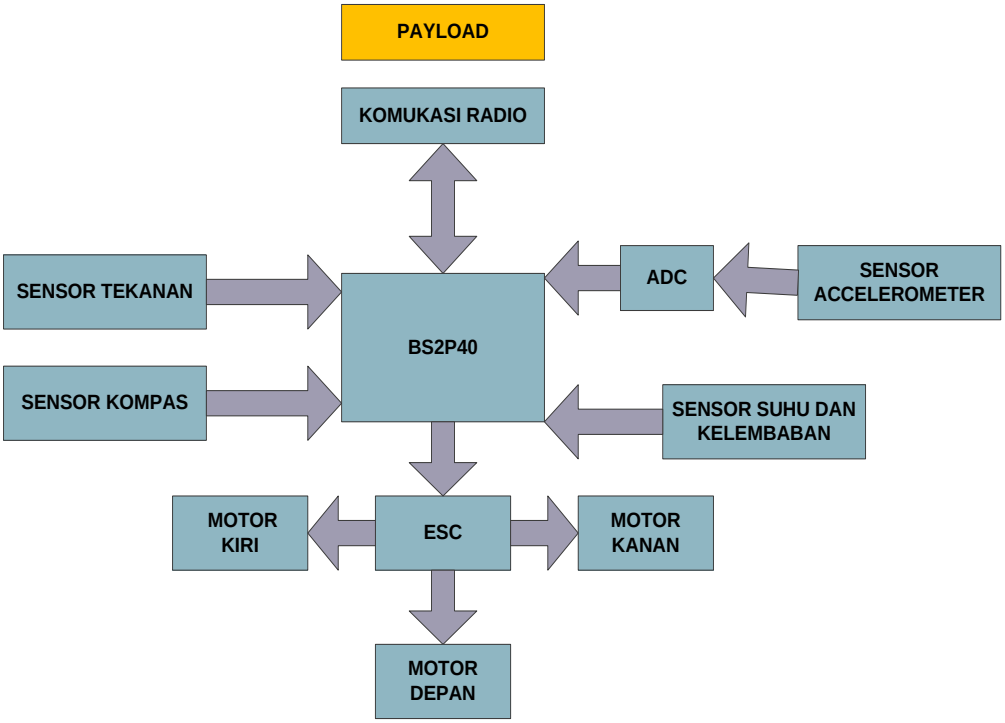
BAB III
PERANCANGAN SISTEM

Pada bab ini akan dijelaskan mengenai komponen–komponen apa saja yang membangun sistem telemetri suhu dan kelembaban pada *payload* roket. Dibawah ini merupakan keseluruhan blok sistem yang dibuat.



Gambar III.1 Blok Diagram Sistem Payload

Dapat dilihat pada diagram blok yang dibuat terdapat sebuah mikrokontroler yang berfungsi sebagai pengendali seluruh aktifitas *payload*. Seluruh sensor yang tepasang berfungsi sebagai pengindera besaran-besaran fisis yang datanya dikirimkan ke mikrokontroler untuk diolah kemudian dijadikan sebagai pengambil keputusan kerja dari mikrokontroler untuk menggerakkan aktuator yang terpasang pada *payload* atau langsung dikirimkan ke *ground segment*.

Tabel III.1 Keterangan diagram blok sistem payload

No	Blok	Keterangan
1	Komunikasi radio	Berfungsi sebagai blok komunikasi yang menghubungkan seluruh blok sistem pada *ground segment*
2	BS2P40	Mengatur seluruh kerja blok sistem, mengolah data, mengambil keputusan kendali gerak aktuator.
3	Sensor Suhu Dan Kelembaban	Mengukur besaran suhu dan kelembaban disekitar *payload*
4	Sensor Kompas	Mencari besaran sudut arah mata angin, 0^0 dari utara
5	Sensor Accelerometer	Berfungsi mengukur besaran axelerasi yang dialami *payload* serta kecepatan *payload*
6	Sensor Tekanan	Mengukur besaran tekanan udara berdasarkan ketinggian *payload*.
7	ADC	Mengubah besaran analog dari sensor accelerometer menjadi besaran digital yang dapat dibaca mikrokontroler
8	ESC	(Elektronik Speed control) mengatur kecepatan putar motor.
9	Motor Depan	Mendorong *payload* setelah menemukan arah yang ditentukan
10	Motor Kanan	Menentukan arah berdasarkan bacaan dari sensor kompas
11	Motor Kiri	Menentukan arah berdasarkan bacaan dari sensor kompas

3.1 Perancangan *Hardware*

Perancangan *hardware* meliputi perancangan desain, pemilihan komponen mikrokontroler, sensor yang digunakan, aktuator serta catu daya yang digunakan.

