NEW YORK UNIVERSITY

DEPARTMENT OF ELECTRICAL AND COMPUTER ENGINEERING

REAL TIME EMBEDDED SYSTEMS

EL-GY 6483

PROJECT REPORT

STUDENT NAME AND ID :

AKSHTA SURESH - N11857317

POOJA GUPTA - N17756215

Contents

1 GOAL OF THE PROJECT

The goal of the project was to design a robot car that would move towards a wifi beacon avoiding the obstacles and stop at the beacon.

2 INTRODUCTION

An embedded system is an electronic system that are designed to perform a dedicated function within a larger system. Real-time systems are those that can provide guaranteed worst-case response times to critical events, as well as acceptable average-case response times to noncritical events. When a real-time system is designed as an embedded component, it is called a real-time embedded system.

3 FEATURES

1.Can move to the left.
2.Can move to the right.
3.Can move front.
4.Can move backwards.
5.Can avoid obstacles.
6.Can reach an access point by using Wi-fi for communication.

4 TOOL'S DESCRIPTION

i.Adafruit HUZZAH ESP8266

The Adafruit HUZZAH ESP8266 is a microcontroller board based on the ESP8266. The ESP8266 processor from Espressif is an 80 MHz microcontroller with a full WiFi front-end (both as client and access point) and TCP/IP stack with DNS support as well. It has the following pins:
Power Pins:
1.GND - this is the common ground for all power and logic
2.BAT - this is the positive voltage to/from the JST jack for the optional Lipoly battery
3.USB - this is the positive voltage to/from the micro USB jack if connected
4.EN - this is the 3.3V regulator's enable pin. It's pulled up, so connect to ground to disable the 3.3V regulator 3V - this is the output from the 3.3V regulator, it can supply 500mA peak

Logic Pins:

This is the general purpose I/O pin set for the microcontroller. All logic is 3.3V
Serial Pins:
RX and TX are the serial control and bootloading pins, and are how you will spend most of your time communicating with the ESP module.
The TX pin is the output from the module and is 3.3V logic.
The RX pin is the input into the module and is 5V compliant (there is a level shifter on this pin)
These are connected through to the CP2104 USB-to-Serial converter so they should not be connected to or used unless you're super sure you want to because you will also be getting the USB traffic on these!

SPI and I2C Pins:

In theory you can use any pins for I2C and SPI but to make it easier for people using existing Arduino code, libraries, sketches we set up the following:
I2C SDA = GPIO 4 (default)
I2C SCL = GPIO 5 (default)

If you want, you can connect to I2C devices using other 2 pins in the Arduino IDE, by calling Wire.pins(sda, scl) before any other Wire code is called (so, do this at the begining of setup() for example
Likewise, you can use SPI on any pins but if you end up using 'hardware SPI' you will want to use the following:
SPI SCK = GPIO 14 (default)
SPI MOSI = GPIO 13 (default)
SPI MISO = GPIO 12 (default)

GPIO Pins:

This breakout has 9 GPIO: 0, 2, 4, 5, 12, 13, 14, 15, 16arranged at the top edge of the Feather PCB
All GPIO are 3.3V logic level in and out, and arenot 5V compatible.
The maximum current drawn per pin is 12mA. These pins are general purpose and can be used for any sort of input or output. Most also have the ability to turn on an internal pullup. Many have special functionality:
GPIO 0, which does not have an internal pullup, and is also connected a red LED. This pin is used by the ESP8266 to determine when to boot into the bootloader. If the pin is held low during power-up it will start bootloading! That said, you can always use it as an output, and blink the red LED - note the LED is reverse wired so setting the pin LOW will turn the LED on.
GPIO 2, is also used to detect boot-mode. It also is connected to the blue LED that is near the WiFi antenna. It has a pullup resistor connected to it, and you can use it as any output (like 0) and blink the blue LED.
GPIO 15, is also used to detect boot-mode. It has a pulldown resistor connected to it, make sure this pin isn't pulled high on startup. You can always just use it as an output
GPIO 16 can be used to wake up out of deep-sleep mode, you'll need to connect it to the RESET pin.

