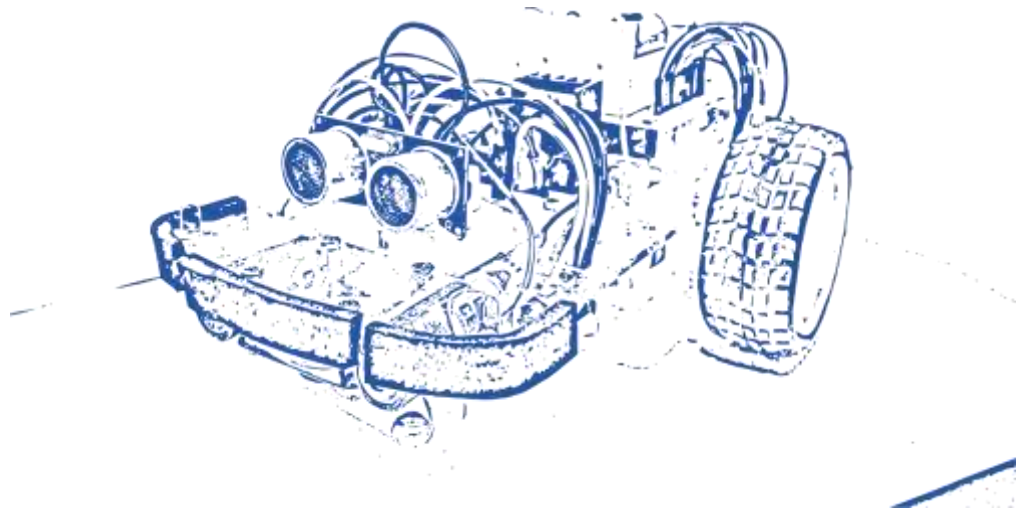


zidCreative R&D

Karya I , 1 Februari 2023



‘ Robot beroda dengan kemampuan menghindari halangan ‘

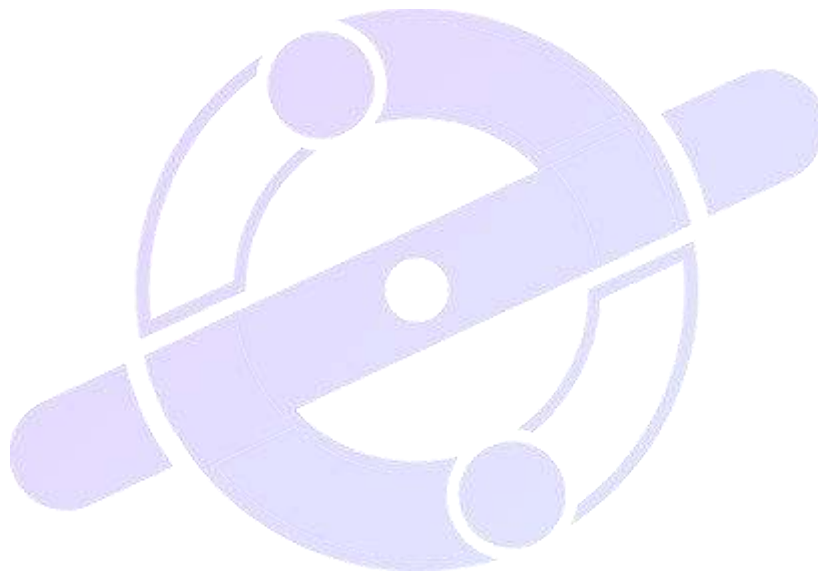
Rancang Bangun

Robot Avider I



Daftar Isi

Daftar Isi	2
Definisi	3
Versi dan Spesifikasi Inovasi	3
Skema	3
Coding	5
Pengujian dan hasil.....	8
Kesimpulan	9
Lampiran	9
Referensi dan Tools	10



