Soil Sensing Robot Utilizing GPS and Bluetooth

***Abstract*– This report outlines the process of constructing and programming an agricultural data collection robot based around the Arduino microcontroller board. It is designed to measure subsoil moisture and temperature readings in order to allow farmers and agricultural automation specialists to gain a better understanding of gradual changes in the quality of soil and the health of crops. The key features of this robot are the auger drill to remove the topsoil layer, the dropper which introduces the sensor to the subsoil layer, and the Bluetooth and GPS modules that allow the robot to be controlled remotely through a mobile app and sent GPS coordinates for drill locations. Furthermore, the robot will be able to avoid obstacles through the use of an ultrasonic sensor. This report outlines the process for selecting, implementing, and optimizing the drilling process, the sensors, Bluetooth communication, and the android phone app.**

***Keywords*- Autonomous, Robot, Automation, Agriculture, Environment, GPS, Bluetooth, Arduino, MIT App Inventor, CAD, analysis**

I. INTRODUCTION

Modern agricultural techniques require homogenized fields that consume vast amounts of land with minimal to extremely negative contributions to the health of the ecosystem. They necessitate deforestation, displace millions of gallons of water from rivers and streams, and utilize pesticides, herbicides, and fertilizers, which leech back into the water table and create danger for humans and the environment. To add to that matter, traditional farming has mostly been instinctual, where a farmer would spend vast amounts of resources on farmland without adequately knowing the requirements for the plants. As a result, there is a huge loss in material resources that are expended to help regulate farmland. With the help of autonomous robots in the field, farmers can dynamically measure the requirements for the crops at the root levels to detect the exact amount of water and nutrients that a plant needs to survive [1]. Furthermore, the moisture and temperature levels can also be used to indicate the presence of other environmental problems such as mineral and aquifer depletion. Accurately predicting and diagnosing these problems before they get out of hand will save the agricultural industry millions of dollars in wasted resources.

II. ENGINEERING REQUIREMENTS

A. OBJECTIVE

The purpose of this project is to build an autonomous robot to take measurements of the soil at the root level of agricultural plants. This robot will be used to collect soil data such as temperature and moisture to be sent to farmers so that they can have a better understanding of the health of their field. The robot will cover a designated amount of land in 1 hour by the use of GPS detection and ultrasonic sensing for obstacles. Along with that, the robot will have an auger to drill a 5-inch hole in the soil where a temperature and moisture sensor will be inserted for the measurements to be at the root level of the plants [2]. This initiative is taken for the convenience of farmers to monitor large spans of farmland.

Figure 1: The Soil Probing Robot

B. MARKET REQUIREMENTS

The target market for this project is going to be the farmers and technology enthusiasts who are more inclined towards implementing new robotic technology against conventional farming in agriculture. One of the aspects to be considered with importance is the ease with which the robot can be controlled. The designed robot is an autonomous robot that will run operations based on GPS and ultrasonic sensors. Along with the control, the portability of the robot has to be taken into consideration. To make the robot more portable it is designed to be a mid-sized robot that is a little larger than most small robots. In terms of the operation time, the robot is set to run for a duration of 1 hour. Cost-effectiveness is taken into consideration and is also one of the most important factors as the robot will have to be available at a competitive price compared to contemporary robots in the industry with better durability so it can last in the wet and dusty conditions of the farmland, and the sensors will have to work in rough conditions as well. The robot will have to be user-friendly, so even if a farmer who is not very experienced with technology and operates it can have ease of access.

C. Dimensional Requirements

The robot is built with dimensional details in mind as it had to function without any complication in a certain way in a specific environment in a field. The details of the dimensions and material for each of the components used to construct the robot is displayed in table 1.

Table 1: Mechanical Components and Dimensions

Component	Dimensions	Material
Base	**17.5” L * 17.5” W**	**Steel**
Legs	**4” L * 4” W * 5” H**	**PLA**
Wheels	**4” Diameter * 1” Thickness**	**Plastic**
Motors (Drivetrain/ Drilling Auger)	**0.236” Diameter * 5.087 L**	**Steel**
Auger	**3” Diameter * 10” L**	**Stainless Steel**
Auger and Actuator Controller	**2.5” Length**	
	Holes with 0.376” and 0.236” Diameter	**Steel**
Actuator	**10” Stroke Length**	**Steel**
Tower	**2” L * 2” W * 36” H**	**Stainless Steel**
Triangular Support with Arc	**5” Vertical Length, 5.90” Horizontal Length**	
	1” Thickness	**PLA**
L-Shape Supports	**4” Vertical and Horizontal**	
	1” Thickness	**Stainless Steel**
Battery	**5.94” L * 3.86 W * 3.98” H**	

The electrical components are shown in table 2. The table shows the microcontrollers, drive-motor controller, obstruction detection sensor, GPS and Bluetooth module.

