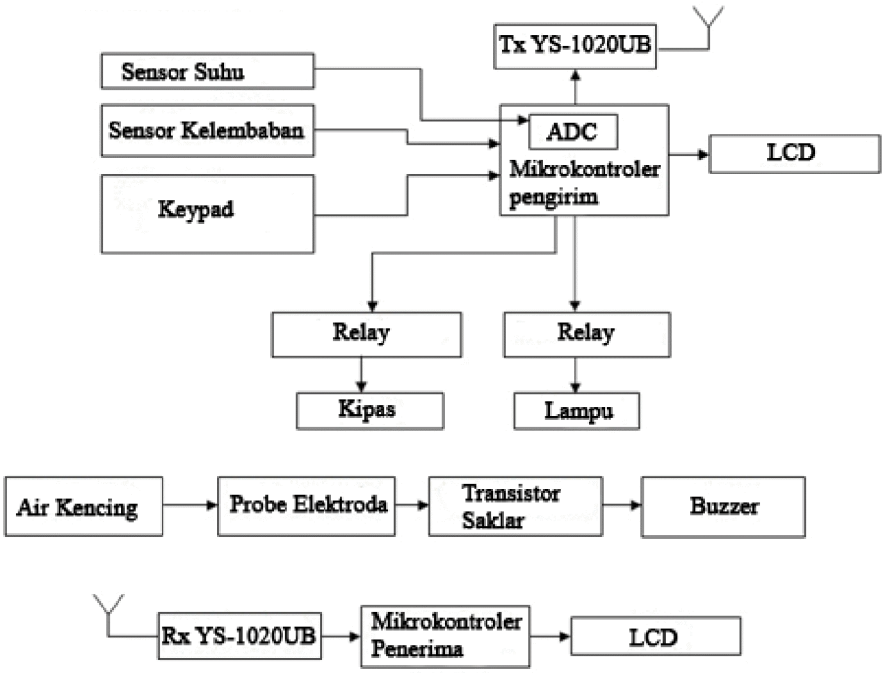
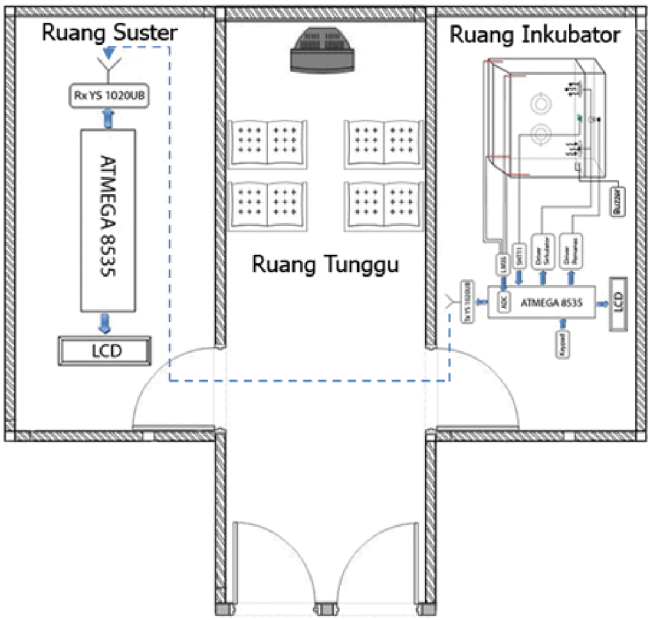
BAB III
PERANCANGAN SISTEM



Gambar 3.1. Diagram blok sistem.

Perancangan merupakan tahap yang terpenting dari keseluruhan proses pembuatan alat. Perancangan sistem inkubator bayi berbasiskan *wireless* ini terbagi dalam tiga bagian yaitu perancangan modul sensor suhu dan kelembaban, modul sensor basah, serta modul penerima.



Gambar 3.2. Diagram blok sistem secara keseluruhan.

Pada diagram blok sistem secara keseluruhan diatas dapat dijelaskan, serta penempatan komponen.

