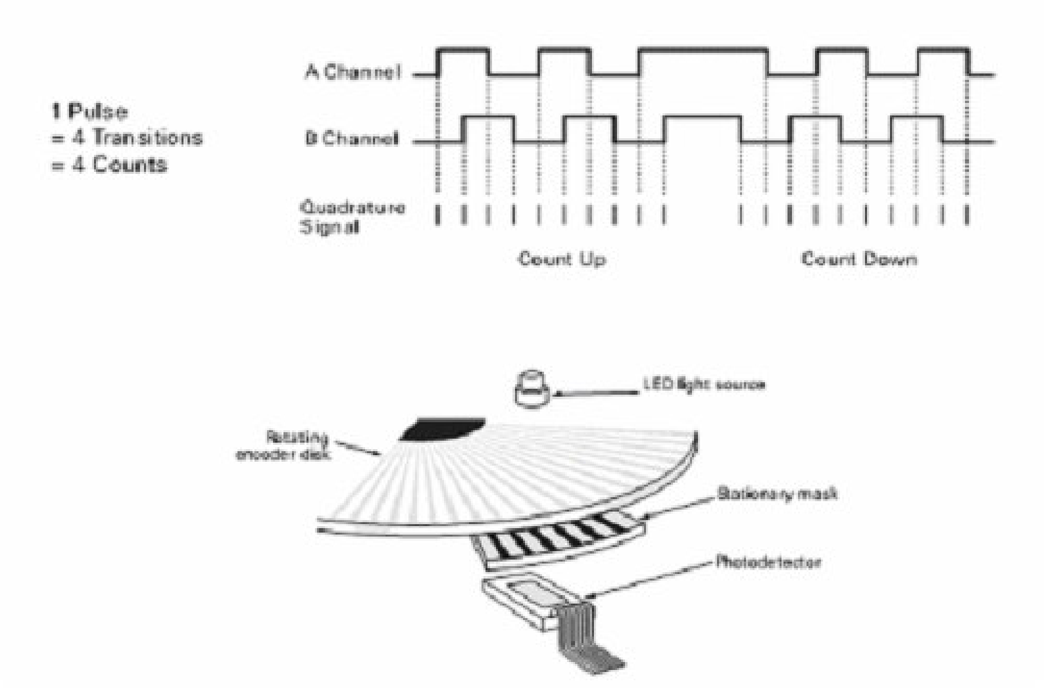

Figure 3.1.2 Quadrature encoder output waveform and quadrature optical encoder setup
(from RobotEQ AX2550 User Manual)

This system cannot provide position estimates with the same accuracy and precision as the DGPS. On the other hand, this system does provide positioning through dead-reckoning that can serve as redundant measurements to ensure the integrity of the RTK GPS. Also, the wheel encoders are not limited by the 10 Hz update rate of the RTK GPS, providing the control algorithms with the necessary position data at times when valid RTK GPS data is not available. The RTK GPS is used to correct the errors that would accumulate if only wheel encoders were used to determine position.

3.2 Drive and Power System

The drive system consists of two Power Chair (NPC) model T64 24-Volt DC motors. The DC motors are voltage-controlled with a low RPM-torque of approximately 300 in-lbs. Furthermore, the motors are equipped with a 20:1 gear ratio to give suitable RPM ranges for operation.

The power system is comprised entirely of rechargeable sealed lead acid batteries. The motor which powers the cutting blade is also 24V. There are separate batteries for the blade motor, the NPC motors, and the entire electrical system. It was important to keep the electrical system, which includes all of the sensors, as isolated as possible from the NPC motors because they can have regenerative feedback which could damage some of the electrical components. Also, within the power system, we had to be able to shutdown the RobotEQ and the blade motor whenever the emergency stop was hit (wired or wirelessly). The wiring diagram for the connections to the RobotEQ are shown below in *figure 3.2.1*

