RECIEVER

#define datain A0 #define ledPin 12 unsigned int temp = 0; const unsigned int upperThreshold = 600; const unsigned int lowerThreshold = 50; void setup() { pinMode(ledPin, OUTPUT); } void loop(){ temp=analogRead(datain); if(temp<lowerThreshold) { digitalWrite(ledPin, LOW); } else if(temp>upperThreshold) { digitalWrite(ledPin, LOW); } }

TRANSMITTER

#include <VirtualWire.h>const int ledPin = 7;char \*data;void setup() { pinMode(ledPin,OUTPUT); vw\_set\_ptt\_inverted(true); vw\_set\_tx\_pin(12); vw\_setup(4000);}void loop(){ data="1"; vw\_send((uint8\_t \*)data, strlen(data)); vw\_wait\_tx(); digitalWrite(ledPin,HIGH); delay(2000); data="0"; vw\_send((uint8\_t \*)data, strlen(data)); vw\_wait\_tx(); digitalWrite(ledPin,LOW); delay(2000);}/\*\*/void setup() { }void loop() { }

TRAFFIC COTROL

//Tutorial by RoboIndia on Traffic Light Controller

//https://www.roboindia.com/tutorials/

int RED1 = 4; //lane-1

int YELLOW1 = 5;

int GREEN1 = 6;

int RED2 = 7; //lane-2

int YELLOW2 = 8;

int GREEN2 = 9;

int sensepin=10;

#include <NewPing.h>

#define TRIGGER\_PIN 3 //Trig

#define ECHO\_PIN 2 //Echo

#define MAX\_DIST 200 //200 centimeter

NewPing sonar(TRIGGER\_PIN, ECHO\_PIN, MAX\_DIST);

void setup()

{ Serial.begin(115200);

pinMode(TRIGGER\_PIN, OUTPUT);

pinMode(ECHO\_PIN, INPUT);

pinMode(RED1, OUTPUT);

pinMode(YELLOW1, OUTPUT);

pinMode(GREEN1, OUTPUT);

pinMode(RED2, OUTPUT);

pinMode(YELLOW2, OUTPUT);

pinMode(GREEN2, OUTPUT);

pinMode(sensepin, OUTPUT);

}

void loop()

{ delay(50); //50 millisecond delay

unsigned int ultrSonic = sonar.ping();

if ((ultrSonic / US\_ROUNDTRIP\_CM) < 10) {

Serial.println("SENSED");

digitalWrite(sensepin, HIGH);

digitalWrite(GREEN2, HIGH);

delay(7000);

}

//turn opposite red and green on

digitalWrite(RED1, HIGH);

digitalWrite(GREEN2, HIGH);

delay(5000);

//turn both yellow lights on, indicates green1 and red2 is going to be on

digitalWrite(RED1, HIGH);

digitalWrite(YELLOW1, HIGH);

digitalWrite(YELLOW2, HIGH);

digitalWrite(GREEN2, LOW);

delay(3000);

//turn opposite red and green on

digitalWrite(RED1, LOW);

digitalWrite(YELLOW1, LOW);

digitalWrite(YELLOW2, LOW);

digitalWrite(RED2, HIGH);

digitalWrite(GREEN1, HIGH);

delay(5000);

//turn both yellow lights on, indicates green2 and red1 is going to be on

digitalWrite(GREEN1, LOW);

digitalWrite(YELLOW1, HIGH);

digitalWrite(YELLOW2, HIGH );

digitalWrite(RED2, HIGH);

delay(2000);

// turn opposite green and red on

digitalWrite(YELLOW1, LOW);

digitalWrite(RED1, HIGH);

digitalWrite(RED2, LOW);

digitalWrite(YELLOW2, LOW);

digitalWrite(GREEN2, HIGH);

delay(5000);

}

NODEMCU FOR SENDING MESSAGE

#include <ESP8266WiFi.h>

#include <WiFiClient.h>

#include <ESP8266WebServer.h>

// Replace with your network credentials

const char\* ssid = "batman";

const char\* password = "batman123";

ESP8266WebServer server(80); //instantiate server at port 80 (http port)

String page = "";

String dataamb = "";

String datahsp = "";

void setup(void){

pinMode(D2, INPUT);

delay(1000);

Serial.begin(115200);

WiFi.begin(ssid, password); //begin WiFi connection

Serial.println("");

// Wait for connection

while (WiFi.status() != WL\_CONNECTED) {

delay(500);

Serial.print(".");

}

Serial.println("");

Serial.print("Connected to ");

Serial.println(ssid);

Serial.print("IP address: ");

Serial.println(WiFi.localIP());

server.on("/", [](){

page = "<h1>Ambulance Detector for Hospital X</h1><h3>Ambulance named:</h3> <h4>"+String(dataamb)+"</h4> <h3>has reached traffic Signal NAMED:</h3> <h4>"+String(datahsp)+"</h4>";

server.send(200, "text/html", page);

});

server.begin();

Serial.println("Web server started!");

}

void loop(void){

if(digitalRead(D2)==HIGH){

dataamb="Ambulance A";

datahsp= "Traffic Signal Y";

}

delay(1000);

server.handleClient();

}