Table 2: Electrical Components

Component	Type
Microcontroller	**Arduino Mega 2560**
Obstruction Detecting Sensor	**HR-S04 Ultrasonic Sensor**
Soil Probing Sensor	**SHT20 Soil Air Temperature and Humidity Sensor**
Drive-motor Controller	**Cytron MD10 Dual Motor Controller**
Bluetooth Module	**HC-05 Bluetooth Module**

D. Base Requirements

The base of the robot has to be strong enough to support the weight of legs, wheels, motors, batteries, and sensors while also having enough space for wiring of the microcontrollers and sensors.

E. Drivetrain Requirements

The drivetrain of the robot will consist of four legs on the four corners of the base to house the wheels and the motors for the wheels. These legs will be 3D-printed using PLA material. The wheels of the robot should be able to support treading in all kinds of fields that can be found on a typical farm. The wheels are kept below 5” as along with the leg so that once mounted on the leg, the robot has six inches of ground clearance.

F. Tower Requirements

The tower is one of the most important aspects of the robot as it will hold all the components involved in the mechanism (drilling Auger, motor and Actuator) in place. The support for the tower has to be durable enough to withstand a push weight of at least 35lbs which is the weight of the entire robot without failure and also hold everything connected to it without any sort of jerking.

G. Auger Drill Requirements

To take the measurements from the soil, a hole has to be made into the soil for the probing sensor to be dropped. To do so an Auger drill is used which will be making a 4” hole into the soil with at least 3” diameter as anything more than that can damage the field and leave unwanted holes in the ground across the crop field. The Auger drill selected has to be able to drill in any type of soil typically found in a crop field and it can withstand the friction from the soil.

H. Actuator Requirements

An actuator will be used to push the auger drill down into the soil. Hence the actuator used for this robot will have to have a pushing force of at least 50lbs so the drilling mechanism can penetrate the soil

I. Sensor Dropper Mechanism Requirements

The soil probing sensor of the robot is placed right across the tower as it will be dropped into the soil to take the measurements for the temperature and moisture of the soil after the hole in the soil is made. A rack and pinion mechanism are used to drop the sensor into the soil that is controlled by a stepper motor.

J. MOTOR REQUIREMENTS

The motors used for the different components should have enough stall torque range to work the components without failure. Hence the motors are selected after conducting the calculations to determine the torque that will be required by each of the wheels and the auger drill operating in the worst-case scenario.

L. POWER REQUIREMENTS

The power system will consist of two different kinds of battery systems. The primary battery which will be powering most of the extensive and heavy-duty operation such as running the auger drill and the drivetrain motors while the secondary battery which is smaller in size will power the Arduino board, computers, controllers and sensors.

M. SENSOR REQUIREMENTS:

The sensors utilized in the robot must be compatible with the Arduino platform. This limits us to common communication types such as I^2C and serial communication methods. Furthermore, the communication types for the sensors must work at the same voltage levels to reduce the component count. For example, we must ensure that all I^2C components run on 3.3v communication and not 5v communication because then it would require a level shifter which is a component that normalizes 5v to 3.3v communication. Furthermore, any sensors that are exposed to the environment such as the ultrasonic and the soil sensor must have cases that prevent them from being damaged by weather or contact with wind, soil, and moisture.

N. THE MICROCONTROLLER

The microcontroller should have the ability to manipulate over 12 PWM devices as well as accommodate at least 2 I^2C devices. It should be powered through a 5V power source but be able to use 5V and 3.3V logic to control a variety of sensors.

III. MECHANICAL COMPONENTS

The mechanical components for the robots were selected after the calculation for the forces, torques and power requirement was conducted. All computations were made to find the worst-case requirements in the cases of motors and then components that far exceeded the limitations were chosen for the actual robot.

A. MOTORS

The motors to be selected required to have a stall torque of at least 1.34 Nm and the auger motor had to sustain a stall torque of 30 Nm. After the calculations, the motor selected for the auger and the motors were the 76-rpm Spur Gear motors with encoders shown in fig # 2. The reason for the selection of this motor is so it can sustain the torque for both the drivetrain and auger drill motors. It is also taken into consideration that the motor not only is working during the final running phase but also survives during the test phase of this project and can help the robot tread through different soil environments.

Figure 2: 76 RPM Spur Gear Motor with Encoders

B. AUGER DRILL

The auger drill bit used is made of stainless steel so it can survive through friction while drilling in different types of soil. The auger has a length of 10" and a diameter of 3" shown in fig # 3 so it can easily make a 5-inch hole into the soil where the soil sensor will be dropped to take measurements.

