**AUTOMATIC RAILWAY GATE CONTROL SYSTEM**

**CODE FOR RAILWAY GATE CONTROL**

#include < Servo .h>

Servo gate Servo;

cons int sensor1 = 2;

cons int sensor2 = 3;

cons int led Open = 8;

cons int led Close = 9;

void setup () {

pin Mode (sensor1, INPUT);

pin Mode (sensor2, INPUT);

pin Mode (led Open, OUTPUT);

pin Mode (led Close, OUTPUT);

gate Servo. attach (10);

gate Servo. write (90);

digital Write (led Open, HIGH);

digital Write (led, LOW);

Serial. begin (9600);}

void loop () {

int sensor1State = digital Read (sensor1);

int sensor2State = digital Read (sensor2);

if (sensor1State == HIGH) {

close Gate ();

}

if (sensor2State == HIGH) {

open Gate ();

Delay (100);

}

void close Gate () {

gate Servo. Write (0);

digital Write (led Open, LOW);

digital Write (led Close, HIGH);

Serial. print ("Gate Closed");

}

void open Gate () {

gate Servo. Write (90);

digital Write (led Open, HIGH);

digital Write (led Close, LOW);

Serial. print ("Gate Opened");

}

**Code for Automatic fruit juice system**

#include < Servo. h>

#include < Keypad. h>

cons byte ROWS = 4;

cons byte COLS = 4;

// Define the keymap

char keys [ROWS][COLS] = {

{'1', '2', '3', 'A'},

{'4', '5', '6', 'B'},

{'7', '8', '9', 'C'},

{'\*', '0', '#', 'D'}

};

byte row Pins [ROWS] = {9, 8, 7, 6};

byte col Pins [COLS] = {5, 4, 3, 2};

// Create the Keypad

Keypad key pad = Keypad (make Keymap(keys), row Pins, col Pins, ROWS, COLS);

Servo juiceServo1;

Servo juiceServo2;

Servo juiceServo3;

cons int trig Pin = 10;

cons int echo Pin = 11;

cons int led Pin = 12;

cons int buzzer Pin = 13;

void setup () {

Serial. Begin (9600);

// Attach servo motors to pins

juiceServo1.attach(A0);

juiceServo2.attach(A1);

juiceServo3.attach(A2);

juiceServo1.write(0);

juiceServo2.write(0);

juiceServo3.write(0);

// Setup the ultrasonic sensor pins

Pin Mode (trig Pin, OUTPUT);

Pin Mode (echo Pin, INPUT);

Pin Mode (led Pin, OUTPUT);

Pin Mode (buzzer Pin, OUTPUT);

Digital Write (led Pin, LOW);

Digital Write (buzzer Pin, LOW);

}

void loop () {

char key = keypad. Get Key ();

if (key) {

Serial. Print (key);

if (is Cup Present ()) {

switch (key) {

case '1':

dispense juice (juiceServo1);

break;

case '2':

dispense Juice (juiceServo2);

break;

case '3':dispense Juice (juiceServo3);

break;

default:

Serial. print ("Invalid selection");

break;

}

} else {

Serial. Print ("Place a cup!");

Digital Write (buzzer Pin, HIGH);

Delay (500)

Delay Microseconds (2);

Digital Write trig Pin, HIGH);

Delay Microseconds (10);

Digital Write (trig Pin, LOW);

long duration = pulse (echo Pin, HIGH);

long distance = (duration / 2) / 29.1; // Convert to cm

if (distance < 10) { // Adjust threshold as needed

digitalWrite(ledPin, HIGH);

return true;

} else {

digitalWrite(ledPin, LOW);

return false;

}

}

void dispense Juice (Servo & juice Servo) {

Serial. Print ("Dispensing Juice...");

Juice Servo. Write (90);

delay (3000);

juice Servo. write(0);

Serial. print("Dispensing Complete");

Delay (1000);

}