

BAB III

METODOLOGI

PERANCANGAN ALAT

3.1 Umum

Pada bab ini akan menjelaskan mengenai konsep perancangan alat Pendeteksi Harmonisa Arus Pada Sistem Tenaga Listrik dengan menggunakan metode *FFT* berbasis *Lab View*. Perancangan ini terdiri dari dua bagian, yaitu perancangan Perangkat Keras (*Hardware*) dan Perancangan Perangkat Lunak (*Software*). Perancangan Perangkat Keras ini meliputi Sistem Minimum Arduino Uno R3 dan pembuatan rangkaian pengolah sinyal dari sensor SCT-019. Kemudian untuk Perancangan Perangkat Lunak meliputi Algoritma pemrograman Arduino Uno, komunikasi Arduino Uno R3 dengan PC atau Komputer, Algoritma Pemrogram Grafis menggunakan Lab View serta proses pengolahan data yang dikirimkan dari Arduino Uno untuk ditampilkan dalam bentuk grafik dan *User Interface* oleh *Lab View*.

3.2 Blok Diagram Sistem

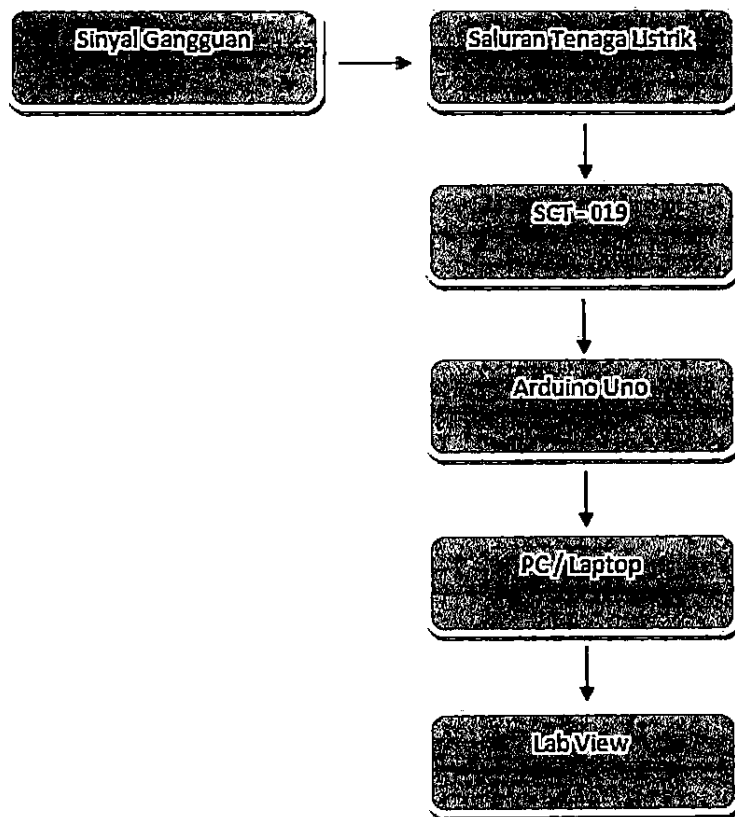
Perancangan alat akan memberikan kinerja maksimal pada hasil sistem ketika perancangan alat dijalankan sesuai prosedur. Pendayagunaan alat menjadi akhir tujuan dikarenakan menentukan keberhasilan perancangan alat. Memperhatikan karakteristik tiap-tiap bagian sangat penting, terkait dengan fungsi dan kinerja alat untuk dapat bekerja secara maksimal.

Secara keseluruhan perancangan alat Pendeteksi Harmonisa Arus Pada Sistem Tenaga

... ..

terdiri dari rangkaian antarmuka yang berfungsi untuk menampilkan keluarannya yang berupa data digital.

Dalam tahapan perencanaan diperlukan suatu gambaran secara menyeluruh dari sistem yang akan dibuat. Dengan adanya gambaran tersebut, diharapkan dapat memberikan wawasan berkaitan dengan alat yang dibuat. Untuk menggambarkan sistem tersebut dapat dilakukan dengan menggunakan blok diagram. Blok diagram Sistem dapat dilihat pada Gambar 3.1.



