Final Project Device Proposal

# Summarized Proposal

The device that has been chosen to improve performance in the competition is the Ultrasonic Module HC-S404 Distance Sensor. This sensor will aid in controlling which path on the track is taken by the robot. The distance sensor will allow for the robot to scan the field in front of it and detect if there is another robot on the path. When the robot comes to a corner where it needs to make a decision on which direction to go, the distance sensor will aid in making that decision. The robot detects that it has reached the end of a path it will spin until a black line is found. Once that black line is found the distance sensor will send out a pulse to detect if there is another robot on that path. If another robot is found on the path, our robot will continue to spin until a new black line is found. This process will be repeated until a black line is found that contains no other robots, in which case our robot will move forward down that path in order to earn points. This strategy is a purely defensive one and should allow for the maximum number of points to be earned while avoiding potential collisions.

# Part Details

|  |  |
| --- | --- |
| Device Name | Ultrasonic Module HC-SR04 Distance Sensor |
| Product Number | SN-HC-SR04 |
| Price | $5.99 |
| Distributor | Sunfounder |
| Communications Interface | N/A |
| Hardware Interface | 4 Pins, Vcc, Echo Trig, GND  Mounted onto a vector board |



# Summarized Use

The operation of the Ultrasonic sensor is fairly simple. Four pins from the sensor are connected to the microcontroller (Vcc, Trig, Echo, GND). The sensor works by sending a pulse of high voltage to the Trig pin for 10us, this will initiate the sensor to transmit out 8 cycles of ultrasonic burst at 40kHz and wait for the reflected ultrasonic burst. When the sensor detects ultrasonic burst from the receiver, It will set the Echo pin to high voltage for a certain period of time proportional to the distance. The distance the sensor is from the object can then be calculated based on the time. If no object is found the Echo pin will be set to high voltage for 38ms. In order to find the time that the Echo pin is high a timer will be started once the Echo pin first goes high. A while loop will then be started and will end once the Echo pin is low. Once the while loop exits the timer will stop and the time will be read and used for calculating the distance. Example software for the device is shown below.

#include "Arduino.h"

#include "Ultrasonic.h"

Ultrasonic::Ultrasonic(int TP, int EP)

{

pinMode(TP,OUTPUT);

pinMode(EP,INPUT);

Trig\_pin=TP;

Echo\_pin=EP;

}

long Ultrasonic::Timing()

{

digitalWrite(Trig\_pin, LOW);

delayMicroseconds(2);

digitalWrite(Trig\_pin, HIGH);

delayMicroseconds(10);

digitalWrite(Trig\_pin, LOW);

duration = pulseIn(Echo\_pin,HIGH);

return duration;

}

long Ultrasonic::Ranging(int sys)

{

Timing();

distacne\_cm = duration /29 / 2 ;

distance\_inc = duration / 74 / 2;

if (sys)

return distacne\_cm;

else

return distance\_inc;

}

/\*

Ultrasonic.h - Library for HR-SC04 Ultrasonic Ranging Module.

Created by ITead studio. Alex, Apr 20, 2010.

iteadstudio.com

\*/

#ifndef Ultrasonic\_h

#define Ultrasonic\_h

#include "Arduino.h"

#define CM 1

#define INC 0

class Ultrasonic

{

public:

Ultrasonic(int TP, int EP);

long Timing();

long Ranging(int sys);

private:

int Trig\_pin;

int Echo\_pin;

long duration,distacne\_cm,distance\_inc;

};

#endif

# Proposed Tasks

Task 1: The first task that will be performed with this device is to output the distance being read from the sensor onto the LCD. In order to do this the value being read from the sensor will be converted into a distance using the following formula: (Duration of high level)\*(340)/2, which will give us the distance in cm. Once this value has been read in and converted it will be passed into a function that prints values onto the LCD. We can compare this value with the actual distance measured from the sensor to the object.

Task 2: The second task will be to interface the distance sensor with motors. The objective of this task is to use the distance sensor in order to control the direction of the robot. The distance sensor will sense weather there is an object in the direct path of the robot. Using this information the motors can be adjusted in order to spin the robot until it finds a clear path and then continue forward.

Task 3: The final task is to interface the distance sensor, motors and the IR sensor. The purpose of this task is to keep the robot on the black lines during the competition whilst avoiding other robots. The distance sensor will be used in order to scan the field in front of the robot when the robot detects that it has come to a corner of the track, it will spin and scan the field in front of it. In order for the robot to continue moving forward it will need to find a black line that contains no other robot on it. In terms of software, the distance sensor will be returning NULL and the IR sensor will detect black.