Figure 3: Auger Drill

C. ACTUATOR

The actuator selected for the robot has a 10" length and a 900 N of maximum lift force. The actuator shown in fig # 4 The actuator works by having a 12v motor turn a helical screw causes a rod to push out an apply pressure to create a lifting action.

Figure 4: Linear Actuator

D POWER SYSTEM

The battery selected for the robot is displayed in fig # 5 supplies power to all the motors involved in the heavy-duty activity of the robot. A battery higher than the required capacity is used to power the robot during the different test phases.

Figure 5: 12V 18Ah Battery

IV. ROBOT BASE DESIGN

A. LEG DESIGN

The legs of the robot consist of a motor, two 4in wheels, two gears, and spacers. The wheels are designed in such a way so they can house all the components for it to run while supporting the base of the robot. This design is displayed in fig # 6 the leg has a 4” length x 4” width and a height of 5”. The leg is designed with a 1:1 gear ratio for power transference between the motor and the leg and it utilizes custom 3D-printed gears that attach to a special flanged coupler that allows for a set screw to press down on the key cut-out of the motor to allow for power transmission. The flange of the couple has holes for screws that are driven into holes on the plastic gear to complete the assembly.

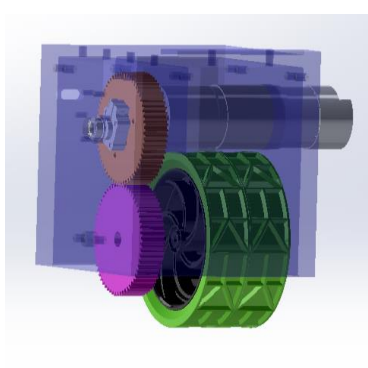

Figure 6: Leg Design

B. BASE DESIGN

The legs will be attached to the corners of the base of the robot to create the drivetrain system for the movement of the robot. The base should not exceed the 17.5" Length x 17.5" width dimension as the minimum distance between rows of crops is 18in. The dimensions of the base are selected so it has enough room to house all the other parts such as the tower, sensor, battery and all other electrical components.

C. BASE CONSTRUCTION

The outline of the base displayed in fig # 7 is built using Aluminum U-Shaped and L-Shaped channels. Aluminium is used as it has a good strength to weight ratio and is resistant to corrosion compared to steel and is relatively cheap to acquire. This will give the robot a hardy structure while being able to run operations in dusty and wet conditions. The base is also covered with a Lexan sheet to prevent water from entering the inside of the robot and harm the electronic components and computers.

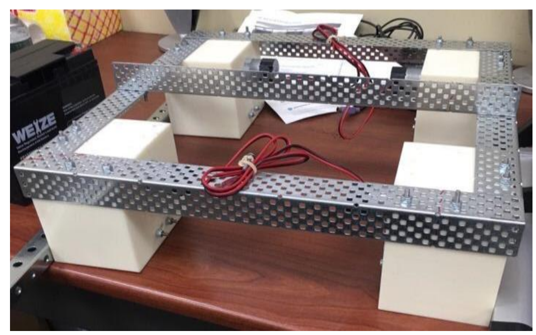

Figure 7: Base of the Robot

V. TOWER DESIGN

A. MOUNTING THE ACTUATOR

The tower that will be set in the middle of the robot and will consist of the actuator and the auger drill. The middle point of the robot is selected to house the tower at the center of gravity is the highest at that point hence, it will have proper weight distribution and prevent the robot from toppling over. The actuator, auger motor and the auger drill bit will be connected on the same line to minimize shearing when the actuator is pressing the drill into the ground. The tower itself is made of a steel square tube with holes through it for mounting. The linear actuator is connected to the robot with a series of clamps on main housing which contains the extending rod. The actuator is then secured in place with a stand-off through the mounting hole on the top to the holes in the square tube. Several nuts are also used to space actuator out further from the square tube so the drill does not hit the base when it is lowered.

B. Motor to Auger Connector

The dimensions of the motor shaft and auger drill bit shaft are different so the part in fig # 8 was designed and fabricated using aluminum as the material so the part is strong enough to hold the components together while in motion. The actuator to auger connector is designed so that a series of set screws can be added to both the top and bottom sections and push down on the sides of the hex power drill adapter on the auger and the screw key on the motor shaft respectively. These connections were tested by hand and by repeatedly plunging the drill into the ground to ensure that the mechanism is secure.

