Embedded System Project Report
CS 684 Project

"Re -Plantation bot"

Team no - 12
Devendra Singh 123050056
Ravi Kumar 123050054
Manoj Gedam 133050077

Department of Computer Science and Engineering
INDIAN INSTITUTE OF TECHNOLOGY BOMBAY
Powai,Mumbai,India -400076

Spring Semester 2013

Contents

List of Figures

1 Introduction

The aim of our project is to design a Re-planting robot which can be used to replant the plants which are not growing properly due to over-utilisation of resources in certain areas. It takes input from user to know the location at which it is need to be replanted and then by using camera it detects the plants. After this it picks up the plant and move it to specified location by following black line.It uses Zigbee module to co-ordinate between camera and bot.

1.1 Definitions, Acronyms and Abbreviations

- **FireBird**: A robot indigenously designed at ERTS laboratory, IIT Bombay.
- **ZigBee** is a specification for a suite of high level communication protocols using small,lowpower digital radios based on an IEEE 802 standard for personal area networks.
- **Wheel Encoders** The wheel encoder is a sensor attached to a rotating object (such as a wheel or motor) to measure rotation. By measuring rotation your robot can do things such as determine displacement, velocity, acceleration, or the angle of a rotating sensor.

2 Problem Statement

To design a bot using FireBird V for replantation of plants from one place to another as specified by the user by following black line and then coming back to same position to pick up another plant.

3 Requirements

3.1 Functional Requirements

Robot uses camera for detecting the specified plant and if it doesn't finds specified plant then it moves little forward and searches for plant and repeats this till it finds the plant.Then it uses sharp sensors to detect the plant and then adjusting itself to maintain minimum distance to plant.then, it moves slowly some distance and again repeats the same process till it finds and picks up the plant and moves 180 degree and moves forward till it finds the black line and then it follows black line till reaching the specified distance and then again turns itself 180 degree and moves back to same position to collect another plant.Then it repeats whole procedure again for collecting another plant.

3.2 Non-Functional Requirements

- Accuracy in determining the specified plant.
- Accuracy in determining and transmitting the value of distance by the user.
- Less delay in transmitting and receiving the signals.
- The bot working should be accurate and repeatable.

3.3 Hardware Requirements

- Zigbee transmitter and receiver modules.
- Digital IR sensor
- Fire-bird ATMEGA bot
- 3 Servo motor
- Plant picking up arm

4 System description

The system consists of a Firebird V Robot,a camera and a remote Zigbee Controller. All the Image Processing and Instructions for the robot while traversing the trough are being calculated on the PC. The camera mounted on the robot is used for video capturing. The input from the camera is processed in PC and instruction is send to robot using XBEE, where robot will recieve message as packets and will get inialised. There is no need of sending any data from Robot to PC. The details of implementation of the system has been described in implementation part. The final implented systems picture is shown in fig. 1

4.1 Big picture

If looked at a large scale, the System can be in five states. These states when looked with their details, can be further divided into some more states. The five states on the bigger scale are described as below:-

