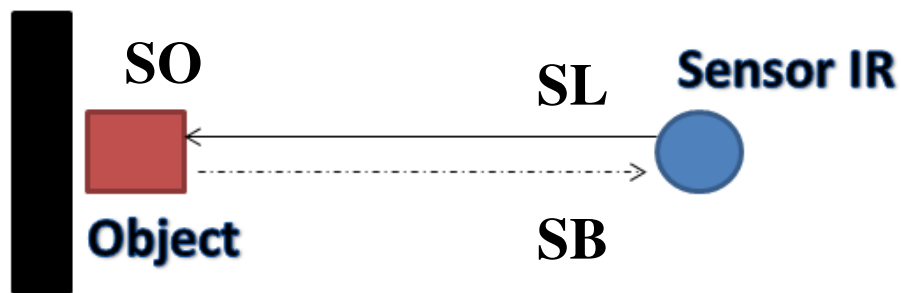


III. METODE PENELITIAN

3.1. Metode Penelitian



Keterangan :

SO = Suhu Objek

SB = Suhu Balik

SL = Suhu Lingkungan

Gambar 3.1 Konsep cara kerja sensor infra merah

Gambar 3.1 menggambarkan prinsip cara kerja dari sensor infra merah yang dimana terdapat sebuah objek yang akan diukur. Suatu objek yang mempunyai suhu diatas 0°C akan memancarkan sinar Infra merah. Cara kerja sensor ini yaitu akan menangkap sinar infra merah yang dipancarkan suatu benda. Suhu yang ditangkap akan memiliki variasi tinggi rendahnya suhu benda yang diukur. Untuk

membedakan antara suhu lingkungan dengan suhu objek, sensor MLX 90620 memiliki internal sensor didalamnya. Sehingga sensor ini dapat mengukur suhu lingkungan yang kemudian akan dibandingkan dengan suhu benda objek yang diukur. Suhu balik kemudin dapat diketahui dengan penambahan antra suhu lingkungan dengan suhu objek yang diukur. Suhu balik dapat didefinisikan sebagai berikut :

$$\mathbf{SB = SL + SO} \quad (3-1)$$

dimana ; $SL > TO$

$SL = TO$

$SL < TO$

Dari rumus (3-1) dapat didefinisikan suhu balik yaitu suhu sumber ditambah suhu objek, Dimana suhu sumber dapat lebih besar dari suhu objek, suhu sumber samadengan suhu objek atau suhu sumber dapat kurang dari suhu objek. Selisih dari suhu sumber dan suhu balik dapat diterjemahkan objek benda yang sedang diukur apakah suhunya lebih tinggi ataupun lebih rendah.

3.2.Waktu dan Tempat Penelitian

Penelitian dan perancangan tugas akhir ini dilaksanakan mulai dari Juli 2014 sampai Februari 2015, bertempat di Laboratorium Terpadu Teknik Elektro, Jurusan Teknik Elektro, Universitas Lampung. Jadwal kegiatan penelitian rancang

bangun pengukuran panas peralatan listrik dengan *thermopille array* MLX 90620 berbasis *mikrokontroller* Atmega 2560 dan Raspberry Pi adalah sebagai berikut :

Tabel 3.1 Tabel kegiatan penelitian

Kegiatan	Juli 2014				Agustus 2014				September 2014			
	I	II	III	IV	I	II	III	IV	I	II	III	IV
Studi Literatur												
Perancangan Hardware												
Pembuatan Program												
Ujicoba Hardware & program												
Seminar Proposal												
Pengambilan Data I												
Pengambilan Data II												
Pengambilan Data III												
Pengambilan Data IV												
Analisa & Pembahasan												
Seminar Hasil												
Komprehensif												

Kegiatan	Oktober2014				November 2014				Desember 2014			
	I	II	III	IV	I	II	III	IV	I	II	III	IV
Studi Literatur												
Perancangan Hardware												
Pembuatan Program												
Ujicoba Hardware & program												
Seminar Proposal												
Pengambilan Data I												
Pengambilan Data II												
Pengambilan Data III												
Pengambilan Data IV												
Analisa & Pembahasan												
Seminar Hasil												
Komprehensif												

3.3.Alat dan Bahan

Adapun alat dan bahan yang digunakan dalam ini penelitian rancang bangun ini adalah sebagai berikut :

