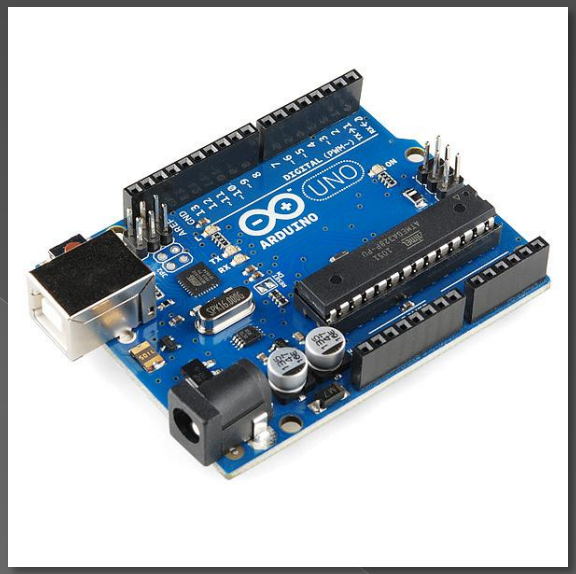
Sensor Data 1

(Temp)

Sensor Data 2

(Light)



Sensor Data 4

(Alarm)

Sensor Data 3

(Motion)

- Light-weight server

- php

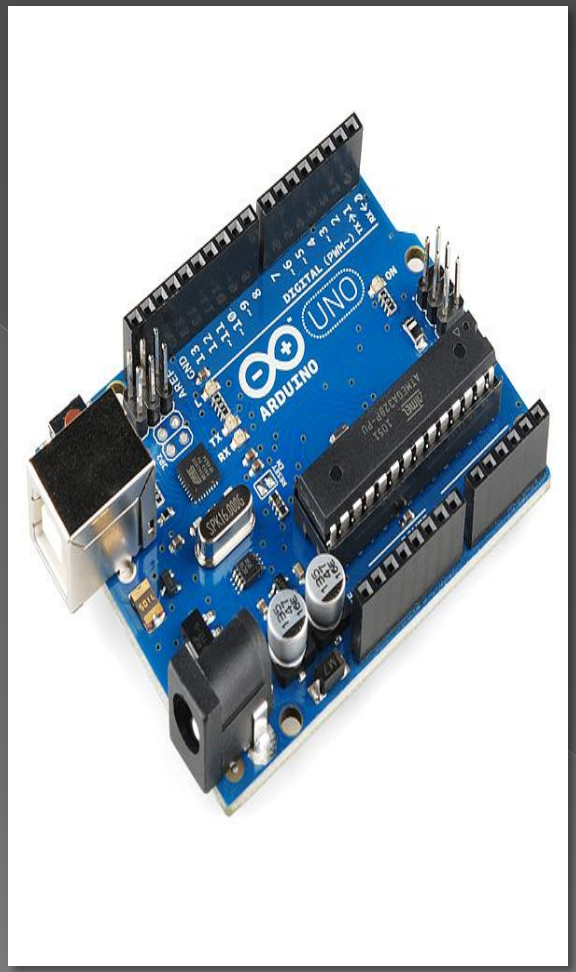
- Data Processing

- Responsive Web App handling

Web Application

- Most recent data
- Data history

Temp

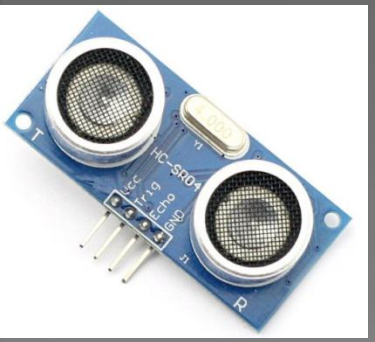


Motor / LED

Turn LEDs of certain rooms on and off through the Raspb Pi or have a Motor operate a task.

