Course Project Documentation

CS101 Project

SPYBOT

TEAM ID 424

Ritwick Chaudhry 14D070063
Ritvik Mathur 14D070018
Aman Dixit 14D070053
Preshit Wazalwar 140040003

Table Of Contents

1. Introduction

This project is based on a spying robot, which can provide visual feed to the user via camera of a remote current location of the bot, and also return back automatically to the initial location.

This device saves the user the trouble to keep track of the path followed by the bot, as it will provide the coordinates of its current position. It prevents the user from the hassles of retracing the bot back to the initial position, as it can return by itself. It has accident protection system, namely the BCAS, which prevents accidents with obstacles.

2. Problem Statement

The AIM of our project is to create a spybot which can travel as commands are given and retreat whenever necessary.

1. On giving commands, the bot should move accordingly and also keep track of its coordinates with respect to the starting point.
2. There should be a camera mounted on the bot, which is able to give live feed of the current position of the bot.
3. The bot should override the user's commands in case of an obstacle ahead and also inform the user about.
4. On specifying retreat command, the bot should automatically retreat to the starting position, avoiding all obstacles in the way and reducing the path length.

3. Requirements

A) Hardware Requirements

1. **FireBird V**: Bot for spying purposes
2. **JMK WS-309AS** : To get live feed of the bot's location
3. **Zigbee** : To maintain communication between attacker and system
4. **Laptop** : For controlling the bot for sending to a remote location

B) Software Requirements

1. **AVR Studio 6** : To program instruction onto a given bot
2. **AVR Bootloader** : To load the program in the bot
3. **X-CTU** : To control the bot via laptop, this software, this software is used to give direct commands to the bot
4. **Ulead VideoStudio SE** : Software for getting feed from the camera

4. Implementation

1. Commands are given wirelessly to the bot though XBee module
2. While manoeuvring, the bot keeps track of its current direction and also its current coordinates.
3. The bot has a mounted camera which provides live feed of the terrain to the user, helping him/her navigate the bot.
4. BCAS (Bot Collision Avoidance System) is also incorporated which prevents the bot from colliding with obstacles while manual control (by overriding the user's input) and also while retreating.
5. ARA (Auto Return Algorithm) is used to recall the bot back to the starting position. It uses the coordinate tracking system to get the current position of the bot and use the BCAS to avoid obstacles.
6. Also the ARA tries to reduce the return path length (which saves power) and also ensures no physical impact to the bot.

5. Testing Strategy and Data

1. Testing the Xbee wireless connection with the bot.
2. Testing the Coordinate Tracking System by checking x-coordinate, y-coordinate and angle with respect to origin.
3. Testing the BCAS Part 1, by placing an obstacle in the path of the bot, and then again giving forward commands.
4. Testing the ARA for test cases, noting the final coordinates.
5. Calibrating the bot by changing the PWM of the motor, and changing the convert functions accordingly.
6. Testing the BCAS Part 2, by placing an object in path of the bot during retreat mechanism, and calibrating the angle and distance of the off-course path.
7. Testing the camera, using Ulead software and its wireless tuner.
8. Testing the final code with all integrated features, and final calibrations.

6. Discussion of System

A) Implementation According to Initial Plan

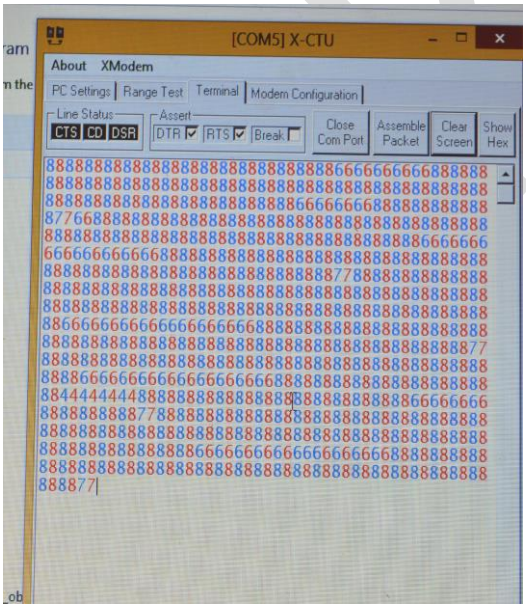
The Coordinate Tracking System, ARA and the BCAS have been implemented as discussed, according to our initial plan.

