ROMO

Kevin Moriarty, CSE, Aaron Stam, EE, Collin Timmerman, EE, and Leonardo Luchetti, EE

***Abstract*— We introduce ROMO (Robotic Mower), an autonomous lawn mower that will allow users to relax while a robot mows their lawn for them. The system uses real time kinematic GPS to precisely position itself within the user's lawn. It turns by varying the speed of its two rear drive wheels allowing it to mow into corners.**

I. Introduction

The average American spends about 70 hours a year on lawn care, according to the American Time Use Survey[3]. This time, considered to be a chore, could be spent more productively on other tasks. Additionally, there is a level of physical effort required to operate a standard push lawn mower. If a person does not possess this strength, they are left with two options, purchase an expensive riding mower, or pay for a lawn care service. There currently exist several autonomous lawn mowers on the market. The problem is that these solutions can easily cost over $1000[18], require time consuming setup, and have poor battery life, requiring multiple charges to mow a lawn.

Our system requirements have been developed based on the requirements of an average consumer. As such we have determined that the mower should be able to mow a 1500 sq. ft. lawn on a single battery charge. The optimum speed for mowing a lawn is 2.5 to 3.5 miles per hour[6]. As such we have targeted a mowing speed of 3.5 mph. In order to mow the lawn with a hypothetical 12-inch blade we have determined that we need to be able to resolve the relative position of the mower with an accuracy of 5cm or better. In order to simplify the initial design, we are assuming a rectangular, level, obstacle free lawn with a known starting position.

Requirement	Specification
Lawn Area	1500 sq. ft.
Mowing Speed	3.5 +/- 1.0 mph
Battery Life	1 charge = 1500 sq. ft.
Position Accuracy	Better than 5cm

Table 1: List of System Requirements and Specifications.

II. Design

A. System Overview

In order to meet the above requirements, we need to plot a path through the lawn with a high level of precision. This requirement facilitates the need for a two-part solution. A base station acts as a reference point from with a mowers position is known. It also serves as the reference for the points which enclose the lawn. The second component of the system is a mower which travels along a path through the lawn mowing a swath.

The base station, to be placed in a static location with a clear view of the sky, contains a power supply, a GPS antenna and receiver, and a Wi-Fi module. Its function is to make measurements of the GPS signal and to transmit these measurements via Wi-Fi to the mower.

The mower traverses the lawn mowing as it travels. It consists of a platform with casters mounted at the front and two powered drive motors mounted at the back. In a final version a mowing blade would be attached in the center of the platform. This has been omitted from the development prototype due to safety concerns. Locomotion is enabled by motors which are driven via H-Bridges. The motors contain quadrature encoders which allow us to determine their speed. Turning is accomplished by differentiating speed of the motors which generates the necessary cornering force. Additionally, the mower contains a 12-volt battery, a GPS antenna and receiver, a microcontroller, a W module, and an I.M.U. (Inertial Measurement Unit).

The GPS module uses correction data received via the Wi-Fi module as well as its own measured position in order to determine its location relative the base station. The IMU generates position relative to the most recent corrected GPS measurement. The microcontroller uses the outputs from the GPS module and the IMU to determine its location and control the speed of the motors in order to traverse a predetermined path through the lawn.

