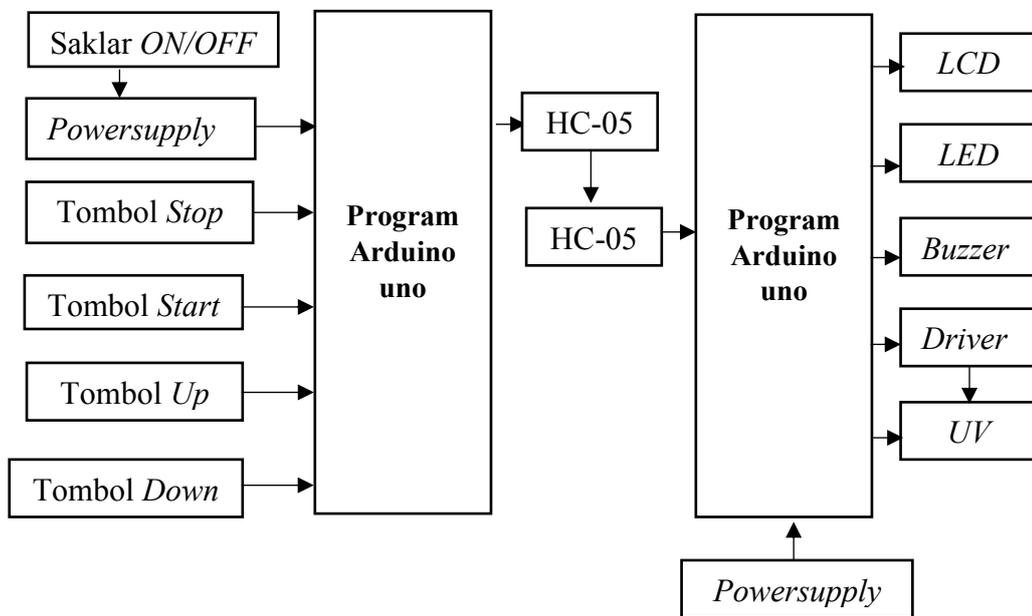


BAB III METODE PENELITIAN

3.1 Blok Diagram *Sterilisator UV*

Untuk melihat rancang bangun rangkain dari alat *sterilisator UV*, maka penulis merencanakan blok diagram dari rangkain yang di rencanakan seperti Gambar 3.1



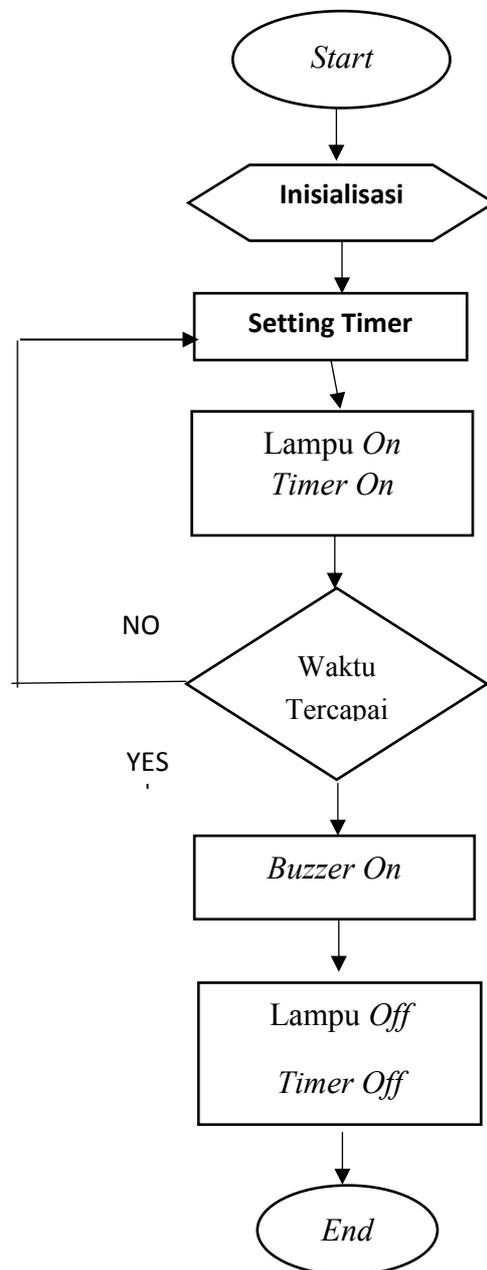
Gambar 3.1. Diagram Blok *sterilisator UV*