Yang prinsip kerjanya pada diagram blok adalah sebagai berikut :

a. Mikrokontroler *(transmitter)* berfungsi sebagai pengontrol ADC, masukan *set point* suhu atas, suhu bawah dari *keypad*, dan juga berfungsi sebagai perantara dalam pengiriman data yang diambil dari ADC menuju YS-1020UB, serta sebagai pengtrol *relay* pemanas dan kipas.
b. 4 buah sensor LM35DZ berfungsi mengubah besaran suhu menjadi besaran listrik, yang nantinya hasil rata-rata dari LM35DZ akan ditampilkan ke LCD pengirim dan penerima.
c. Sensor SHT11 berfungsi mengubah besaran kelembaban menjadi besaran listrik dan di tampilkan ke LCD pengirim dan penerima.
d. 2 *Relay* pendingin berfungsi sebagai media pengantar panas kedalam inkubator,dan pembuang panas dari dalam inkubator.
e. *Relay* pemanas berfungsi sebagai pemanas didalam inkubator.
f. *Keypad* berfungsi sebagai masukan *set point* suhu atas dan suhu bawah inkubator, serta sebagai sistem pengaman inkubator.

g. *Buzzer* berfungsi sebagai indikator suara ketika bayi buang air.

h. LCD berfungsi sebagai *display* suhu dan kelembaban pada bagian sistem pengirim dan penerima.

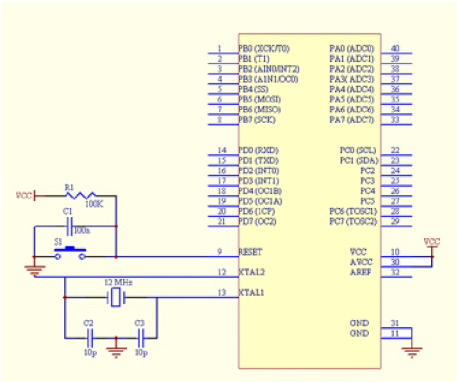
i. YS-1020UB berfungsi sebagai modul pemancar *(transmitter)* dan penerima *(receiver)* data melalui gelombang radio berfrekuensi tinggi.

j. Mikrokontroler *(receiver)* berfungsi sebagai perantara antara data yang diterima dari YS-1020UB untuk ditampilkan hasilnya ke LCD.

3.1. Perancangan Perangkat Keras

3.1.1. Mikrokontroler ATmega8535

Blok mikrokontroler ATmega8535 berfungsi sebagai alat untuk memproses masukan yang berasal dari sensor SHT11, sensor LM35DZ serta dari *keypad* untuk masukan nilai *set point* suhu atas dan suhu bawah. Selain sebagai alat proses, mikrokontroler tersebut juga pengontrol *relay* dan *display* yang digunakan sebagai keluaran informasi dari perubahan suhu didalam inkubator yang nantinya akan dikirim ke mikrokontroler penerima. Berikut ini rangkaian dari blok sistem minimum mikrokontroler ATmega8535.



Gambar 3.3. Rangkaian mikrokontroler ATmega8535

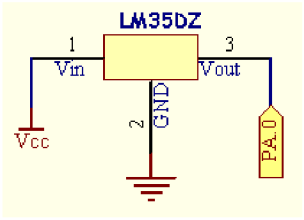
3.1.2. ADC *(Analog To Digital Vonverter)*

ADC berfungsi sebagai pengkonversi *output* sensor LM35DZ dari analog ke digital, dalam alat ini ADC bekerja dalam mode *single convertion* yaitu pengguna harus mengaktifkan setiap kali ADC akan digunakan.

3.1.3. Sensor Suhu

Sensor ini bekerja berdasarkan perubahan tegangan atau arus dengan sensitivitas yang sangat baik. IC ini mempunyai linieritas yang baik terhadap perubahan suhu. Kemampuan untuk mengkonversi suhu terhadap tegangan dengan keluaran 10 mV/10C.

Dalam sistem ini, menggunakan 4 sensor LM35DZ. Penggunaan pin V*out* pada kaki LM35DZ, dihubungkan dengan *Port*A.0-*Port*A.3 pada Mikrokontroler, pin +Vs dihubungkan dengan +5 VDC, dan pin GND dihubungkan dengan *ground*.

