CS 308 Embedded Systems Lab

Project Documentation

The Shopping Robot

By:-

Team 5

Rahil Shah rahils@cse.iitb.ac.in 09005011

Palllav Dhobley pallavdh@cse.iitb.ac.in 09005012

Namit Rawal namitrawal@cse.iitb.ac.in 09005014

Aditya Gupta adityagupta@cse.iitb.ac.in 09005017

Under the guidance of

Prof. Kavi Arya kavi@it.iitb.ac.in

Department of Computer Science and Engineering

Indian Institute of Technology Bombay

April 18 2012

Table of Contents

1. Introduction

Imagine a busy departmental clothing store; say a Westside or a Lifestyle. There exist a lot of discarded clothes outside the changing room. It is this problem that we want to address!

Instead of making an employee sort these clothes and put them back in the correct racks, we envision a future where the entire process is automated! Thus, we wish to automate the process of sorting clothes and placing back in the correct racks.

Here, we come up with system in which there is a Sorting bot (The gripper bot) which sorts different clothes based on the RFid tags present on them and then Delivery bots delivers the clothes to their correct shelves.

This particular process can be a generalized to a simple sorting and delivering mechanism. Although we have used a departmental store as an example the same system can be used in garbage sorting and disposal, at factories and at many other places. Thus we are attempting to automate the generic problem of sorting and transporting.

2. Problem Statement

This project aims to automate the generic problem of sorting and delivering objects.

Sorting is done by a **gripper bot**, which has **RFid scanner** connected to itself. It scans the RFids tags attached to the objects and sends this data to the coordinator wirelessly using zigbee module attached on it. The coordinator instructs the gripper bot to place the object on a specific delivery bot.

The **delivery bot** then **delivers** the object to the appropriate **destination** based on the information given to it by the coordinator again wirelessly via zigbee.

All the bots are moving in the same arena, and hence **path planning** and **collision avoidance** is also an integral part of the project.

3. Requirements

A. Hardware Requirements

- Firebird — Requires 1 bot as the gripper bot.
- Zigbee — Requires 4 modules, one each on a bot and one connected to the laptop on a serial port, to maintain communication between the bots and the coordinator.
- Gripper Arm — Requires 1 arm to pick objects from their location and drop them to their respective delivery bots.
- Spark — Requires 2 bots (Delivery bots).
- RFID Scanner — Requires 1 module to scan the RFID tags on the objects.
- RFID Tags — Requires 5 to 10 modules to distinguish between different objects.

B. Software Requirements

- Microsoft Visual C++ 2010 Express — To run the code of the coordinator on a windows laptop/pc.
- E-CTU — To configure Xbee modules so that they can communicate with each other.
- AVR Studio — To program instruction onto a given bot
- AVR Boot Loader — To boot a program on a given bot.

4. Implementation

A. Functionality:

- **The Gripper Bot:**

Pick up an item from location of dumped items and communicate the RFid to the master laptop. Depending on the master laptop's response, the gripper bot will either drop the item on the required delivery bot (which bot will be told to the gripper bot by the laptop) or will return the item into the original location.

- **The Delivery Robots:**

The helper robots when not deployed are in a parking yard! They wait for a signal from the master laptop. On receiving a particular signal, the helper robots will pick up the required basket and drop it off at the needed point. This will be done based on the path told to the robot by the laptop. The helper robots will also respond to interrupts to prevent collisions with other robots.

- **The Coordinator:**

The master laptop controls the entire process. On receiving an RFid from the gripper bot, it will look for an available delivery bot. If no bot is available the laptop instructs the gripper to return the item to the original location. Otherwise it asks the gripper to put the item in the correct delivery bot. When enough items are present in a particular delivery bot, the laptop instructs it to take the items to their correct location in the store i.e. grid point.

