

Kelompok

Atha Aulia shidiq 101032300197
Muhammad Faiz Rohadi 101032300126
Nazmi Hilmi Nuruzzaman 101032300164
I Made Marcello Dharmaguna 101032300092
Maulana Alfadhilah Ekaputra 101032300172
Satrio mukti Wibowo 101032300210

Latar belakang

Kebiasaan bangun pagi masih merupakan hal yang berat dilakukan bagi sebagian orang. Bagi sebagian orang lagi, bangun pagi merupakan kebiasaan atau kebutuhan, banyak manfaat yang bisa diperoleh dari bangun lebih awal. Orang biasanya nyetel alarm untuk dapat bangun pagi dan beraktivitas, biasanya orang menaruh alarm tersebut di tempat yang terjangkau dari tempat tidurnya, namun kebanyakan orang hanya akan mematikan alarm tersebut yang dan kembali tidur. Maka fungsi alarm itu sendiri itu hampir tidak berguna.

Spesifikasi

Mikrokontroler	ATmega328P
Tegangan Pengoperasian	5V
Tegangan Input(Rekomendasi)	7-12V
Batas Tegangan Input	6-20V
Pin I/O Digital	14 (6 diantaranya dapat di gunakan sebagai output PWM)
Pin Digital PWM	6
Pin Input Analog	6

Arus DC Tiap Pin I/O	20 mA
Arus DC untuk pin 3.3V	50 mA
Flash Memory	32 KB (ATmega328P) Sekitar 0.5 KB digunakan untuk bootloader
SRAM	2 KB (ATmega328P)
EEPROM	1 KB (ATmega328P)
Clock Speed	16 MHz
LED_BUILTIN	13
Panjang	68.6 mm
Lebar	53.4 mm
Berat	25 g

Alasan memilih solusi

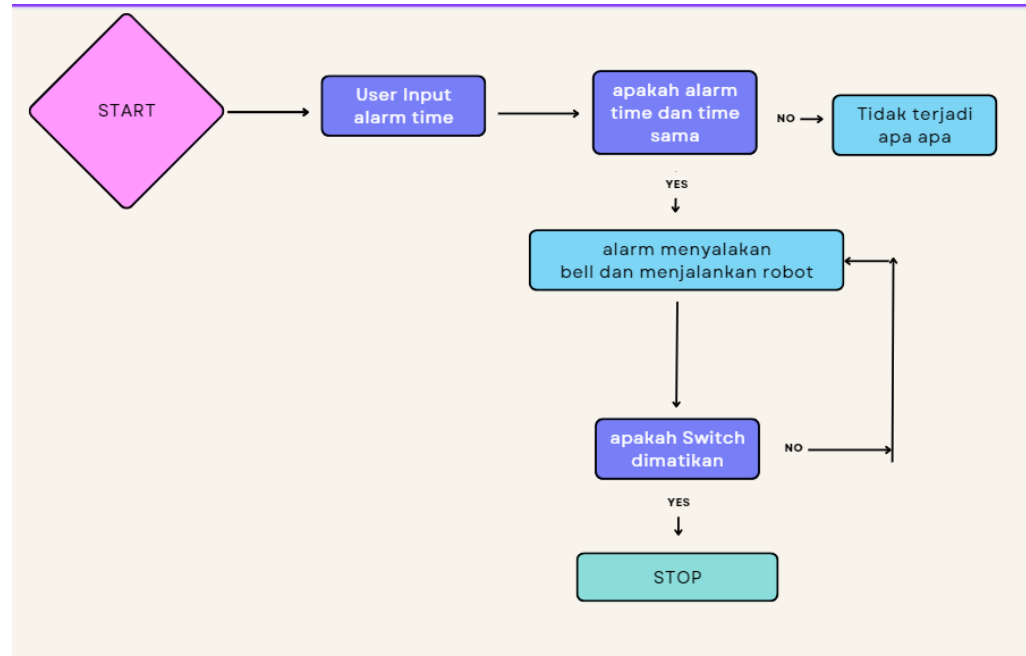
Alasan kita memilih solusi ini karena dengan alarm yang berjalan orang tidak akan dapat langsung mematikan alarm dan kembali tidur namun harus ngejar alarm tersebut yang kemungkinan besar membuat akan membuat orang tersebut terbangun.

Desain

A. Komponen yang dibutuhkan

- Arduini uno R3 Atmega328p
- Battery holder paralel
- HC-SR04 Ultrasonic sensor
- Module L298N
- Gearbox
- Breadboard
- Switch
- LCD 1602 +I2C Serial interface
- Kabel jumper

B. Flow chart

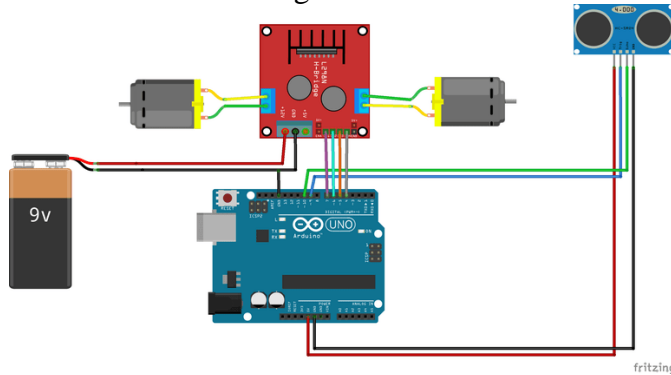


Maksud dari flowchart tersebut adalah pada saat rangkaian dinyalakan user diminta untuk menginput jam saat alarm akan menyala dan saat jam pada lcd sudah sama dengan jam alarm maka alarm tersebut akan menyala dan saat alarm nyala buzzer akan berbunyi dan robot akan bergerak jika user mematikan alarm seluruh rangkain akan mati

Implementasi

A. Gambar Tinker card

Ultrasonic sensor dan gearbox



B. Cara kerja alat

user menyalakan rangkaian, lalu user akan menginputkan alarm pada suatu waktu pada saat waktu sama dengan set waktu yang di tentukan user maka robot akan mengeluarkan suara dari dan mulai berjalan dengan menggunakan gearbox dan sensor agar tidak menabrak hingga user mematikan switch atau menekan tombol reset.