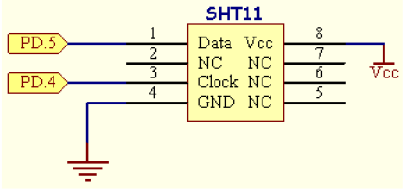


Gambar 3.4. Rangkaian LM35DZ.

Fungsi LM35DZ pada inkubator yang saya rancang adalah sebagai masukan data nilai suhu dalam inkubator. Adapun hubungan LM35DZ pada inkubator yang saya rancang ialah sebagai sumber data nilai suhu.

3.1.4. Sensor Kelembaban

SHT11 adalah *single chip* sensor suhu dan kelembaban relatif yang mempunyai banyak sensor module yang terdiri sebuah pengkalibrasi output digital. Pada pengukuran suhu data yang dihasilkan 14 *bit* sedangkan untuk kelembaban data yang dihasilkan 12 *bit*. Keluaran dari SHT11 adalah digital sehingga untuk mengaksesnya diperlukan pemrograman dan tidak diperlukan pengkondisi sinyal atau ADC.



Gambar 3.5. Rangkaian SHT11.

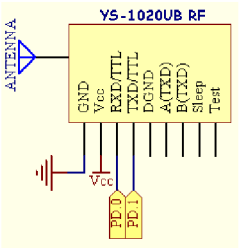
Pada alat ini digunakan sebuah sensor SHT11 sebagai sensor kelembaban, untuk membaca nilai kelembaban didalam inkubator. Penggunaan pin SHT11 pada inkubator ini ialah pin Data SHT11 dihubungkan ke *Port*D.5 pada mikrokontroler, pin Clock dihubungkan ke *Port*D.4 mikrokontroler, pin Vcc dihubungkan ke +5 VDC, dan pin GND dihubungkan pada *ground*.

Fungsi SHT11 pada inkubator ini adalah sebagai masukan data nilai kelembaban bayi. Adapun hubungan SHT11 dengan inkubator ini adalah sebagai sumber data kelembaban.

3.1.5. Modem Radio Frekuensi YS-1020UB

Modem radio frekuensi YS-1020UB adalah alat yang dapat digunakan sebagai pengirim (*transmitter*) dan penerima (*receiver*) data yang akan dikirim dan diterima oleh mikrokontroler secara *wireless*, dengan jarak kirim dan terima data antara 0-800 meter. Dan berfungsi juga sebagi rangkaian *modulator* dan *demodulator*.

Cara penggunaan modem radio frekuensi YS-1020UB ini adalah pada pin pertama yaitu GND diberi *ground*. Pin kedua yaitu Vcc diberi tegangan +DC 5volt. Sedangkn pin keempat dan kelima adalah pin Rx dan Tx yang akan di hubungkan ke *port*D.0 dan *port*D.1 pada mikrokontroler.



Gambar 3.6. Rangkaian modem radio frekuensi YS-1020UB.

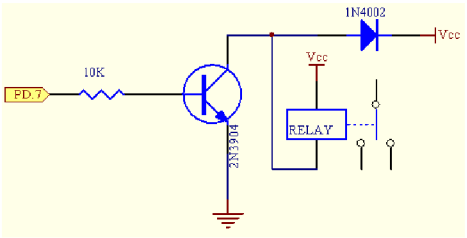
Pada alat ini digunakan 2 buah modem radio frekuensi YS-1020UB berfungsi sebagai media pengirim data suhu dan kelembaban dari inkubator ke penerima, yang nantinya data akan diterima oleh mikrokontroler penerima dan di tampilkan ke layar LCD. Adapun

hubungan YS-1020UB pada inkubator ini adalah sebagai media pengirim dan penerima data.

3.1.6. *Relay*

Pengkondisian hidup matinya *relay* ditentukan dengan besarnya nilai suhu atas dan suhu bawah diinkubator yang diinput melalui *keypad*. Bila nilai suhu dalam inkubator lebih besar atau lebih kecil dari batas pengaturan *set point* temperatur maka *relay* akan mati atau hidup.