B. The step-by-step algorithm for the entire process is given by the following:

1) The initial condition of all the bots is at rest at their origin.
2) If the coordinator finds an object which is not in its place, it instructs the Gripper bot to pick up that item from its initial position.
3) The Gripper Bot starts moving towards the object as instructed by the coordinator.
4) On reaching the location of the object, it picks up the object using its gripper arm and scans the RFid tag of the object from the RFid scanner present on the gripper arm itself.
5) The Gripper bot communicates to the Coordinator the RFid tag of the picked up object.

6) If RFid is not scanned due to some error, the Coordinator instructs the Gripper bot to place the object back again at its position. The algorithm then proceeds from step 2.
7) Based on the RFid scanned, the Coordinator checks if an empty delivery bot is available. If no delivery bot is available, the Coordinator instructs the Gripper bot to place the object back again at its position. The algorithm then proceeds from step 2.
8) Based on the RFid scanned, the Coordinator instructs the gripper bot to place the object on a particular delivery bot.
9) The gripper bot places the object on the correct bot and informs the coordinator about the same.
10) The Coordinator checks if any delivery bot has required number of objects. If a delivery bot has the required number of objects, the coordinator instructs it to deliver the basket to the desired location. It also plans the path for the delivery robot to minimize delivery time and prevent collisions.
11) The algorithm returns to step 2.
12) If no objects remain to be sorted and delivered then all the bots return to their initial positiosn.

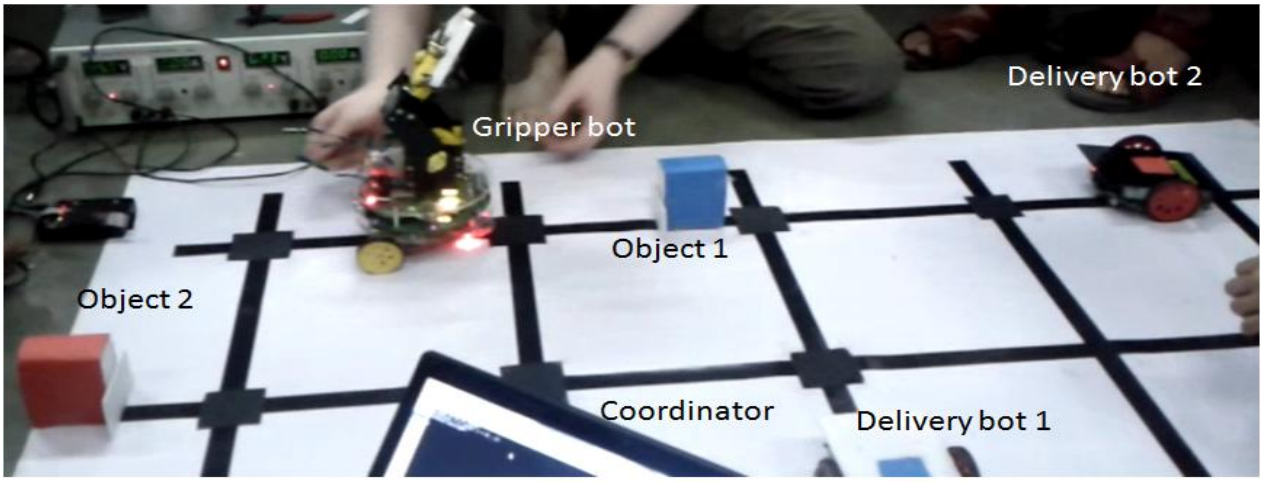
5. Testing Strategy and Data

Stage 1

Initially, the **gripper bot** and both the **delivery bots** are at their specified positions on the grid.

Coordinator knows thier positions and also of the objects to be delivered.

The Blue object is to be placed on Delivery Bot 1, and the Red one on Delivery Bot 2.