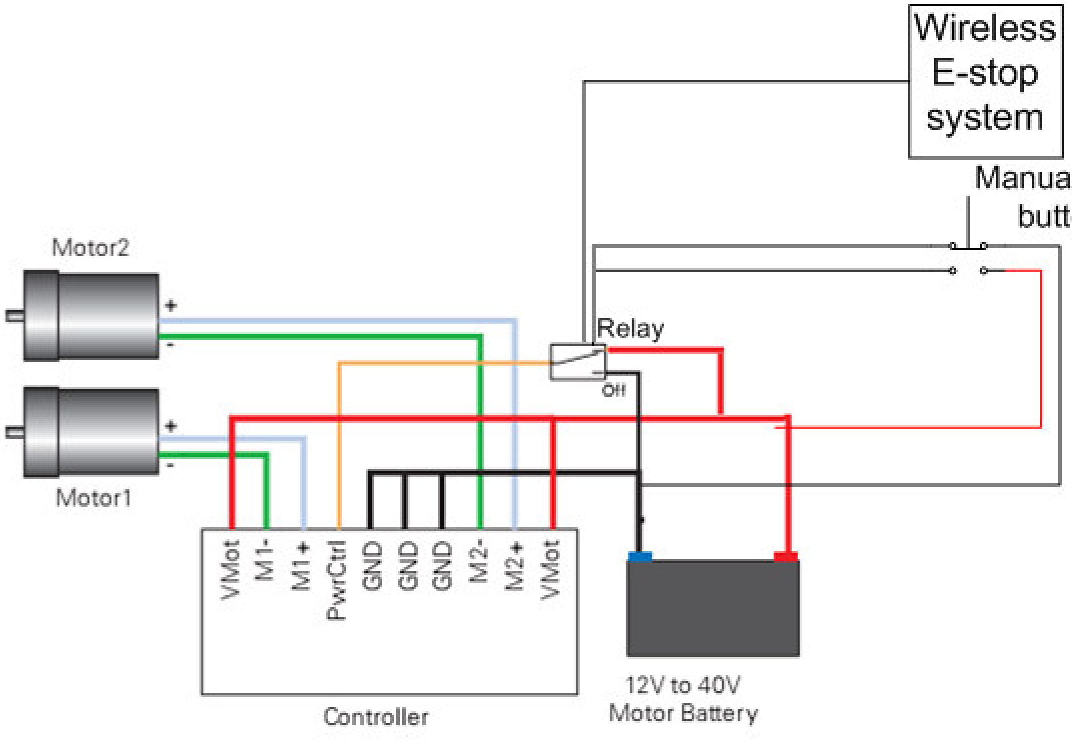

Figure 3.2.1 Wiring Diagram for RobotEQ

4 Software Strategy

4.1 Dynamic Path Planning

Last year, an entirely static path planning approach was taken meaning that all of the waypoints the mower was set to go to were pre-determined. An example of a basic static path planning approach is shown below in *figure 4.1.1* which shows how waypoints would have been pre-computed given the static location of two different sized obstacles. While an advantage of this approach is knowing exactly where the mower will go and in what order, that advantage really does not lend itself to a dynamic environment with moving obstacles.

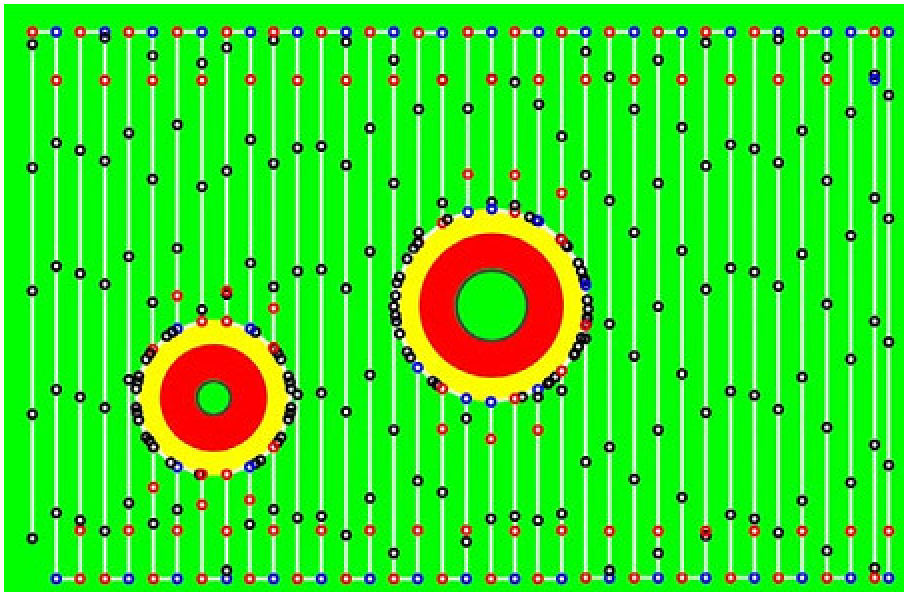

Figure 4.1.1 Static Path Planning Approach

With a dynamic environment in mind, using a fully dynamic control system and obstacle avoidance was the clear choice. However, because this particular approach is one of the most difficult to implement, a simulator environment had to be developed to test our dynamic path planning algorithm.