Mengacu pada gambar Blok Diagram Alat dapat dijelaskan sistem kerja alat *sterilisator UV*. Terdapat 2 arduino pada rangkaian *sterilisator UV* yaitu untuk rangkaian *Remote* dan untuk rangkaian inti. Pada saat saklar rangkaian *Remote* pada posisi *ON* maka *power supply* memberikan tegangan 5V ke arduino uno. Terdapat 4 tombol pada arduino uno di rangkaian *Remote* yaitu tombol *Start*, tombol *Stop*, tombol *Up*, tombol *Down* yang berfungsi untuk mengatur waktu lamanya

sterilisasi . Ketika Arduino uno pada *Remote* mendapatkan perintah, maka perintah tersebut dikirimkan oleh *bluetooth HC-05 transceiver* dan diterima oleh *Bluetooth HC-05 receiver*. Perintah dari *Bluetooth HC-05 receiver* diolah oleh Arduino uno rangkaian inti. Sehingga tampilan pengaturan jam pada *LCD* akan berubah sesuai dengan masukan dari rangkaian tombol pada *Remote*. Rangkaian *LCD* berfungsi sebagai *indikator* waktu pemilihan jam yang diinginkan. Terdapat 4 *LED* pada rangkaian tersebut yaitu satu *LED* untuk *indikator* alat sudah hidup, satu *LED* untuk *indikator* pemilihan waktu 2 jam, satu *LED* untuk *indikator* pemilihan waktu 4 jam dan satu *LED* untuk *indikator* pemilihan waktu 6 jam. Jika tombol *Start* pada *Remote* ditekan maka arduino uno pada rangkaian inti mengaktifkan *driver relay* untuk menyalakan lampu *UV*.