The Initial State

Stage 2

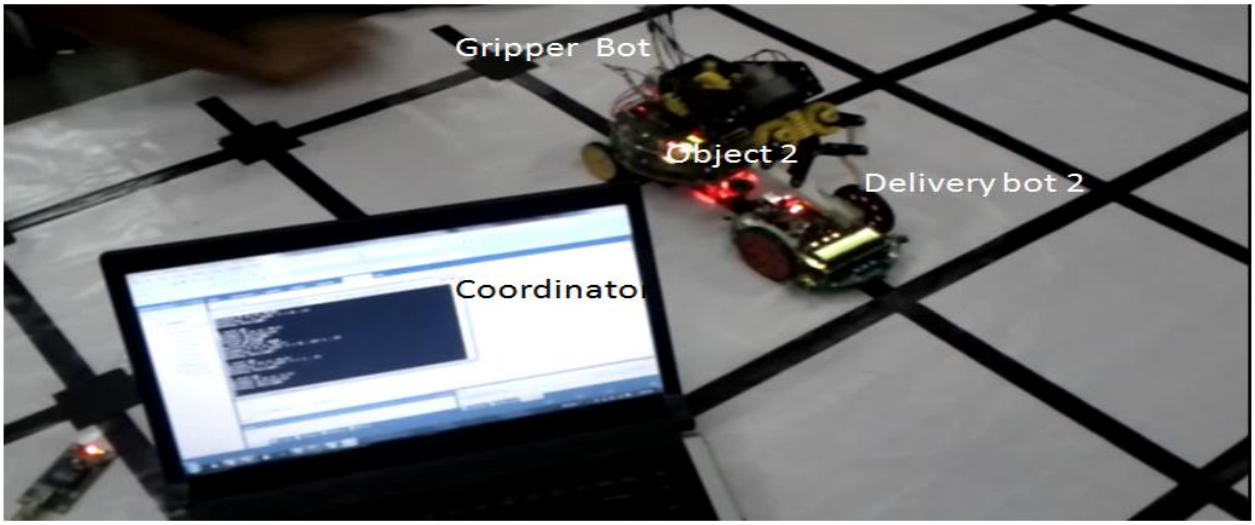
Following coordinators instructions, the gripper bot picks up the Red object (object 2). At the same time, the RFid scanner, fixed on the gripper arm, scans the RFid of the object and sends it to the coordinator.

Gripping of object and scanning the RFID tag

Stage 3

On receiving the RFid tag, the coordinator instructs the gripping bot to place the Red Object(object 2) on delivery bot 2.

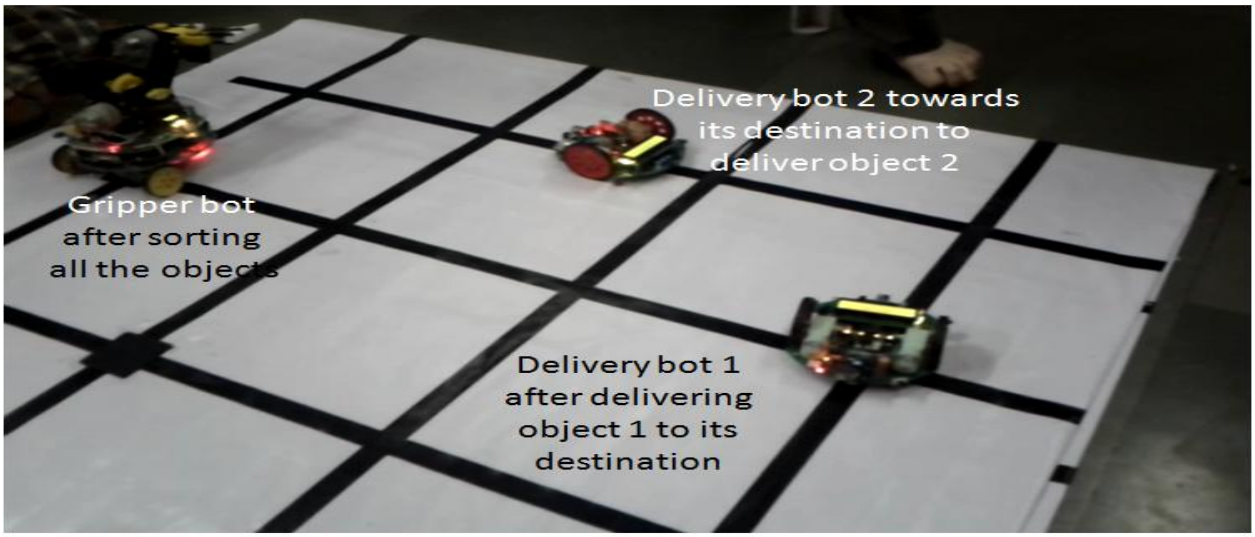
As per the instruction, the Red Object is placed.