Definisi

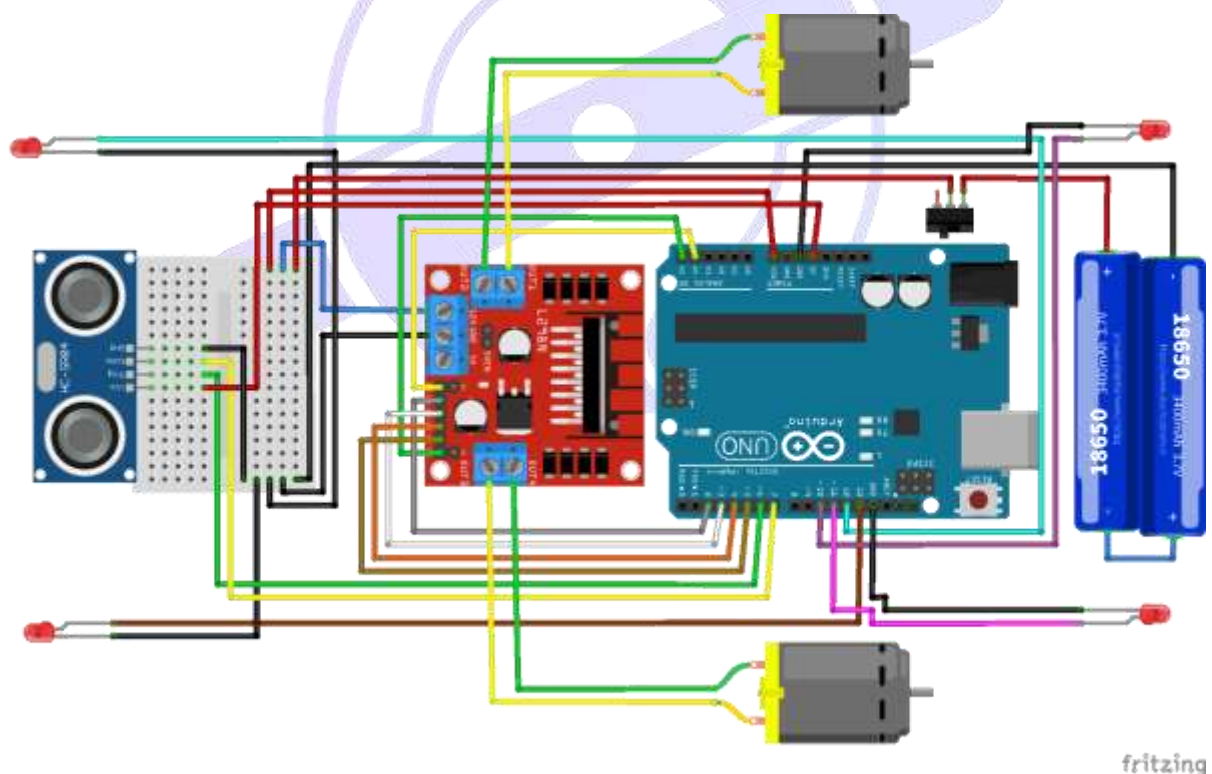
Robot Avider adalah jenis robot yang memiliki kemampuan untuk menghindari objek atau rintangan yang ada di sekitarnya. Biasanya robot ini dilengkapi dengan sensor seperti ultrasonik atau infrared untuk membantunya menentukan lokasi dan jarak objek, sehingga dapat membuat keputusan untuk menghindari objek tersebut.

Versi dan Spesifikasi Inovasi

Dalam versi **Karya I** kali ini dibuat robot avoider sederhana dengan hanya menggunakan :

- Satu buah sensor ultrasonik HC-SR04
- Dua buah Penggerak Roda Geared DC Motor 6 Volt dengan Driver L298N
- Mikrokontroller Arduino Uno
- Input daya dua baterai 18650 3.7 Volt DC diseri (7.4 Volt DC)
- 4 buah LED indikator gerak
- Akrilik body

Skema

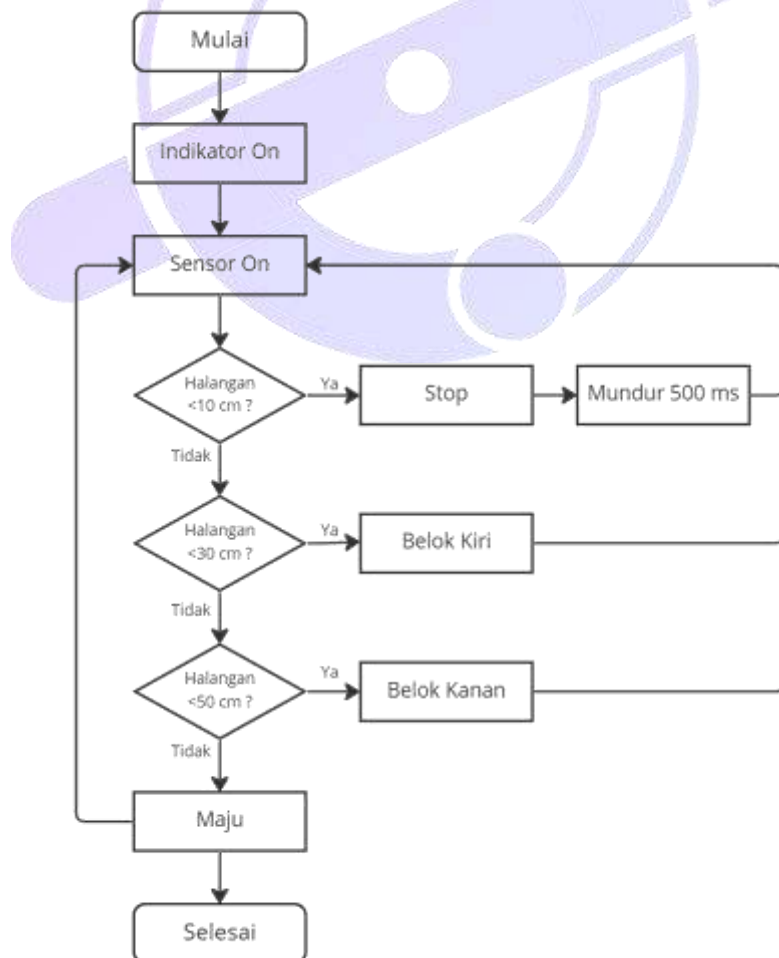


Berikut detail skema :