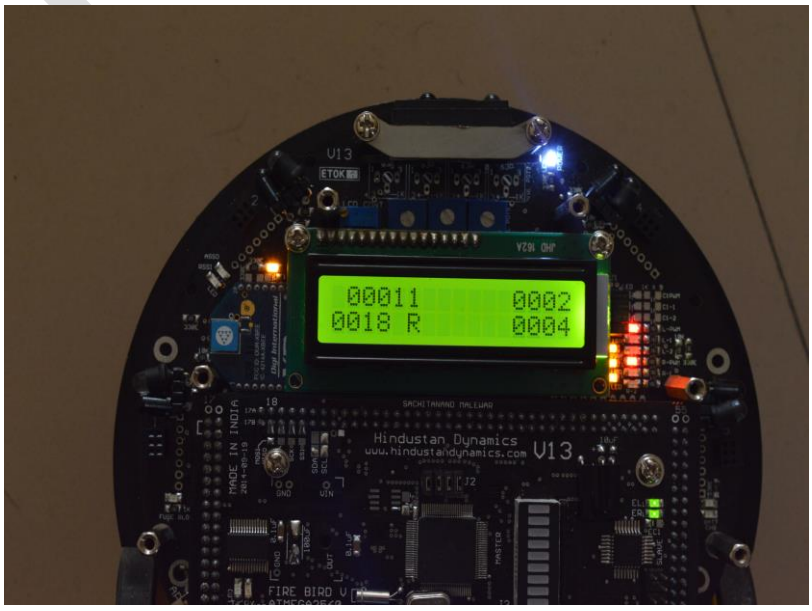
B) Extra Inputs with respect to Initial Plan

Installation of Camera on the bot for Video Feed of the unknown area.

C) Changes in Initial Plan

Initially, we planned to control our bot by hand gesture motion sensor. For that purpose, we were provided a WiCed Sense device, which had an in-built accelerometer. However, due to unavailability of serial inputs of the device, we were unable to transfer its values to our bot. Hence we were unable to make a motion controller. So we gave the controlling commands by the keyboard, though wireless Xbee interface.





7. Future Work

This project has various prospects of future improvements.

1. It can be enhanced in connectivity by satellite communications for military use, as it can be used to scan enemy territories, scout an area for explosives using explosive detectors, etc.
2. It can be used to make an automated goods transfer system, for use in factory, where goods can be transferred easily to a distant location and return back.
3. We can incorporate multiple proximity sensors for obstacle detection, thus making BCAS more accurate.

8. Conclusions

This project is useful for spying purposes, in both small and large scale use, or it can be used in a goods transfer mechanism. There are many other uses in industry for this type of device, such as mobile surveillance camera, movable repairing bot, etc.

9. References:

1) E-yantra website http://www.e-yantra.org
2) FireBird V Atmega2560 Robotic Research Platform Hardware
manual.IIT Bombay and NEX Robotics Pvt. Ltd.
3) FireBird V Atmega2560 Robotic Research Platform Software manual.IIT Bombay and NEX Robotics Pvt. Ltd.
4) Exploring Unseen Arena By E-Yantra Team : https://github.com/eyantra/CS684_Exploring-unseen-arena-using--swarm-robots_2011
5) Gesture Controlled Bot By E-Yantra Team https://github.com/eyantra/CS684_Gesture-Controlled-Robot_2011

Our project videos:

1) Main Working Project Video https://www.youtube.com/watch?v=qSJGZCg6S-o&feature=share

2) Screen Cast Video for Installing Softwares https://www.youtube.com/watch?v=ebYXcuHAF2s&feature=share