C. Bagaimana menerapkan solusi untuk menjawab latar belakang

Harapkan kami dengan proyek boombot ini orang yang biasanya hanya mematikan alarm dan kembali tidur harus bangun dari kasur terlebih dahulu sebelum bisa mematikan alarm sehingga dapat membuat orang lebih terbangun

Pengujian

A. Troubleshooting

Pada saat pertama kali menyambungkan rangkaian seluruh rangkaian tidak dapat menyala setelah kita cek lagi terjadi kesalahan pada penghubungan kabel positif dan negatif. Tempat baterai yang kita gunakan dirangkai secara paralel dan baterai yang digunakan hanya 3,7 V oleh karena itu saat kita mencoba menggunakan baterai output yang dikeluarkan gearbox tidak terlalu besar

B. Masalah yang terjadi

Karena terjadi miskom dalam pembelian bahan kita tidak dapat membeli buzzer dan pada saat akan membeli secara langsung pada toko offline, toko tersebut tutup dan toko online membutuhkan waktu yang melebihi batas oleh karena itu hasil rangkaian yang kita dapat hanya robot arduino dengan sensor dan display jam pada LCD

Codingan

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

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

```
#include <DS3231.h>
```

```
DS3231 rtc(SDA, SCL);
```

```
LiquidCrystal_I2C lcd(0x27,16, 2); //alamat i2c 0x27
```

```
int trigPin = 9;    // trig pin of HC-SR04
```

```
int echoPin = 10;   // Echo pin of HC-SR04
```

```
int revleft4 = 4;    //REVerse motion of Left motor
```

```
int fwdleft5 = 5;    //ForWarD motion of Left motor
```

```
int revright6 = 6;   //REVerse motion of Right motor
```

```
int fwdright7 = 7;   //ForWarD motion of Right motor
```

```
long duration, distance;
```

```
void setup() {
```

```
    Serial.begin(9600);
```

```
    rtc.begin();
```

```
    lcd.begin();
```

```
    lcd.setCursor(0,0);
```

```
    lcd.print("Rtc");
```

```
lcd.setCursor(0,1);
```

```
lcd.print("DS3231");
```

```
delay(3000);
```

```
lcd.clear();
```

```
delay(random(500,2000)); // delay for random time
```

```
Serial.begin(9600);
```

```
pinMode(revleft4, OUTPUT); // set Motor pins as output
```

```
pinMode(fwdleft5, OUTPUT);
```

```
pinMode(revrigh6, OUTPUT);
```

```
pinMode(fwdright7, OUTPUT);
```

```
pinMode(trigPin, OUTPUT); // set trig pin as output
```

```
pinMode(echoPin, INPUT); //set echo pin as input to capture reflected waves
```

```
}
```

```
void loop() {
```

```
    lcd.setCursor(0,0);
```

```
    lcd.print(rtc.getDOWStr(1));
```

```
    lcd.setCursor(3,0);
```

```
    lcd.print(",");
```

```
    lcd.setCursor(4,0);
```

```
lcd.print(rtc.getDateStr());
```

```
lcd.setCursor(00,1);
```

```
lcd.print(rtc.getTimeStr());
```

```
lcd.setCursor(10,1);
```

```
lcd.print("t:");
```

```
lcd.setCursor(12,1);
```

```
lcd.print(rtc.getTemp());
```

```
delay (1000);
```

```
digitalWrite(trigPin, LOW);
```

```
delayMicroseconds(2);
```

```
digitalWrite(trigPin, HIGH); // send waves for 10 us
```

```
delayMicroseconds(10);
```

```
duration = pulseIn(echoPin, HIGH); // receive reflected waves
```

```
distance = duration / 58.2; // convert to distance
```

```
delay(10);
```

```
// If you dont get proper movements of your robot then alter the pin numbers
```

```
if (distance > 19)
```

```
{
```

```
digitalWrite(fwdright7, HIGH); // move forward
```

```
digitalWrite(revright6, LOW);
```

```
digitalWrite(fwdleft5, HIGH);
```

```
digitalWrite(revleft4, LOW);  
}
```

```
if (distance < 18)
```

```
{  
    digitalWrite(fwdright7, LOW); //Stop  
    digitalWrite(revright6, LOW);  
    digitalWrite(fwdleft5, LOW);  
    digitalWrite(revleft4, LOW);  
    delay(500);  
    digitalWrite(fwdright7, LOW); //movebackward  
    digitalWrite(revright6, HIGH);  
    digitalWrite(fwdleft5, LOW);  
    digitalWrite(revleft4, HIGH);  
    delay(500);  
    digitalWrite(fwdright7, LOW); //Stop  
    digitalWrite(revright6, LOW);  
    digitalWrite(fwdleft5, LOW);  
    digitalWrite(revleft4, LOW);  
    delay(100);  
    digitalWrite(fwdright7, HIGH);  
    digitalWrite(revright6, LOW);  
    digitalWrite(revleft4, LOW);  
    digitalWrite(fwdleft5, LOW);  
    delay(500);  
}
```


}

}