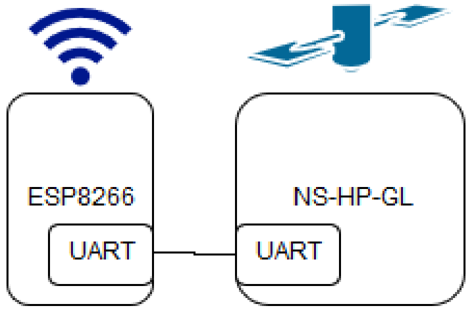

Figure 1: Block Diagram of Base Station

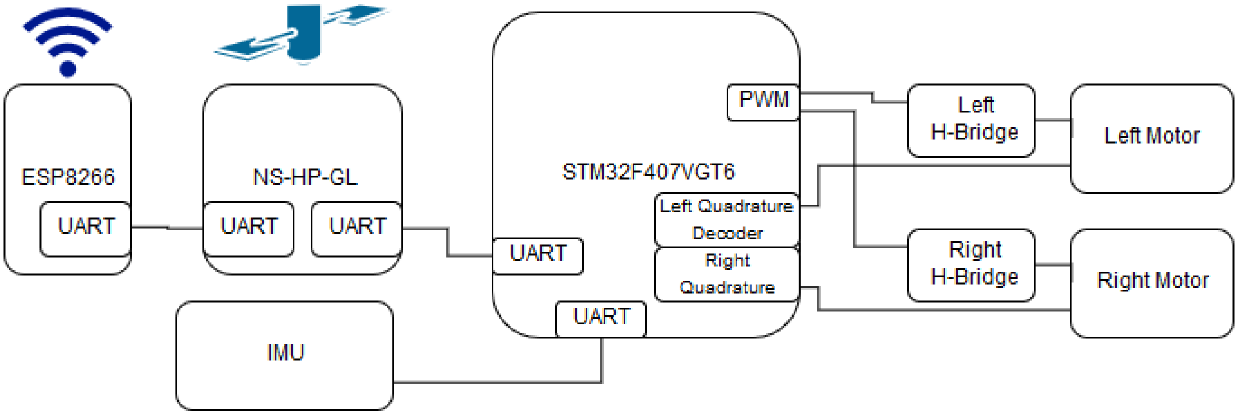

Figure 2: Block Diagram of Mower

B. RTK GPS System

Our requirement for the GPS system is to be able to determine the mowers differential position from the base station within 5 cm. Traditional GPS receivers are simply not accurate enough for this purpose. The extension of traditional GPS positioning that enables this level of position is known as real time kinematic GPS (RTK). Traditional GPS receivers calculate their position using the code information transmitted by multiple GPS satellites. [7] RTK GPS differs in that it used the measurement of carrier phase. [7] This measurement leaves several unknown variables. These are the receiver clock offset, the satellite clock offset, the hardware biases, and the number of wavelengths between the satellite and the receiver.[7] If the carrier phase measurement is taken in two locations the first two unknown variables cancel leaving only the number of wavelengths between the satellite and receiver unknown.[7] By fixing these at an integer number it then becomes possible to calculate the differential position between the two receivers[7] If one of these receivers is in a fixed location the position of a mower in reference to this location becomes known.

The implementation of RTK GPS we are using consists of two RTK capable GPS receivers connected via a wireless link. The base station has a GPS module with outputs connection data via a serial port running at 57,600 baud. [13] This data is transmitted via the wireless link to the mower where it is input to the mower's GPS receiver. The GPS receiver on the mower uses its own measurements as well as those from the base station to calculate its position relative to the base station. [13] The RTK receivers we are using are rated for "1cm+1ppm" accuracy. [13] This equates to accuracy relative to the base station of better than 2cm when within 10km of the base station. The mowers GPS module outputs the differential position over another serial port running at 115,200 baud to the microcontroller up to 5 times per second. [13]

C. IMU Sensor

Our requirements for an IMU were that it contain both a 3-axis accelerometer and a 3-axis gyroscope, and that a breakout board be available to avoid soldering the QFN packages typically associated with IMUs. These requirements guided us to a limited selection of devices. The final selection was a Bosch BNO055. This IMU was selected because it met all the above requirements and additionally has onboard processing capability. [1] This processing capability means that rather than giving us acceleration and yaw in three-axis it outputs a direction vector and an acceleration vector. This offloads a significant amount of computation from the main microcontroller and reduces the software complexity of the system. The output rate of the sensor is 100 Hz. [1]

D. Dead Reckoning (MOVE TO MOTOR CONTROL SOFTWARE SIDE, IMU, AND GPS)

Neither the GPS system or the IMU are sufficient to position the mower alone. The GPS updates position at a maximum rate of 5 Hz [13] which would not be sufficient when negotiating turns. The IMU only generates direction and acceleration vectors relative to the last measurement taken and is susceptible to drift over time [1]. The weaknesses of these two systems can be overcome using a dead reckoning algorithm. The purpose is to enable us to position the mower with the accuracy of the GPS system but at the rate of the IMU. The algorithm works by forward interpolating the position of the mower from the previous GPS position and velocity measurements.

Velocity and heading are interpolated as follows:

$$v_t = v_{t-1} + a_t * \Delta t$$

$$\theta_t = \theta_{t-1} + \frac{d\theta}{dt} * dt$$

E. Wireless Link:

The requirements for the wireless link are that it must support a baud rate greater than or equal to 57,600 baud and it must be able to function correctly at the maximum distance between the mower and base station. The baud rate requirement is derived from the baud rate of the GPS correction data which is 57,600 baud [13]. For the distance requirement we assume the maximum aspect ratio of a lawn to be 4:1 giving us a required operating radius of 77.5 ft. Wi-Fi was chosen as the wireless link of choice as it meets the above requirements while allowing the use of commodity hardware which keeps us within budget. Wi-Fi also allows the use of standard internet protocols to ensure the error free reception of correction data. With this in mind we implemented the wireless link using two NodeMCU dev kits [12] each containing an ESP8266 Wi-Fi module [4]. These two units are programmed as a server/client pair communicating through websockets on port 81. The server unit generates a wireless network which the client connects to and transmits correction data. Functionality of the wireless link was tested by using Script Communicator [17] to transmit serial data to the client module via UART and confirming that the same data was transmitted by the server's UART in return. We confirmed the wireless link successfully transmits data at a rate of 1 Hz and meets the requirements of the GPS system using this method.