4.2 Simulator

The simulator we developed allows us to rapidly test changes to our control algorithm. The simulator was coded in Java with OpenGL using JoGL extensions and it has a relatively small footprint with it being just a couple thousand lines of code. The simulator allows us to change sensor refresh rates, noise levels on the different sensors, and how fast we can react to changes in our environment. It also allows us to monitor both the robots perceived state and the actual "truth" state. A screenshot of the simulator running one of our earlier control algorithms can be seen in *figure 4.2.1*.

Figure 4.2.1 Simulator Running Early Control Algorithm

In *figure 4.2.1* the various different geometric shapes in the simulator each represent something. The circle represents a moving obstacle which can be controlled where it goes by the user, and the square filled in with a gradient represents the mower. The shape of the field is laid out like the field will be in the competition. Also, when the LIDAR from the mower hits an obstacle, you can see the red dots on the obstacle where the mower "sees" an obstacle. The green represents un-mowed grass and the brown is mowed grass. The dark red area is considered out of bounds. The two different x and y coordinates represent where the mower thinks it is at, and where the mower "actually" is at. Lastly, the yellow circle in front of the mower represents where the algorithm has selected as the next location to go to.

4.3 Dynamic Path Planning

In order to select the next "waypoint" to traverse to, we have developed a heuristic function which returns a value on how "desirable" a possible future location is. To do this, we look at areas a little larger than the size of the mower in front of us. The dynamic path planning algorithm uses the heuristics function to rate almost 100 different areas around it. It will then adjust the velocity of the robot to go towards the new area. An example of how the heuristics function works is shown below in *figure 4.3.1*