1. Hardware

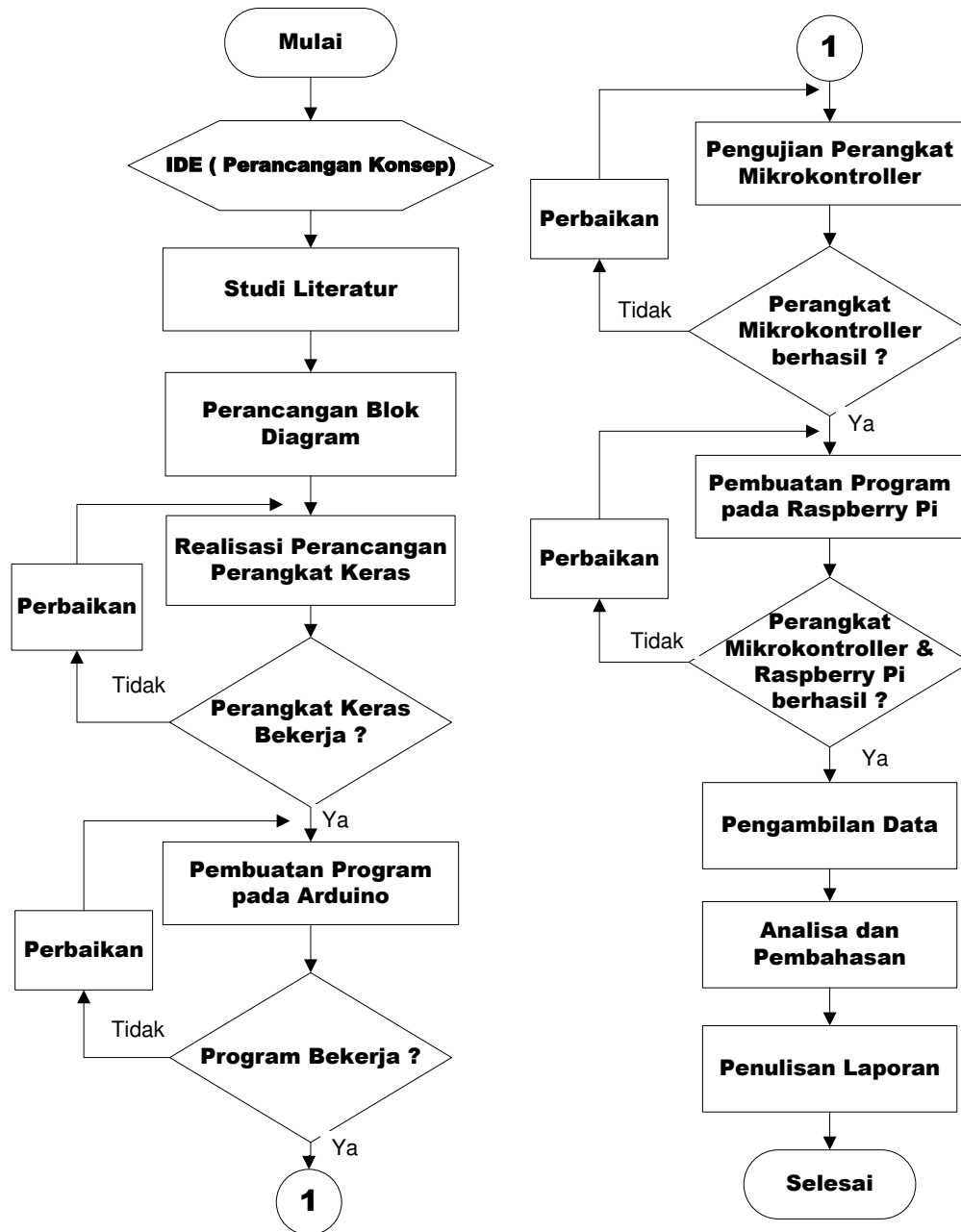
- a. Personal komputer / Laptop
- b. Arduino Mega
- c. Raspberry Pi
- d. 1 Sensor Suhu LM 35
- e. 1 Sensor Thermopille Array MLX 90620
- f. 2 sensor Hc – SR04
- g. 1 Data logger shield
- h. 1 buah catu daya power bank 10.000 mAh
- i. LED indikator

2. Software

- a. Ide Arduino digunakan untuk proses pemrograman mikrokontroller Arduino Uno.
- b. Microsoft excell 2007 digunakan sebagai pencatatan data yang dilakukan oleh data logger.
- c. Python 2.7 digunakan sebagai proses pemrograman pada Raspberry Pi untuk melakukan pengolahan citra matriks.

3.4. Metode/Prosedur Kerja

Dalam penelitian ini, langkah-langkah kerja yang dilakukan adalah sebagai berikut:



Gambar 3.2. Diagram alir penelitian

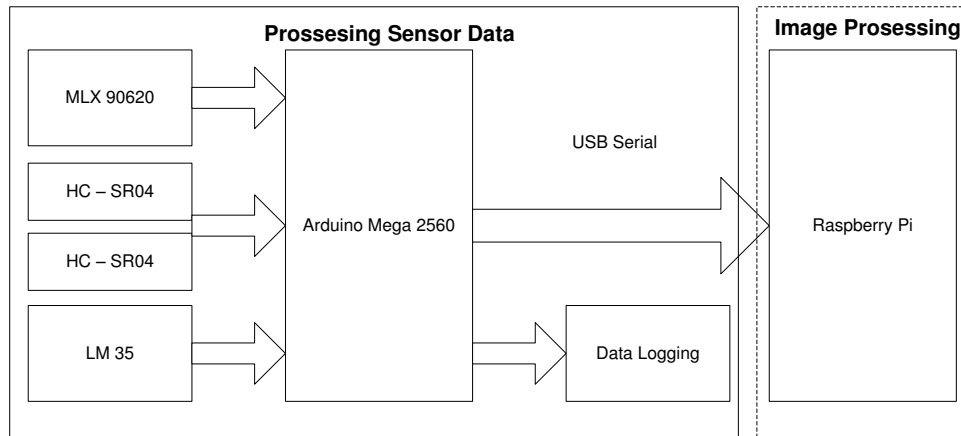
3.4.1. Studi Literatur

Studi literatur dilakukan untuk mempelajari berbagai sumber referensi atau teori yang berkaitan dengan sistem perancangan pengukuran panas peralatan listrik menggunakan sensor *thermopille array* MLX90620, antara lain :