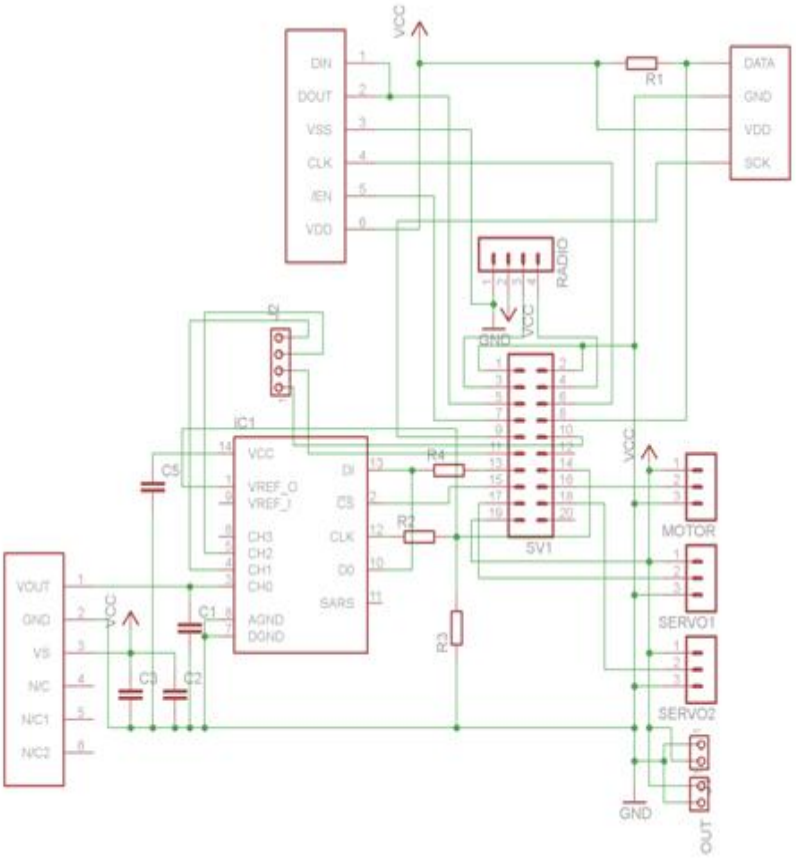
3.1.1 Rangkaian *Matherboard* Sensor-Sensor

Mikrokontroler yang digunakan adalah sebuah modul Basic Stamp 2P40 yang telah banyak dipasaran, alasan penggunaan mikrokontroler ini adalah karena jumlah pin pada basic stamp yang mencukupi kebutuhan *hardware*-nya, kemudahan dalam pemrograman khususnya sebagai pengendali aktuator yang diguanakan, berikut gambar dari basic stamp 2P40 yang digunakan:



Gambar III.2 Basic Stamp 2P40

Untuk mempermudah koneksi antara sensor dengan pin-pin pada basic stamp maka dibuatlah sebuah PCB untuk meletakkan semua sensor yang langsung dapat terkoneksi dengan modul basic stamp



Gambar III.3 Skema Rangkaian Sensor-Sensor

Berikut merupakan tabel penggunaan pin pada basic stamp mulai dari pin 0 s/d pin 15.