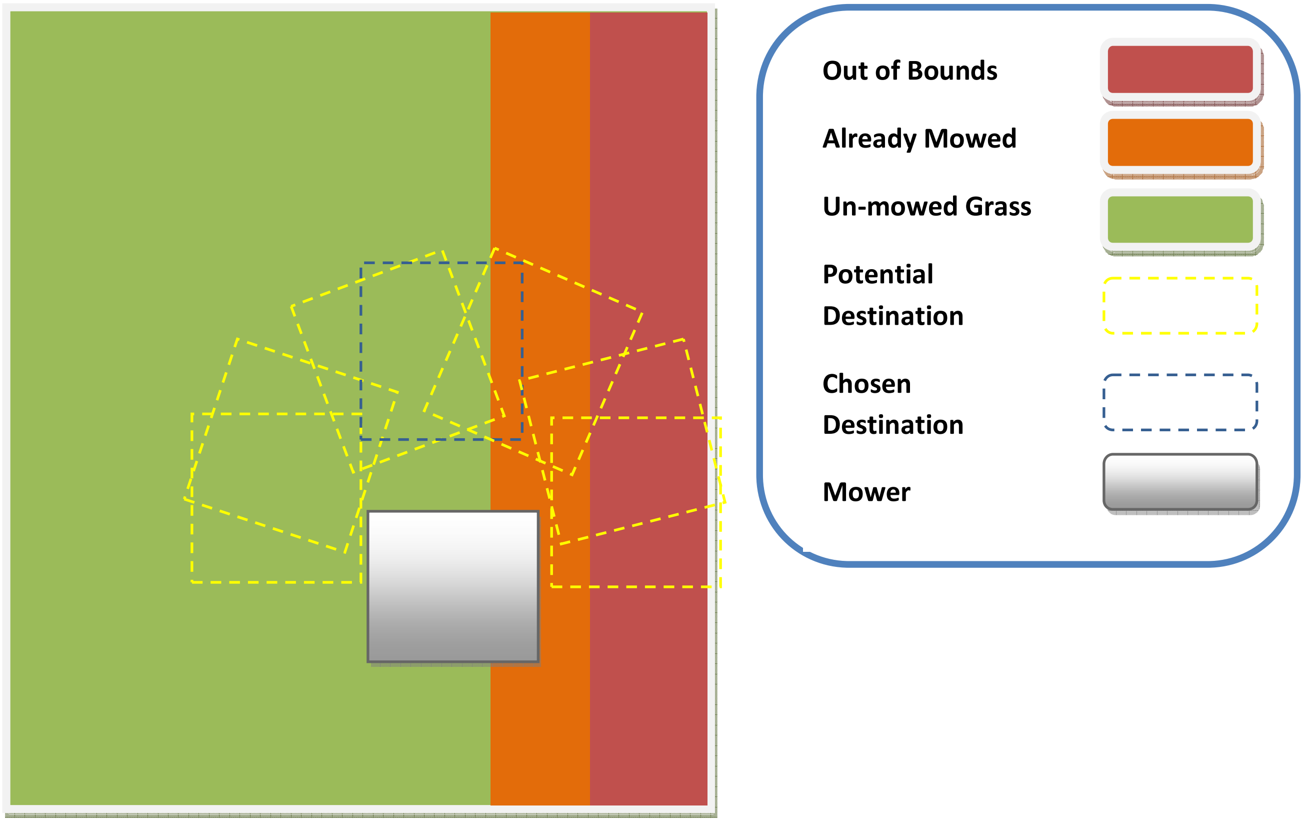

Figure 4.3.1 Heuristics Function

In the above example, all of those dotted box areas would be passed into the heuristic function and be rated on the most desirable area. The dotted blue box would end up being the most desirable place to go because we would want the majority of the area to be uncut grass, but we would still want a small portion of it to be cut grass so that there is some overlap in the cutting. Sometimes; however, the mower can get into a place where there no longer are any “good” places to go to next. If this happens, then the mower is in one of two situations. The first situation would be that the entire lawn is done being cut, and there is nowhere else to mow. The second situation would be that the mower needs to expand how far in advance it is looking ahead. Assuming the second situation is true, the entire field is examined to see if there does exist a better place to go to, if there is such a better place, the mower will head there; if not, then the mower will consider its job done. In *figure 4.3.2* below, the overall flow chart of the dynamic path planning algorithm is shown.

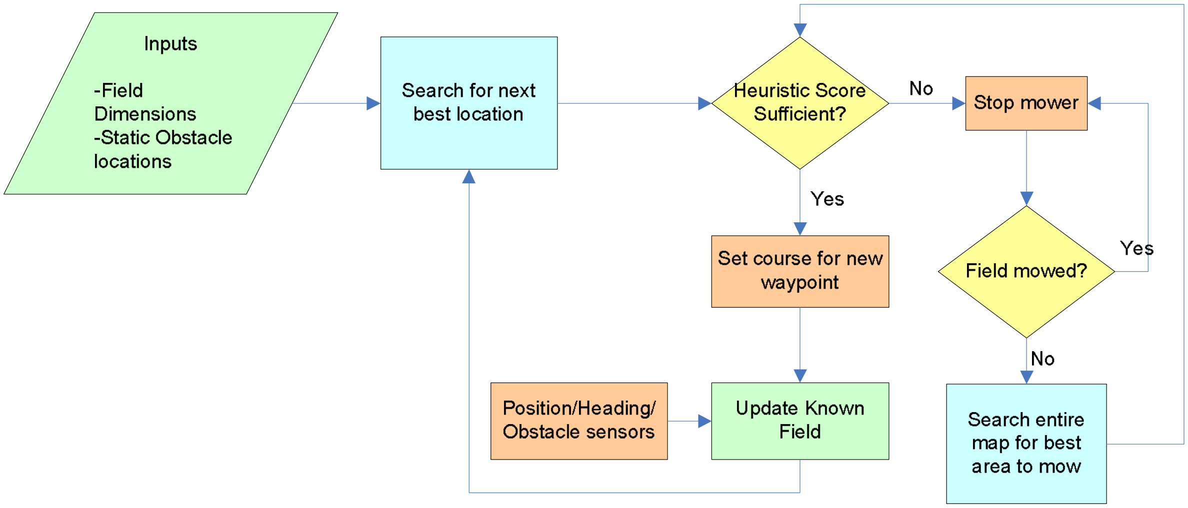

Figure 4.3.2 Dynamic Path Planning Control Algorithm Flow Chart

5 System Integration

The entire control system is run on an AMD X2 platform that is on the mower. All of the various electronics, motor controllers, and sensor communicate with the onboard computer using RS232 connections.

5.1 Overview and System Integration

The control software consists of four major components: (1) High-level path planning and control; (2) a control loop to determine and correct the current lawnmower position with regard to the path-planning, sensor inputs, and obstacle detection; (3) low-level communication interfacing between the control loop and the sensors, the actuators, and the remote base station; and (4) a PID controller to direct the lawnmower while it is moving. *Figure 5.1.1* shows a flow diagram of the integration of these systems.