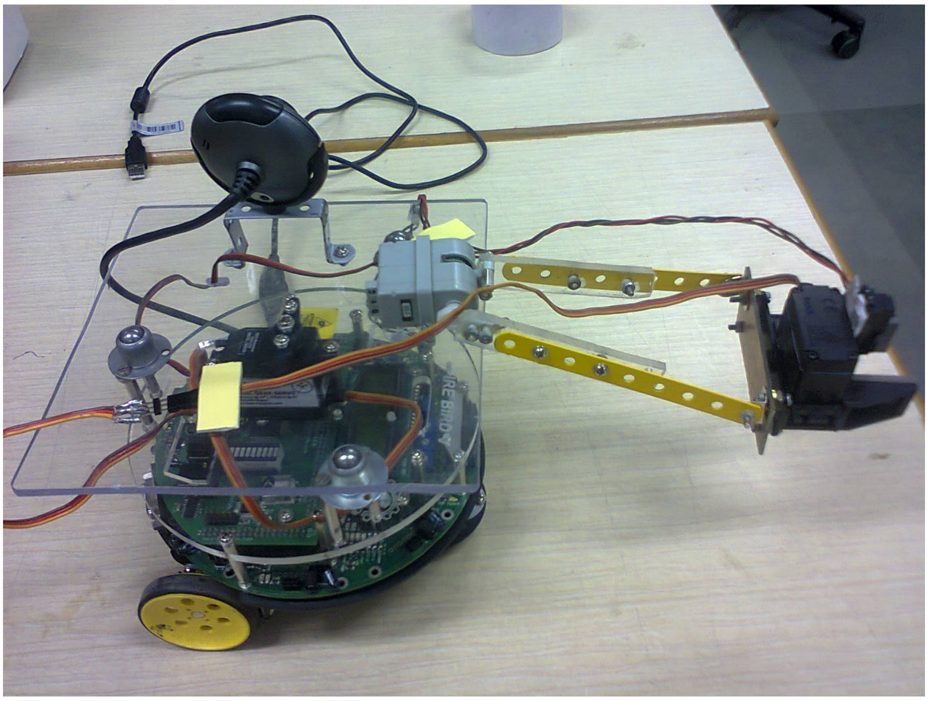

Figure 1: Bot

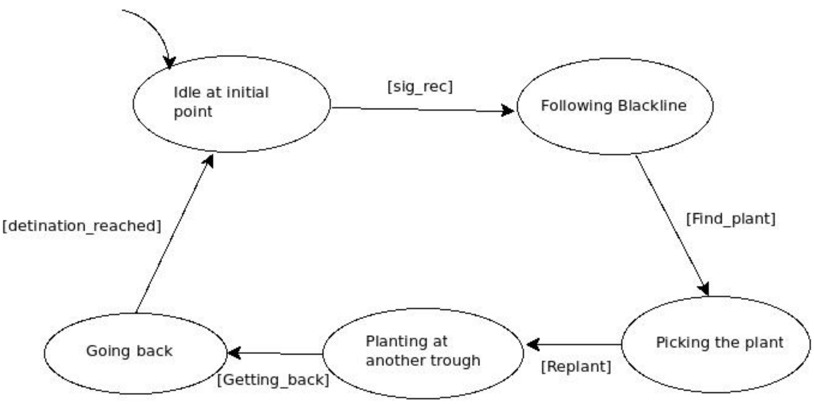

Figure 2: Big Picture of the system

4.1.1 Idle at initial position

The system will be initially in this state only. In this state, the bot will be listening for the command from the PC. Once initialized the robot will leave this state and will start moving.

4.1.2 Following blackline

The bot will be moving on the blackline and camera will be taking pictures and sending them to PC where they are processed. Once a plant is spotted, PC gives instruction to robot to pick the plant.

4.1.3 Picking the plant

After a plant is spotted, robot takes a left turn and searches the plant using sharp IR sensors and move towards the plant. When it reaches near the plant, robot will pick the plant.

4.1.4 Replanting

After picking the plant robot will move back to the blackline and it will put the plant on another trough.

4.1.5 Going back

After reaching to its desired location, robot will put the plant there and follow the same path back to the initial position from where it started. And it can do the same work again and again.

4.2 Following blackline in detail

In this state robot will be moving on blackline and searching for the plant. It will move until it finds a plant or the blackline ends. It is shown in fig. 3. Various states in this states are following:

4.2.1 Searching the plant

Camera on the robot is continously taking pictures and sending them to PC and moving forward. PC will be processing those images and whenever it finds any plant it will send a signal to bot to stop moving forward and take a left turn until plant comes in front of sharp IR sensor mounted on the picking arm. It is shown in fig. 4