Tabel III.2 Pin – pin yang digunakan pada Basic Stamp 2P40

PIN	KETERANGAN
0	RX radio
1	TX radio
2	Data kompas
3	Clock kompas
4	Enable kompas
5	*Clock* Sht75
6	Data SHT75
7	Status MMA
8	ST MMA
9	Indikator LED
10	Data ADC
11	*Clock* ADC
12	CS ADC
13	Motor kiri
14	Motor kanan
15	Motor depan

3.1.2 Komunikai Serial

Proses pengiriman data dari mikrokontroler ke *ground segment* menggunakan modul radio, dimana dalam perancangan ini pin pada basic stamp harus terhubung pada pin modul radio yang terpakai agar dapat mengirim data maupun menerima data dari *ground segment*. Pada peraturan KORINDO *baudrate* yang harus digunakan adalah 9600. Jadi perlu adanya penyesuaian antara basic stamp dengan modul radio yang digunakan. Untuk penggunaan *baudrate* pada basic stamp dapat dilihat pada tabel dibawah ini.

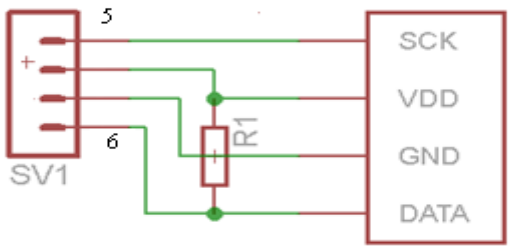
Tabel III.3 Pemakaian baudrate pada Basic Stamp 2P40[6]

Baud Rate	8-bit No Parity INVERTED	8-bit No Parity TRUE	7-bit Even Parity INVERTED	7-bit Even Parity TRUE
1200	18447	2063	26639	10255
2400	17405	1021	25597	9213
4800	16884	500	25076	8692
9600	16624	240	24816	8432

Pin RX pada modul radio digunakan sebagai pin untuk mengirimkan data dengan perintah dari basic stamp yaitu; *Serout, Pin, Baudrate,[data yang dikirimkan]*, sedangkan untuk membaca data yang diterima maka pin TX pada modul radio disambungkan dengan pin basic stamp dan digunakan perintah; *Serin, Pin, Boudrate, variabel penyimpan data*.

3.1.3 Rangkaian Sensor SHT75

Sensor SHT75 memiliki satu pin untuk mengeluarkan dua data yaitu data suhu dan data kelembaban, untuk mengaktifkan sensor Sht75 perlu diberikan sebuah *command* awal sebesar 00000101 untuk kelembaban dan 00000011 untuk temperatur. Untuk pin data diberikan sebuah resistor sebesar 10Kohm sebagai *pull-up.*



Gambar III.4 Rangkaian Sensor Suhu dan Kelembaban

Untuk mendapatkan nilai besaran suhu dan kelembaban dari keluaran sensor yang berupa data digital digunakan persamaan berikut :

1. Sensor kelembaban

   $RH_{linier}=C_1+C_2.SO_{RH}+C_3.SO_{RH}^2$

   Dengan nilai C_1, C_2, dan C_3:

Table III.4 Konstanta pengali untuk mencari nilai kelembaban [9]

SO_{RH}	c_1	c_2	c_3
12 bit	-4	0.0405	$-2.8 * 10^{-6}$
8 bit	-4	0.648	$-7.2 * 10^{-4}$

2. Sensor suhu

Temperature = $d_1+d_2.SO_T$

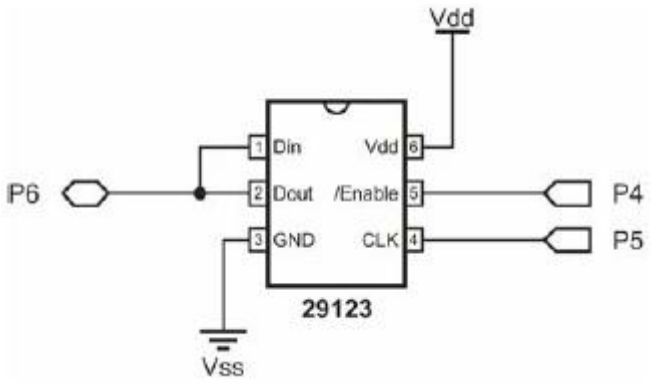
Table III.5 Konstanta pengali untuk mencari nilai suhu [9]