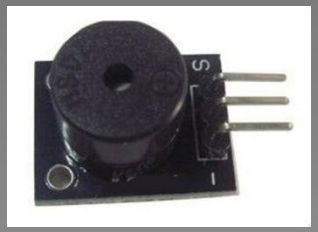
Alarm

Motion

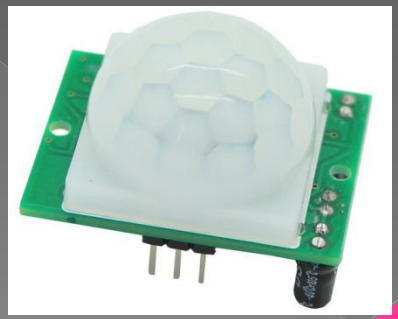


HC-SR04

+



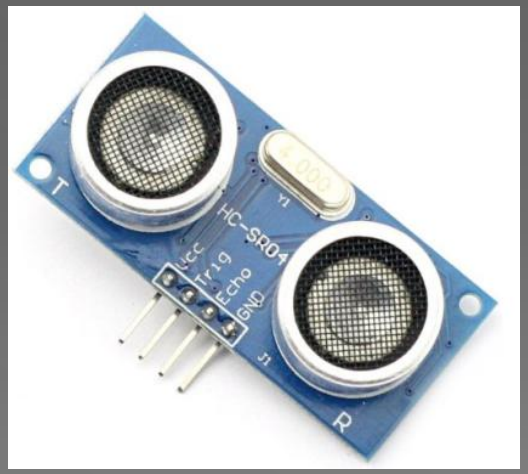
Buzzer

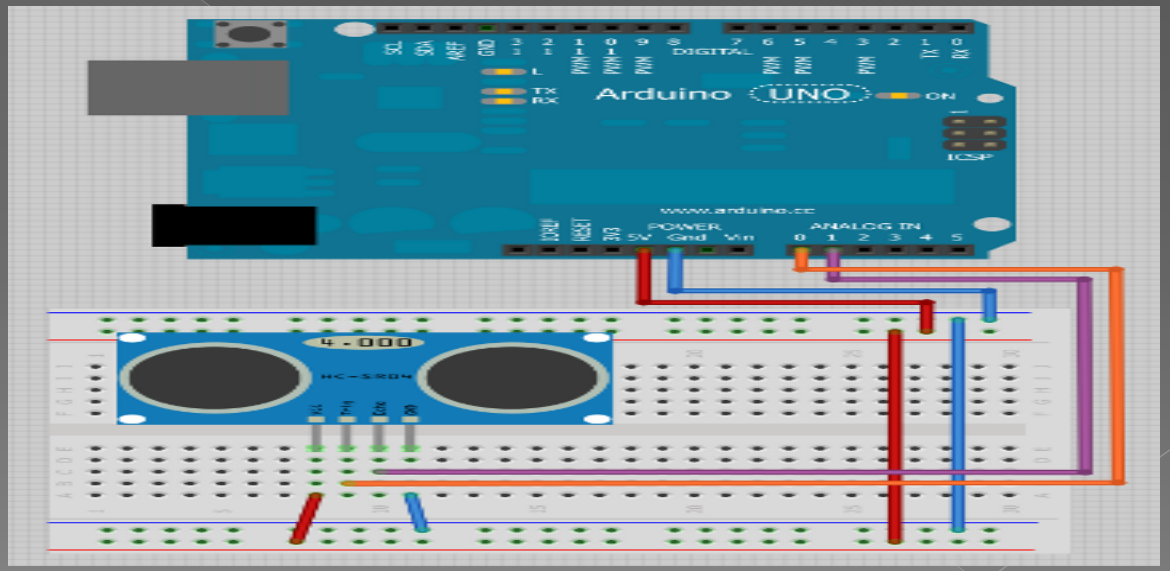


HC-SR501

Alarm Sensor (Distance Sensor)

- Goal : Alarm will go off when a certain pre-defined distance limit is passed.