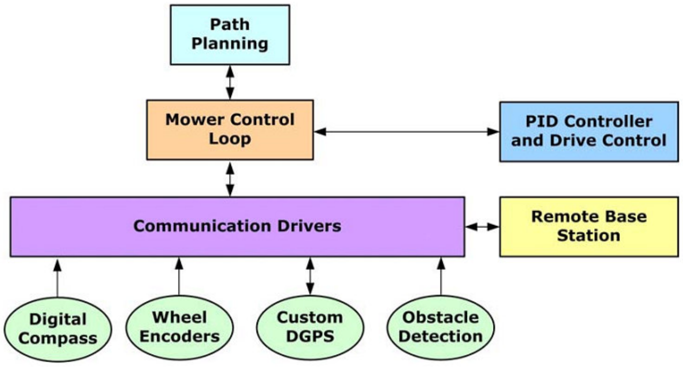

Figure 5.1.1 Control System Integration flow diagram

The control software provides the option to control the path planning through the remote base station for testing and monitoring purposes. All of the software is written in Java. An object-oriented approach was implemented to provide the most flexibility for the project. Extensive class libraries were created for the systems described above, and a detailed model description of these libraries is available upon request.

6 Safety System

The safety system has two main parts: the manual pushbutton to stop the unit, and the wireless safety system. The manual pushbutton is relatively simple and it grounds the RobotEQ to stop the mower from moving and it shuts off the motor for the blade. The wireless system has the same functionality, but instead the pushbutton is in a wireless transmitter unit. We developed a custom wireless emergency stop system for this project.

6.1 Wireless Emergency Stop System

A custom wireless emergency stop system was developed using two PIC18 microcontrollers and two Xbee RF modules. There is a receiver unit and a transmitter unit. The receiver unit is on the actual mower and the transmitter is contained in a portable hand held unit. The transmitter is constantly sending out an “OK” signal through the Xbee RF module and the receiver unit keeps the mower on as long as it is receiving the “OK” signal. If the mower goes out of range or the transmitter is set to stop sending the signal, the receiver will shut off the mower if it has not received the “OK” signal for more than 1/10 of a second. After testing the systems on a breadboard, the system was fabricated on custom printed circuit boards (PCB). *Figure 6.1.1* shows the schematic for the custom PCB, and *Figure 6.1.2* is a picture of the transmitter unit.

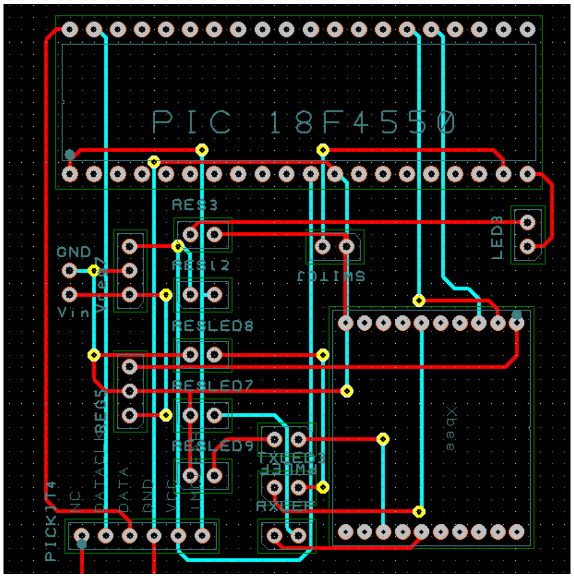

Figure 6.1.1 Custom Wireless Stop PCB Schematic

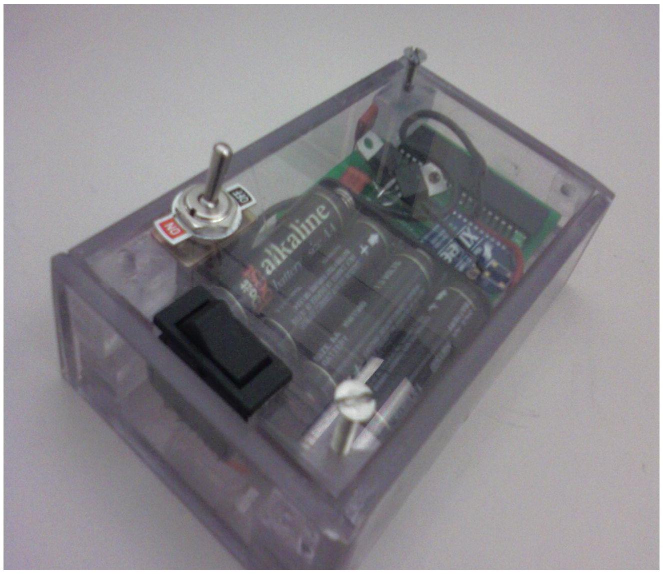

Figure 6.1.2 Wireless Stop Transmitter Unit

7 Overall System Specifications

7.1 Cost

Below in *table 7.1.1* is a cost breakdown of the entire project. This list contains not only the electrical parts, but the mechanical parts and the sensors for the Redblade. These prices include no educational discounts used, and these are prices if anyone would wish to buy these components. The actual price paid is significantly less; however the educational price does not accurately reflect the amount of money required to make a working prototype.