Gambar 3.1 Blok Diagram Sistem

Secara garis besar sistem akuisisi sinyal listrik untuk monitoring harmonisa arus listrik dapat dilihat seperti pada gambar 3.1. Sinyal gangguan atau sinyal yang diduga mengandung harmonisa listrik akibat pemakain beban tak linear diambil dari saluran tenaga listrik.

Gelombang pada sistem tenaga yang akan dideteksi masih mempunyai amplitudo yang sangat besar, maka amplitudo dari gelombang tersebut harus diturunkan terlebih dahulu. Gelombang yang diamati disini adalah gelombang arus. Agar dapat melihat bentuk gelombang arus maka digunakan sensor arus.

Agar gelombang tersebut dapat digunakan sebagai input pada *ADC* maka terlebih dahulu dimasukkan kedalam rangkaian *differensial amplifier*. Dari besaran analog yang merupakan output dari rangkaian *differensial amplifier* diubah menjadi besaran digital oleh *A/D Converter* sehingga dapat diproses oleh komputer (*PC*). Didalam *PC* dilakukan proses pengolahan sinyal dengan metode *FFT*.

3.3 Perancangan Perangkat Keras (*Hardware*)

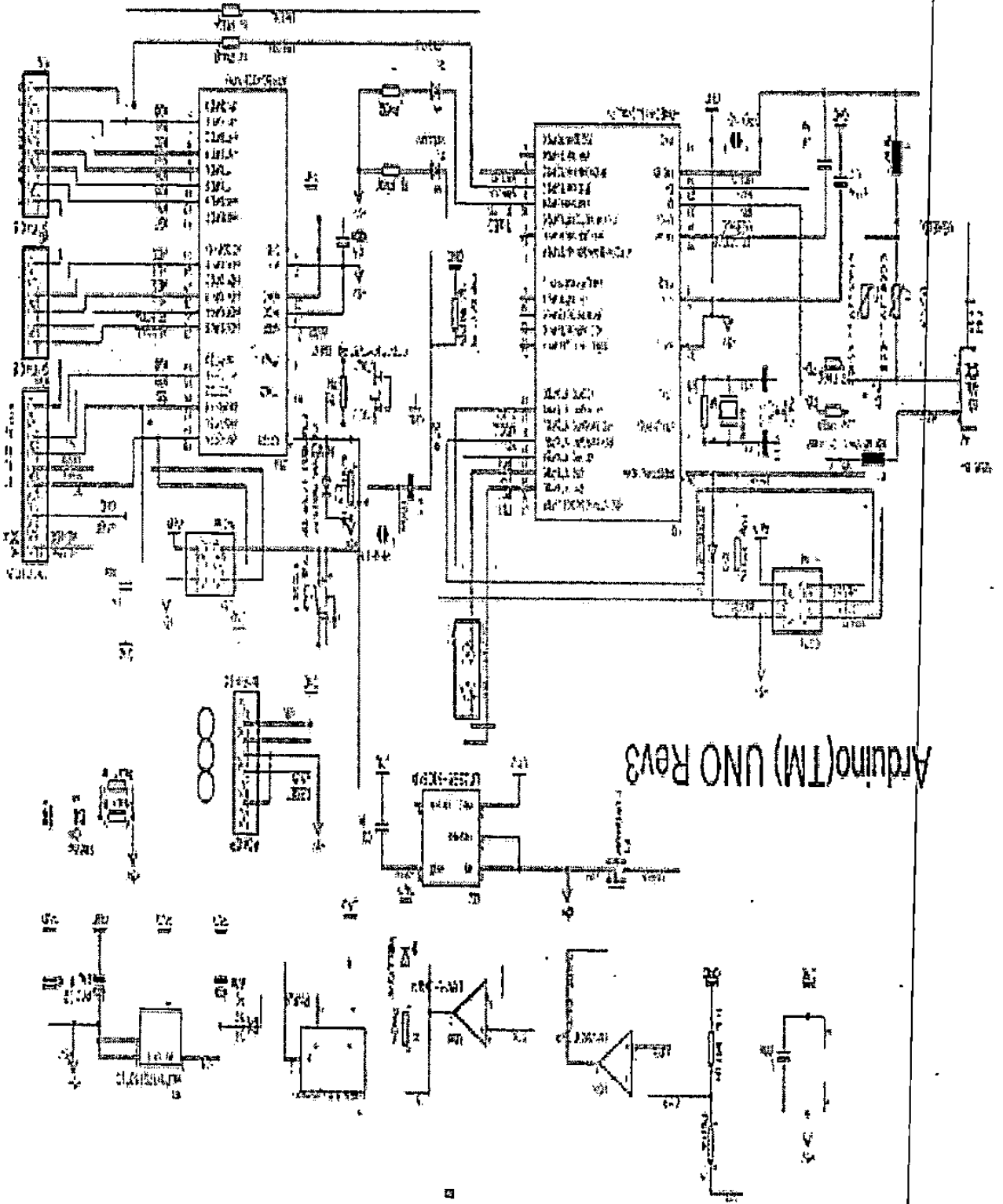
Perangkat keras pada alat ini yaitu Sistem Minimum Arduino Uno R3 sebagai pemroses dan pengolahan sinyal atau data dari output sensor SCT-019. Karena output dari sensor SCT-019 masih berupa sinyal analog yang sangat kecil outputannya, sehingga diperlukan rangkaian tambahan upaya bisa terbaca oleh input ADC Arduino.