VDD	d_1 [°C]	d_1 [°F]
5V	-40.00	-40.00
4V	-39.75	-39.55
3.5V(3)	-39.66	-39.39
3V(3)	-39.60	-39.28
2.5V(3)	-39.55	-39.19

SO_T	d_2 [°C]	d_2 [°F]
14bit	0.01	0.018
12bit	0.04	0.072

3.1.4 Rangkaian Sensor Kompas

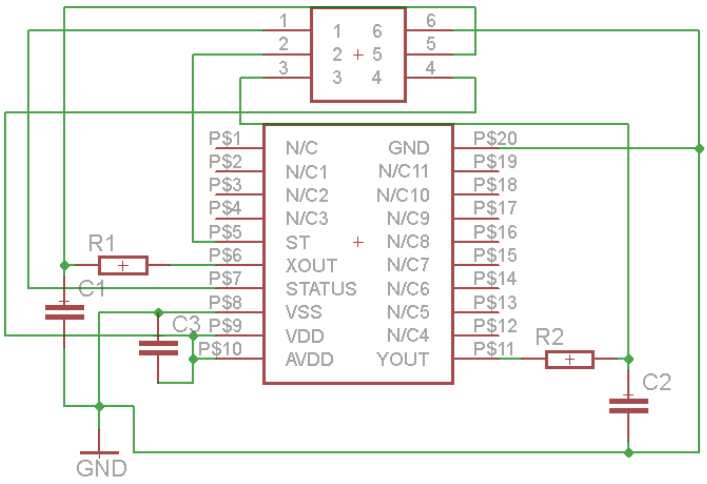
Sensor kompas bekerja pada tegangan 5VDC, pin data input dan data output disatukan untuk jalur data keluarannya, diperlukan juga sebuah data *clock* untuk dapat mengambil data dari kompas karena datanya bersifat serial



Gambar III.5 Skematik Sensor Kompas [4]

3.1.5 Rangkaian Sensor Percepatan

Sensor ini akan mengukur akselerasi gerak dari *payload* dengan satuan g. Data yang digunanakan adalah data dari *axis* X dan juga *axis* Y, apabila kedua data ini akan dijadikan dalam besaran derajat maka dapat dihitung menggunakan rumus *angle*= ArcusTAN (X/Y), serta apabila data yang sudah menjadi besaran derajat ingin dirubah menjadi besaran dalam g kembali, maka dapat digunakan rumus x=sin*angle* dan y=cos*angle*. Berikut gambar dari rangkaian *accelerometer* 3201.



Gambar III.6 Rangkaian Accelerometer 3201

3.1.6 Catu Daya

Catu daya merupakan faktor yang penting dalam perancangan sebuah sistem telemetri suhu dan kelemababan ini, karena tanpa adanya sebuah catu daya maka sistem ini tidak akan mampu untuk bekerja. Catu daya ini berfungsi untuk mencatu mikrokontroler, semua sensor yang digunakan, aktuator serta radio komunikasinya.

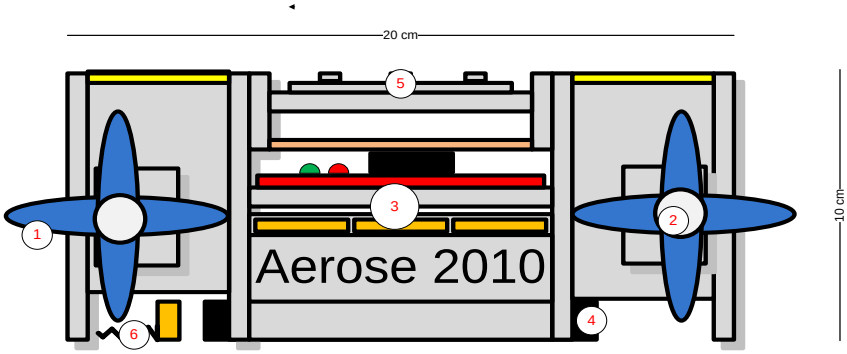
Catu daya pada aktuator menggunakan sumber tegangan sebesar 11,1Vdc dan 5Vdc sedangkan pada mikro dan sensor membutuhkan sumber sebesar 5Vdc. Pada mikrokontroler telah tersedia regulator 5V sebagai penurun tegangan sedangkan pada aktuator tidak diperlukan sebuah rangkaian penurun tegangan.