Fungsi *relay* pada inkubator ini sebagai saklar otomatis pemanas dan pendingin. Adapun hubungan *relay* pada alat ini untuk mengaktifkan lampu dan kipas agar suhu inkubator tetap stabil. Rangkaian *driver relay* digunakan sebagai element kendali akhir. *Relay* yang digunakan pada alat ini sebanyak 3 buah dan bekerja pada tegangan 5 Volt. Cara penggunaan *relay* ini adalah pada pin Vcc dihubungkan ke tegangan DC 5 volt, pin GND dihubungkan pada *Ground*, untuk pin data sebagai input perintah dari mikrokontroler yang dihubungkan ke *port*D.7.



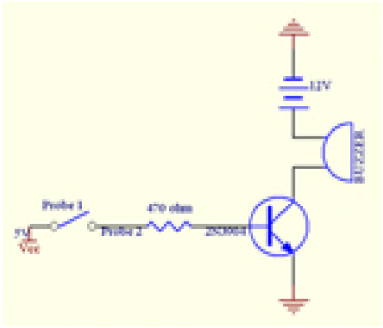
Gambar 3.7. Rangkaian pengkondisi *relay* untuk lampu.

Gambar untuk rangkaian pengkondisi *relay* kipas sirkulasi dan kipas buang sama dengan gambar 3.7. hanya saja melalui *port*D.6 untuk kipas sirkulasi, *port*D.3 untuk kipas buang.

3.1.7. Sensor Basah

Rangkaian sensor basah terdiri dari probe elektroda, transistor 2N3904 (NPN) yang bekerja aktif *high* '1'. Dalam perancangan rangkaian sensor basah *probe* elektroda dibagi menjadi dua bagian, adapun cara penggunaan dimana salah satu *probe* dihubugkan dengan Vcc 5 Volt dan *probe* berikutnya dihubungkan ke basis transistor, kemudian kaki emiter

dihubungkan ke *ground*, sedangkan kaki kolektor dihubungkan dengan kaki *buzzer* (-) dan kaki *buzzer* (+) dihubungkan dengan tegangan 12 V. Fungsi dari sensor basah adalah ketika bayi yang berada didalam inkubator tersebut buang air maka akan mengenai kedua *probe elektroda* yang disimpan di bawah bayi dan menghasilkan suara, kemudian tegangan akan mengalir pada kaki basis transistor sehingga berfungsi sebagai saklar, maka saklar akan bekerja aktif *high* menyebabkan *buzzer* berbunyi, dimana arus listrik mengalir dari kaki kolektor ke emiter.

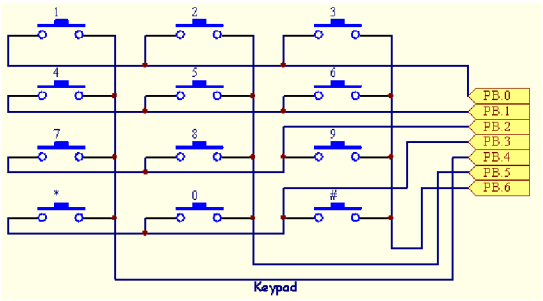


Gambar 3.8. Rangkaian indikator suara *buzzer*.

Hubungan sensor basah pada sistem ini sebagai tanda ketika bayi buang air kecil yang akan member informasi ke perawat.

3.1.8. *Keypad*

Hubungan *keypad* dalam perancangan alat ini sebagai input ke mikrokontroler yang berfungsi untuk menjalankan aplikasi yang sudah jadi, dengan adanya *keypad* input suatu besaran nilai yang dinginkan dan selain itu dapat memerintahkan suatu aplikasi agar dapat diproses.



Gambar 3.9. Rangkaian *keypad*.

Cara kerja *keypad* ini adalah pada pin 1- pin 7 dihubungkan ke *port*B.0- *port*B.6 pada mikrokontroler. Fungsi *keypad* disini sebagai input nilai *set point* dan sistem *password*. Untuk input dari sebuah *keypad* agar dapat terbaca oleh mikrokontroler dikondisikan seperti pada Tabel berikut.