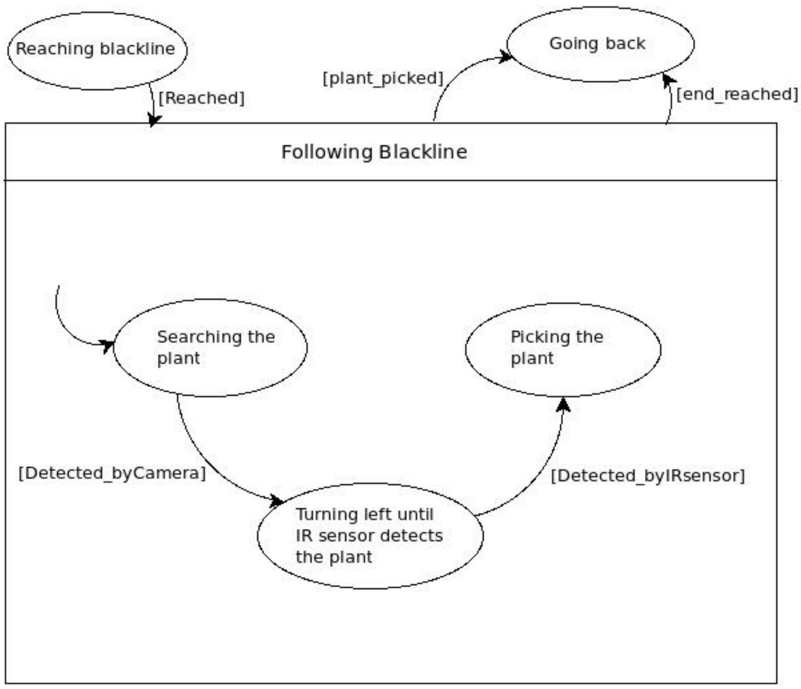

Figure 3: Following blackline

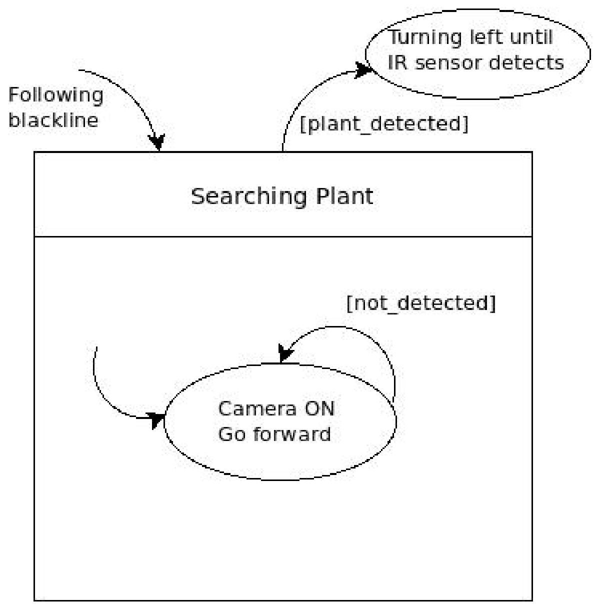

Figure 4: Searching the plant

4.2.2 Turning left

Now the robot will be taking left turn and searches the plant using sharp IR sensor and will be turning left until it detects the plant.

4.2.3 Picking the plant

Once the plant is detected by the sharp IR sensor, robot will be moving towards the plant. After reaching near the plant, it stops and pick the plant using the arm mechanism. It is shown in fig. 5

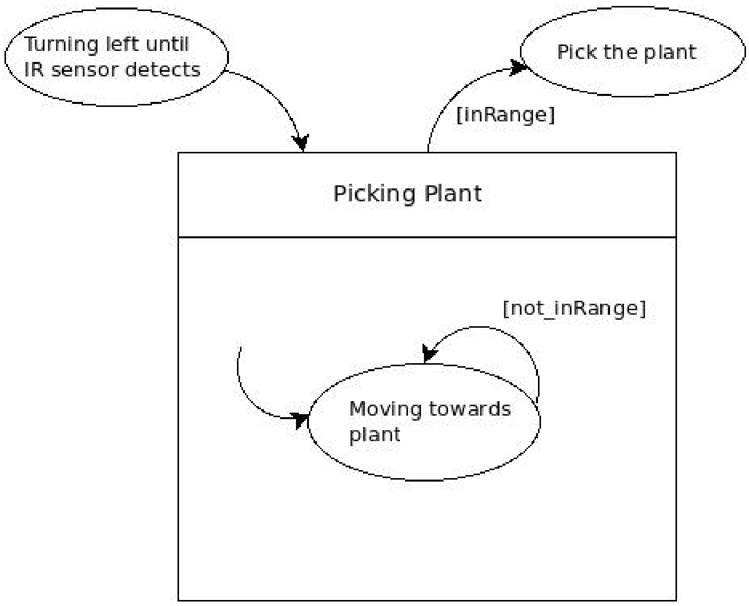

Figure 5: Picking the plant

4.3 Finding the plant in detail

When the front sharp IR sensor mounted on picking arm mechanism detects the plant, robot will be moving towards it to pick it up. It will be moving a slight forward and again calculates the distance because the distance calculated earlier can chnage due to error encountered by slipping of wheels of the robot. And it follows the same procedure until it reaches near it and finally picks it. It is shown in fig. 6