3.1.7 Desain *Payload*

Penggunaan semua komponen mempengaruhi bentuk dari *payload*, namun pada dasarnya desain *payload* ini mengacu pada aturan KORINDO. Setiap komponen yang dipakaipun mempengaruhi kinerja dari *payload*, desain yang dibuat diupayakan untuk membuat *payload* yang handal serta kuat dalam sistem kontruksinya. Rangka *payload* roket dibuat dari bahan tembaga yang kuat agar tahan tehadap hentakan saat roket diluncurkan serta saat roket mengalami sparasi. Rangka *payload* juga dilengkapi dengan peredam yang terbuat dari bahan busa untuk menahan getaran yang dihasilkan saat roket diluncurkan. *Payload* memiliki

tiga buah *propeller* sebagai pengendali gerak *payload*, *propeller* kiri dan kanan berfungsi untuk mencari sudut yang telah ditentukan, sedangkan bagian depan berfungsi sebagai pendorong *payload* jika sudut yang dicari telah ditemukan.

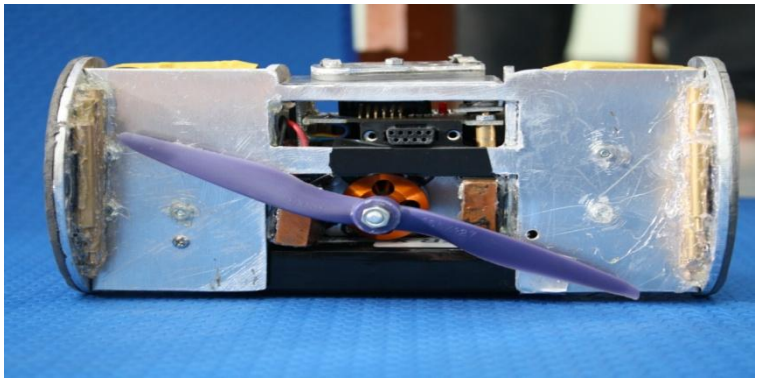
(a)



(b)



(c)



Gambar III.7 Rancangan Desain Payload, (a)Desain awal, (b) Tampak depan, (c) Tampak belakang

Keterangan gambar III.7 :

1. *Propeller*
2. Motor
3. Mikro dan sensor-sensor
4. Baterai
5. Tempat gantungan tali
6. Radio