- a. Bahasa pemrograman yang digunakan dalam mikrokontroler AVR, pemrograman mikrokontroller Arduino, dan bahasa python.
- b. Aplikasi-aplikasi mikrokontroler AVR.
- c. Mempelajari Thermografi.
- d. Penggunaan FLIR sebagai pencitraan visual panas.
- e. Mempelajari pengolahan citra melalui python.
- f. Pengolahan citra matriks output data melalui python.
- g. Pengolahan output sensor MLX90620 untuk pencatatan data secara berkala.
- h. Pengolahan output sensor lainnya sebagai pembanding data sensor MLX90620.

3.4.2 Perancangan Blok Diagram

Perancangan blok diagram ini dilakukan agar lebih mudah merealisasikan alat yang akan dibuat. Secara umum gambar blok diagram sistem dapat di lihat pada gambar 3.3. dibawah ini



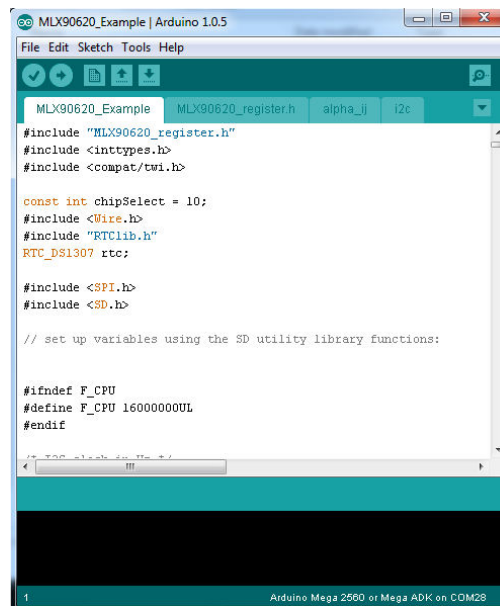
Gambar 3.3 Blok diagram sistem

3.4.3. Perancangan Perangkat Keras

Implementasi rangkaian dilakukan dengan tahapan-tahapan sebagai berikut:

- Menentukan sensor dan komponen yang digunakan dalam setiap bagian perangkat yang akan dibuat.
- Merangkai perbagian sensor dan komponen ke dalam board Arduino mega 2560.
- Melakukan pengujian terhadap rangkaian yang telah dibuat pada masing-masing bagian sensor dan komponen tersebut.
- Menggabungkan rangkaian perblok yang telah diuji pada sebuah papan percobaan (*project board*), melakukan pengujian ulang setelah dilakukan penggabungan rangkaian.

3.4.4. Pembuatan Program Mikrokontroller



```

MLX90620_Example | Arduino 1.0.5
File Edit Sketch Tools Help
MLX90620_Example MLX90620_register.h alpha_ij i2c
#include "MLX90620_register.h"
#include <inttypes.h>
#include <compat/twi.h>

const int chipSelect = 10;
#include <Wire.h>
#include "RTClib.h"
RTC_DS1307 rtc;

#include <SPI.h>
#include <SD.h>

// set up variables using the SD utility library functions:

#ifndef F_CPU
#define F_CPU 16000000UL
#endif
  
```

Gambar 3.4. Pemrograman pada IDE Arduino

Pemrograman pada IDE Arduino ini menggunakan bahasa C. Pada IDE Arduino terdapat *library-library* yang dapat membantu penulisan pemrograman. *Interface* pada IDE arduino ini sangat memudahkan bagi pemula dikarenakan arduino sudah *open source*. Sebelum tulisan pemrograman yang kita buat di upload ke *mikrokontroler* harus melewati *compiler*. *Compiler* ini yang akan memverifikasi apakah pemrograman yang ditulis sudah benar ataupun salah. Jika salah *compiler* akan memberitahukan kepada pengguna letak kesalahan penulisan pada pemrograman. *Compiler* itu sendiri berguna untuk merubah file C ke dalam bentuk HEX sehingga tulisan yang dibuat dapat diidentifikasi oleh *mikrokontroler*.