Komponen 1		Komponen 2	
HC-SR04	Vcc	+5v	Arduino Uno
	Gnd	Gnd	
	Trig	6	

	Echo	7	
LED Indikator	Depan Kanan	12	
	Depan Kiri	13	
	Belakang Kanan	10	
	Belakang Kiri	11	
Baterai 18650	+	Vin	
	-	Gnd	
Driver Motor L298N	ENA	A4	
	ENB	A5	
	IN 1	2	
	IN 2	3	
	IN 3	4	
	IN 4	5	
	GND	Gnd	
	+12 v	+	Baterai 18650
	Output 1	+ / -	Motor DC 1
	Output 2	+ / -	Motor DC 2

Berikut adalah detail alur kerja / Flowchart :



Coding

```
//Driver_2 - Penambahan lampu dan perbaikan kecepatan

#include <HCSR04.h>

#define ENA 18          // Enable/speed motors kanan
#define ENB 19          // Enable/speed motors kiri
#define IN_1 2          // motor kanan maju
#define IN_2 3          // motor kanan mundur
#define IN_3 4          // motor kiri maju
#define IN_4 5          // motor kiri mundur
#define SDKI 13
#define SDKA 12
#define SBKI 11
#define SBKA 10

UltraSonicDistanceSensor distanceSensor(6, 7);    //Trig 6 Eco 7
int speedCar = 300;          // 400 - 1023.
int speed_Coeff = 3;

void setup()
{
    Serial.begin(9600);

    pinMode(ENA, OUTPUT);
    pinMode(ENB, OUTPUT);
    pinMode(IN_1, OUTPUT);
    pinMode(IN_2, OUTPUT);
    pinMode(IN_3, OUTPUT);
    pinMode(IN_4, OUTPUT);
    pinMode(SDKI, OUTPUT);
    pinMode(SDKA, OUTPUT);
    pinMode(SBKI, OUTPUT);
    pinMode(SBKA, OUTPUT);

    for (int i = 1; i < 3; i++)
    {
        allnyala();
        delay(200);
        allmati();
        delay(200);
    }
}

void loop()
{
    distanceSensor.measureDistanceCm();
    Serial.println(distanceSensor.measureDistanceCm());

    if ((distanceSensor.measureDistanceCm() < 10) and
        (distanceSensor.measureDistanceCm() > -1))
```

```

{
    henti();
    delay(1000);
    mundur();
    delay(500);
    henti();
    delay(1000);
}
else if ((distanceSensor.measureDistanceCm() < 30) and
(distanceSensor.measureDistanceCm() > 9))
{
    belka();
    delay(100);
}
else if ((distanceSensor.measureDistanceCm() < 50) and
(distanceSensor.measureDistanceCm() > 29))
{
    belki();
    delay(100);
}
else
{
    maju();
}

delay(20);
}

//----- Roda

void maju()
{
    lamdep();
    digitalWrite(IN_1, HIGH);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, HIGH);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar);
}

void mundur()
{
    lambla();
    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, HIGH);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, HIGH);
    analogWrite(ENB, speedCar);
}

```

```

}

void belka()
{
    lamka();
    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, HIGH);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, HIGH);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar);
}

void belki()
{
    lamki();
    digitalWrite(IN_1, HIGH);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, HIGH);
    analogWrite(ENB, speedCar);
}

void henti()
{
    allnyala();
    digitalWrite(IN_1, LOW);
    digitalWrite(IN_2, LOW);
    analogWrite(ENA, speedCar);

    digitalWrite(IN_3, LOW);
    digitalWrite(IN_4, LOW);
    analogWrite(ENB, speedCar);
}

//----- lampu

void sendeki()
{
    digitalWrite(SDKI, HIGH);
}

void lamdep()
{
    digitalWrite(SDKI, HIGH);
    digitalWrite(SDKA, HIGH);
    digitalWrite(SBKI, LOW);
}