3.2. Diagram Alir Sterilisator UV

Berikut merupakan blok diagram alir alat *sterilisator UV*

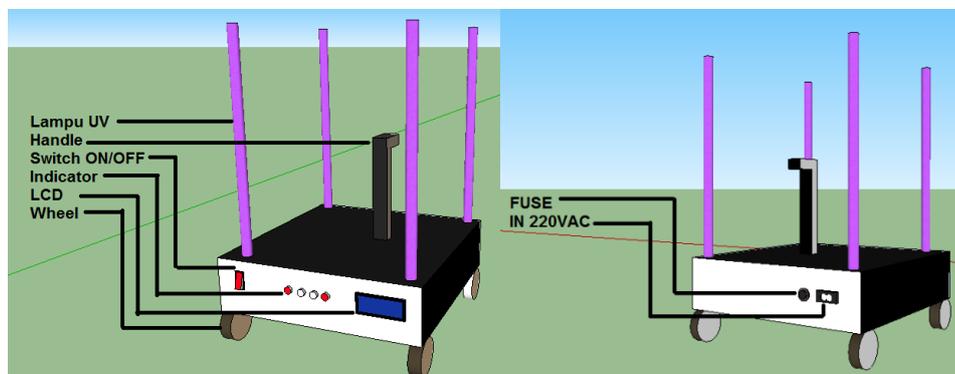


Gambar 3.2 Diagram Alir

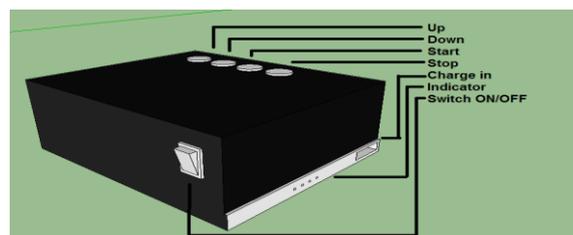
Ketika *ON/OFF* ditekan alat dalam keadaan menyala menginisialisasikan *LCD* serta modul *Bluetooth*. Jika tombol *Start* tidak ditekan maka masuk fungsi pengaturan waktu .Ada 3 pilihan waktu yaitu 2 jam, 4 jam dan 6 jam. Jika ditekan tombol *Start* maka waktu mulai mencacah mundur selama waktu yang ditentukan dan lampu *UV* nyala.Jika ditekan tombol *Stop* maka waktu berhenti mencacah, lampu *UV* mati dan kembali ke fungsi *setting* waktu . Jika waktu habis maka waktu berhenti mencacah , lampu *UV* mati dan kembali ke fungsi *setting* waktu.

3.3 Diagram Mekanis Alat

Berikut ini adalah rancangan atau diagram mekanis alat *sterilisator UV*



Gambar 3.3 *Diagram Mekanis Alat Tampak Depan*



Gambar 3.4 *Diagram Mekanis Remote*

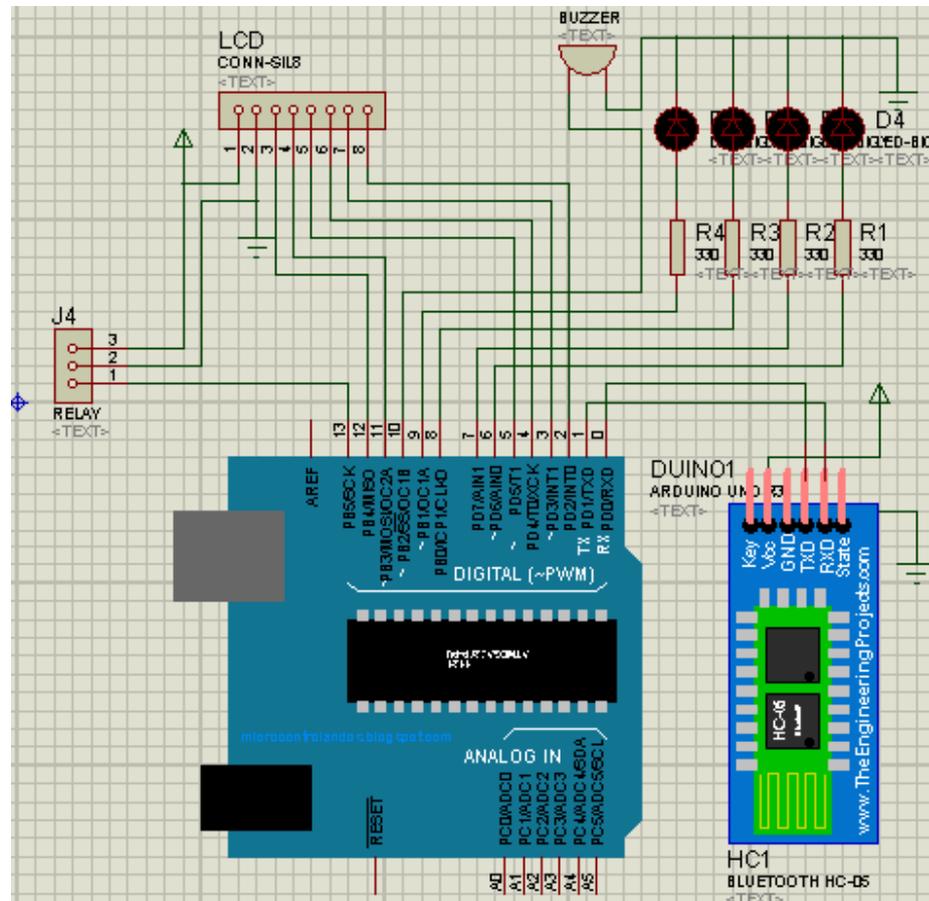
1. Tombol kotak adalah tombol *SWITCH*
2. Persegi panjang adalah lampu *UV*
3. Lampu merah adalah lampu *Indicator*
4. Persegi panjang bengkok adalah *HANDLE*
5. Kotak persegi panjang warna biru adalah *LCD*
6. Lingkaran yang paling bawah adalah *WHEEL*
7. Lingkaran kecil warna hitam adalah *FUSE*
8. Persegi panjang memiliki dua lingkaran di dalamnya adalah *IN220VAC*