Analog Pins:

There is also a single analog input pin called A. This pin has a 1.0V maximum voltage, so if you have an analog voltage you want to read that is higher, it will have to be divided down to 0 - 1.0V range

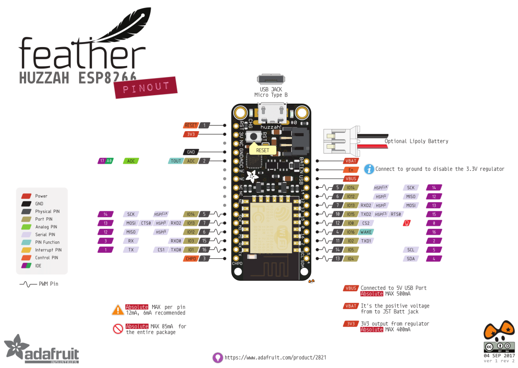
Other control pins:

RST - this is the reset pin for the ESP8266, pulled high by default. When pulled down to ground momentarily it will reset the ESP8266 system. This pin is 3.3V logic only

EN (CH$_PD) - ThisistheenablepinfortheESP8266, pulledhighbydefault. Whenpulleddowntogroundm$

NC Pins:

The rest of the pins are labeled NC which means Not Connected - they are not connected to anything and are there as placeholders only, to maintain physical compatibility with the other boards in the Feather line!

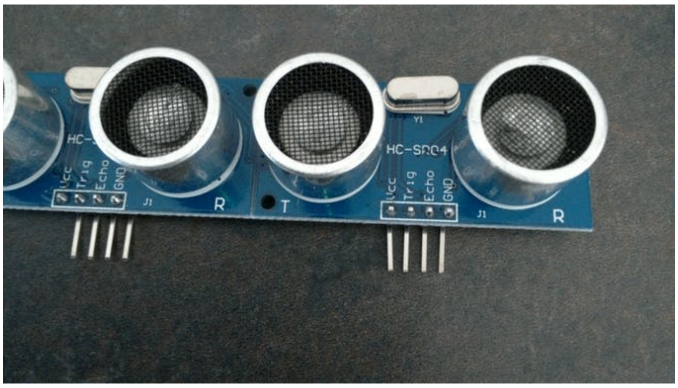


Adafruit Feather Huzzah Pinout

ii.Ultrasonic Ping Sensor

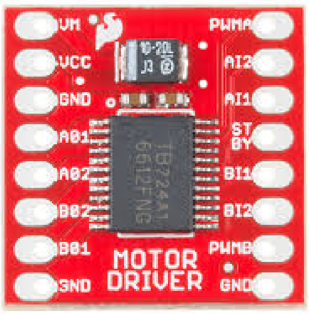
This is a very simple and useful ultrasonic sensor. There are four pins that you would use to interface with the sensor: VCC, Trig (signal output pin), Echo (signal input pin), and GND.
Each of the four pins are connected to the Arduino: VCC to 5v, Trig to a digital pin, Echo to a digital pin, and GND to GND (ground). No resistors are needed, just pin to port.
The sensor outputs a muted sound wave out of its speaker (one of the silver barrels) that will bounce off an object and be reflected back to the sensor's microphone (the other silver barrel). After the sound wave is sent out, the sensor will time how long it takes for the reflected sound to get back to the ultrasonic sensor. The operation is kind of like echolocation. There are many quick math equations you can use to

convert the pulse time into a unit of measurement (note that these results may not be very accurate).



iii.TB6612 Driver

The TB6612FNG is an easy and affordable way to control motors. The TB6612FNG is capable of driving two motors at up to 1.2A of constant current. Inside the IC, you'll find two standard H-bridges on a chip allowing you to not only control the direction and speed of your motors but also stop and brake. This guide will cover in detail how to use the TB6612FNG breakout board. The library for this guide will also work on the RedBot Mainboard as well since it uses the same motor driver chip.



iV.DC Motor and Wheel

DC Gear Motor and Wheel. Our DC gear motor and wheel set for making robots! These motors are light weight, high torque and low RPM. They can climb hills and have excellent traction, plus you can mount the wheel on either side of the motor with its double-sided output shaft.

V.Mini Breadboard

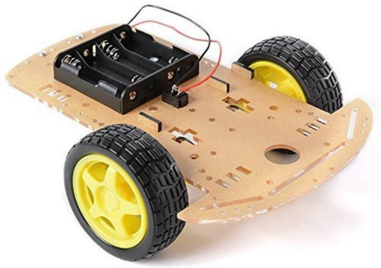
Anatomy of a Mini Breadboard. ... These tie points take the form of holes within the breadboard, into which wires and components can be pushed. They're useful for basic prototyping, but breadboards don't accommodate anything with two closely spaced rows of pins, such as the header on the Raspberry Pi.