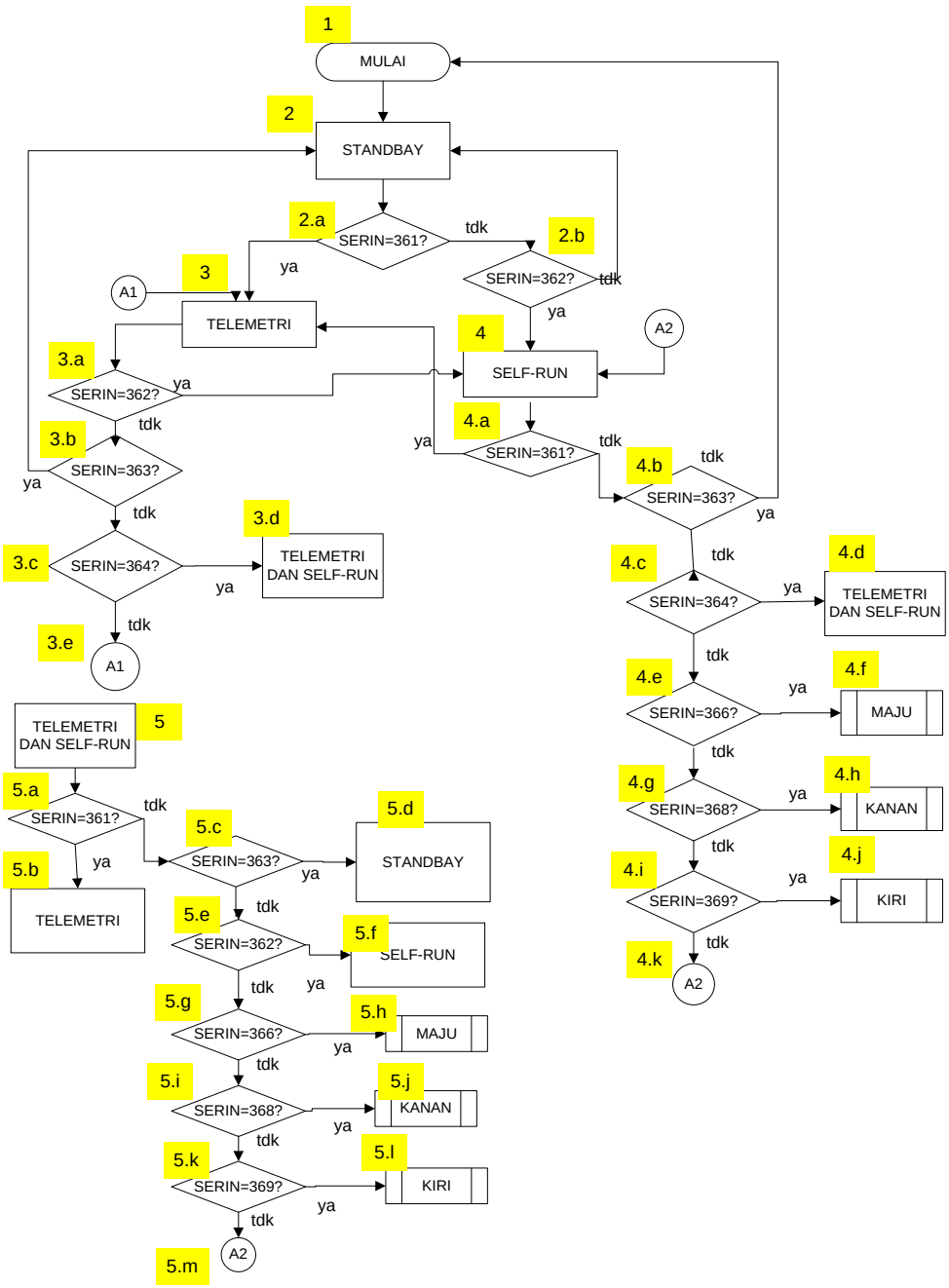
3.2 Perancangan *Software*

Perancangan *software* merupakan hal yang penting dalam perancangan sistem kendali pada *payload* ini. Pada bagian ini yang nantinya akan dijadikan sebagai algoritma kendali dari keseluruhan kerja *payload*. Seluruh algoritma dituliskan pada sebuah Basic Stamp Editor v2.2 dan kemudian disimpan pada sebuah IC mikrokontroler BS2p40, maka secara otomatis mikrokontroler BS2p40 akan mengerjakan seluruh program yang dimasukkan kedalamnya.

3.2.1 Algoritma Dasar

Merupakan sebuah algoritma yang mendasari seluruh algoritma yang ada dalam *payload*. Algoritma inilah yag nanti akan dikembangkan menjadi algoritma-agoritma yang lebih spesifik lagi.

Ketika awal dijalankan maka mikro akan menunggu interuksi dari user terlebih dahulu tanpa mengerjakan apapun, selanjutkan akan mengerjakan perintah-perintah sesuai dengan interuksi yang diberikan. Berikut ini merupakan bagan *flowchart* sederhana dari seluruh algoritma yang ada.