Alat *sterilisator UV* mempunyai 4 lampu dan dilengkapi *Remote* dan *indicator timer*. *Setting timer* terdiri atas 2 jam, 4 jam dan 6 jam. Selain itu pada modul terdapat saklar *ON/OFF* tegangan PLN menuju catu daya, saat *ON* ditekan (alat dalam keadaan terbuka) maka tegangan PLN akan masuk ke catu daya, kemudian masuk ke seluruh rangkaian dan *driver*. terdapat satu buah *Remote* berfungsi sebagai pengatur lamanya lampu menyala.

3.4 Rangkaian Penyusun Modul

Dibawah ini terdapat rangkaian yang dapat menyusun sistem kerja alat *Sterilisator UV* dengan sistem *Remote* kontrol berbasis *microcontroller atmega328* sebagai berikut:

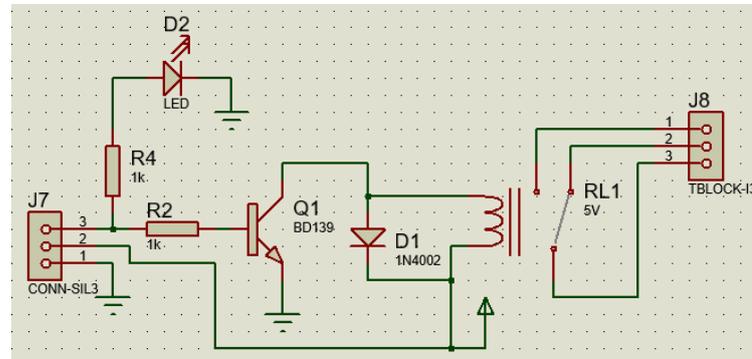
3.4.1 Rangkaian Minimum Sistem



Gambar 3.4. Rangkaian Minimum Sistem

Rangkaian minimum sistem berfungsi sebagai kontrol rangkaian. Rangkaian ini merupakan pengendali segala kerja dari alat. Gambar diatas merupakan rangkaian minimum *system* Arduino uno yang telah di lengkapi dengan *indicator LCD*.

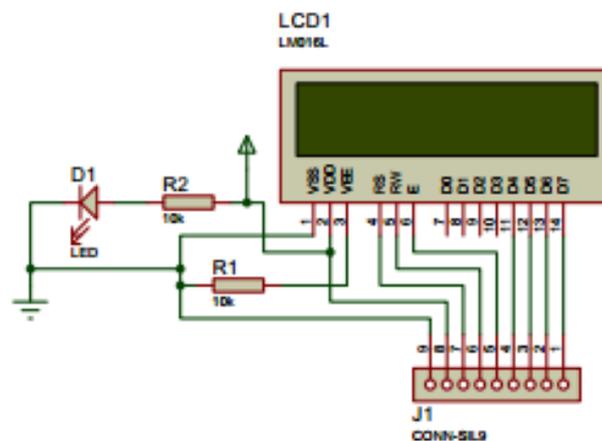
3.4.2 Rangkaian *Driver Relay*



Gambar 3. 1 Rangkaian *Driver Relay* (5 VDC)

Rangkaian diatas merupakan rangkaian *driver relay* yang berfungsi sebagai pengendali lampu *UV*. Rangkaian *driver relay* masuk *PIN 13* pada rangkaian minimum system. *Pin 13* berfungsi memberikan perintah *HIGH* atau *LOW* ke rangkaian *relay*. Jika *PIN13* memberikan logika *HIGH* maka kaki basis pada transistor *BD139* mendapatkan tegangan dan kaki *emitor* terhubung ke kolektor sehingga *ground* terhubung ke *relay 5V* dan tegangan pada kaki *COM relay* terhubung ke lampu *UV*.