3.4.5. Pengujian Perangkat Keras dan Program

Pengujian perangkat keras dan program ini dilakukan untuk mengetahui seberapa besar nilai pembacaan suhu pada sensor MLX 90620. Pengujian ini dilakukan beberapa tahap diantaranya :

- a. Komunikasi *serial kontroler* dengan komputer.
- b. Pengendalian pengukuran dengan MLX90620.
- c. Penyimpanan data melalui data *logging shield*.
- d. Komunikasi *serial* antara Raspberry Pi dengan Arduino Mega 2560
- e. Pengolahan citra dari data hasil sensor MLX 90620.
- f. Pengujian temperature ruangan dengan sensor MLX 90620.
- g. Pengujian temperature ruangan menggunakan sensor MLX 90620 dibandingkan dengan sensor LM 35.
- h. Pengujian temperature suatu objek misalnya *solder*, cangkir dengan air hangat, dan komputer dengan sensor MLX 90620. Dari pengujian ini didapat data seharusnya objek yang diujikan *terdetect* dengan sensor MLX 90620.

3.4.6. Pembuatan Program Python pada Raspberry Pi

```

xmp5.py - C:\Users\Akhmad Harry Susanto\Desktop\python TA\wmp5.py
import matplotlib.pyplot as plt
from pylab import *
import numpy as np
import time

def draw(splits, i):
    fig = figure(1)
    plt.imshow(np.array(splits), interpolation='nearest', cmap='jet')
    plt.xticks(np.arange(0,16), ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J'])
    plt.yticks(np.arange(0,4), ['1', '2', '3', '4'])
    plt.show()
    plt.savefig(str(i)+'nearest'+'.png')

    fig = figure(2)
    plt.imshow(np.array(splits), interpolation='bilinear', cmap='jet')
    plt.xticks(np.arange(0,16), ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J'])
    plt.yticks(np.arange(0,4), ['1', '2', '3', '4'])
    plt.savefig(str(i)+'bilinear'+'.png')

    fig = figure(3)
    plt.imshow(np.array(splits), interpolation='bicubic', cmap='jet')
    plt.xticks(np.arange(0,16), ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J'])
    plt.yticks(np.arange(0,4), ['1', '2', '3', '4'])
    plt.savefig(str(i)+'bicubic'+'.png')

##conf_arr = [[200.21, 31.29, 31.09, 32.09, 31.21, 29.67, 29.67, 30.28, 33.08, 3
##
## [31.81, 31.58, 33.60, 32.11, 32.26, 31.85, 31.97, 35.78, 32.08, 30
##
## [30.18, 30.10, 30.74, 33.02, 30.31, 32.53, 32.26, 31.98, 30.03, 32
##
## [27.95, 30.37, 29.57, 32.96, 28.22, 30.00, 30.56, 34.69, 31.15, 30
##
###plt.imshow(np.array(conf_arr), interpolation='nearest', cmap='jet')
###plt.xticks(np.arange(0,16), ['A', 'B', 'C', 'D', 'E', 'F', 'G', 'H', 'I', 'J'])
###plt.yticks(np.arange(0,4), ['1', '2', '3', '4'])
##
##figure(1)

nyoba.py - C:\Users\Akhmad Harry Susanto\Desktop\python TA\nyoba.py
import serial
import re
ser = serial.Serial('COM28', 9600)
from xmp5 import draw
iter = 0
#import matplotlib.pyplot as plt
#plt.ion()

while iter < 5 :
    baca = ser.readline()
    splits = baca.split(" ")
    splits.remove(splits[-1])
    print '#####', splits, '\n', len(splits)
    n = 4
    b0 = []
    b1 = []
    b2 = []
    b3 = []
    for i in range(64):
        if (i+1) % 4 == 1:
            b0.append(float(splits[i]))
        elif (i+1) % 4 == 2:
            b1.append(float(splits[i]))
        elif (i+1) % 4 == 3:
            b2.append(float(splits[i]))
        elif (i+1) % 4 == 0:
            b3.append(float(splits[i]))
    splits = [b0,b1,b2,b3]
    print splits
    draw(splits, iter)

#my_string = '\n'
#try:
#    x = float(my_string)
#    print splits
#except ValueError:
#    print "terjadi kesalahan mengubah float untuk : %s" % my_strin
iter += 1

```

Gambar 3.5. Pemrograman pada Python

Pembuatan program Raspberry Pi ini dilakukan dengan menggunakan program Python 2.7. Pembuatan program ini dilakukan untuk pemrosesan citra dari data yang diterima melalui serial menjadi matriks berwarna yang menginterpretasikan suhu dari benda yang diukur. Matriks berwarna ini mengidentifikasi tingkat kepanasan suatu objek yang dituju atau yang diukur. Warna biru menandakan objek tersebut mempunyai *temperature* normal sedangkan *temperature* merah menandakan objek yang diukur memiliki tingkat *temperature* yang tinggi.