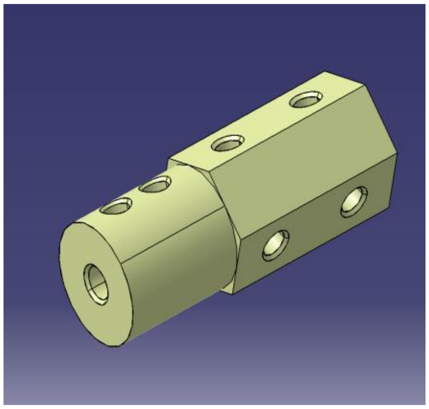

Figure 8: Auger to Motor Connector

C. The Actuator and Motor Connector

The actuator and the auger drill motor have to be connected so it can push the running auger drill to the ground to make the hole. Two separate parts were designed for the sole purpose of connecting the auger motor and the actuator shown in fig # 9. The parts were fabricated with aluminum and they enclose the auger motor with a small slot on the top so that the wires can escape the enclosure.

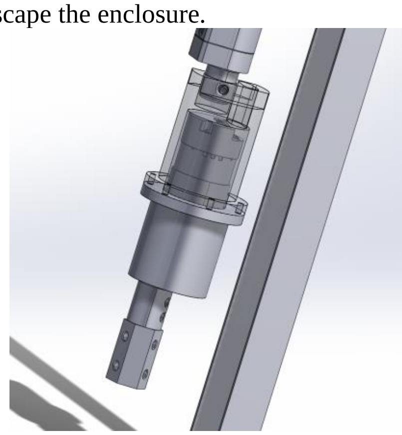

Figure 9: Actuator and Motor Connector

D. The Support Brackets for the Tower

Support brackets are used to hold the tower in place. These support brackets help hold the tower in place so the tower remains rigid and resists the force of plunging the drill into the ground. Three brackets are used to hold the tower in place, two L-Shaped brackets on the sides and one arc-shaped bracket in the rear. The bracket is designed in SolidWorks and necessary FEA analysis was conducted on it to confirm if it can withstand the force exerted on it from the robot displayed in fig # 10.

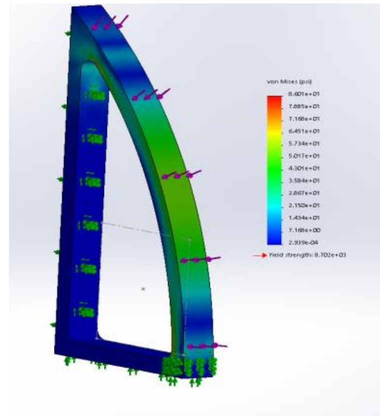

Figure 10: Stress Analysis of Arc Support Bracket

E. Soil Sensor Dropper Mechanism

The soil probing sensor dropping mechanism of the robot is set up across the tower shown in fig # 11. This was done so the sensor is near the drill so that the robot does not have to adjust its position more than a few inches when it is ready to insert the sensor.

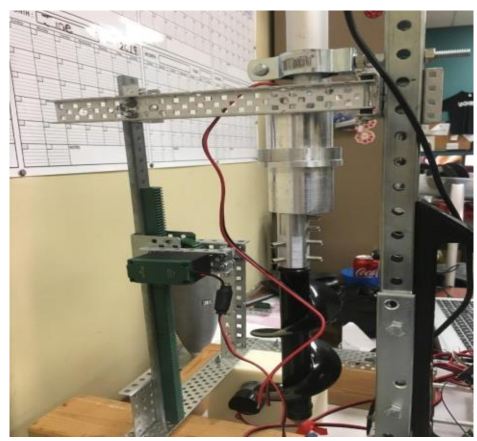

Figure 11: Sensor Dropper Mechanism

The soil dropper system was built using a rack and pinion mechanism and had to be set up after the auger was installed. This was done so that the dropper descent could be easily controlled and have limits placed in the raising and lowering motions.

F. EQUATIONS

Some equations were calculated before designing the components to determine the torque requirements required by each of the motors controlling the wheels to move the weight of the entire robot and auger to drill a hole in the soil. The robot has to move forward at an acceleration of $0.2m/s^2$ as moving faster might disrupt the overall mechanism of the robot. The power and capacity calculations were also done to select the battery for the robot. The calculation for the drivetrain motors was done by determining the forward and upward force, the forward and upward torque to account for the forward motion of the robot and in case the robot gets trapped in a pothole [3]. Another factor to take into consideration for the drivetrain was to consider the worst-case scenario in which the robot has to tread and for the case of typical farmland, this is a 20° inclined path. For the motor of the auger drill bit, the motor used in a regular drill was sufficient. The calculation for the power and capacity primary battery was required as it had to power the heavy-duty operations of the robot.