F. Wheel Speed Control

In order to control the speed of the wheels as required to generate cornering force we needed to be able to measure the speed of the wheels. With this in mind we selected geared brushed D.C. motors which have quadrature encoders built in. The encoders in our selected motors generate 64 pulses per revolution [2]. Given the 30:1 gearbox ratio [2] and the maximum output speed of 366 rpm [2] the encoders will generate 11,733 pulses per second.

We concluded that it would not be practical to use interrupts and software to handle motor speed calculation. This influenced our selection of an STMicroelectronics ARM cortex -M4F microcontroller. It was chosen because its timers have quadrature decoder functionality in hardware [16]. The built in timers increment or decrement a counter based on the pulse pattern [16]. As a result, we can use the change in counter value over a known time period to determine the direction and speed of the motors. The microcontroller uses the measured speed as the input to a P.I.D. controller [14] which is used to generate the P.W.M. signals which drive the motors through H-bridges.

G. Path Generation

Path Generation is an example of a Motion Planning problem [13] and its solution lies in the Motion Planning solutions domain. The Romo starts in one corner of a rectangular lawn with knowledge of; a) the dimensions of the lawn (L_{lawn} x W_{lawn}), b) radius of the Romo's blade (R_{blade}, c) turning radius of the Romo (R_{turn}), and d) desired overlap width ($W_{overlap}$) of the path.

$$W_{eff} = R_{blade} - W_{overlap} \tag{1}$$

$$TURN_x \mid TURN_y = L_{lawn} - R_{eff} \mid W_{lawn} - R_{eff} \tag{2}$$

$$N = L_{lawn}/W_{eff} \mid W_{lawn}/W_{eff} \tag{3}$$

The effective width of the Romo (W_{eff}) is calculated in equation (1). This effective width in turn is used to calculate the effective turning radius (R_{eff}) which determines when the mower will turn to ensure full lawn coverage. The mower travels in a straight line from its starting position until it reaches its first flagged coordinate as calculated in (2) whence the mower performs a 180 degree turn and continues to the next un-mowed strip of lawn. This pattern repeats until the entire lawn is mowed, with the mower alternating clockwise and counterclockwise turns until N strips of lawn have been mowed with N defined in (3) This gives us a simple multi-step algorithm which first calculates the required number of turns using the lawn dimensions and effective mower width, subtracts the turning radius from the width or length of the lawn as applicable, calculates the position of the first turn with this distance, subtracts the turning radius again to account for all passes of the lawn past the first, and use this distance to calculate coordinates for all other turns.

H. Path Following

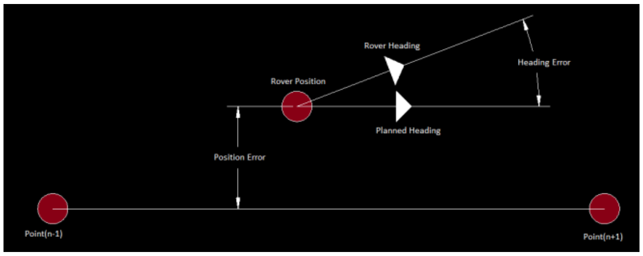

Figure 3: Error Vector Diagram

Path following is a complex part of the control system which enables the mower to traverse a known path. Using the position output from the dead reckoning system we first determine which path points lie ahead and behind the mowers current position. We then generate a course vector from the point immediately behind the mower through the point immediately ahead of the mower. We define the heading error to be the difference between the mowers measured heading and the heading of this vector. We then construct a vector perpendicular to the course vector which intersects the mowers position. This vector is the position error vector. See Figure 3 for a visual reference.