3.4.7. Pengujian Sistem Pengukuran

Pengujian sistem pengukuran ini dilakukan dalam beberapa tahap. Hal ini dimaksudkan agar pembacaan yang dilakukan oleh sensor MLX 90620 merupakan nilai yang benar dari *temperature* objek yang diukur. Untuk menguji kebenaran nilai tersebut dilakukan beberapa cara diantaranya :

a. Melakukan kalibrasi dengan sensor LM 35

Pengujian ini dimaksudkan melihat akurasi pembacaan temperature lingkungan dari sensor MLX 90620 terhadap sensor LM 35. Dalam pengukuran ini sistem tidak melakukan pengukuran terhadap benda panas melainkan sistem akan melakukan pengukuran terhadap suhu lingkungan. Hasil yang didapat nantinya akan merepresentasikan akurasi dari pembacaan setiap pixel sensor MLX 90620.

b. Melakukan kalibrasi dengan *Infrared Temperature* Fluke

Dalam melakukan kalibrasi dengan *Infrared Temperature* Fluke ini menggunakan sebuah plat besi berukuran 18 cm x 58 cm. Dalam melakukan kalibrasi dengan *Infrared Temperature* ini berbeda dengan kalibrasi dengan sensor LM 35. Dalam kalibrasi ini, plat besi diberi ukuran 1 cm x 1 cm dengan menyesuaikan matriks yang dihasilkan dari sensor MLX 90620. Matriks yang dihasilkan dari MLX 90620 ini yaitu 4 baris dengan 16 kolom. Sebuah plat besi dalam pengujian kali ini menggunakan benda seterika akan ditempelkan ke plat besi tersebut. Kemudian plat besi ini akan diukur menggunakan *Infrared Temperature* Fluke setiap kotak satu per satu. Selanjutnya sistem akan melakukan pengukuran dengan cara memindahkan jarak pengukuran antara sistem dengan objek plat ini. Sehingga nantinya akan mendapatkan jarak optimal yang menyerupai pengukuran dari *Infrared Temperature* Fluke.

3.4.8. Analisa dan Pembahasan

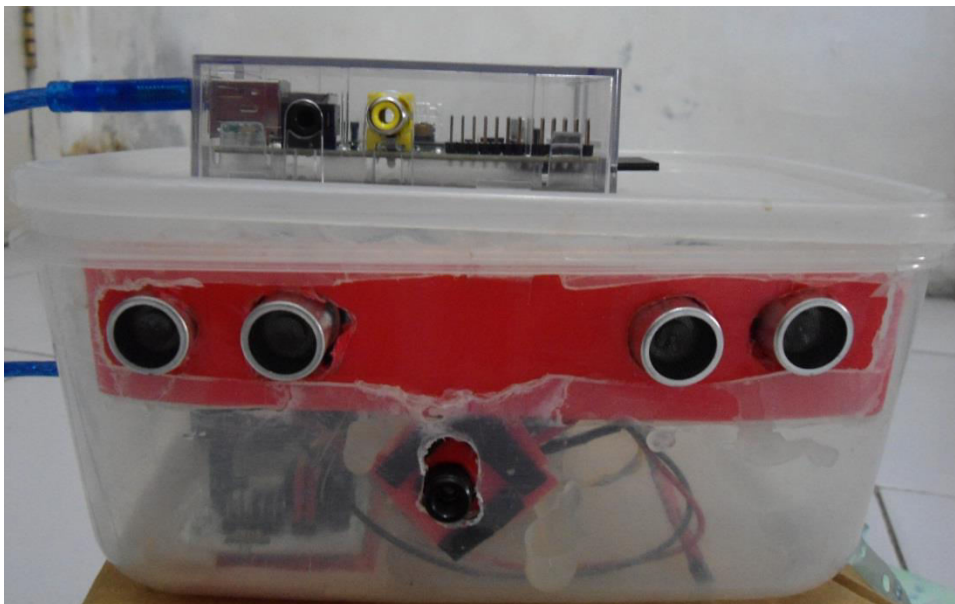
Analisa dilakukan dengan cara membandingkan antarahasil pengukuran panas transformator yang didapat dengan hasil *datasheet* dari transformator distribusi yang diukur sehingga di dapat kesimpulan mengenai sistem pengukuran panas transformator ini efektif dan efisien.

3.4.9 Pembuatan Laporan

Akhir dari tahap penelitian ini adalah pembuatan laporan dari semua kegiatan penelitian yang telah dilakukan.

3.5. Spesifikasi dan Perancangan Sistem

3.5.1. Karakteristik Sistem



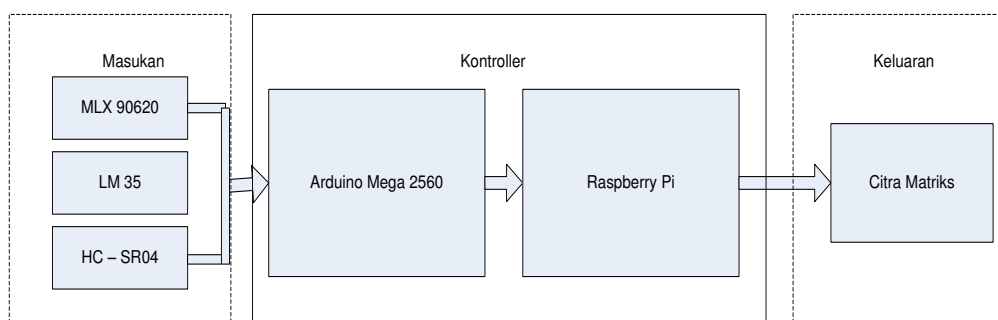
Gambar 3.6. Perancangan bentuk sistem pengukuran panas peralatan listrik