Before calculating the torque, the forward force of the moving robot at an acceleration of 0.2 m/s^2 solved in equation (1).

$$Force\ Required\ to\ move\ the\ robot\ forward$$

- $\sum F_x = M * a$ (1)
- $F_x = 14.894\ Kg * 0.2\ m/s^2 = 2.978\ N\ \approx 3\ N$

Following equation 1, the force and the torque of the robot have to be calculated at an incline of 20 degrees as that is the worst-case scenario for a typical field used for agriculture. The torque required by the wheels of the robot shown and solved in equation (2).

$$Torque\ Required\ by\ all\ the\ wheels$$

- $Tall = r * M * (a + gsin\ 20°)$ (2)
-
- $Tall = \left(4in * 0.0253\frac{m}{in}\right) * 14.894\ Kg * \left(0.2\frac{m}{s^2} + 9.81sin\ 20\right) = 5.359\ Nm$

After finding the torque required by each of the wheels the individual torque for each of the wheels is calculated using equation (3).

$$Torque\ Required\ by\ the\ individual\ wheels$$

- $Tind = \frac{Tall}{4}\ Nm$ (3)
- $Tind = \frac{5.359}{4}\ Nm = 1.34\ Nm$

To find the torque requirement for the auger drill bit the stall torque for a regular drill is taken as the auger would require the same torque as the drill to make a hole into the soil. The calculation is shown in equation (4).

$$Torque\ Required\ by\ the\ Auger\ Drill$$

- $Taug = 256\ in\ lb * 0.11298\ Nm = 29.9397\ Nm = 30\ Nm$ (4)

After calculating the torque for each of the motors the power required to operate them had to be calculated. To do so, first, the angular velocity had to be calculated as shown in equation (5) which is then used to find the power required by all the motors shown in equation (6).

$$Angular\ Velocity\ of\ the\ Wheels$$

- $\omega = \frac{v}{r}$ (5)
- $\omega = \frac{2}{0.0508*r} = 12.532\frac{rad}{s}$

$$Power\ Requirement\ of\ the\ System$$

- $P = T * \omega$ (6)
- $P = 2.992\ Nm * 12.532\frac{rad}{s} = 37.500\ W$

VI. ELECTRICAL COMPONENTS

A. ARDUINO

The Arduino Mega board is a specialized board design to be used to accommodate a large number of external components. The Arduino Mega board has 54 digital input/output pins, and 15 of these can be used for PWM. The board also has 256KB of flash memory, which is more than sufficient to hold a large size program to control all of the different components on our robot. The Arduino and all the peripheral sensors are powered by a 9v rechargeable battery which provides the robot with enough power to run the peripherals for one hour.

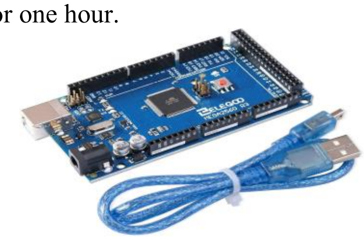

Figure 12: Arduino Mega 2560

B. MOTOR CONTROLLERS

The Cytron MD10 dual motor controller is the ideal motor controller for our project because it can be used to operate 12v motor in range of .2-20A as well up to 30A. This means that our motors will operate in a wide variety of conditions and resistance without burning out. On the robot, three of the boards are used to control the left drive motors, right drive motors, auger, and the linear actuator. The boards that are used to control the drive motors utilize a fan-out of PWM signals from the main Arduino board that only two pins control an entire side of the robot. This is done to minimize the number of pins used on the Arduino and reduce the number of wires coming out of the Arduino.

Figure 13: Cytron MD10A Motor Driver

C. Compass

The compass is the sensor on the robot that is used to ensure that the robot is going in the correct orientation. The compass/magnetometer measures the strength and direction of the local magnetometer with reference to its own orientation. The compass must be properly calibrated first so that internal error in the measurements can be accounted for. The process for properly calibrating the compass involves wiring it up to the Arduino and then running a simple code to receive the magnetic field data. The sensor is then spun around in all directions until the results of the X and Y offsets begin to normalize and settle. The process of reading the sensor results and spinning the sensor is then repeated and the average of the results is taken to minimize the error. The sensor utilizes 3.3v I^2C protocols to communicate with the Arduino through the SCL and SDA pins and sends information along with its unique address to ensure that multiple I^2C devices can be used on the same pins.

