#include <Wire.h>

#include <LiquidCrystal\_I2C.h>

LiquidCrystal\_I2C lcd(0x27,16,2); //Change the HEX address

#include <Servo.h>

Servo myservo1;

int IR1 = 5;

int IR2 = 6;

int Slot = 4; //Enter Total number of parking Slots

int flag1 = 0;

int flag2 = 0;

void setup() {

lcd.begin();

lcd.backlight();

pinMode(IR1, INPUT);

pinMode(IR2, INPUT);

myservo1.attach(7);

myservo1.write(100);

lcd.setCursor (0,0);

lcd.print(" ARDUINO ");

lcd.setCursor (0,1);

lcd.print(" PARKING SYSTEM ");

delay (2000);

lcd.clear();

}

void loop(){

if(digitalRead (IR1) == LOW && flag1==0){

if(Slot>0){flag1=1;

if(flag2==0){myservo1.write(0); Slot = Slot-1;}

}else{

lcd.setCursor (0,0);

lcd.print(" SORRY :( ");

lcd.setCursor (0,1);

lcd.print(" Parking Full ");

delay (3000);

lcd.clear();

}

}

if(digitalRead (IR2) == LOW && flag2==0){flag2=1;

if(flag1==0){myservo1.write(0); Slot = Slot+1;}

}

if(flag1==1 && flag2==1){

delay (1000);

myservo1.write(100);

flag1=0, flag2=0;

}

lcd.setCursor (0,0);

lcd.print(" TRUST SPHERE ");

lcd.setCursor (0,1);

lcd.print("Slot Left: ");

lcd.print(Slot);

}

**CODE #2**

#include <Wire.h>

void setup()

{

Wire.begin();

Serial.begin(9600);

while (!Serial); // Leonardo: wait for serial monitor

Serial.println("\nI2C Scanner");

}

void loop()

{

byte error, address;

int nDevices;

Serial.println("Scanning...");

nDevices = 0;

for(address = 1; address < 127; address++ )

{

// The i2c\_scanner uses the return value of

// the Write.endTransmisstion to see if

// a device did acknowledge to the address.

Wire.beginTransmission(address);

error = Wire.endTransmission();

if (error == 0)

{

Serial.print("I2C device found at address 0x");

if (address<16)

Serial.print("0");

Serial.print(address,HEX);

Serial.println(" !");

nDevices++;

}

else if (error==4)

{

Serial.print("Unknown error at address 0x");

if (address<16)

Serial.print("0");

Serial.println(address,HEX);

}

}

if (nDevices == 0)

Serial.println("No I2C devices found\n");

else

Serial.println("done\n");

delay(5000); // wait 5 seconds for next scan

}