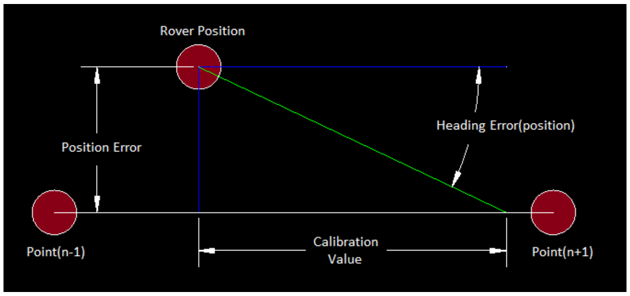

Figure 4: Secondary Heading Error Diagram

In order to reduce the above errors into a course alteration is necessary to reduce the position error to a heading error. We do this by calculating the alteration in angle necessary to intersect with the course vector over a set distance. This distance is not fixed and is a calibration value. (See Figure 4) After this step the final heading error becomes the sum of the measured heading error and the converted position error.

After the final heading error has been generated the last step is to convert the heading error into a differential in wheel speed. This is done using a P.I.D. controller [14] which takes the combined heading error as its output and generates a positive or negative number which represent turning left or right respectively.

III. Project Management

MDR Deliverable	**Progress**
Finalize System Design	100%
Get Boards Designed & Produced	100%
Complete Preliminary Software	100%
Build Mower Chassis	100%

Table II: A list of MDR Deliverables and progress

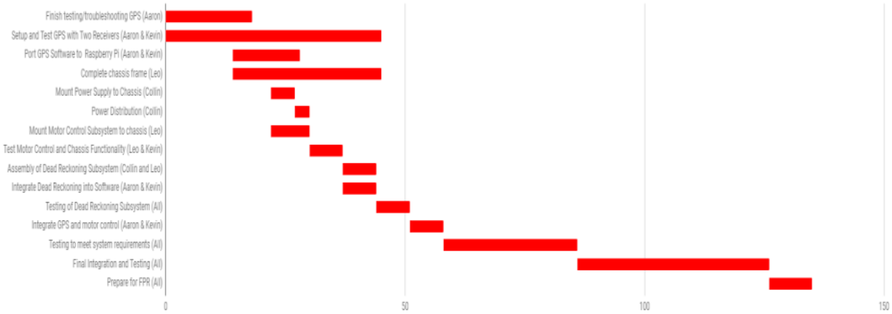

Figure 5: Gantt Chart

We have accomplished all of the goals we set for ourselves for MDR. All of our parts are here, and are ready to be assembled. Kevin programmed and tested the wireless modules that are used for GPS positioning, and together with Aaron is in the process of testing them with the GPS modules. The code to interpret and track the corrected GPS data has also been written. The motor control software is complete and ready for testing once it is integrated with the dead reckoning subsystem. We chose to build our chassis out of plywood instead of aluminum due to the ability to more easily and cheaply correct any errors in the chassis discovered during testing. Collin mounted the casters and motor hubs on a 1.5 ft. x 1.0 ft. board and drilled out the wheel hubs to fit our motor shafts. Leo has researched, designed, and is now assembling an H bridge.

Our team members have interests in different subfields of the project, so dividing up the work has proved to be straightforward. Selecting parts was the most difficult step when designing the project and where we helped each other the most. Discovering that none of us knew much about applying design requirements to part selection coming into the project, we ordered parts by working as a team. We've been meeting with our advisor, Professor Kwon once per week, and we are meeting regularly as a team. Kevin is our CSE on the team so his responsibilities involve writing, integrating, and debugging software. His primary focus is programming the wireless link and dead reckoning subsystems.

Aaron is designing the RTK GPS and motor control subsystems. His focus is on PCB design to collect GPS signals, and writing software that will allow those signals to be interpreted and integrated with the rest of our systems. Leo is in charge of assembling the H-Bridge and connecting the motor control system to the power supply. Collin's primarily responsibility is the chassis. He is in charge of constructing and modifying the chassis to meet the needs of the project. Additionally, he is constructing on the power distribution system. His focus is on wiring the battery to appropriate nodes, and generating a 3.3V power supply through the use of the TLE42744 voltage regulator. Once the system is fully integrated, everyone will work together to test and debug the mower to ensure that it functions properly.

IV. Conclusion

The project is starting to make some steady progress. We fell behind by approximately a fortnight early in the project due to a lack of face to face communication. Our electronic communications suffered from an error we were unaware of until shortly before MDR resulting in a loss of communications. We have undergone design changes a few times now concerning our choice of motors, power source, and onboard processing. However, we are beginning to demonstrate cohesion, and have seen a marked increase in productivity as a team. We have constructed and programmed all subsystems which are ready for integration and testing.