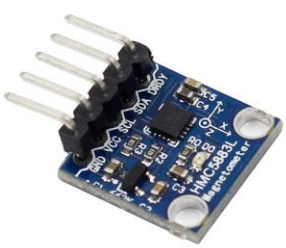

Figure 14: QMC5883L Compass/Magnetometer

D. Ultrasonic Sensors

The main obstacle avoidance sensors are the HC-SR04 ultrasonic sensors. This sensor can detect objects which are 2-400cm away without coming into contact with them. The working principle is that the module sends out eight 40 kHz sound waves. If there is an object, the waves will return and trigger a high output. The duration of the high output is equivalent to the time it takes the waves to return. Then the data from the sensor is repeated back to the Arduino, where a special library on the Arduino Mega decodes the signal received into a value that represents the time the transmission was sent and received back. From this information, the distance of the object can be determined.

Figure 15: HR-S04 Ultrasonic Sensor

E. GPS

The neo-6m GPS module is a 4800 baud GPS that is capable of receiving satellite NMEA code a deciphering to return a user's latitude and longitude within 15 feet of the actual destination. On the robot, the GPS is the main navigational sensor. The GPS requires certain conditions to meet. The weather must be optimal so that there is little to minimal cloud cover as the GPS relies on direction connection with satellites. Other types of GPS used cell phone towers to strengthen the connection to the satellites and allow for a stronger connection. The GPS must also be connected to at least four different satellites, of which three are used to triangulate the current location of the module and the fourth is used to verify the result from the other three. The GPS is relied upon for course-plotting and ensuring that the orientation of the robot is correct. In the program, the GPS and compass are continually queried to know the difference between the desired and actual heading of the robot. Those two values are then constantly subtracted from each other to find the actual value. Once the actual value of the heading is known then the robot will either perform a short left or right turn by giving the appropriate motor side more power.

Figure 16: NEO-6M GPS Module

F. HC-05 Bluetooth Module

The HC-05 Bluetooth module is a 5V compatible IC with breakout board that interfaces with the TX and RX pins on the Arduino to send and receive serial communication through Serial Port Protocol (SSP) [4]. This particular module can function as either the master or slave when connected to a network of Bluetooth devices which is why it was selected for application on the robot along with its ease of integration into Arduino projects. The ability to send and receive information is useful in conjunction with an app as it will allow the user to send commands and also receive live updates on whether those commands have been followed. To use the HC-05 with a phone, it must first be paired. To pair the HC-05, it is powered on with no signal to the TX and RX pins on the IC. Once on, an LED will be illuminated, signaling that it is ready for connection. Then the user must open their Bluetooth connections on their phone or laptop if the app is being emulated on PC and select the HC-05 to be connected. The connection is password-protected, usually "1234", and once entered, the module is ready to be of use for the mobile app on the phone or computer.

Figure 17: HC-05 Bluetooth Module

G. SHT20 Soil Temperature and Moisture Sensor

The SHT20 soil temperature and moisture sensor is the main environment sensor on the robot. This is the sensor that will be inserted into the ground to read the soil temperature and moisture levels. The sensor also sends information to the Arduino through 3.3v I^2C protocols. The sensor is an IC chip with two common types of sensors integrated together. The first is a band-gap temperature sensor that reads the change in forward voltage caused by the temperature on the diode of a bipolar junction transistor. The second part of the SHT20 is a capacitive moisture sensor. This sensor works by reading the change in capacitance caused by the presence of water particles entering the space between the IC and the copper-particle sintered cap. The cap is there to ensure that only a small amount of water can enter the gap and that the amount entering is equivalent to the amount of external moisture.

Figure 18: SHT20 Soil Temperature and Moisture Sensor

VII. Arduino Program

A. Arduino setup

Arduino Programming is done in C++ with a few special functions to make the hardware programming simpler for new users. The IDE that is used to program the Arduino comes with many tools for the reading of sensor data that were utilized for this project. These tools include the serial monitor and serial plotter which are used to display serial data that the Arduino outputs. This information is used to calibrate sensors and troubleshoot the robot. The Arduino IDE also allows for the import of sensor specific libraries that enable complicated tasks such as querying the GPS for its current location and the number of connected satellites to be done in a single command. The third-party libraries made by the corporations that developed the sensor and motor controllers used in the project are DFT_SHT20, TinyGPS++, and Cytron Motor Driver libraries. Many Arduino IDE native libraries were also used such as the servo library which enables us to use the dropper. The main section of the code includes all of the libraries and the setup conditions for each of our sensors. The main section of the program involves the robot continuously checking for serial input from the Bluetooth sensor as well as querying the GPS for its current location reading.