Distance_Sensor_Program | Arduino 1.0.5

File Edit Sketch Tools Help

Distance_Sensor_Program §

```
#include <NewPing.h>

#define Dis_Pin 14
#define ECHO_PIN     15
#define MAX_DISTANCE 200

NewPing DistanceSensor(Dis_Pin, ECHO_PIN, MAX_DISTANCE);
int LED = 13;

void setup()
{
    Serial.begin(9600);
    pinMode(LED,OUTPUT);
}

void loop()
{
    unsigned int cm = DistanceSensor.ping_cm();  //DISTANCE IN CM
    if ( cm <= 4 )
    {
      digitalWrite(LED, HIGH);  // LED(ALARM) turns on

    }
    else
    {
    digitalWrite(LED, LOW); // LED(ALARM) turns off
    }
    Serial.print("Distance: ");
    Serial.print(cm);
    Serial.println("cm");
    delay(1000);                //gives you the reading of the Distance sensor every sec
}
```

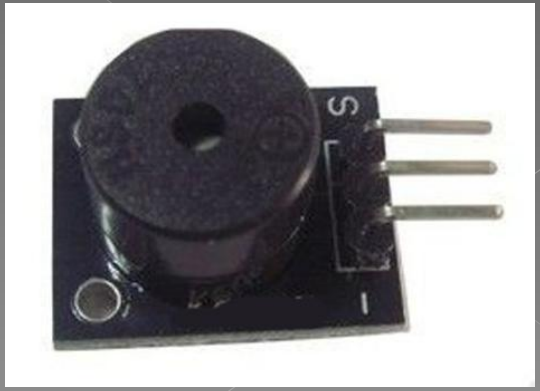
Done compiling.

```
Binary sketch size: 3,672 bytes (of a 32,256 byte maximum)
```

13 Arduino Uno on COM3

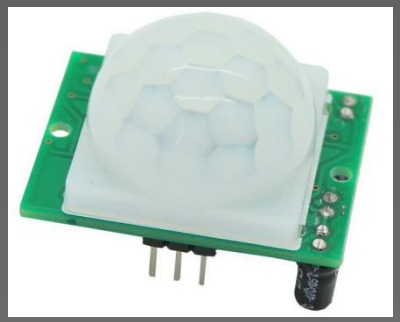
Distance Program

- Pre-defined limit = 4cm
- LED turns on Pin 13's LED
- Serial Monitor Controls data. In the future would like to send data to Raspb Pi
- Raspb Pi records Distance Data and when Alarm is turned on.

