Irribot

Mahen Kinnur 140040024
Shivank Kumar 140010024
Raj Krishnan 140010007
Jagesh Golwala 14D070005

Problem Statement and Description

- We want to turn into reality the thought of a self – irrigated field.
- This would not only reduce the farmer’s work but also ensure that the entire field is irrigated efficiently.

- For this, our bot will follow a white line and stop when an pot is detected on the right. It will then detect the moisture content and water the pot as necessary.

Task Specifications

- We first aim that our robot follows a white line accurately. It can take sharp turns as and when necessary.
- Then, we aim to stop a robot on seeing a pot on the right
- Since it moves a little distance ahead after detecting a pot, we bring our robot a little back, following which, one of the robotic arms attached to a servo motor with a soil moisture is rotated to an appropriate level so that the soil moisture sensor is properly inserted in the soil. We then wait for a few seconds allowing the soil moisture sensor to measure the soil moisture.
- We now move the bot a little ahead so that our second robotic arm, to which we have connected a pipe, rotates to an appropriate level and waters the soil. We have moved the robot a little forward to adjust to our mechanical limitations so as to cover a distance between the two robotic arms.
- Depending on the readings, we will rotate the second robotic arm for a time of either of 0 seconds, 1.5 seconds or 3 seconds.

Project Plan

- Mahen Kinnur : Ideation of the concept of project, initialisation of constants for the bot(the optimum level, water dispersal constant, etc). Initialisation (default values) of all the ports. Forming the test cases and distributing among the members.Will be doing the part of the shaft count (for stopping the bot at the regular distance with the shaft parameter) and the ADC part, the watering arm design.
- Raj Krishnan: Forming the basis for all with the code in main program (which includes the running of the bot and coordinating all the ports with each other). Taking control for the part of soil moisture sensor. To look into the sharp sensor part (for the right obstacle detection, where the pots will be placed in the plantations).
- Shivank Kumar: Will be doing the part of the motion of the bot by completing the white line follower (using the white line sensor values and coordinating to make it complete). Even looking the part of designing the robotic-arm for using the moisture sensor, and the remaining mechanical aspects.
- Jagesh Golwala: Will be looking into the part of lcd (i.e. displaying all the current functions and the sensor values on the LCD screen. He will even be working on the part of Servo motors which is then used for the watering arm and the robotic arm with the soil moisture sensor.

Project Plan

- In the first week we had started up with learning the embedded C. The basic motion and the programs had been tested by us. Exploring the functioning of various sensors to be used.
- The second week we will be separately finishing our individual task assigned
- The third week we will combine the written code till then for the purpose of the prototype submission.
- The fourth week we have planned to commit our time for the additional tasks decided and performed tasks which were left incomplete before the prototype. Each member worked upon his task.
- The fifth week we have planned to combine all our work, coordinate each and everything and then debugging of the final code and checking all the test cases properly.

Innovation

- ➢ Our project is actually a combination of extremely innovative ideas as it connects the two very different fields, farming and robotics, with each other.
- ➢ It all begins with the very own ideation of the project i.e. the concept of irrigation (which will be a major problem in coming decades).
- ➢ We have used the optimization of amount of the water being used for the plantations.
- ➢ We have used a method of watering plants just by using the potential energy of the water thereby saving power by eliminating the use of water pumps.

Challenges

- One challenge that we faced was to stop the robot on seeing an obstacle.
  - 1] Initially we came up with an alternate solution, of stopping the robot either on regular time intervals and at regular distances.
  - 2] Later on, we were successful in stopping the robot on seeing on obstacle in the front or on the right.
  - 3] When it will see an obstacle in the front, it will make a 90 degree turn right ways.
  - 4] And when it sees an obstacle on the right, it stops, reverses a bit and then servo motor starts rotating.
- We have had a problem using the formula for calculating the duration for which the water arm should remain down. This is because the time delay function in the robot can take only compile time constant values. Hence we are now using two constant delays based on water content

Challenges

- The other challenge that we faced was to turn on the buzzer when the water level in the tank is below a certain level. For that, initially we thought of using an ultrasonic sensor, but due to multiple reflections from all the sides of the tank it was to difficult to use the readings to determine the water level. Since using an extra sensor and its corresponding circuit will lead to an increased power consumption, we decided to instead add an extra variable to turn the buzzer on. Assuming water dispersal rate is constant, we will be adding a constant to the variable for each second there is water output, and once the variable exceeds a pre-decided value based on container capacity, we will turn the buzzer on to indicate “Low Water Level”
- Also watering the soil was a bit difficult as we were not able to get the DC water pump required.
  - We managed to avoid using this by using a simple gravity powered mechanism due to which water flows when the arm is lowered below the water level in the tank.

Tasks Completed

1. **The White Line follower :**

Getting the ADC values from pin 1,2,3 is very crucial as it is used to sense the white line underneath . We selected values less than 16 in decimal and less than 10 in “hex” to follow the white line and it did well. Lighting conditions were optimal so the white line following part executed perfectly only excluding those extreme conditions of lighting-too bright and too light as well as too hot or too cold environments as the white line sensors are infact IR sensors.

