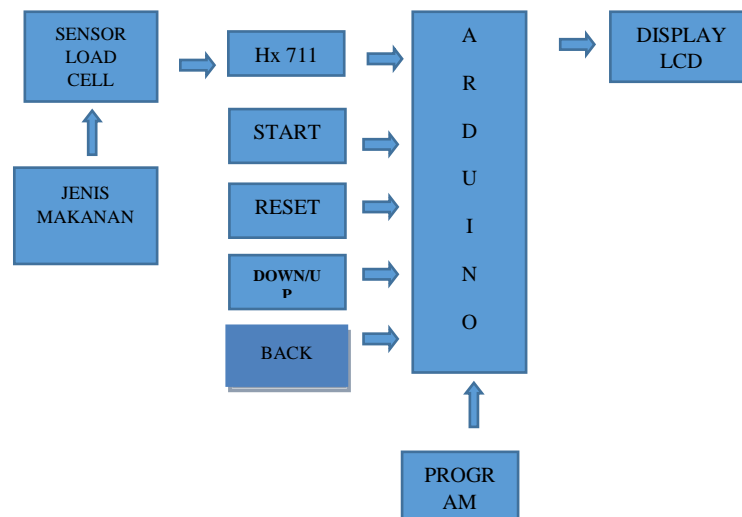


BAB III

METODOLOGI PENELITIAN

3.1. Diagram Blok Sistem

Diagram blok sistem ditunjukkan oleh Gambar 3.1. berikut.



Gambar 3.1. Diagram blok sistem.

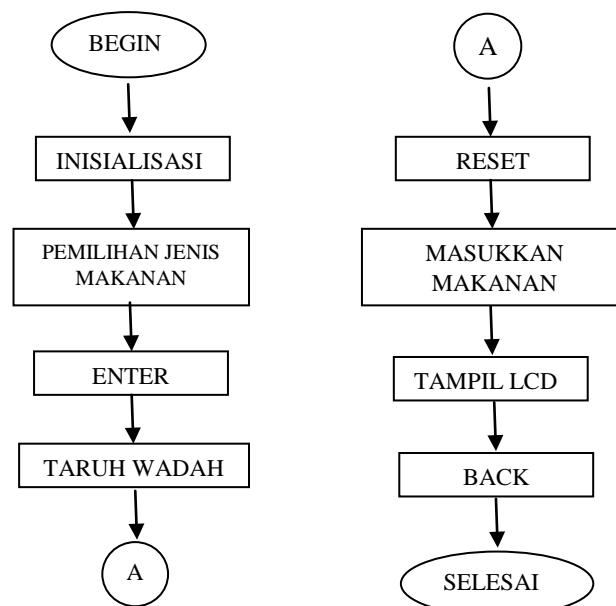
3.1.1. Cara kerja blok diagram

Awalnya dilakukan pemilihan jenis makanan. Berat jenis makanan akan di *sensor* oleh *load cell*. *Output* dari *load cell* masih kecil. Kemudian diolah oleh penguat HX 711 agar bisa terbaca oleh *Arduino Uno*. Kemudian *Arduino Uno* mengolah data yang telah diterima. Sebelumnya, *Arduino Uno* telah diprogram sehingga hasilnya dalam satuan kalori. Hasil tersebut, kemudian ditampilkan oleh LCD karakter. Selain itu, arduino juga menerima perintah dari tombol *start*, *reset*, *down/up*, dan *enter*.

Tombol – tombol tersebut berfungsi sebagai pemilihan jenis makanan, memasukkan jenis makanan, memberikan perintah memulainya proses perhitungan kalori dan sebagai zero atau mengencolkan kembali alat agar ketika dipakai hasilnya akurat dan tepat.

3.2. Diagram Alir Proses

Diagram alir proses ditunjukkan oleh Gambar 3.2. berikut.



