Robotic Solar Farm Grass Cutting Design Challenge

Tyler Bradley, Daniel Corbellini, Nathaniel McKain, Joseph Morelli

University of Central Florida, Department of Electrical Engineering and Computer Science, Orlando, Florida, 32816, U.S.A

**Abstract** — **The objective of this project is to reduce the cost and maintenance required to keep ground-mounted solar arrays maintained. The solution is expected to turn the maintenance of the solar array farm into an automated process and reduce the amount of chemicals used to keep the grass away from the solar panels. The group chose this project as our senior design project because it was a challenge in autonomously moving robots. This was also a chance to work with other engineers in different fields to practice working together to get a task done. It tested our teamwork, creativity, and resourcefulness.**

***Index Terms*** — **Sonar, Transmitting Antennae**

I. Introduction

The Orlando Utilities Commission and Duke energy are sponsors for an inter-university competition between the University of Central Florida and the {University of South Florida}. They are sponsoring this competition to try and determine the best solution to reducing the amount of human maintenance, pesticide usage, and cost to maintain the landscape around their ground-mounted solar panel arrays. The positioning of the solar panels makes it difficult to cut the grass using conventional mowing techniques, as shown in Figure 1, so the bot must be able to travel around the pillar of the panels and under the structure.

The expectations for the senior design members are to research previous products that are similar in scope to what is required and design and build an AI-assisted, autonomous rover-based robot to articulate and guide electric weed whackers. It must cut the whole area of 50x10 feet in under 15 minutes. The bot must also have a return-to-sender function that ensures it returns to where it started so that it may be charged or stationed.



*Figure 1: Typical view of a solar panel farm*

II. System Components

The main challenge with this project was picking the components for the robot’s design. It was a challenge to make sure all parts worked well together and have each part accomplish all required tasks to complete the desired goal. This section will provide all necessary details that need to be known about each component chosen for this project.

1. *Microcontroller*

Three types of microcontrollers were used in the design. They are the Arduino Uno, Arduino Nano, and the Arduino Mega. The Arduino Nano is used for the remote kill switch. It communicates with the Arduino Uno using a wireless signal from a remote control to whether the bot is on or off. The power consumption is 19mA and has an operating voltage of 5V. The Arduino Mega is what regulates the speed and direction of the motors. It takes an input from the ultrasonic sensors and controls the servo. This will dictate the course of the robot. The clock speed is 16MHz and has an operating voltage of 5V. The Arduino Uno is used to power on the relay to power the Arduino Mega on or off. It has an operating voltage of 5V and a clock speed of 16MHz. The Arduino Uno uses SPI to interface with the nRF scanner.

1. *Sensors*

The sensor picked was the Ultrasonic Sonar Distance Sensor (HC-SR04). It's very simple and commonly used in robot projects. They are used for automation and motion sensing. They can sense from 2cm to 400cm away but are recommended at 10cm to 250cm away. The power required to run it is 5V DC and 15mA of current. Weighing only 8.7g makes them very light and they can measure at a 15° angle. They can be written with Arduino, CircuitPython, or MakeCode.

1. *Motor Controller*

The motor controller chosen was a modified version of the DROK PWM DC Motor Speed Controller. It has a voltage range of 10-50V and a current range of 60A. It controls the speed of the wheels by regulating the voltage received by the relays to the motors. The PCB was cut so that it could be interfaced with the Arduino Mega’s analog Arduino inputs. The reason being was that so the robot could adjust its speed accordingly to any input it receives. This used a 4-channel relay to control direction of the robot.

1. *Batteries*

Three kinds of batteries were used to power the design. The trimmer motors are powered by 1 12V lead-acid battery. They are basic 12V, 7Ah/20Hr, ExpertPower batteries. The batteries that originally came with the trimmers were 20V that we repurposed to be used to power the wheel motors. They are Black and Decker 20V with 2Ah that were put in parallel to create a 4Ah battery capable of powering the wheel motors. The last set of batteries used were 3.7V lithium-ion with 1500mAh. These power the kill switch remote.