2. **Operating the right Sharp IR sensor**:

Initially we had a difficulty in operating the sharp IR sensor as its calibration was difficult. But we were finally successful in doing that. We were able to detect the obstacle at a max range of 30cm and make the bot stop by setting its velocity to be zero.

3. **Moisture Sensor interface:**

Successfully able to interface the bot with the moisture sensor in analog mode and getting the moisture level values after ADC conversion in real time.

4. **Operating the servos**(Water arm and the moisture arm):

Simultaneous and independent control of both the servos was done successfully to operate the water arm and the moisture sensor arm with precision

Extra tasks Completed

Precise position of lowering of the moisture arm and the water arm:

In order to successfully water small plants in pots , we had to take the moisture level reading at the same position in the pot and lower the water arm at the same position also .this was done by first measuring the moisture level and then moving backwards for 300 ms (calibrated after a no.of attempts) and then lowering the water arm.

Changes made in plan:

1)Water arm for watering :

We didn’t get 5V dc motor for watering and thus we had to resort to alternative methods for watering which was just using the gravity

2]Stopping the bot at regular intervals:

Initially we thought of using the shaft count algorithm to decide the distance moved by the bot so as to stop the bot at required distances. But then using the Right sharp IR sensor we could easily identify the obstacle and stop to enhance use cases.

Testing

- The white line sensors of the fire-bird constantly monitors its path whether it’s following the white line or not. So the first testing strategy was to make the bot follow white line in all possible orientations for eg- when the bot is tilted from its path to right and to the left(not large deviation)
- The bot is expected to turn and properly align itself along the white line drawn on the flex.Accordingly every command execution is displayed in the LCD display(for eg-taking a left turn , moving forward)

- This test was completed successfully.

Testing

- The second part was to asses the working of the ‘right’ sharp IR sensor that it indicates lower ADC values after an obstacle comes in the right and accordingly the line following fire-bird V has to stop.
- The bot stopped as required on detecting the pot

- The moisture sensor interface is checked after proper connections were made.
- We dipped the moisture sensor in soil containing varying amount of water
- Correspond ADC values in the range 0-255 were returned according to the moisture content

Performance metrics

- The sharp sensors on the right take a bit of time to detect obstacles and give the readings, in which time our bot moves a bit forward.
- So we overcame that by measuring the distance after experimenting a few times and moved the bot backward accordingly.
- Same is the problem with the soil moisture sensor, for which we increased the time for which the sensor will remain inserted in the soil.
- Also, we need to determine the amount of water to be dripped based on the sensor readings.
- For that we set ranges of the soil moisture sensor values and water the soil for 0 seconds, 1.5 seconds and 3 seconds.

Testing

We had to augment all the systems and now as the bot senses the right obstacle (pot) it should stop and consecutively take 3 readings of the moisture level in the soil and then lower the water arm to water the soil as per the following algorithm:

I. Low moisture (ADC{190-255}): need to water the soil for longer duration.
II. Medium moisture(140-190): need to water for a shorter duration.
III. Optimum moisture(<140): no need to lower the water arm as watering is not required.

The test was completed successfully

Reusability Features

- The code can be reused and made more efficient in terms of quantity of water.
- It can also be used to implement many other tasks that we have mentioned in the future enhancements.
- Also by adding another sensor on the left and its corresponding code, the bot can be made to stop on seeing a left obstacle.
- In terms of hardware, we can add some more sensors using the same number of robotic arms but some additional motors to make it a multipurpose bot.
- The hardware can be modified in case the farmer wishes to have sprinkle irrigation facility in the field, thus serving both drip and sprinkle irrigation facilities.

Future Enhancements

- There are many expansions that can be done with respect to this robot, which are mainly oriented towards helping the farmers.
- Since the robot is going to move around the field, a web cam can be added, which is used to detect the presence of rodents or any other harmful bodies to the plants.
- Can also be used as a moving scare crow which also makes noises, a huge improvement over the present case.
- It can also be used to spray pesticides as desired by the farmer.
- There are other sensors available in the market which give the nutrient content of the soil. These can be attached to the robot, so as to get thee corresponding nutrient level.

Future Enhancements

- Also, a user interface (probably a user – friendly app) can be made, which can be used by the farmers to get the appropriate information.
- Also, they can set the optimum water level of the soil as per their wish and control the pesticide amount to be given to the soil.
- Add a water level sensor and a wireless module to alert the user when the water tank needs to be refilled.
- We can directly check the water level of the tank by using “Water level measurement Sensors”.
- By placing them at the required level in the tank, once the water level falls below that level, the buzzer will indicate that.
- The robot can be made self – sufficient in terms of power by placing a solar panel on the robot, since the robot will carry out its work mainly during the daytime.