Component	**Cost**	**Quantity**	**Total**
6061 -T6 Aluminum	350	1	350
3003 Aluminum sheet	150	1	150
Fasteners	35	1	35
Brackets	25	1	25
Casters	16	4	64
Microstrain IMU	1500	1	1500
NPC T64 24V DC motors	300	2	600

Roboteq DC motor controller	700	1	700
E7MS Optical Encoders	47	2	94
SICK LMS 200 Laser Finder	5000	1	5000
Misc. Wires/cables/etc.	50	1	50
Computer components	470	1	470
Xbee wireless transmitters	20	2	40
Wireless PCB	33	1	33
Power-Sonic SLA 12V batteries	35	4	140
Topcon RTK DGPS System	49295	1	49295
		total:	$58,606.00

Table 7.1.1 Cost Breakdown of Project

7.2 Max Speed

The max speed of the mower is approximately 10mph, however the control algorithm limits the speed to the allowed 10km/hr as outlined in the rules.

7.3 Cutting Width

The cutting width of our main cutting blade is 19 inches.

7.4 Dimensions

The mower frame is 2’x2’x1’. The wheels extend approximately 7” from each side of the frame. The frame rides approximately 5” above the ground.

7.5 Energy Usage

The robot uses approximately 1.68kW while cutting. The robot can provide approximately 45 minutes of usage per charge.

8 Conclusions

The Redblade is Miami University’s fourth generation autonomous lawnmower. This generation, the lawnmower has been completed rebuilt and coded from the ground up. Some of the new features implemented this year include the dynamic path planning which allows the mower to avoid moving obstacles that could be coming from any direction and moving in any arbitrary direction. There is also a new commercial RTK GPS system which will be much more robust, accurate, and faster than the custom DGPS system developed last year.

This project has accomplished much more than just providing a candidate for an autonomous lawnmower competition. In fact, from an educational standpoint, the Redblade demonstrates the culmination of an undergraduate education in engineering. We have had to pull knowledge and experience from several disciplines in order to create a functional product that goes way beyond anything we could have done in the classroom. From high level object oriented programming to embedded controllers to mechanical design, our team feels that we are confident not only with the systems that we have designed and implemented for the lawnmower, but also for any engineering project we may encounter in the future. Lastly, we feel it is important to note that we are very happy to have worked with the Redblade, but more significantly, we enjoyed the whole process of the project and look forward to where it will develop in the future.

9 References

B. McNally, M. Stutzman, C. Koranda, C. Mantz, J. Macasek, S. Miller, A. Walker, J. Morton, S. Campbell, J. Leonard. “The Miami REDBLADE: Technical Report," Institute of Navigation Autonomous Lawnmower Competition, 2004.

M. French, J. Russler, J. Smith, L. Smith, T. Walters, J. Morton, S. Campbell, J. Leonard. “The Miami REDBLADE II: Technical Report," Institute of Navigation Autonomous Lawnmower Competition, 2005

D. Anderson, K. Green, M. Lang, G. Newstadt, “The Miami REDBLADE III: Technical Report”, Institute of Navigation Autonomous Lawnmower Competition, 2007

“RobotEQ: AX2550." Internet: http://www.roboteq.com/ax2550-folder.html, [May 14, 2008].

“Metal Optical Kit Encoder - E7M." Internet: http://www.usdigital.com/products/e7m/ ,[May 16, 2007].

10 Acknowledgements

The following individuals and organizations have provided valuable support in various forms. This work is not possible without their contributions.

- Miami University School of Engineering and Applied Science
- Miami University Department of Electrical and Computer Engineering
- Miami University Department of Mechanical and Manufacturing Engineering
- Miami University Physical Facilities
- Mr. Jeff Peterson (Miami University)
- Mr. Karl Reiff (Miami University)
- Mrs. Michele Lea (Miami University)
- Dr. Qihou Zhou (Miami University)