

Sistem detectie viteza autoturism

Introducere

Un sistem ce masoara viteza de deplasare a vehiculului si o afiseaza pe un display cu indicatii sonore. Acesta este util pentru deplasarea soferilor care conduc cu viteza si atentionarea acestora.

Descriere generală

Componentele principale:

- Arduino Uno: Reprezentat în centrul schemei, este unitatea centrală de control
- Un display LCD cu 16 pini conectat la Arduino
- Buzzer: Un buzzer conectat pentru a emite sunete
- 2 senzori IR: Detecteaza miscarea



Hardware Design

- Display
- 2X senzor IR
- Fire
- Buzzer activ
- Arduino Uno R3

De rezistente nu a fost nevoie.

Software Design

Codul a fost scris in Arduino IDE.

```
while (1) {
  if ((PINC & (1 << PINC0)) == 0 && flag1 == 0) {
    timer1 = TCNT1;
    flag1 = 1;
  }

  if ((PINC & (1 << PINC1)) == 0 && flag2 == 0) {
    timer2 = TCNT1;
    flag2 = 1;
  }

  if (flag1 == 1 && flag2 == 1) {
    if (timer1 > timer2) {
      Time = (timer1 - timer2) / (F_CPU / 1024.0);
    } else if (timer2 > timer1) {
      Time = (timer2 - timer1) / (F_CPU / 1024.0);
    }
    speed = (distance / Time) * 3600 / 1000;
    flag1 = 0;
    flag2 = 0;
  }

  if (speed == 0) {
    lcd_set_cursor(0, 1);
    if (flag1 == 0 && flag2 == 0) {
      lcd_print("No car detected ");
    } else {
      lcd_print("Searching... ");
    }
  } else {
    lcd_clear();
    lcd_set_cursor(0, 0);
    lcd_print("Speed:");
    char buffer[10];
    dtostrf(speed, 4, 1, buffer);
    lcd_print(buffer);
    lcd_print(" Km/Hr ");
    lcd_set_cursor(0, 1);
    if (speed > 50) {
      lcd_print(" Over Speeding ");
      PORTB |= (1 << PORTB5);
    } else {
      lcd_print(" Normal Speed ");
    }
    _delay_ms(3000);
    PORTB &= ~(1 << PORTB5);
    speed = 0;
    lcd_clear();
  }
}
```

```
}
```

Cod complet

```
#include <Wire.h> #include <LiquidCrystal_I2C.h>
```

```
LiquidCrystal_I2C lcd(0x27, 16, 2);
```

```
int timer1; int timer2;
```

```
float Time;
```

```
int flag1 = 0; int flag2 = 0;
```

```
float distance = 3.5; float speed;
```

```
int ir_s1 = A0; int ir_s2 = A1;
```

```
int buzzer = 13;
```

```
void setup() {
```

```
pinMode(ir_s1, INPUT);  
pinMode(ir_s2, INPUT);  
pinMode(buzzer, OUTPUT);
```

```
lcd.begin(16, 1);  
lcd.init();  
lcd.backlight();  
lcd.clear();  
lcd.setCursor(0, 0);  
lcd.print("VYCTOR MASHINA");  
delay(2000);  
lcd.clear();
```

```
}
```

```
void loop() {
```

```
if (digitalRead(ir_s1) == LOW && flag1 == 0) {  
    timer1 = millis();  
    flag1 = 1;  
}
```

```
if (digitalRead(ir_s2) == LOW && flag2 == 0) {
```

```
timer2 = millis();
flag2 = 1;
}
```


```
if (flag1 == 1 && flag2 == 1) {
  if (timer1 > timer2) {
    Time = timer1 - timer2;
  } else if (timer2 > timer1) {
    Time = timer2 - timer1;
  }
  Time = Time / 1000;
  speed = (distance / Time);
  speed = speed * 3600;
  speed = speed / 1000;
}
```

```
if (speed == 0) {
  lcd.setCursor(0, 1);
  if (flag1 == 0 && flag2 == 0) {
    lcd.print("No car detected");
  } else {
    lcd.print("Searching... ");
  }
} else {
  lcd.clear();
  lcd.setCursor(0, 0);
  lcd.print("Speed:");
  lcd.print(speed, 1);
  lcd.print("Km/Hr ");
  lcd.setCursor(0, 1);
  if (speed > 50) {
    lcd.print(" Over Speeding ");
    digitalWrite(buzzer, HIGH);
  } else {
    lcd.print(" Normal Speed ");
  }
  delay(3000);
  digitalWrite(buzzer, LOW);
  speed = 0;
  flag1 = 0;
  flag2 = 0;
}
```

}

Concluzii

Download

O arhivă (sau mai multe dacă este cazul) cu fișierele obținute în urma realizării proiectului: surse, scheme, etc. Un fișier README, un ChangeLog, un script de compilare și copiere automată pe uC crează întotdeauna o impresie bună .

Fișierele se încarcă pe wiki folosind facilitatea **Add Images or other files**. Namespace-ul în care se încarcă fișierele este de tipul **:pm:prj20??:c?** sau **:pm:prj20??:c?:nume_student** (dacă este cazul).
Exemplu: Dumitru Alin, 331CC → **:pm:prj2009:cc:dumitru_alin**.

Jurnal


Puteți avea și o secțiune de jurnal în care să poată urmări asistentul de proiect progresul proiectului.

Bibliografie/Resurse

Listă cu documente, datasheet-uri, resurse Internet folosite, eventual grupate pe **Resurse Software** și **Resurse Hardware**.

[Export to PDF](#)

From:
<http://ocw.cs.pub.ro/courses/> - **CS Open CourseWare**

Permanent link:
<http://ocw.cs.pub.ro/courses/pm/prj2024/vstoica/victor.ionescu0812> 

Last update: **2024/05/30 06:41**