B. Reading SHT20 Sensor data

The SHT20 soil temperature and moisture sensor utilize I^2C communication to send information that it receives to the Arduino. The Arduino then sees this information as serial data followed by the unique address that identifies which I^2C device the information is coming from. Once received, the Arduino is programmed to use the special terms "temperature" and "humidity" in the DFRobot_SHT20 library to extract the environmental information from serial data and print it in the serial monitor so that the user can see what the sensor readings are. These readings are shown in fig # 19.

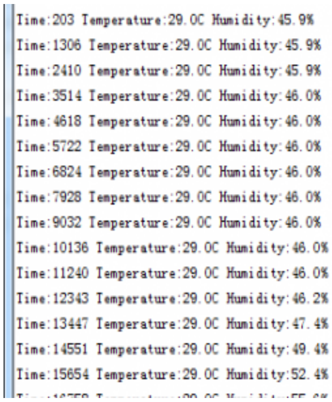

Figure 19: SHT20 Sensor Readings

C. GPS WAYPOINT

The main way that the robot navigates through user input is through GPS waypoints. GPS waypoints are locations that the user has been too and stored as GPS coordinates through the use of the app. These GPS locations are then stored as an array in the program and up to five locations can be stored to be navigated to at a later time. The process for navigation works by first accessing the information on the robot's current orientation and comparing it to the desired orientation to reach the GPS waypoint. The robot will calculate a minimum turn radius by first checking to see if the robot is within 15 degrees of the desired location in which case it will continue to go straight. If the robot is not within 15 degrees of the GPS course, it will check to see if the heading value is between 0 and 180 in which case it will perform a slow left turn and if the heading value is anything else it will perform a slow left turn until it is within the 15 degrees of the GPS waypoint.

D. BLUETOOTH COMMUNICATION

The robot receives commands from the app via the Bluetooth module which controls all aspects of movement and navigation. To control the robot with an android device, first, the device must be paired using the Bluetooth connection features on most modern mobile phones. Once the connection has been made, the app also needs to be paired so that it knows which bound devices to anticipate data from. Once all connections are established the HC-05 sensor and the phone begin to share data. Since the phone is primarily sending data, it is considered the master and the HC-05 is primarily receiving data so it is considered the slave in terms of Bluetooth connection terminology [4]. When the user is ready to send data to the HC-05 they simply press a button that triggers the phone to send a number in ASCII code to the Bluetooth module. The Bluetooth controlling code on the Arduino is written to be constantly checking certain numbers which trigger functions for specific actions. For example, "5" will cause the robot to stop moving and break out of the go-to-waypoint function, and "17" will cause the actuator to lower. Each unique movement related function is controlled in a similar process. The Bluetooth code also includes a way to receive input from a slider for controlling the turn speed during moving towards a waypoint. The slider is set with a range of 150 to 255 on the app, which means that the slider controls the turning speed of the robot from 60% to 100% power.

E. COLLISION AVOIDANCE

Collision avoidance on the robot done via the ultrasonic sensor which acts as the main collision avoidance sensor. The ultrasonic sensor works by using the Millis function on the Arduino to count how much time has elapsed since the program has elapsed. The program then compares the current time with the time value received by the ultrasonic sensor and does a calculation to convert it to an approximate distance. If the distance measured is less than a certain value it will cause the robot to stop. If the robot measures a value that is less than the prescribed stop value which can happen when it is moving but cannot slow down in time due to momentum then, in that case, it will print out "crash" to the notification window on the app and attempt to stop the robot. The collision avoidance protocol can be toggled on and off through a button on the app.

VIII. APP DEVELOPMENT

A. MIT APP INVENTOR

MIT App inventor is an app development platform developed by Massachusetts Institute of Technology that allows app development beginners to create simple apps for IOS and Android utilizing a programming language similar to Scratch which allows the user to drag and drop blocks to create programs. Another key feature of App inventor is its hidden objects which function in a similar way to the libraries in Arduino and allow for more specialized blocks to be used. In this project, the Bluetooth client and clock hidden objects were used to set-up and receive Bluetooth signals as well as continually check for Bluetooth signals to send commands along. The app creation station is split into two main components which are the designer area where the user can create buttons, sliders, lists, and other elements for the app user to interact with and the blocks section where the programming blocks are stored that control what the buttons do. The blocks in fig # 20 use the Bluetooth client hidden object to allow the user to connect to a paired Bluetooth device and then specify what to do after the connection has been established such as read that the connection has been established using a text to speech tool and change the colour of the button so that the user knows the connection has been made.