Spesifikasi alat yang dibuat adalah sebagai berikut :

1. Pengolahan sensor *MLX90620*, sensor *LM35*, dan *Sensor HC-SR04* dilakukan di mikrokontroler Arduino Mega 2560.
2. Sensor *MLX 90620* memiliki 64 *pixel* sensor yang terdiri dari 4 baris dan 16 kolom *pixel*.
3. Hasil keluaran sensor *MLX90620* kemudian dikirim ke *Raspberry Pi* melalui *USB serial* untuk diolah citra yang dihasilkan oleh sensor tersebut.
4. Citra yang di peroleh kemudian di interpolasi dengan 3 cara yaitu interpolasi *Nearest*, interpolasi *Bilinear*, dan interpolasi *Bicubic*.
5. Hasil pengolahan sensor *MLX90620* ini menggunakan *i²c* dalam komunikasi dengan *mikrokontroller* *Atmega 2560*.
6. Hasil dari pengukuran oleh sensor *MLX 90620*, *LM 35*, dan sensor *HC-SR04*
7. *SR04* ini disimpan di *data logger* berupa file *.txt* .

3.5.2.Cara Kerja Sistem

Secara umum, cara kerja sistem pengukur panas peralatan listrik ini akan menggunakan sistem kendali *open loop*. Ilustrasi sistem dapat dilihat pada gambar 3.7.



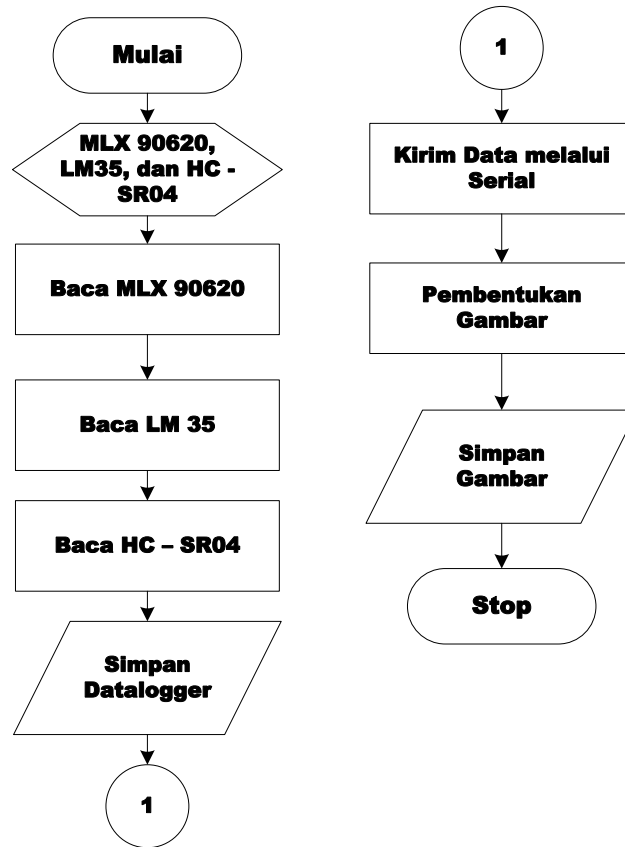
Gambar 3.7. Cara kerja sistem

Masukan dari sistem ini yaitu sensor MLX 90620, sensor LM 35, dan Sensor HC-SR04. Sensor Ketiga sensor ini kemudian diolah dengan *mikrokontroler* Arduino Mega 2560. Selanjutnya data ketiga sensor ini akan disimpan dengan sebuah datalogger. Di *datalogger* ini akan tersimpan nilai dari pembacaan sensor MLX 90620, Sensor LM 35, sensor HC – SR04, dan waktu melakukan pengukuran. Sesudah disimpan dalam *datalogger* data MLX 90620 akan dikirimkan ke mini komputer Raspberry Pi melalui USB serial. Nilai data sensor MLX 90620 diharapkan mendapatkan nilai yang ideal dengan cara pengukuran jarak antara alat yang dibuat dengan objek yang akan diukur. Selanjutnya data yang diterima akan diolah menjadi sebuah matriks yang berukuran 4 baris dan 16 kolom yang kemudian akan diubah menjadi citra berwarna yang merepresentasikan suhu dari objek yang diukur dengan *software* Python 2.7. Citra berwarna yang di hasilkan kemudian diinterpolasi dengan 3 cara interpolasi yaitu interpolasi *Nearest*, interpolasi *Bilinear* dan interpolasi *Bicubic*.

3.6. Diagram Alir Sistem Pengukuran Panas

Dari gambar 3.8. dapat dilihat diagram alir dari sistem pengukuran panas dengan thermopille array MLX 90620. Diagram alir ini tidak memiliki caption atau meriview dari pembacaan masing - masing sensor dikarenakan sistem yang dibangun adalah sistem dengan kendali openloop.