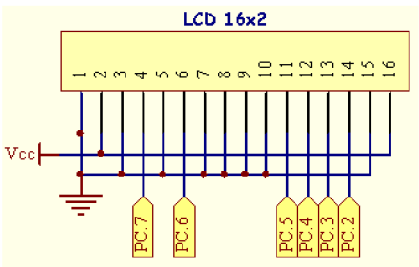
Tabel 3.1. Kondisi Input Keypad

Input Keypad							**Output (Hasil)**
Port B.0	Port B.1	Port B.2	Port B.3	Port B.4	Port B.5	Port B.6	
0	1	1	1	0	1	1	1
0	1	1	1	1	0	1	2
0	1	1	1	1	1	0	3
1	0	1	1	0	1	1	4
1	0	1	1	1	0	1	5
1	0	1	1	1	1	0	6
1	1	0	1	0	1	1	7
1	1	0	1	1	0	1	8
1	1	0	1	1	1	0	9
1	1	1	0	0	1	1	*
1	1	1	0	1	0	1	0
1	1	1	0	1	1	0	#

3.1.9. LCD *(Liquid Crystal Display)*

Pada perancangan alat ini menggunakan (LCD) *Liquid Crystal Display* yang bertipe JHD162A yang merupakan piranti *display* yang mampu menampilkan karakter 16 kolom dan 2 baris (16x2).

Penggunaan pin LCD pada inkubator ini ialah pin Vss, pin V0, pin R/W, pin DB.0 sampai pin DB3, pin A/Vee dihubungkan ke ground. Pin Vdd dan pin K dihubungkan ke tegangan DC 5 volt. Pin Rs dihubungkan pada *port*C.7 mikrokontroler. Pin Enable dihubungkan ke *port*C.6 mikrokontroler. Pin DB.4 sampai pin DB.7 masing-masing dihubungkan pada *portC.*5 sampai *port*C.*2.*

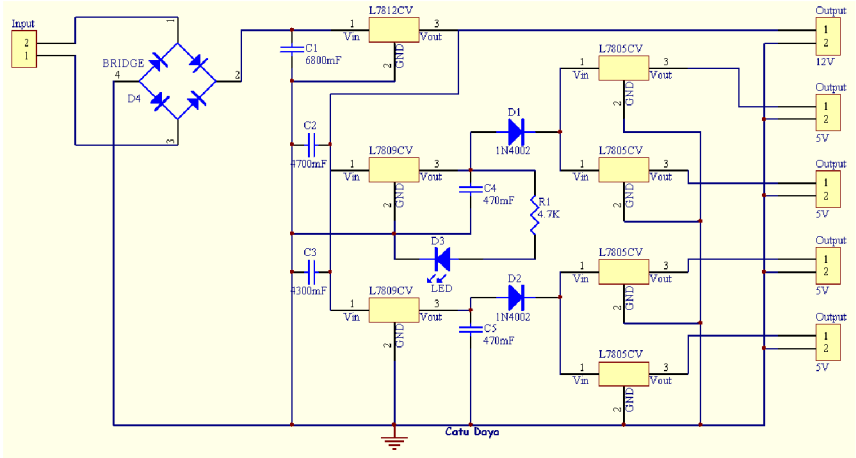


Gambar 3.10. Rangkaian karakter LCD 16x2.

Pada alat ini digunakan 2 buah LCD 16x2 masing-masing berfungsi sebagai *display* data suhu dan kelambaban dari mikrokontroler pengirim dan penerima. Adapun hubungan LCD pada sistem ini sebagai media *display* informasi suhu dan kelembaban inkubator ke perawat atau dokter.

3.1.10. Catu Daya

Untuk mengaktifkan rangkaian-rangkaian elektronika pada *thermometer* digital ini dibutuhkan *supply* tegangan dari catu daya sebesar +5 V dan +12 V. Rangkaian catu daya ini terdiri dari sebuah transformator dengan keluaran 12 Volt, penyearah arus dengan *Brige Diode* dan IC L7812, L7809, L7805.



Gambar 3.11. Rangkaian catu daya.

Tegangan keluaran dari transformator masih dalam bentuk tegangan bolak-balik (AC), sehingga harus disearahkan oleh penyearah gelombang penuh. Sebelum tegangan diregulasi, dilewatkan dulu pada tapis atau *filter* untuk memperkecil *ripple*. Komponen *filter* yang digunakan yaitu kapasitor polar.