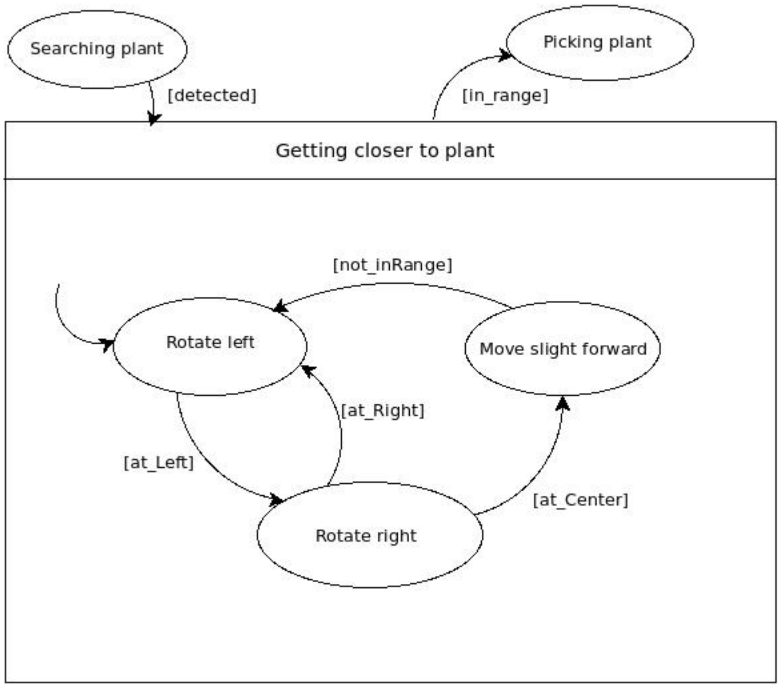

Figure 6: Finding the plant

4.3.1 Rotate left

The robot will be rotating left while searching the plant. When it comes in front of sharp IR sensor, distance calculated by the sensor will start decreasing and it continues rotaing left. Distance will be minimum when the plant will be exactly in front of sensor. As it rotates more left, diastance starts increasing and now the robot will stop taking left turn here.

4.3.2 Rotate right

Now the robot will take a right turn slight more than the last left turn and moves a little forward. This process is again repeated untill the robot reaches near the plant and picks it.

5 Implementation

5.1 Scilab code description

Scilab code is used for detection of plant.

- It opens up the camera and takes image of surrounding.

- This image is analysed and centroid of image is calculated.
- In our case,we are collecting plants having bigger size so centroid of bigger plant will be above than other plants.
- If y axis of centroid is above 125 then send detected through Zig-bee otherwise send not detected and bot will move 2 cm.
- Repeat the above process again till it finds the plant.

5.2 C code description

One of the important features of the ATMEGA 2560 processor, present on the Firebird V , is the interrupt handler. Interrupt handlers have been utilized in our C code.So let's have a look at working of ISR.

An interrupt handler, also known as an interrupt service routine (ISR), is a callback subroutine in microcontroller firmware, operating system or device driver whose execution is triggered by the reception of an interrupt. Interrupt handlers have a multitude of functions, which vary based on the reason the interrupt was generated and the speed at which the interrupt handler completes its task. An interrupt handler is a low-level counterpart of event handlers. These handlers are initiated by either hardware interrupts or interrupt instructions in software, and are used for servicing hardware devices and transitions between protected modes of operation such as system calls.

In our code, interrupt handler has been used to know that plant to be collected it detected or not detected by camera and perform a

Detailed explanation:

1. USART0 is initialized and then controller goes into an infinite loop waiting for an interrupt to arrive from the Zigbee module.
2. As soon as some data is received by the Zigbee module, the interrupt handler starts executing.
3. If data received is 8 then it shows plant is not detected then bot will move 2 cm forward.
4. If data received is 1 then it shows plant is detected so it will perform whole task.
5. Then it turn 90 degree to left and IR sensor will calculate distance.

6. Bot will adjust itself till the distance calculated by IR sensor is minimum and less than 40 cm.
7. Then bot will move forward by 5 cm
8. Step 6 and 7 are repeated till bot distance calculated by IR sensor is less than 10 cm.
9. Then it moves its arm to pick up the plant.
10. It moves forward till it white line sensors detects black line.
11. After this bot follows black line till it reaches specified distance.
12. Then it turns back 180 degree and moves back to same position by following black line till it reaches initial position
13. Then step 1 to 12 are repeated again for replantation of next plant.

6 Testing strategy and data

We tested the bot for following conditions:

- **Plants are located at random distance from blackline**
  For this test case our robot picks the plants well if they are within range of sharp IR sensor.
- **Plants are closely spaced**
  Runs fine for sufficiently spaced plants.
- **Plants are located in line at a particular diatance**
  For this test case our robot picks the plants perfectly if the line is within range of sharp IR sensor.
- **Plants are closely spaced**
  BOT moves along the whole blackline and stops at the end of blackline without doing anything.