Our next step is to begin testing the functionality between subsystems and construct a full working prototype system consisting of a functioning base station and mower. We are currently testing functionality between the GPS units and the wireless link to ensure correction data is received correctly, as well as the functionality of the IMU unit and its interface with the STM32 microcontroller. We expect that getting the completed dead reckoning system working with our motor control to be the most difficult obstacle to completion due to the differences between real and ideal operation. We are optimistic that the project will be completed on time and within budget, albeit with considerable effort on our part.

V. References

[1] “BNO055,” *Bosch Sensortec*. [Online]. Available: https://www.bosch-sensortec.com/bst/products/all_products/bno055. [Accessed: 05-Feb-2018].

[2] CQRobot, “12V DC Geared Motor w/Encoder-366RPM+12kg.cm, Gear Ratio is 30:1,” AngelANQIOOO1GB datasheet.

[3] C. Ingraham, "Lawns are a soul-crushing timesuck and most of us would be better off without them," in Chicago Tribute, chicagotribune.com, 2015. [Online]. Available: http://www.chicagotribune.com/news/opinion/commentary/ct-stop-mowing-your-lawn-20150805-story.html. [Accessed: Dec. 21, 2017]

[4] “ESP8266 Datasheet,” *Espressif Systems*. [Online]. Available: https://www.espressif.com/sites/default/files/documentation/0a-esp8266ex_datasheet_en.pdf. [Accessed: 05-Feb-2018].

[5] Espressif, “espressif/ESP8266_NONOS_SDK,” *GitHub*. [Online]. Available: https://github.com/espressif/ESP8266_NONOS_SDK/releases/tag/v2.1.0. [Accessed: 05-Feb-2018].

[6] E. Perratore, “Cutting the grass in slow mow produces better results,” *Consumer Reports*, 10-Jun-2013. [Online]. Available: https://www.consumerreports.org/cro/news/2013/06/cutting-the-grass-in-slow-mow-produces-better-results/index.htm. [Accessed: 21-Dec-2017].

[7] GMK, “RTK Fundamentals,” Sept 18, 2014. [Online]. Available: http://www.navipedia.net/index.php/RTK_Fundamentals. [Accessed Nov. 14, 2007].

[8] G.W. Lucas, “A Tutorial and Elementary Trajectory Model for the Differential Steering System of Robot Wheel Actuators,” 2001 [Online]. Available: http://rossum.sourceforge.net/papers/DiffSteer/. [Accessed Dec. 20, 2017].

[9] Infineon, “Low Dropout Linear Voltage Regulator,” TLE42744 datasheet, July 2014.

[10] J.-C. Latombe, *Robot motion planning: a distributed representation approach*. Stanford, CA: Dept. of Computer Science, Stanford University, 1989.

[11] Links2004, “Links2004/arduinoWebSockets,” *GitHub*, 16-Oct-2017. [Online]. Available: https://github.com/Links2004/arduinoWebSockets. [Accessed: 05-Feb-2018].

[12] “NodeMCU Documentation.” [Online]. Available: https://nodemcu.readthedocs.io/en/master/en/. [Accessed: 05-Feb-2018].

[13] “NS-HP-GL : GPS/GLONASS RTK Receiver,” *NavSpark*. [Online]. Available: https://navspark.mybigcommerce.com/ns-hp-gl-gps-glonass-rtk-receiver/. [Accessed: 05-Feb-2018].

[14]“PID Theory Explained,” *National Instruments*, 29-Mar-2011. [Online]. Available: http://www.ni.com/white-paper/3782/en/. [Accessed: 05-Feb-2018].

[15] “S2525F8-GL-RTK : GPS/GLONASS RTK Reciever Module,” *NavSpark*. [Online]. Available: https://navspark.mybigcommerce.com/s2525f8-gl-rtk-gps-glonass-rtk-receiver-module/. [Accessed: 05-Feb-2018].

[16] “STM32F407VG ,” *STMicroelectronics*. [Online]. Available: http://www.st.com/en/microcontrollers/stm32f407vg.html . [Accessed: 05-Feb-2018].

[17] “S. Zieker, “ScriptCommunicator / serial terminal,” *SourceForge*. [Online]. Available: https://sourceforge.net/projects/scriptcommunicator/. [Accessed: 05-Feb-2018].

[18] WORX WG794 Landroid Pre-Programmed Robotic Lawn Mower with Rain Sensor and Safety Shut-off : Garden & Outdoor https://www.amazon.com/dp/B00SJEUFF4/. [Accessed: 21-Dec-2017]