Konsep dasar kerja kapasitor ini untuk menyimpan muatan-muatan listrik (Q) sehingga jika diberi tegangan, kapasitor tersebut akan terisi namun tidak seketika penuh muatan, sebaliknya ketika kapasitor tersebut akan terisi namun tidak seketika penuh muatan.

Untuk meregulasi tegangan digunakan regulasi tegangan positif yaitu tegangan 12 V (L7812), 9 V (L7809) dan 5 V (L7805). Konfigurasi

kaki IC regulator tegangan positif adalah kaki 1 berfungsi sebagai input, kaki 2 terhubung ke ground dan kaki 3 sebagai keluaran tegangan positif.

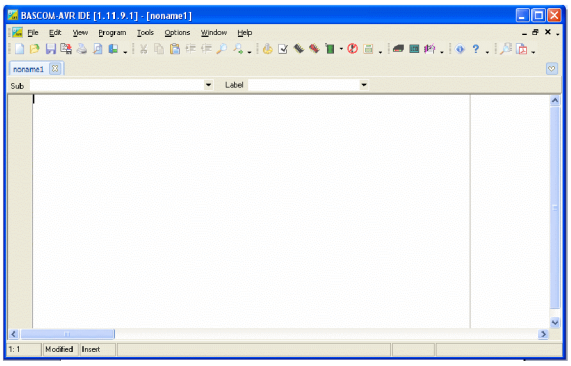
3.2. Perancangan Perangkat Lunak

3.2.1. Bascom-AVR

Bascom-AVR adalah program *basic compiler* berbasis windows untuk mikrokontroler keluarga AVR merupakan pemrograman dengan bahasa tingkat tinggi " *Basic* " yang dikembangkan dan dikeluarkan oleh MCS *electronics* sehingga dapat dengan mudah dimengerti atau diterjemahkan.

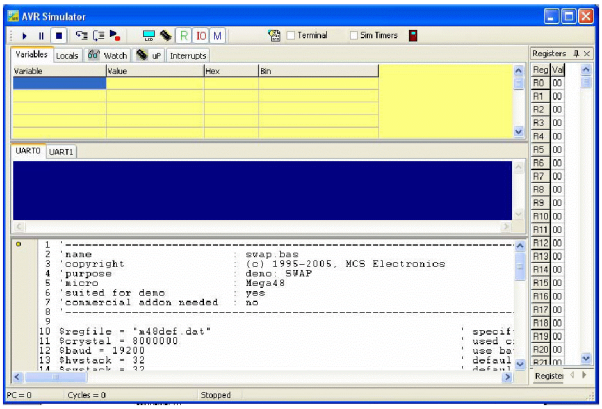
Dalam program Bascom-AVR terdapat beberapa kemudahan, untuk membuat program ATmega8535, seperti program simulasi yang sangat berguna untuk melihat, simulasi hasil program yang telah kita buat, sebelum program tersebut kita *download* ke mikrokontroler.

Ketika program Bascom-AVR dijalankan dengan mengklik *icon* Bascom-AVR, maka jendela berikut akan tampil :



Gambar 3.12. Tampilan jendela program *Bascom-AVR.*

Bascom-AVR menyediakan pilihan yang dapat mensimulasikan program. Program simulasi ini bertujuan untuk menguji suatu aplikasi yang dibuat dengan pergerakan LED yang ada pada layar simulasi dan dapat juga langsung dilihat pada LCD, jika kita membuat aplikasi yang berhubungan dengan LCD.