The releasing mechanism

Stage 4

The delivery bot is now instructed by the coordinator to carry the Red Object to the red marked region on the grid. Delivery bot 2 delivers the Red Object to that place.



The delivering mechanism

6. Discussion of System

A. What all worked as per planned?

- Zigbee Communication:
    - Successfully able to communicate between multiple bots and the coordinator using zegbee modules. We defined our own protocol to communicate with all bots using the same zegbee on the coordinator. There is one zegbee, connected with the coordinator which is in broadcast mode and all other zigbees which are on the bots, are in unicast mode.
- Path Planning/Optimization and collision avoidance:
    - Successfully able to decide a optimal path for a bot to move on the grid without colliding with other bots or any other obstacle.
- Delivery Bot mechanism:
    - The delivery bots are programmed to carry objects to their final destination as per instructed by the coordinator.
- RFID identification and sorting of objects:
    - The RFID tag attached to a object is unique and hence it is the identifier of the object. The coordinator sorts the objects based on their RFID tag value.

B. What we added more than discussed in the SRS?

- RFID Scanner on Firebird V (ATMega2560):
    - We were successful in adding a new feather in the firebird V robot. We added RFID scanner on the firebird itself. This has been done for the first time in any of the projects. The scanner can scan any RFid tag which is in its proximity of about 5-7 cm.
- Gripper Arm:
    - We added a gripper arm to the firebird V which has two DC motors, one for vertical motion and one for horizontal motion. The task of the gripper arm is to lift or drop an object as instructed by the coordinator.

C. Changed made in plan:

- Robotic arm:
    - We didn't use the robotic arm to sort the objects as written in the SRS.

  Reason:
    - The arm due to its non mobility restricted the area in which the objects can be searched which are to be placed in their correct locations.
    - The Gripper bot had a benefit of having mobility and hence not getting restricted in one particular region.

7. Future Work

- Android apps can be developed using which users can specify the objects they want to try. The delivery bots can get all the objects for the user.
- A Robotic Arm can be used instead of the gripper to pick up objects with better precision.
- Cameras can be installed on the delivery bots so that they do not require a coordinator for the movements avoiding the problem of single point failure.
- Bots to unload objects from the delivery bot and stack them up.

8. Conclusion

Our project was successful in demonstrating that generic tasks like sorting and delivering objects can be automated by using well coordinated systems.

In our case, we were able able to identify the objects, determine the locations they should have been at and subsequently, place them at their appropriate places.

This is precisely what is needed at places like shopping malls where objects (like clothes) that have been tried by people need to replaced at their original positions. In gereral, such automated systems can be extended to simplify any task invloving sorting and delivery (eg. garbage segregation etc.)

9. References

- Serial Communications in Win32
  - (http://msdn.microsoft.com/en-us/library/ms810467.aspx)
- X-CTU Configuration & Test Utility Software
  - (http://ftp1.digi.com/support/documentation/90001003_A.pdf)
- E-Yantra WebSite
  - (http://www.e-yantra.org/)
- ATMega2560 datasheet
- Fire Bird V ATMEGA2560 Hardware Manual 2010-12-21.pdf
- Fire Bird V ATMEGA2560 Software Manual 2010-12-27.pdf
- Zigbee Wireless
  - (http://www.kanda.com/zigbee-wireless.html)
- RFID Reciver and Tags
  - (http://www.acs.com.hk)
  - (http://en.wikipedia.org/wiki/Radio-frequency_identification)