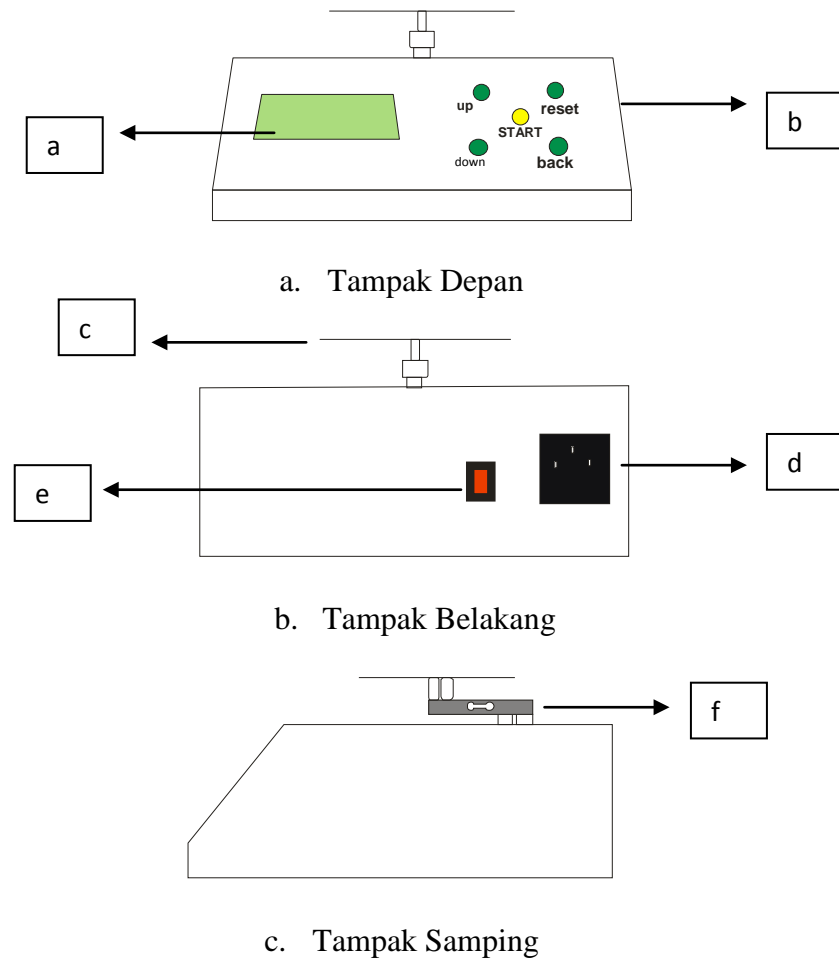
Gambar 3.2. Diagram alir.

3.2.1. Cara kerja diagram alir

Pertama nyalakan alat kemudian pilih jenis makanan yang akan dihitung kalorinya. Setelah dipilih tekan *enter* kemudian letakan terlebih dahulu wadah makanan yang akan ditimbang kemudian tekan tombol *reset* yang berfungsi sebagai pengencol timbangan. Setelah tampilan muncul angka 0 maka masukkan jenis makanan kemudian tekan tombol lalu hasil akan langsung muncul pada LCD.

3.3. Diagram Mekanis Sistem

Pada Gambar 3.3. berikut ini akan dibahas rancangan alat penghitung kalori pada makanan berupa desain dimensi alat tampak depan, belakang dan samping.



Gambar 3.3. Dimensi rancangan alat.

Keterangan :

- a : Tampilan untuk kalori dan berat.
- b : Tombol *up/down, reset, enter, back*.
- c : Tempat untuk meletakkan makanan
- d : Tempat kabel power

e : Tombol *on/off*.

f : *Sensor load cell*.

Dimensi Alat :

Panjang : 25 cm

Lebar : 20 cm

Tinggi : 15 cm

3.4. Pembuatan Casing *box* alat

3.4.1. Bahan :

- a. Plat aluminium
- b. Cat pilok putih dan hitam
- c. Akrilik
- d. Plat besi

3.4.2. Alat :

- a. *Cutter*
- b. Pemotong plat
- c. Penggaris besi
- d. Bor listrik
- e. Amplas halus
- f. Dempul
- g. Mesin las

3.4.3. Langkah pembuatan :

- a. Gambar pola pada plat aluminium sesuai desain yang kita inginkan.
- b. Sesuaikan dengan pola komponen yang akan di pasang.
- c. Potong dan lubangi pola pada plat menggunakan pemotong plat, lalu disesuaikan bentuk nya dan di bor.
- d. Potong plat besi untuk dudukan *sensor Load Cell* sesuai pola.
- e. Lalu las dengan *box* plat aluminium yang sudah dibentuk tadi.
- f. Potong akrilik bentuk bulat untuk meletakkan media makanan yang akan di timbang.
- g. Lubangi akrilik, dan pasang pada tumpuan plat besi yang sudah tergabung dengan *box* plat aluminium.
- h. Rapikan las – lasan dengan dempul agar halus.
- i. Tunggu kering, lalu warnai *box* dengan pilok.
- j. Setelah *box* kering, *box* siap untuk dipasang komponen.