3.3.1 Rangkaian Sistem Minimum Arduino

Arduino Uno R3 dalam perancangan alat pendeteksi ini berfungsi untuk pengendalian sensor dan untuk menerjemahkan input analog ke dalam sistem digital serta mengirimkan data tersebut ke PC atau Laptop melalui komunikasi *serial*. Pada Arduino Uno proses komunikasi data dari Arduino ke PC menggunakan komunikasi serial melalui Port USB yang terdapat pada Arduino. Arduino Uno sendiri menggunakan IC (*Integrated Circuit*) Mikrokontroler ATmega 328. Berikut merupakan rangkaian Sistem Minimum Arduino

senerti ditunjukkan Gambar 3.2 berikut:

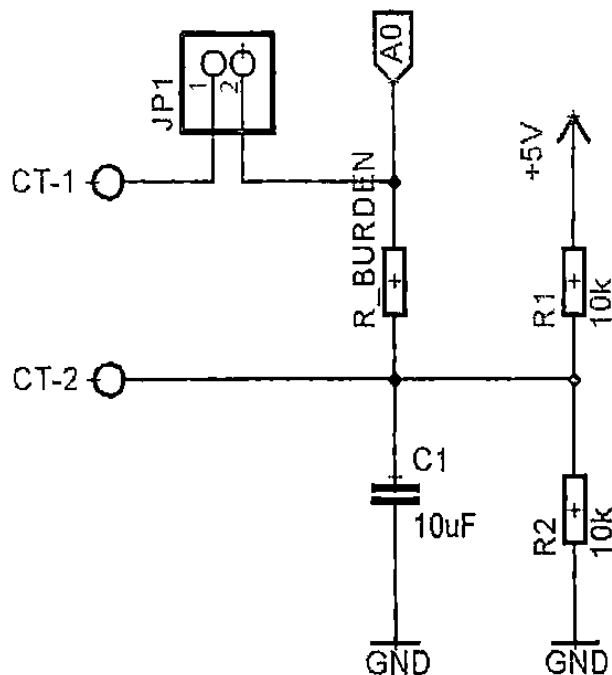
Arduino



Arduino(TM) UNO Rev3

3.3.2 Rangkaian Pengkondisi Sinyal Sensor SCT-019

Untuk menghubungkan output dari sensor CT-019 ke Arduino maka diperlukan rangkaian pengkondisi sinyal agar memenuhi persyaratan masukan dari input analog Arduino yaitu tegangan positif dengan range antara 0V-5V serta tegangan referensi ADC. Rangkaian berikut ini merupakan rangkaian beban resistor (*resistor burden*) maksud dari pemasangan resistor tersebut agar sensor SCT-019 dapat bekerja karena dipengaruhi oleh beban resistor, karena prinsip kerjanya menggunakan transformator arus. Gambar 3.3 menunjukkan gambar rangkaian pengkondisi sinyal dari sensor SCT-019.



Gambar 3.3 Rangkaian Pengkondisi Sinyal Sensor SCT-019

Pada Gambar 3.3 terlihat bahwa *burden resistor* yang terpasang adalah dengan nilai

- Konversi Arus RMS maksimum terhadap arus puncak dengan dikalikan $\sqrt{2}$

$$\text{Primary Peak Current} = \text{RMS} \times \sqrt{2}$$

$$\text{Primary Peak Current} = 200A \times 1.414$$

$$\text{Primary Peak Current} = 282,8A$$

- Pembagian arus puncak dengan jumlah kumparan di CT untuk memberikan arus puncak dalam kumparan sekunder.