7 Design Challenges and Open Issues

- **Installation of OpenCV**: We tried installing opencv in ubuntu but while running it with codeblocks it was giving Error: camera not initialised that we cann't resolved till end.Another problem we faced in installation of opencv is in windows 7.In windows 7 installation of opencv was fine but it requires GNU compiler for which we needed mingw but installation of mingw created a problem while running any c program it was giving Error : libstdc++-6.dll is missing and we could not resolve it even by registering this dll file in windows registry.so in last we opted to use scilab instead of opencv.

- **Achieving Location Accuracy**: Our bot tracks current location using wheel encoders. Hardware inaccuracies introduce small error with every movement. Errors accumulate over the time to significant proportion. To highlight specific problems, positional encoders do not reliably measure movement less than 4 mm. Both DC motors may not run with same speed for same value of PWM. There might be small difference in wheel circumference of each wheel. Response delay of each motor may be different. As result of these hardware inaccuracies, achieving same results with repeated trials is itself a challenge.So we used checkpoints like reaching black or white lines while movement of bot.

- **Detection of Plant**: We used Scilab for image processing but running a image processing program on scilab becomes difficult as we can capture images through camera only once and if we try capturing second time after turning off camera once it will give black image.For this,we need to close the scilab and re-run it again.

8 Future Work

We have identified following possible improvements for future projects.

- **Improving image processing** Our image processing is based on sizes of plant. It can be improved by extracting the exact object(plant) and detecting colour,size and other parameters.

- **Digging Arm** We have not dig a hole before planting a plant.An arm for digging a hole can be included with the bot to make it more realistic.

- **Container** A container can also be attached to bot to reduce the movement of bot so that it can collect all the plant at once and plant them at specified distance.

- **Graphical user interface** A GUI can be used to give commands as we were giving commands through scilab command line.

9 Conclusion

This project demonstrate that it is possible to automate replantation of plants to increase their growth.Although real environment of greenhouse is different from demonstration platform but it can be improved to perform it in real environment. We substantiate this claim for white line. In real greenhouse,we cannot use black lines because of the soil and the dirt on the floor. However, black line for this project is mere abstraction of any mechanism that allows the Bot to move in straight line. We can use gyroscope or even laser guidance system to move along straight line in real greenhouse. The system is developed to detect a specified plant successfully and collecting it even without using white or black line following as it uses only IR sharp sensors for collecting the plant.

References

[1] *E-yantra website.*, `http://www.e-yantra.org.`, Accessed: 2013-10-30.

[2] *Firebird v atmega2560 robotic research platform hardware manual.*

[3] *Firebird v atmega2560 robotic research platform software manual.*

[4] *Scilab tutorials*, https://www.scilab.org/resources/documentation/tutorials.

[5] *Xbee/xbee-pro oem rf modules manual.*

Appendices

A X-bee

XBee and XBee-PRO 802.15.4 OEM RF modules are embedded solutions providing wireless end-point connectivity to devices. These modules use the IEEE 802.15.4 networking protocol for fast point-to-multipoint or peer-to-peer networking. They are designed for high-throughput applications requiring low latency and predictable communication timing. While Bluetooth focuses on connectivity between large packet user devices, such as laptops, phones, and major peripherals, ZigBee is designed to provide highly efficient connectivity between small packet devices. As a result of its simplified operations, which are one to two full orders of magnitude less complex than a comparable Bluetooth device, pricing for ZigBee devices is extremely competitive, with full nodes available for a fraction of the cost of a Bluetooth node. ZigBee devices are actively limited to a through-rate of 250 Kbps, compared to Bluetooth's much larger pipeline of 1Mbps, operating on the 2.4 GHz ISM band, which is available throughout most of the world.

What is X-CTU

Digi International offers a convenient tool for Xbee module programming - X-CTU. With this software, the user be able to upgrade the firmware, update the parameters, perform communication testing easily. The basic operations are listed below.

How To Use X-CTU?

Before we can talk to an XBee, except USB cable we will need to get an USB adapter for XBee With the USB adapter, we can communicate with Xbee through "USB Serial Port". We may have more than one device on serial port, for example, we want to test the wireless communication and connect two XBee to our PC. So, we will add more devices on serial port. In either case, we need to select the correct one that we want to perform operations.In this case, it only has one Xbee connect to PC and it locates on com port 9 (COM9).