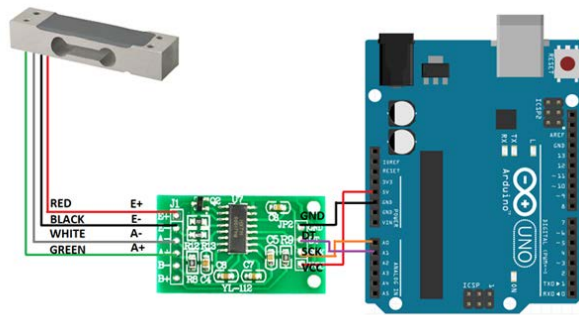
3.5. Merakit *Sensor Load Cell*

3.5.1. Bahan :

- a. Modul *sensor Load Cell*.
- b. Modul penguatan HX 711.
- c. Modul *Arduino Uno*.
- d. Pin sisir 8 buah.
- e. Kabel jumper *female male* 8 buah.
- f. *Casing sensor*.

3.5.2. Langkah perakitan

Rakit *sensor load cell*, modul HX 711, dan *arduino* sesuai dengan Gambar 3.4. berikut.



Gambar 3.4. Perakitan modul *sensor load cell*.

- Kupas kabel pada *sensor load cell*.
- Sambungkan dengan *white housing female*.
- Pasang pin sisir pada modul HX 711 dan *Arduino Uno*.
- Solder pin sisir yang telah terpasang.
- Sambungkan kabel *loadcell* dengan modul HX 711 sesuai gambar di atas.
- Sambungkan modul HX 711 dengan *Arduino Uno* menggunakan jumer *female*.

3.6. Pembuatan *Power Supply*

3.6.1. Bahan :

- Diode* 4a
- Trafo* 0,5 A
- Kapasitor* 3300 μf (2)
- Kapasitor* 470 μf (2)

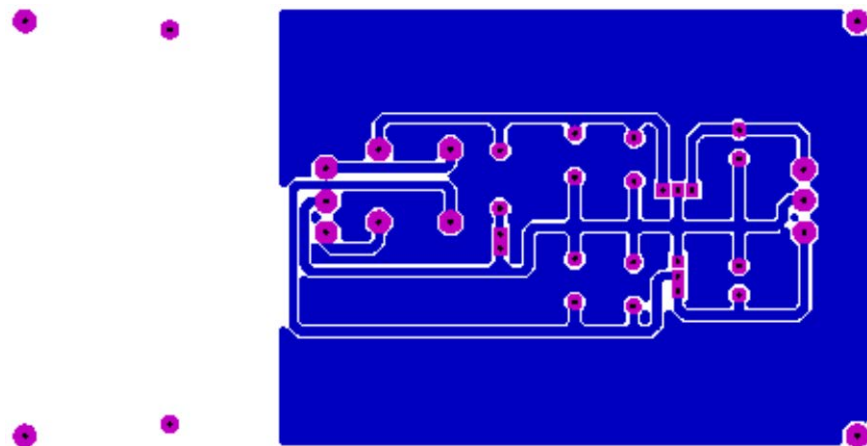
- e. *Kapasitor non polar* 104 (4)
- f. *IC regulator* 7812
- g. *LED* (1)

3.6.2. Alat :

- a. Papan pcb
- b. Solder
- c. Timah
- d. Penyedot timah
- e. Gunting

3.6.3. Langkah perakitan

- a. Rangkai sistematis rangkaian *power supply* dengan menggunakan aplikasi pada laptop, aplikasi yang digunakan pada pembuatan modul ini adalah *proteus*.
- b. Setelah sistematis rangkaian jadi, tahap selanjutnya membuat *lay out* nya dan disablon ke papan *pcb*. Untuk gambar *lay out power supply* pada papan *pcb* dapat dilihat pada gambar 3.5. di bawah ini:



Gambar 3.5. *Layout Power supply.*

c. Rakit komponen yang dibutuhkan dengan menggunakan solder.

Rangkaian *power supply* pada modul ini berfungsi sebagai *supply* tegangan ke semua rangkain yang menggunakan tegangan DC. Prinsip kerja *power supply* adalah mengubah tegangan AC menjadi tegangan DC dengan menggunakan *transformator* sebagai penurun tegangan dan dioda sebagai komponen yang berfungsi sebagai penyearah tegangan. Pada modul ini *power supply* akan mengubah tegangan AC menjadi DC sebesar 12 VDC dengan menggunakan *IC regulator 78012*. Adapun tegangan 12 VDC digunakan untuk inputan pada *Arduino Uno*.