SCT-019 memiliki 6000 kumparan, sehingga puncak arus sekunder akan:

$$\text{sekunder peak current} = (\text{primer peak current} - \text{banyak lilitan})$$

$$\text{sekunder peak current} = (282,8A / 6000)$$

$$\text{sekunder peak current} = 0,0471A$$

- Untuk memaksimalkan resolusi pengukuran tegangan diatas, *burden resistor* pada puncak arus maka tegangan AREF Arduino harus dibagi 2.

$$\text{Burden Resistor Ideal} = \left(\frac{\text{AREF}}{2} \right) / \text{Sekunder peak current}$$

$$\text{Burden Resistor Ideal} = \left(\frac{2.5}{0.0471A} \right)$$

$$\text{Burden Resistor Ideal} = 53\Omega$$

Karena dipasaran atau ditoko elektronik susah mendapatkan resistor dengan nilai 53Ω maka nilai tersebut diganti dengan resistor yang banyak terdapat dipasaran yaitu diganti dengan nilai 56Ω .

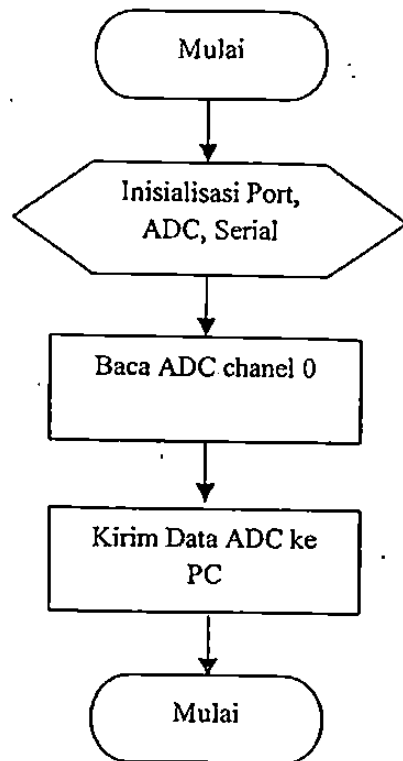
3.4 Perancangan Perangkat Lunak (*Software*)

Pada Perangkat Lunak (*Software*) terdiri dari pembuatan Algoritma pemrograman pada Board Arduino Uno R3. Proses pemrograman pada Board Arduino meliputi dari proses pembacaan ADC dari output sensor SCT-019 serta proses pengiriman data ke PC atau Laptop melalui Port USB menggunakan komunikasi serial. Selain itu pemrograman pada Board Arduino, serta proses pembuatan *User Interface* menggunakan *Lab View*.

3.4.1 Perancangan Pemrograman Pada Board Arduino

Karena output dari sensor SCT-019 masih berupa sinyal analog dengan range yang sangat kecil, namun setelah melewati rangkaian pengkondisi sinyal sensor SCT-019, data dari sensor SCT-019 dapat terbaca oleh input ADC Board Arduino. Syarat sinyal input yang dapat terbaca oleh input ADC Board Arduino adalah tegangan dengan maksimal tegangan sampai 5V. sebelum membuat program untuk membaca input ADC, terlebih dahulu membuat *flow chart* diagram sebagai acuan dalam pembuatan program tersebut. Gambar 3.4

memplihatkan *flow chart* diagram pada pemrograman Board A 1.



Gambar 3.4 Flow Chart Diagram Program Arduino

Pada Gambar 3.4 terlihat bahwa setelah proses pembacaan ADC maka data tersebut dikirimkan langsung ke Komputer. Proses pengiriman data ADC akan *update* setiap 100ms.

Potongan program berikut merupakan sebagian program yang ditanamkan pada Board Arduino untuk membaca tegangan output dari sensor SCT-019.