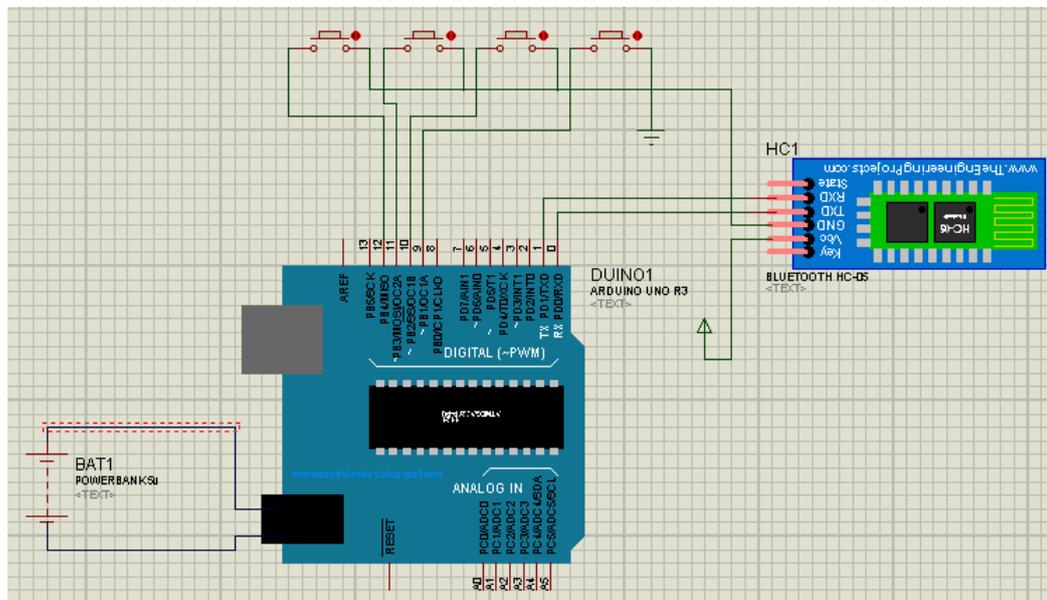
3.4.3 Rangkaian *LCD*



Gambar 3. 2 Rangkaian *LCD*

Rangkaian diatas merupakan rangkaian *LCD* karakter 4 x 16 dimana *LCD* sebagai *display* yang menampilkan *setting* waktu dari alat yang telah di jalankan oleh *program* serta menampilkan waktu nyala lampu *UV (hourmeter)*. Untuk dapat menyalakan *LCD* membutuhkan tegangan +5V, pada pin *VDD* mendapatkan +5V, pin *VSS* mendapatkan *ground*, pin D4 D5 D6 D7 mendapatkan *PORT* yang di tentukan pada rangkaian minimum sistem.