```

```

    digitalWrite(SBKA, LOW);
}

void lambla()
{
    digitalWrite(SDKI, LOW);
    digitalWrite(SDKA, LOW);
    digitalWrite(SBKI, HIGH);
    digitalWrite(SBKA, HIGH);
}

void lamka()
{
    digitalWrite(SDKI, LOW);
    digitalWrite(SDKA, HIGH);
    digitalWrite(SBKI, LOW);
    digitalWrite(SBKA, HIGH);
}

void lamki()
{
    digitalWrite(SDKI, HIGH);
    digitalWrite(SDKA, LOW);
    digitalWrite(SBKI, HIGH);
    digitalWrite(SBKA, LOW);
}

void allnyala()
{
    digitalWrite(SDKI, HIGH);
    digitalWrite(SDKA, HIGH);
    digitalWrite(SBKI, HIGH);
    digitalWrite(SBKA, HIGH);
}

void allmati()
{
    digitalWrite(SDKI, LOW);
    digitalWrite(SDKA, LOW);
    digitalWrite(SBKI, LOW);
    digitalWrite(SBKA, LOW);
}

```

Pengujian dan hasil

Berikut adalah data pengujian penggunaan robot berdasarkan voltase baterai Robot dengan menggunakan baterai yang berbeda yaitu :

- Baterai 18650 3.7 Volt bekas powerbank 15000 mAH
- Baterai 18650 3.7 Volt Ultrafire 9000 mAH

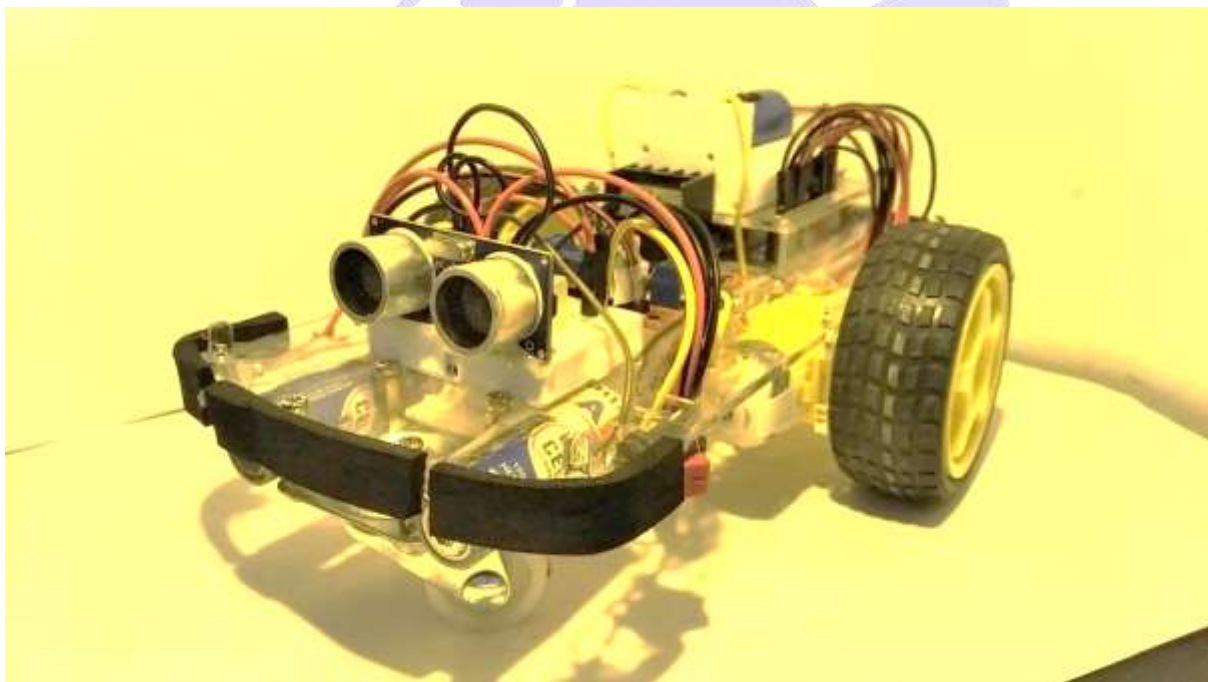
Percobaan ke	mAH	Voltase Awal	Durasi (Menit)	Voltase Akhir
1	15000	8.3	195	5.7
2	9000	8.3	75	6.2

Kesimpulan

Robot avoider arduino ini adalah robot tingkat sederhana yang tidak terlalu sulit dirancang dan dikembangkan. Robot avoider cocok sekali digunakan media pembelajaran dalam memahami integrasi cara kerja komponen-komponennya khususnya pada robotika tingkat pemula. Bahannya pun kini banyak tersedia dan murah yang didukung juga oleh banyaknya software dan website dengan sumber terbuka yang mampu membuat dan merancang program dan skema robot dengan lebih mudah.

Lampiran

Berikut kenampakan robot setelah jadi :



Referensi dan Tools

- Software Fritzing
- Arduino IDE
- Miro Website
- ChatGPT openAI