"Program Pendeteksi Harmonisa Arus"

```

const int analogInPin = 0;      // Analog input pin that the potentiometer
is attached to
const int analogOutPin = 9;    // Analog output pin that the LED is
attached to
int sensorValue = 0;           // value read from the pot
int outputValue = 0;           // value output to the PWM (analog out)
long previousMillis = 0;       // will store last time LED was updated

```



```

Serial.print(sensorValue);
Serial.print(';');
Serial.print(100);
Serial.print(';');
Serial.print(200);
Serial.print(';');
Serial.print(500);
Serial.print('\n');

void loop()
{
  unsigned long currentMillis;           // read the analog in value:
  sensorValue = analogRead(analogInPin); // map it to the range of
  the analog out:
  outputValue = map(sensorValue, 0, 1023, 0, 255); // change the analog
  out value:
  analogWrite(analogOutPin, outputValue); // print the results to the
  serial monitor:
  currentMillis = millis();
  if(currentMillis - previousMillis > 100)
  {
    sendData();
    previousMillis = currentMillis;
  }

  // wait 2 milliseconds before the next loop
  // for the analog-to-digital converter to settle
  // after the last reading:
  delay(2);
}

```

Karena pada perancangan program pada Arduino mengacu pada *library Emon Lib* maka untuk menghitung nilai kalibrasi digunakan persamaan berikut ini:

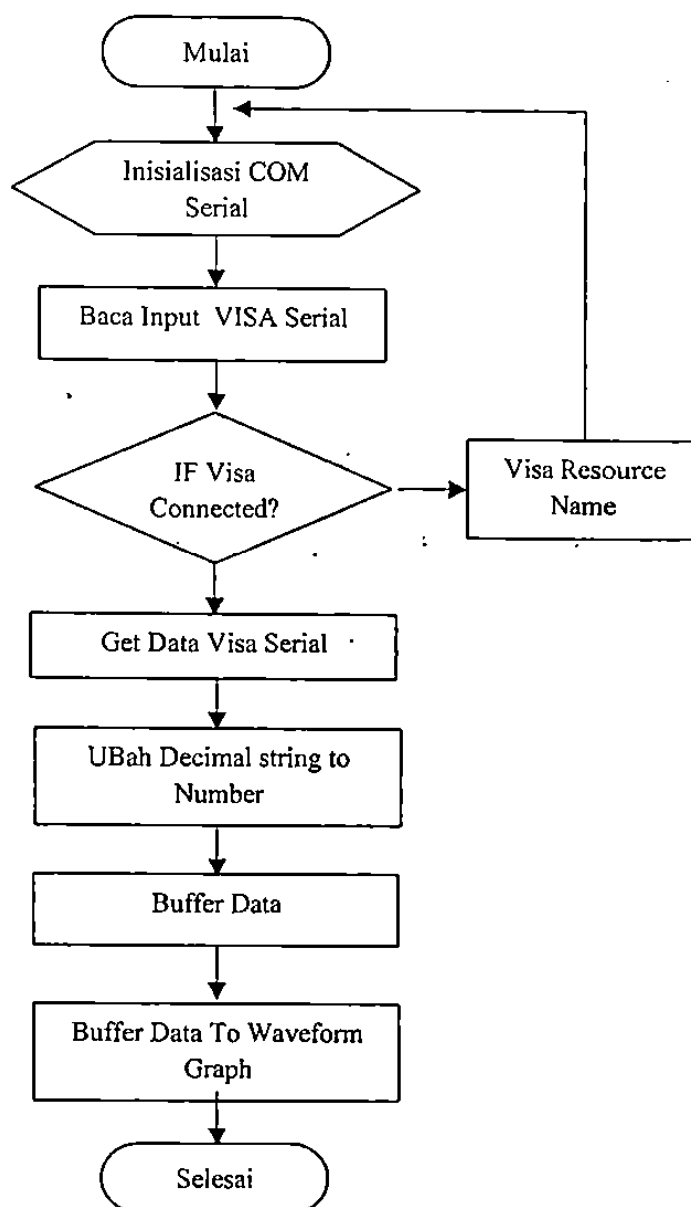
$$\text{Calibratin Value} = (I(\text{measured}) / I(\text{sensor}) / R(\text{burden}))$$

$$\text{Calibratin Value} = (282,8 / 0,04713 / 53)$$

$$\text{Calibratin Value} = 113,21$$

3.4.2 Perancangan Pemrograman pada *Lab View*

Lab View (Laboratory Virtual Instrumentation Engineering Workbench) adalah perangkat lunak komputer untuk pemrosesan dan visualisasi data dalam bidang akuisisi data, kendali instrumentasi serta automasi industry yang pertama kali dikembangkan oleh perusahaan *National Instruments* pada tahun 1986. Sistem pemrograman pada *Lab View* berbeda seperti pemrograman pada Arduino dimana pemrograman pada *Lab View* yaitu menggunakan sistem *Graphical Programming*. Gambar berikut ini menunjukkan *Flow Chart* diagram pada pemrograman di *Lab View*.