3.4.4 Rangkaian *Remote*

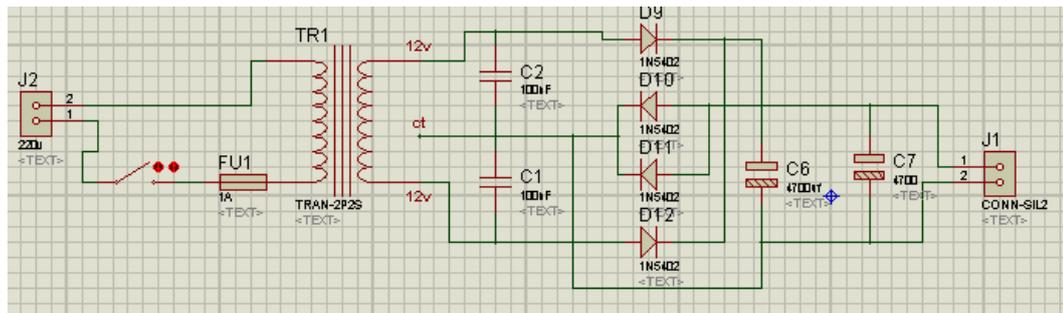


Gambar 3. 8 Rangkaian *Remote*

Rangkaian diatas merupakan rangkaian *Remote* untuk mengirim perintah dari tombol. Terdapat 4 tombol pada rangkaian tersebut yaitu *Up*, *Down*, *Start*, *Stop*. tombol *Up* dan *Down* berfungsi saat pengatur waktu sedangkan tombol *Start* berfungsi untuk memulai atau menyalakan lampu dan *hourmeter*, tombol *Stop* berfungsi sebagai menghentikan atau mematikan lampu dan *hourmeter* rangkaian

tersebut menggunakan *powerbank* 5v Terdapat 1 modul *HC-05* untuk mengirim perintah ke *HC-05* lainnya.

3.4.5 Rangkaian *Power supply*



Gambar 3.9 Rangkaian *Powersupply*

Rangkaian diatas merupakan rangkaian *Power supply* Terdapat transformator yang berfungsi merubah tegangan $220VAC$ menjadi tegangan $12VAC$. $12VAC$ tersebut masuk ke diode untuk disearahkan sehingga keluaran dari 4 dioda tersebut $12VDC$

3.5. Pembuatan Program

Berikut ini adalah inti dari program modul tugas akhir menggunakan Arduino:

a. Program Utama

```

seting(); //memanggil fungsi setting
digitalWrite(uv,LOW); //uv mati
digitalWrite(led1,HIGH) ; //led1 hidup

```

b. Program fungsi Setting

```
void seting()
if(state=='1'){jam=jam+2;delay(500);}
//jika terbaca nilai 1 dari remot maka jam bertambah
2
if(state=='2'){jam=jam-2;delay(500);}
//jika terbaca nilai 2 dari remot maka jam berkurang
2
if(state=='3'){mains();}
//jika terbaca nilai 3 dari remot maka memanggil
fungsi mains
lcd.setCursor(0,0);
lcd.print("  Seting waktu :");
lcd.setCursor(0,1);
lcd.print(jam);
lcd.print(" jam");
//menampilkan jam
```

c. Program Fungsi Mains

```
digitalWrite(uv,HIGH); //uv nyala
lcd.clear();
lcd.setCursor(0,0);
lcd.print(jam); //menampilkan jam
lcd.print(":");
lcd.print( menit); //menampilkan menit
lcd.print(":");
lcd.print( detik); //menampilkan detik
delay(10);
```

d. Program Perintah Tombol

```

if(digitalRead(up)==LOW)
{delay(500);Serial.print('1');}
//jika ditekan tombol up maka mengirim angka 1
if(digitalRead(down)==LOW)
{delay(500);Serial.print('2');}
//jika ditekan tombol down maka mengirim angka 2
if(digitalRead(ok)==LOW)
{delay(500);Serial.print('3');}
//jika ditekan tombol ok maka mengirim angka 3
if(digitalRead(hop)==LOW) {Serial.print('4');}
//jika ditekan tombol hop maka mengirim angka 4

```

e. Program menyimpan waktu *hourmeter*

```

if(state=='4' || jam<0)
{
EEPROM.write(0, hourjam);
EEPROM.write(1, hourmenit);
EEPROM.write(2, hourdetik);
digitalWrite(buzz,HIGH);// buzzer nyala
delay(1000); // jeda 1000ms
digitalWrite(buzz,LOW);// buzzer mati
break;//kembali ke program utama
}

```

f. Program membaca waktu *hourmeter*

```

hourdetik = EEPROM.read(2);
hourmenit = EEPROM.read(1);
hourjam = EEPROM.read(0);
lcd.setCursor(7,1);
lcd.print(hourjam); //menampilkan jam
lcd.print(":");
lcd.print( hourmenit);//menampilkan menit
lcd.print(":");
lcd.print( hourdetik);//menampilkan detik

```

3.6. Perancangan Pengujian

Pada analisa rancangan ada parameter yang akan diujikan yaitu *timer* apakah rancangan sudah sesuai dengan kondisi yang diinginkan atau belum. Pengujian direncanakan akan dilakukan pengambilan data setiap 2 jam dengan 20 kali pengujian, 4 jam 20 kali pengujian dan 6 jam 20 pengujian. Pada modul ini terdapat parameter yang akan diuji yaitu lama waktu penyinaran.