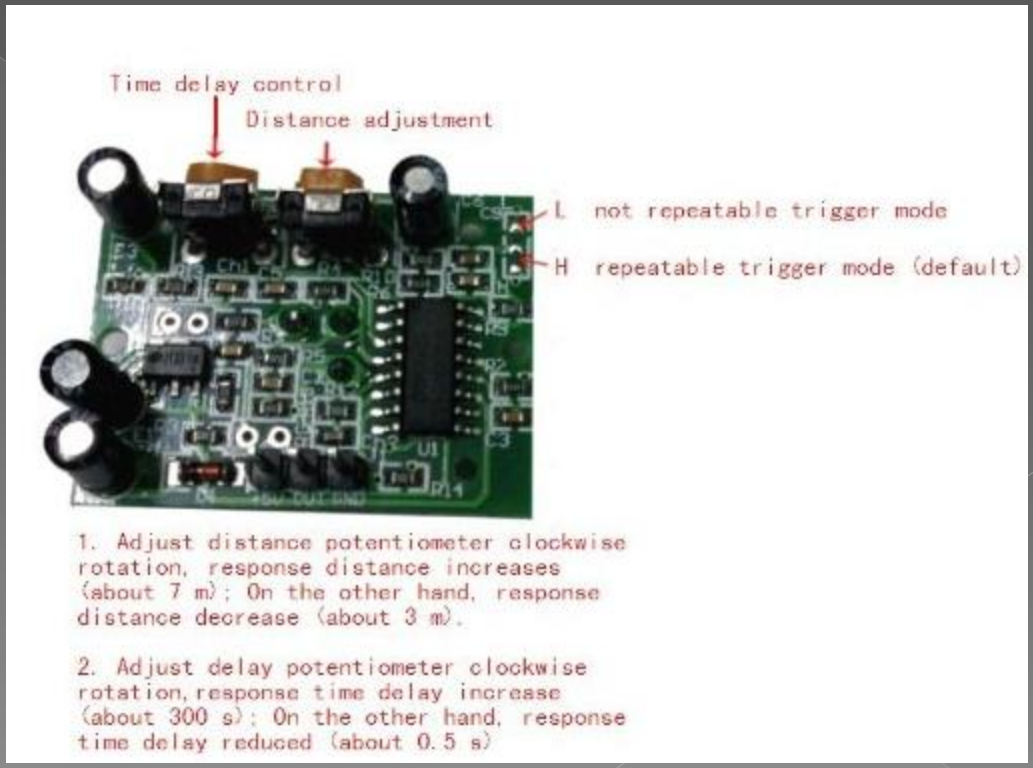


Motion Detector

- Goal : HC-SR501 will illuminate an LED when a certain limit or lower is passed.



- Time : how long the LED will stay high
- Distance : how far the HC-SR501 will read in a signal for the LED to turn on.



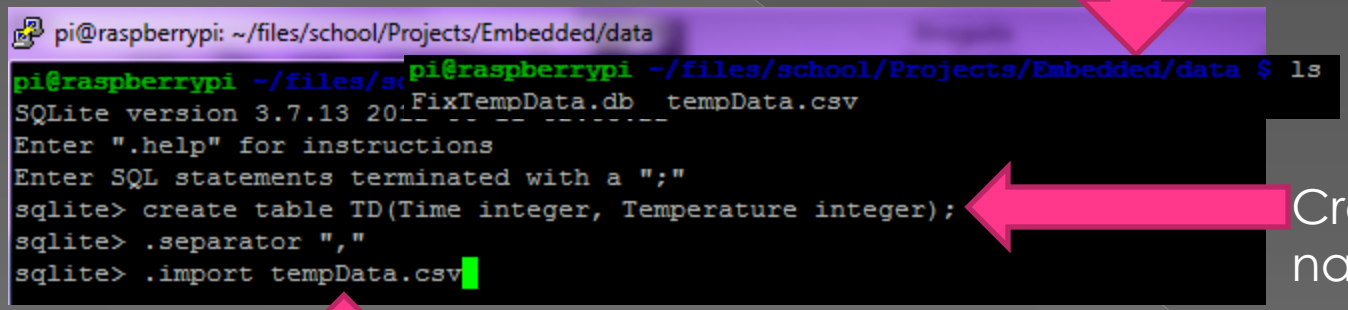
Sqlite3

Setting up database on Raspberry Pi

- Sqlite doesn't have database functions
- downloading packages only through desktop

SQLITE3- How to create a DB

DB name



Creates Table for DB named TD with columns Time and Temperature both of type int

Imports data from tempData.csv file

Checking DB

```
pi@raspberrypi ~/files/school/Projects/Embedded/data $ ls
FixTempData.db  tempData.csv
```

Figure 1

```
sqlite> select * from TD ;
sqlite>
```

Figure 2

Figure 1 shows that the db was created but figure 2 shows that there is no ‘data’ in the database.

Errors received

- Err http://ec2-us-east-mirror.rightscale.com jaunty-updates/main libsqlite3-dev 3.6.10-1ubuntu0.2404 Not Found [IP: 174.129.253.100 80]Err http://ec2-us-east-mirror1.rightscale.com jaunty-updates/main libsqlite3-dev 3.6.10-1ubuntu0.2404 Not FoundErr http://ec2-us-east-mirror2.rightscale.com jaunty-updates/main libsqlite3-dev 3.6.10-1ubuntu0.2404 Not FoundErr http://ec2-us-east-mirror3.rightscale.com jaunty-updates/main libsqlite3-dev 3.6.10-1ubuntu0.2404 Not FoundFailed to fetch http://ec2-us-east-mirror3.rightscale.com/ubuntu/pool/main/s/sqlite3/libsqlite3-dev_3.6.10-1ubuntu0.2_i386.deb 404 Not FoundE: Unable to fetch some archives, maybe run apt-get update or try with --fix-missing?

- file is encrypted or is not a database

Solutions

- Have to use desktop GUI when internet is needed
- Database file being imported have to be in the same directory as database created
- Database from the same name even in previous versions of sqlite cannot exist.