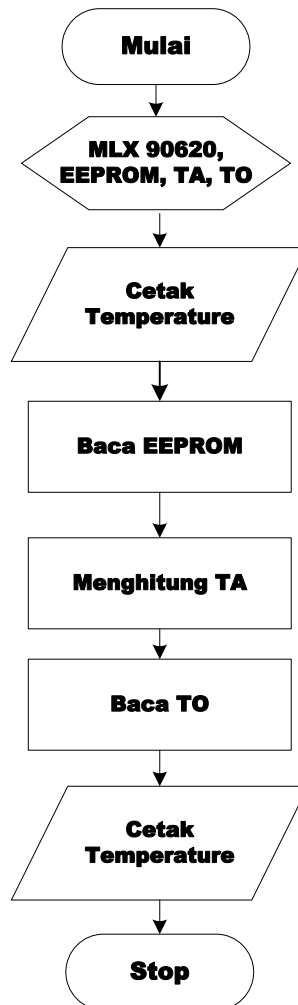
Pada sistem pengukur panas peralatan listrik yang akan dibuat ini, diagram alir sistemnya pada gambar 3.8 berikut ini :



Gambar 3.8. Diagram alir sistem

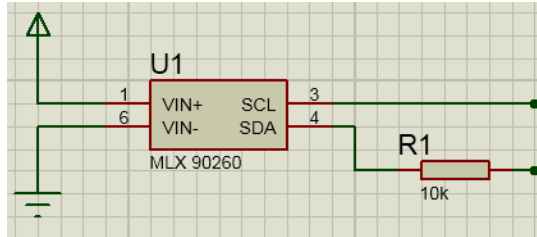
Sistem akan dimulai dengan pembacaan nilai dari temperature MLX 90620. Untuk diagram alir pembacaan *temperature* MLX 90620 dapat dilihat pada gambar 3.8. Selanjutnya yaitu pembacaan berturut-turut adalah pembacaan sensor oleh sensor LM 35 dan sensor HC-SR04. Kemudian semua data yang didapat akan disimpan dalam *datalogger* dan selanjutnya data MLX 90620 akan dikirimkan ke Raspberry Pi untuk dilakukan pembentukan citra. (Rangkaian secara keseluruhan terdapat pada lampiran)

3.6.1 Diagram Alir Sensor MLX 90620



Gambar 3.9. Diagram alir sensor MLX 90620

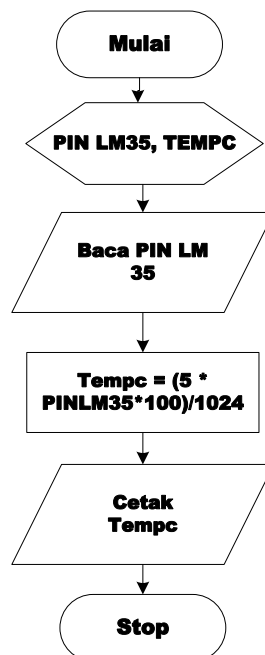
Dari gambar 3.9. yang merupakan diagram alir sensor MLX 90620 terdapat persiapan yaitu diantaranya 64 pixel MLX90620 sebagai masukan, EEPROM yang merupakan penyimpanan dari kalkulasi alpha i dan j, Temperature ambient (TA) yang merupakan temperature lingkungan, dan Temperature Object (TO) yang merupakan temperature dari objek yang diukur. Sensor MLX 90620 menggunakan rangkaian I₂C yang terdiri dari SDA dan SCL. Pulsa dibangkitkan melalui pin SCL dan data diterima melalui pin SDA.



Gambar 3.10 Rangkaian MLX 90620

Gambar 3.10 merupakan rangkaian dari sensor MLX 90620. Rangkaian MLX 90620 terdapat reistor pull up pada pin SDA. Tegangan yang diperlukan untuk mengaktifkan sensor MLX 90620 yaitu 2.6 VDC.

3.6.2 Diagram Alir sensor LM 35



Gambar 3.11. Diagram alir sensor LM 35

Diagram alir sensor LM 35 terdiri dari persiapan yaitu masukan pin analog pin LM35 dan inialisasi dari sensor LM 35. Sensor LM 35 memiliki sensitivitas hingga + 10 mv/°C. Maksudnya adalah setiap kenaikan 10 mv maka suhu yang

dibaca akan naik 1°C. Sensor LM 35 merupakan sensor dengan penggunaan analog. Sehingga pin keluaran dari sensor ini akan menjadi masukan pada mikrokontroler Arduino Mega 2560. Rumus dari LM 35 menjadi :

$$\text{Temperatur} = \frac{(5 * \text{Out} * 100)}{1024} \quad (3-2)$$

Dimana :