1. *Fencing*

The goal of the perimeter fencing is to keep the robot in bounds during the competition. There are two circuits that were going to be used in the perimeter fence. The first is a generator circuit that uses a 555 timer. The goal of this was to inject a signal at a desired frequency (~35 kHz) into the wire surrounding our course. Second, we have a tank circuit that oscillates at our desired frequency. When the sensor circuit is placed close enough to our perimeter wire, it will oscillate at its resonance frequency (~35 kHz). The detection of the wire from our sensor would have notified our system that the robot was approaching the edge. Unfortunately, due to lockdown, we did not have the equipment to complete this part of the project.

1. *Kill Switches*

Two kill switches are being used to ensure the safety of the robot’s design. The first is the onboard kill switch which is a simple push button. It is called the Red Mushroom Emergency Stop Push Button Switch. The job of this switch is to cut power from the main battery and microcontroller when the robot needs to be stopped. The second kill switch is a remote kill switch that was designed using the Arduino Nano. The Arduino Nano is powered by a small 12V battery in the homemade remote box. It has a button to turn the remote itself on and off, and a separate button to turn the robot on and off, acting as a kill switch. The remote is rechargeable by a micro-USB and has LEDs that indicate the battery charge level of the remote.

1. *Power Management*

The Arduino Uno communicates with the 1-channel relay that dictates whether the Arduino Mega will be powered on or off. An Arduino Uno does not have the power to power another Arduino, so there is a connection made from the 12V battery into the 1-channel relay so that when the Arduino Mega is powered on it will receive 12V from the battery. There is also a connection made from the relay to the grass cutters, so if the Arduino Mega is powered off the grass cutters are also turned off. The 4-channel relay is interfaced with the Arduino Mega so that when the Arduino Mega dictates the direction of the goat, it can turn on or off corresponding channels to accomplish this task.

1. *Communication*

The nRF24 Series is a System-on-Chip that enables the implementation of ultra-low power and high-performance communication with low cost host microcontrollers. The nRF24L is the product used as a communication connection between the Arduino Nano and the Arduino Uno.

III. System Concept

To understand the complete system, a state machine of operation is helpful. Below is Figure 2 which shows the flow of information.

A close up of a map

Description automatically generated

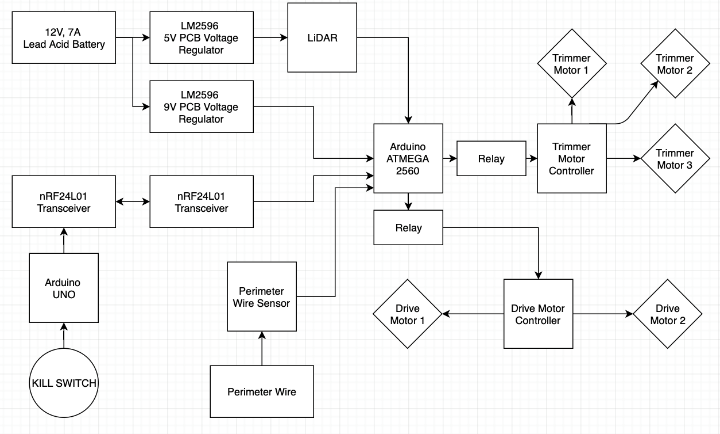
*Figure 2: Five- state state-machine demonstrating the cycling nature of the system*

The hardware design for the e-GOAT consists of batteries, controllers, motors, perimeter wire generator and the sensor for the perimeter wire. The biggest design task was creating our generator wire and generator wire sensor. Most perimeter wires can be expensive, and most GPS perimeter wires can be tedious for exact coordinates. The best route was to design our own. In addition, we were tasked with designing a few step-down voltage regulators. This was essential to the design because it helps minimize battery usage. This is important for increasing battery capacity for all components. The reason we chose a step-down voltage vs a step-up voltage from a smaller battery is because of the reliability. Stepping up voltage can be inefficient due to the fact of the device creating energy. We used multiple batteries for different components to simplify power consumption and design. If one battery was used instead of two, it would need to have the capability to power everything, which can be hard to implement. Overall, we will have 3 batteries: one to power motors, one for the electronics, and another battery for the remote kill switch.

After power management, the next important topic of our e-GOAT is the remote kill switch. This switch was implemented using radio transmitters. This could be considered one of the most important parts of the project. Our design consists of two safety kill switches. One will be present on top of the robot. If you are in the range of the e-GOAT and notice it isn’t operating properly, there is a large button there. This is a kill all switch and will terminate power driven to all components of the system. Furthermore, a second safety kill switch is required. This requirement takes care of the case where the machine is too dangerous to approach. In this case, our separate remote controller can be used to cut the power via another large, red button.