Vi.Jumper Wires

A jump wire (also known as jumper wire, or jumper) is an electrical wire, or group of them in a cable, with a connector or pin at each end (or sometimes without them – simply "tinned"), which is normally used to interconnect the components of a breadboard or other prototype or test circuit, internally or with other equipment or components, without soldering

Vii.Car Chassis Kit

Size: 22 x 12cm (L x W). Wheel Size: 6.5cm (Dia.) x 2.7cm (H) Mechanical structure is simple, it is easy to install. Can be used for distance measurement, velocity Can use with other devices to realize function of tracing, obstacle avoidance, distance testing, speed testing, wireless remote control. Gear Motor reduction radio: 1:48.



Viii.18650 Battery Holder

18650 Battery Holder - Wire - 2 Cell. This 18650 battery case holds two (2) 18650 lithium-ion batteries connected in series, providing approximately 7V output (3.7V x 2). The holder includes output leads approximately 6" long for soldering or connecting to your circuit.

ix.18650 3.7v Li-Poly Battery

3.7V 2500 mAh Li-poly Rechargeable Battery for powering the Node MCU.

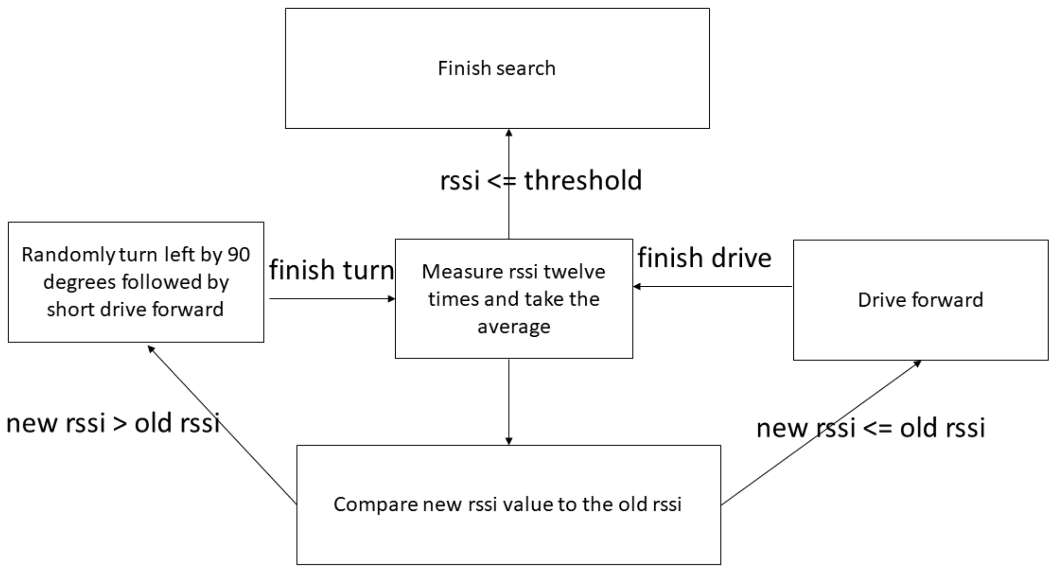
x.Roller Bovine Wheel

Overview of Wheels Used in Robotics. The wheel is the most common moving element among other possibilities including legs, flying, swimming and rolling. A wheel provides at least speed, accuracy and stability for a robot, three characteristics very important in designing and build robots.

5 WORKING PROCESS AND ALGORITHM

In this project, a robot is wifi-enabled to reach an access point by avoiding the obstacles. The working of the robot is explained here.

The algorithm for deciding the path to follow is based on gradient descent. At the start, the robot makes a 360 degree turn to figure out the maximum RSSI strength and then starts moving in the direction of maximum strength by avoiding the obstacles. The RSSI values are measured 12 times(360 degree=30*12) and its average calculated. Then, the robot is made to turn left by 90 degrees when the newly calculated RSSI is greater than the previously calculated value. Otherwise, the robot continues to move forward in the same direction. As soon as the robot comes within a threshold value of the RSSI(which means that it has reached closer to the beacon), it stops.



The algorithm for obstacle avoidance is by using the ultrasonic ping sensor. Each sensor has a threshold value and the robot will avoid the obstacles when the obstacle is within the set threshold value.

6 RESULT

We achieved our objective, which was to reach the access beacon point using Wi-fi and in the path that the robot followed, the obstacles were also avoided.