Gambar III.8 Flowchart Pemrograman

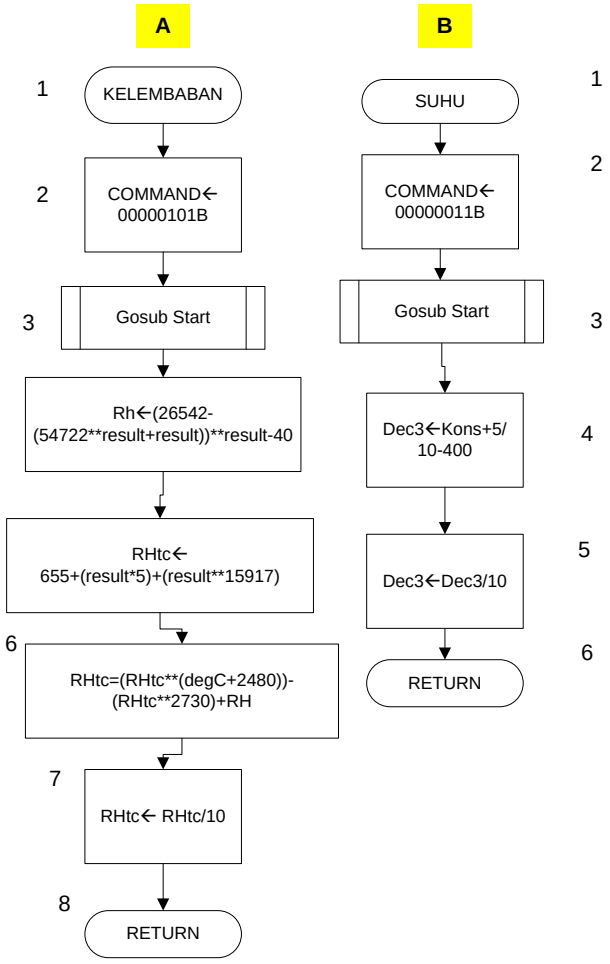
Keterangan gambar dari *flowchart* diatas dapat dilihat pada tabel III.6 dibawah ini. Tabel ini menerangkan alur program yang ada dalam *flowchart* tersebut.