3.7. Pembuatan program

Dalam pembuatan program penulis menggunakan bahasa *arduino* atau bahasa C, berikut adalah isi program yang di buat untuk mengisi *microcontroller* atau dalam bahasa arduino biasa disebut *sketch* :

SKETCH PROGRAM LENGKAP

```
#include <LiquidCrystal.h>
#include <math.h>
#include "HX711.h"

#define tombol1 8
#define tombol2 9
#define tombol3 A3
#define tombol4 A4
#define tombol5 A5

#define __NASI 0 //180
#define __NASI_TIM 1 //120
#define __NASI_MERAH 2 //149
```



```

#define __PEPAYA 3 //46
#define __PISANG 4 //127
#define __APEL 5 //58
#define __BAYAM_REBUS 6 //23
#define __KANGKUNG_REBUS 7 //22
#define __KENTANG_REBUS 8 //62
#define __AYAM_GORENG 9 //270
#define __EMPAL_GORENG 10 //348
#define __TEMPE_GORENG 11 //350
#define __KEJU 12 //326

char str[16];
char strFloat[10];
float kaloriList[13] = {1.8, 1.2, 1.49, 0.46, 1.27, 0.58,
0.23, 0.22, 0.62, 2.70, 3.48, 3.50, 3.26};
char* kaloriName[13] = {"      NASI      ",
                        "      NASI TIM   ",
                        "      NASI MERAH ",
                        "      PEPAYA      ",
                        "      PISANG      ",
                        "      APEL        ",
                        "      BAYAM REBUS ",
                        "      KANGKUNG REBUS ",
                        "      AYAM GORENG  ",
                        "      EMPAL GORENG ",
                        "      TEMPE GORENG ",
                        "      KEJU        "};

char choice=0;

LiquidCrystal lcd(2, 3, 4, 5, 6, 7);
float berat,kalori;

// HX711.DOUT - pin #A1
// HX711.PD_SCK - pin #A0

```

```
HX711 scale(A1, A0); // parameter "gain" is omitted;
the default value 128 is used by the library

void resetTimbangan(){
    scale.tare();
    lcd.clear();
    lcd.setCursor(0,0);
    lcd.print("      RESET      ");
    delay(500);
}

void bacaTimbangan() {
    if(digitalRead(tombol3)==0){
        resetTimbangan();
    }
}

void bacaTombol()
{
    char tombol1Var = digitalRead(tombol1);
    char tombol2Var = digitalRead(tombol2);
    char tombol3Var = digitalRead(tombol3);
    char tombol4Var = digitalRead(tombol4);
    char tombol5Var = digitalRead(tombol5);
    }
}

if(!tombol3Var){
    resetTimbangan();
}

if(!tombol1Var){
    choice++;
}
```

```
        if(choice>12)
            choice=0;

        delay(200);
    }

    if(!tombol2Var){
        choice--;
        if(choice<0)
            choice=12;

        delay(200);
    }
    delay(50);
}

void setup() {
    pinMode(tombol1, INPUT);
    pinMode(tombol2, INPUT);
    pinMode(tombol3, INPUT);
    pinMode(tombol4, INPUT);
    pinMode(tombol5, INPUT);

    digitalWrite(tombol1, HIGH);
    digitalWrite(tombol2, HIGH);
    digitalWrite(tombol3, HIGH);
    digitalWrite(tombol4, HIGH);
    digitalWrite(tombol5, HIGH);

    lcd.begin(16, 2);
    lcd.clear();

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

```

Serial.begin(38400);
Serial.println("HX711 Demo");

Serial.println("Readings:");
delay(2000);
lcd.clear();
}

void loop() {
  lcd.setCursor(0,0);
  lcd.print(" JENIS MAKANAN: ");