Timer/counter adalah fasilitas dari *ATMega328* yang digunakan untuk perhitungan pewaktuan. Pengujian *timer* ini bertujuan untuk memastikan bahwa *timer* sudah berfungsi dengan baik. Fungsi dari *timer* sendiri yaitu untuk mengatur lamanya waktu yang akan digunakan dalam proses penyinaran. Pengujian *timer* dilakukan dengan cara melakukan perbandingan dengan *stopwatch*. Setiap 2 jam, 4 jam dan 6 jam dilihat data jam dan menit apakah sama dengan *stopwatch*. Pengujian *timer* dilakukan sebanyak masing-masing 2 jam 20 kali pengujian sedangkan 4 jam dan 6 jam masing-masing 20 kali pengujian.

3.7. Jenis Penelitian

Jenis penelitian yang penulis gunakan adalah jenis penelitian eksperimental, artinya meneliti, mencari, menjelaskan, dan membuat suatu instrument dimana instrument ini dapat langsung dipergunakan oleh pengguna. *Variabel* yang diteliti dan diamati pada alat bantu *Sterilisator UV* dengan sistem *Remote* kontrol berbasis *microcontroller* *ATMega 328* ini adalah lama waktu penyinaran pada lampu *UV*.

3.8. Sistematika Pengukuran

3.8.1. Rata-Rata

Rata-Rata adalah nilai atau hasil pembagian dari jumlah data yang diambil atau diukur dengan banyaknya pengambilan data atau banyaknya pengukuran.

$$\bar{x} = \frac{\sum x_n}{n} \quad (3-1)$$

dengan :

\bar{x}	= Rata – rata
$\sum x_n$	= Jumlah x sebanyak n
n	= Data Banyak

3.8.2. Simpangan Error %

Simpangan adalah selisih dari rata–rata nilai harga yang dikehendaki dengan nilai yang diukur. Berikut rumus dari simpangan:

Dengan:

$$\text{simpangan} = x_n - \bar{x} \quad (3-2)$$

Dengan:

$Simpangan$	= Nilai <i>error</i> yang dihasilkan
x_n	= Rata – rata data waktu
\bar{x}	= Rata – rata data modul

3.8.3. Presentase Error (%)

Error (kesalahan) adalah selisih antara *mean* terhadap masing-masing data.

Rumus *error* adalah :

$$\text{Persentase Error} = \frac{\text{simpangan}}{x_n} \times 100\% \quad (3-3)$$

Dengan :

: simpangan = *Besarnya nilai simpangan atau error dalam %*
persentase error = rata-rata data *stopwatch*

3.8.4. *Standar Deviasi* (SD)

Standar deviasi adalah suatu nilai yang menunjukkan tingkat (derajat) variasi kelompok data atau ukuran *standar* penyimpangan dari *meannya*.

Rumus standar deviasi adalah:

$$SD = \frac{\sqrt{\sum(x_1 - \bar{x})^2 + (x_2 - \bar{x})^2 + \dots + (x_n - \bar{x})^2}}{n - 1} \quad (3-4)$$

Dengan :

SD = *Standar deviasi*

x = Data x

\bar{x} = Rata-rata

n = Banyak data

3.8.5. Ketidakpastian (Ua)

Ketidak pastian adalah kesangsian yang muncul pada tiap hasil. Atau pengukuran biasa disebut, sebagai kepresisian data satu dengan data yang lain.

Rumus dari ketidakpastian adalah sebagai berikut :

$$\text{Ketidakpastian} = \frac{stdv}{\sqrt{n}} \quad (3-5)$$

Dengan :

STDV = *Standar Deviasi*

n = banyaknya data