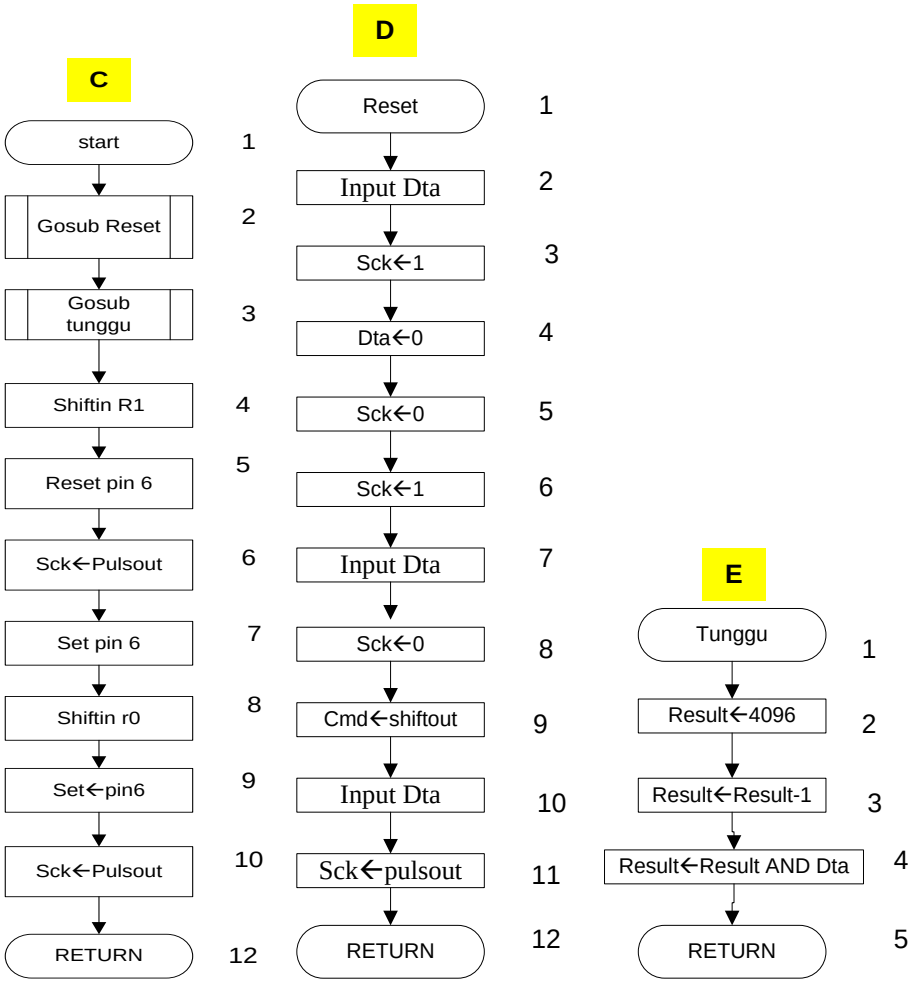
Tabel III.6 Keterangan gambar 3.6

No	Keterangan
1	Mulai program dijalankan
2	Rutin *standby* dimana mikrokontroler tidak mengejakan apapun
a	Jika data = 361 maka lompat ke rutin Telemetri jika tidak lanjut ke b
b	Jika data = 362 maka lompat ke rutin *self-run* jika tidak balik keawal
3	Rutin Telemetri
a	Jika data = 362 maka lompat ke rutin *self-run* jika tidak lanjut ke c
b	Jika data = 363 maka lompat ke rutin *Standby* jika tidak lanjut ke d
c	Jika data = 364 maka lompat ke d jika tidak lanjut ke e
d	Telemetri dan *Self-run*
e	Ulangi dari awal Telemetri
4	Rutin *self-run*
a	Jika data = 361 maka lompat ke rutin Telemetri jika tidak lanjut ke b
b	Jika data = 363 maka lompat ke rutin *Standby* jika tidak lanjut ke c
c	Jika data = 364 maka lompat ke d jika tidak lanjut ke e
d	Telemetri dan *Self-run*
e	Jika data = 366 maka lompat ke f jika tidak lanjut ke g
f	Sub rutin Maju
g	Jika data = 368 maka lompat ke h jika tidak lanjut ke i
h	sub rutin Kanan
i	Jika data = 369 maka lompat ke j jika tidak lanjut ke k
j	Sub Rutin Kiri
k	Mulai dari awal Program *Self-run*
5	Rutin telemetri dan *self-run*
a	Jika data = 361 maka lompat ke b jika tidak lanjut ke c
b	Rutin telemetri
c	Jika data = 363 maka lompat ke d jika tidak lanjut ke e
d	Rutin *standby*
e	Jika data = 362 maka lompat ke f jika tidak lompat ke g
f	Rutin *Self-run*
g	Jika data = 366 maka lompat ke h jika tidak lanjut ke i
h	Sub rutin Maju
i	Jika data = 368 maka lompat ke j jika tidak lanjut ke k
j	Sub rutin Kanan
k	Jika data = 369 maka lompat ke l jika tidak lanjut ke m
l	Sub Rutin Kiri

m	Mulai dari awal Program *Self-run*

3.2.2 Prosedur Sensor Sht75





Gambar III.10 Flowchart Pemrograman Sensor Sht75

Berikut ini adalah tabel keterangan dari *flowchart* pemrograman Sht75, setiap alur program diceritakan dalam tabel dibawah ini.

Tabel III.7 Keterangan gambar 3.7

No	KETERANGAN
A	Rutin Menghitung kelembaban
1	Mulai
2	Berikan nilai awal (command) = 5
3	Lompat ke sub ruin start
4 s/d 8	Lakukan perhitungan mencari rh
B	Rutin Menghitung Suhu
1	Mulai

2	Berikan nilai awal (command) = 3
3	Lompat ke sub rutin start
4 s/d 5	lakukan perhitungan mencari nilai suhu
C	Sub Rutin awal
1	Mulai
2	Lompat ke sub rutin reset
3	Lompat ke sub rutin tunggu
4	Ambildata dengan sebuah clock dan simpan data pada R1
5	Reset pin 6
6	Berikan sebuah pulsa pada pin sck
7	Set pin 6
8	Ambil data dengan sebuah clock dan simpan data pada R0
9	Set pin 6 menjadi 1
10	Berikan pulsa pada pin sck
12	Kembali ke program utama
D	Sub Rutin Reset
1	Mulai sub rutin
2	Pin data aktif dan dapat diambil data saat nilai hight
3	Set pin sck menjadi 1
4	Set pin data menjadi 1
5	Reset pin sck
6	Set pin sck menjadi 1
7	Pin data aktif dan dapat diambil data saat nilai hight
8	Reset pin sck
9	Beri pulsa pada cmd untuk mengeluarkan data
10	Pin data aktif dan dapat diambil data saat nilai hight
11	Berikan pulsa pada pin sck
12	Kembali keprogram utama
E	Delay
1	Sub rutin tunggu
2	Result=4096
3	Result-1
4	Result AND data