  lcd.setCursor(0,1);
  lcd.print(kaloriName[choice]);
  bacaTombol();
}

```

3.8. Jenis Penelitian

Jenis penelitian yang dipakai untuk melakukan penelitian ini adalah *after only design*. Pada rancangan ini, peneliti hanya melihat hasil tanpa mengukur keadaan sebelumnya.

Dalam penelitian terdapat kelompok kontrol. Kelemahan dari rancangan ini adalah tidak diketahui keadaan awalnya, sehingga hasil yang didapatkan sulit untuk disimpulkan.

X	O
Non Random -----	
(-)	O

X = *Treatmen*/perlakuan yg diberikan (*variabel Independen*)

0 = *Observasi* (*variabel dependen*)

(-) = Kelompok control

3.9. *Variabel Penelitian*

3.9.1. *Variabel Bebas*

Sebagai *variabel* bebas yaitu berat jenis makanan.

3.9.2. *Variabel Tergantung*

Sebagai *variable* tergantung adalah *sensor* berat *loadcell*.

3.9.3. *Variabel Terkendali*

Sebagai *variabel* terkendali yaitu *Arduino Uno*.

3.10. *Rumus Statistik*

Pengukuran untuk kalibrasi dilakukan sebanyak 20 kali dalam percobaan dengan membandingkan dengan alat yang berstandar dan dicari nilai standar *deviasi* (*STDV*), angka ketidakpastian dan juga *error* dengan rumus sebagai berikut :

3.10.1. *Rata – rata*

Rata – rata digunakan untuk menghitung hasil pembagian dari jumlah data yang di ambil atau di ukur dengan banyaknya pengambilan data atau banyaknya pengukuran.

$$\text{Rata – Rata } (\bar{X}) = \frac{\sum Xi}{n} \quad [3 - 1]$$

Dimana :

$$\bar{X} = \text{rata – rata}$$

$\sum X_i$ = Jumlah nilai data

N = Banyak data (1,2,3,...,n)

3.10.2. Simpangan

Simpangan digunakan untuk menghitung selisih dari rata-rata nilai harga yang dikehendaki dengan nilai yang diukur. Berikut rumus dari simpangan :

$$\text{Simpangan} = Y - \bar{X} \quad [3-2]$$

Dimana :

Y = berat setting

\bar{X} = rerata

3.10.3. Error (%)

Error (kesalahan) digunakan untuk menghitung selisih antara *mean* terhadap masing-masing data. Rumus *error* adalah:

$$\text{Error\%} = \left(\frac{\text{DataSetting} - \text{Re rata}}{\text{Datasetting}} \right) \times 100\% \quad [3-3]$$

3.10.4. Standar Deviasi

Standar *deviasi* digunakan untuk menghitung suatu nilai yang menunjukkan tingkat (derajat) variasi kelompok data atau ukuran standarpenyimpangan dari *meannya*.

Rumus standar *deviasi* (SD) adalah:

$$SD = \sqrt{\frac{\sum_{i=1}^n (X_i - \bar{X})^2}{(n-1)}} \quad [3-4]$$

Dimana :

SD = standar *deviasi*.

\bar{X} = nilai yang dikehendaki.

n = banyak data.

3.10.5. Ketidakpastian (UA)

Ketidakpastian digunakan untuk menghitung kesangsian yang muncul pada tiap hasil. Atau pengukuran biasa disebut sebagai kepresisian data satu dengan data yang lain.

Rumus dari ketidakpastian adalah sebagai berikut:

$$\text{Ketidakpastian} = \frac{stdv}{\sqrt{n}} \quad [3-5]$$

Dimana :

STDV = Standar *Deviasi*

n = banyaknya data

3.11. Persiapan Bahan

Adapun komponen – komponen penting yang digunakan dalam pembuatan modul, antara lain :

- a. Modul penguatan HX 711

- b. *Sensor Load cell*
- c. *LCD 2x16*
- d. *Lampu LED*
- e. *IC ATmega 328*
- f. *Soket IC ATmega 328*
- g. *Minimum Sistem Arduino*
- h. *USB Female*
- i. *Kabel USB*
- j. *Conector 6 pin*
- k. *Conector 5 pin*
- l. *Conector 4 pin*
- m. *Conector 2 pin*
- n. *Push Button*
- o. *Diode 4a*
- p. *Trafo 0,5 A*
- q. *Kapasitor 3300 μ f (2)*
- r. *Kapasitor 470 μ f (2)*
- s. *Kapasitor non polar 104 (4)*
- t. *IC regulator 7812*

3.12. Peralatan yang digunakan

Sebagai sarana pendukung dalam pembuatan tugas akhir ini, ada beberapa peralatan yang dibutuhkan antara lain sebagai berikut :

- a. *Solder listrik*

- b. *Atractor* (Penyedot Timah)
- c. *Toolset*
- d. Bor PCB
- e. Timah (*Tenol*)
- f. *Multimeter*
- g. Komputer
- h. Lem tembak
- i. Kabeltis