The e-GOAT will consist of an additional two PCB boards aside from the voltage regulator. The other two boards were used to create a generator wire and a sensor to detect the wires signal. This main purpose of this design was to ensure there is a backup safety precaution. The generator circuit uses a 555 timer to create a square wave in the range of 32 kHz to 44 kHz. There is a potentiometer attached in this design to ensure the generated signal can match the sensor’s resonance frequency. The sensor uses an LC tank circuit with a designed resonance frequency of around 34 kHz. If the generator wire creates a signal with a frequency around that of the LC circuit, the LC circuit can resonate this frequency and interrupt our e-GOAT. The interrupt contains the information that the robot is reaching the edge of the zone.

Lastly, the drive motor controller was selected to be external to the designed PCB’s. We decided this would be most efficient when hooking up the system entirely. In addition, the suggestion for the design was to use “off the shelf” motor trimmers and drive motors. This suggestion was used and made our interconnectivity of everything simpler. Our design ensures a safe and very sturdy soldered connection that will not introduce any voltage drop or errors in the voltage signals being provided. A positive aspect of our design is that it is manageable when creating the final layout. The position and location of every component will have easy access and will be easy to differentiate. Our final design of our components diagram is shown below.



IV. PCB Design

Our PCB board is shown below. The goal of designing this PCB was to have it function as our main brain. This PCB is equipped with a voltage regulator (LM2596), the microprocessor (STM32), mini USB, I/O ports, reset button, multiple decoupling capacitors, and a few resistors. Ideally, this PCB would function like our current Arduino MEGA is functioning. It’s responsibility would have been to control the motor controller, the sensors, and the trimmers. Without proper soldering/testing equipment our PCB was put aside and the Arduino MEGA was used. This PCB was designed on Eagle and ordered through JLCPCB.



V. Software Detail

1. *Main System*

The approach taken for our software was to make our e-GOAT as close to “plug and play” as possible. We did this by building a very basic working e-GOAT using last years e-GOAT and parts as well as a members breadboard, and arduino uno. The code was written in such a way that when the final motors are added it can be seamlessly plugged in without much coding being needed. This also applies to the other aspects of the robot. The e-GOAT was made using an HC-SR04 ultrasonic sensor. Additionally, this ultrasonic sensor will be placed on a servo motor. This servo motor will rotate left and right allowing the ultrasonic to view not only in front of it, but angled left and right.

The base of the code currently is if something is seen in front of the e-GOAT, it will stop. After stopping, the servo motor will look to the right. If nothing is seen, it will move right. If the path is blocked, the servo will rotate and view the path to the left. If this path is blocked, the robot will back up to escape from the corner. As long as nothing is in the current path, the robot will continue forward.

1. *Remote Kill Switch*

The remote kill switch is interfaced in a way that when pushed, it will kill power to the motors and to the trimmers. The approach taken for this component was to use a button as an interrupt. Essentially, when the button is pushed, it pings the arduino nano to send a signal to the arduino uno. Upon receiving this signal, power is cut to the motors via a relay.

The Engineers

**Daniel Corbellini-** is a 21 year old Electrical Engineering student from the University of Central Florida. Focused on the Power and Renewable Energy track is dedicated to working in the Power Sector to increase the use and efficiency of renewable energy.

**Nathan McKain-** is a 24 year old Computer Engineering Student. 

He has a passion for programming and hopes to one day work as a project lead. He is currently employed at Lockheed Martin as an intern where he will be starting full-time in May.

**Tyler Bradley-** is 24 year-old Electrical Engineering student. Tyler hopes to eventually pursue a career in the area of analog circuits. His goal is to start a career in a large company as a test engineer to gain experience.

**Joseph Morelli-** is a 22-year old graduating Electrical Engineering student. As a student of the Comprehensive track he has a wide array of skills in the basics of many branches of electrical engineering. He hopes to pursue a career in research and development of stronger electrical components.

Acknowledgement

The authors wish to acknowledge the assistance and support of Dr. Samuel Richie, Lei Wei, Patrick LiKamWa, Aleksander Dimitrovski, Chinwendu Enyioha.

References

[1]