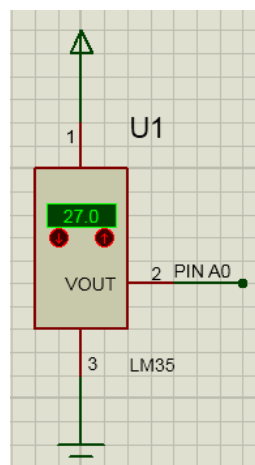
5 = Tegangan input

Out = Tegangan output dari LM 35

100 = Hasil dari pembagian $\frac{1 \text{ V}}{10 \text{ mV}}$

1024 = Hasil bilangan dari desimal dari 10 bit bilangan biner

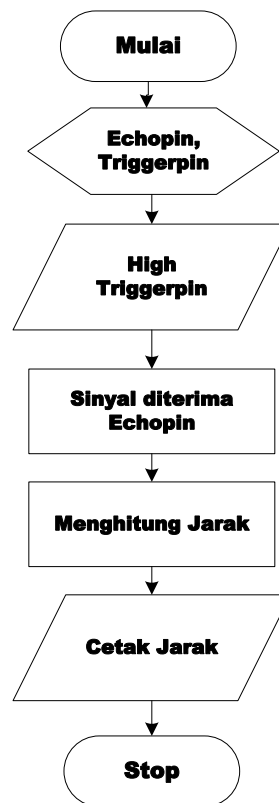
Dapat dijelaskan dari rumus (3-2) tegangan output diubah menjadi temperatur. Dengan mengingat bahwa 10 mV/°C maka didapat nilai 100 yaitu hasil pembagian 1 V dibagi 10 mV. Kemudian 1024 merupakan nilai dari 10 bit bilangan biner.



Gambar 3.12. Rangkaian LM 35

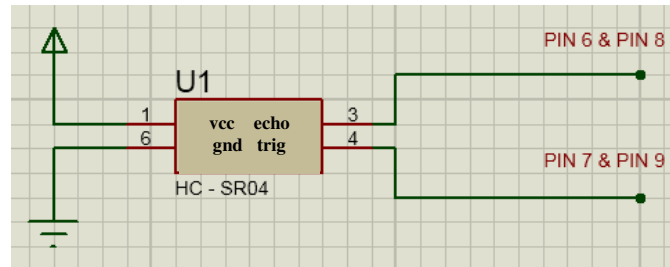
3.6.3. Diagram Alir sensor HC-SR04

Gambar 3.13. merupakan diagram alir dari sensor HC-SR04. Sensor HC-SR04 merupakan sensor jarak yang prinsip kerjanya adalah Pin trigger akan mengeluarkan sonar yang kemudian akan ditangkap melalui pin Echo. Perhitungan ini kemudian menggunakan kecepatan suara sebagai medianya. Selanjutnya hasil jarak ini akan dicetak dan disimpan dalam *datalogger*.



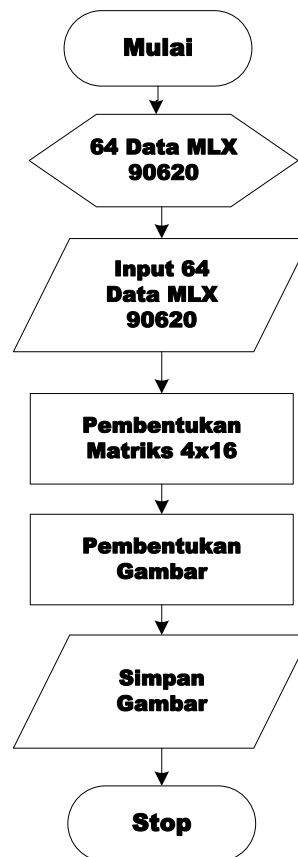
Gambar 3.13. Diagram alir sensor HC – SR04

Pada gambar 3.14 merupakan rangkaian dari sensor HC-SR04. Tegangan yang digunakan untuk mengaktifkan sensor HC-SR04 yaitu 5VDC. Pin echo merupakan pin yang menerima sinyal. Kemudian pin Trigger adalah pin yang mengeluarkan sinyal suara. Kedua pin ini akan dihubungkan dengan pin digital pada mikrokontroler Arduino Mega 2560.



Gambar 3.14. Rangkaian HC – SR04

3.6.4 Diagram Alir Pembuatan Citra pada Software Python 2.7



Gambar 3.15. Diagram alir pembuatan citra pada software Python 2.7

Gambar 3.15. merupakan diagram alir pembuatan citra pada software Python 2.7.

Dari diagram diatas dapat dilihat masukan dari diagram ini yaitu 64 data MLX 90620. Selanjutnya 64 data tersebut dibentuk matriks berbentuk 4 baris dan 16

kolom. Pembentukan ini menggunakan *library* numpy dan matplotlib. Library ini disediakan oleh python yang berguna untuk mengolah gambar. Dengan bantuan library matplotlib pembentukan gambar akan semakin mudah. Dalam proses ini akan dibentuk 3 buah gambar interpolasi yaitu interpolasi *Nearest*, interpolasi *Bilinear* dan interpolasi *Bicubic*. Selanjutnya gambar ini akan disimpan dengan pengkodean nomor didepan nama gambar untuk membedakan gambar-gambar yang diproses.