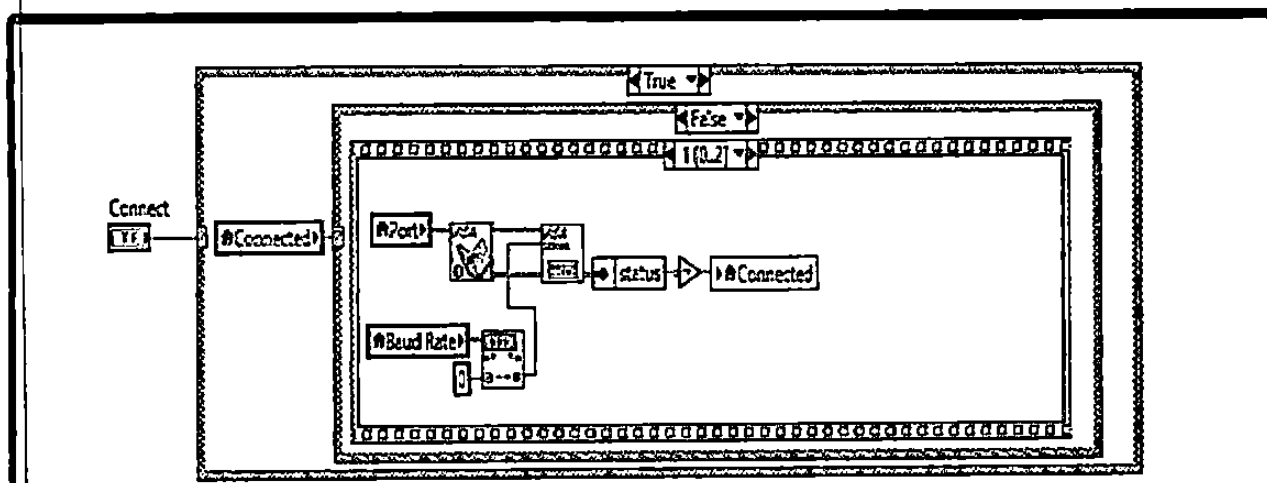


Gambar 3.5 Flow Chart Diagram pada Pemrograman *Lab View*

Pada Gambar 3.5 untuk membaca input data serial melalui Port USB maka digunakan fungsi atau komponen *Visa Serial*. Dimana pada komponen tersebut *Visa Serial* akan *men-scan* *COM Serial* mana yang sedang aktif dengan cara melihat *inputan COM* Arduino pada panel *Device Manager* pada komputer. Jika pada saat proses *connecting* data dari Arduino dengan *User Interface* pada *Lab View* tidak terjadi *error*, maka data selanjutnya akan dibaca dan diubah ke dalam bilangan desimal biasa dari bilangan *decimal string* agar data tersebut bisa dimasukkan ke dalam *buffer* data dan dapat ditampilkan ke dalam bentuk Grafik menggunakan komponen *Waveform Graph* pada *Lab View*.

3.4.3 Fungsi Case Struktur Pengkoneksian COM Serial

Pada saat pengkoneksian *COM Serial*, maka terjadi pembacaan *input Port USB* oleh *Visa Serial*. Pembacaan tersebut meliputi membaca *COM Serial* yang mana sedang aktif yang ada dipanel *device manager* pada komputer. Jika pada saat pengkoneksian tidak terjadi *error* maka data akan diteruskan dan diolah ke dalam bilangan desimal dan dimasukkan ke dalam *buffer* data untuk dapat ditampilkan ke dalam bentuk *Graphik*. Gambar 3.6 berikut menunjukkan potongan program pada *Lab View* untuk proses pengkoneksian *COM Serial USB Arduino* dengan *User Interface* pada *Lab View*.



Gambar 3.6 Potongan Program Pengkoneksian COM Serial USB Arduino dengan User Interface pada Lab View