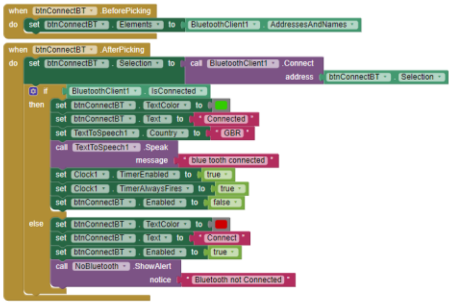

Figure 20: MIT App Inventor Bluetooth Blocks

B. User Interface

The user interface as shown in fig #21 is designed to be simple and clearly lay out all of the information with all functions on the layout having a text to speech feature included so that the user always knows what the robot is supposed to be doing and to ease troubleshooting.

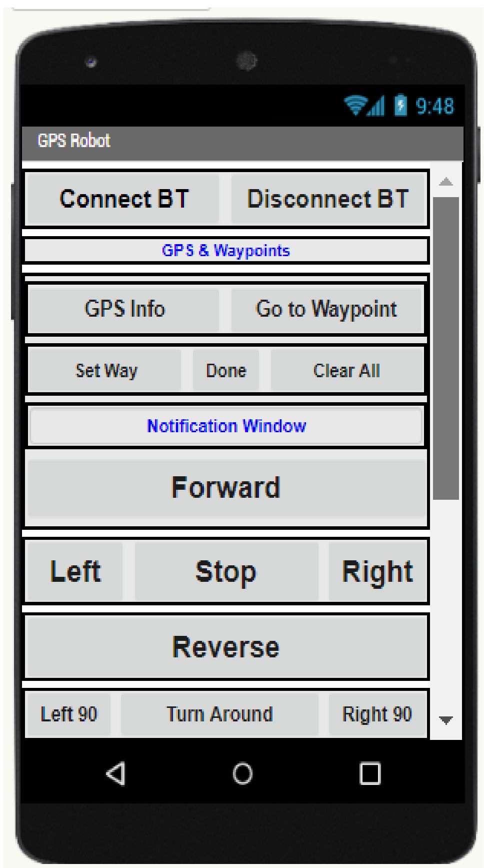

Figure 21: User Interface

IX. Cost Analysis

The cost price of all the components used to construct the robot is given in table 3. The total cost came out to be $361.19.

Table 3: The Total Cost of the Robot

Parts (Purchased)	Quantity	Price
Actuator	1	$39.00
Drilling Auger	1	$10.65
76-rpm Motors	3	$60.72
1/4" ID X 1/2" OD FLANGED BALL BEARING (2 PACK)	4	$11.56
Spur Gear Motor	2	$9.99
Spacers	1	$2.00
5mm Screws	1	$6.45
Poly Carb Sheet (Lexan)	1	$8.99
Shipping (Servo CIty)		$7.00
Soil Probe Sensor	1	$20.00
QMC3558l compass	1	$8.99
Bluetooth module	1	$7.39
Arduino Mega	1	$14.99
4pack Flange Couplers	1	$11.69
8mm Screws	1	$7.09
Ultrasonic Sensors	10	$14.99
Heavy Duty Switch	1	$3.98
Cytron Dual Motor Drivers	3	$19.25
Terminal Position	1	$11.49
BEC	1	$5.99
Battery	1	$38.99
Additional Items (nuts, bolts, etc)	1	$20.00
Total Cost		$341.20

X. Conclusion

The robot has many unique features. The primary features of the robot include a Bluetooth based app to give the robot GPS locations to which it will travel and then drill into the soil to implant a temperature and moisture sensor with which the user can learn valuable information about the quality of the soil. This design is very beneficial to the farmer and those interested in the automation of agriculture as it will help farmers gather more data and apply agricultural resources in a controlled and frugal manner.

XI. Acknowledgment

The authors would like to acknowledge and thank our advisor, Dr.Amir Elzawawy for his continued support. We would also like to thank Dr. Shouling He, Dr. Douglas Jahnke, Dr. Yougashwar Budhoo, Mr. Rachid Nafaa, and the Chair of the Engineering Department, Dr. Hossein Rahemi for their advice and expertise throughout the project

XII. References

[1] Jeff Daniels, "Future of farming: Driverless tractors, ag robots", CNBC News, 16th September 2016
https://www.cnbc.com/2016/09/16/future-of-farming-driverless-tractors-ag-robots.html

[2] Danny Lipford, "How to Measure Soil Temperature for Planting", Today's Homeowner
https://todayshomeowner.com/how-to-measure-soil-temperature-for-planting/

[3] Ferdinand Pierre Beer, "Vector Mechanics for Engineers, Statics and Dynamics-5th Edition", McGraw Hill Education

[4] Jim Blom, "Bluetooth Basics", Learn.sparkfun.com
https://learn.sparkfun.com/tutorials/bluetooth-basics/all