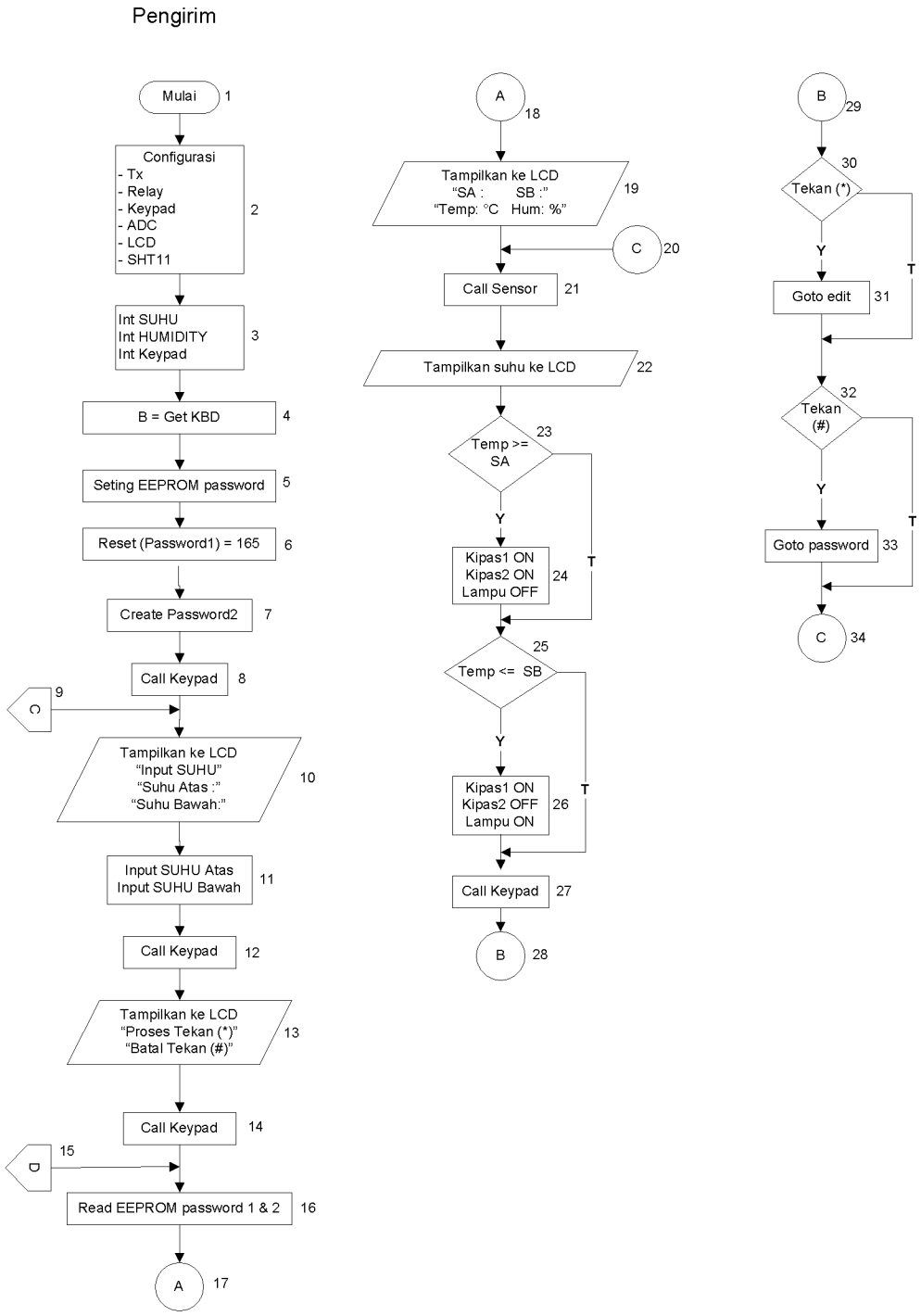
Gambar 3.13. Tampilan simulasi *Bascom-AVR*.

Intruksi yang dapat digunakan pada editor Bascom-AVR relatif cukup banyak dan tergantung dari tipe dan jenis AVR yang digunakan. Berikut ini beberapa instruksi-instruksi dasar yang dapat digunakan pada mikrokontroler ATmega8535.

Tabel 3.2. Beberapa instruksi dasar Bascom AVR

INSTRUKSI	KETERANGAN
DO ..... LOOP	Perulangan
GOSUB	Memanggil Prosedur
IF ..... THEN	Percabangan
FOR ..... NEXT	Perulangan
WAIT	Waktu Tunda Detik
WAITMS	Waktu Tunda MiliDetik
WAITUS	Waktu Tunda MicroDetik
GOTO	Loncat Kealamat Memori
SELECT ..... CASE	Percabangan

Berikut ini adalah gambar 3.14. *flowchart* pengirim.

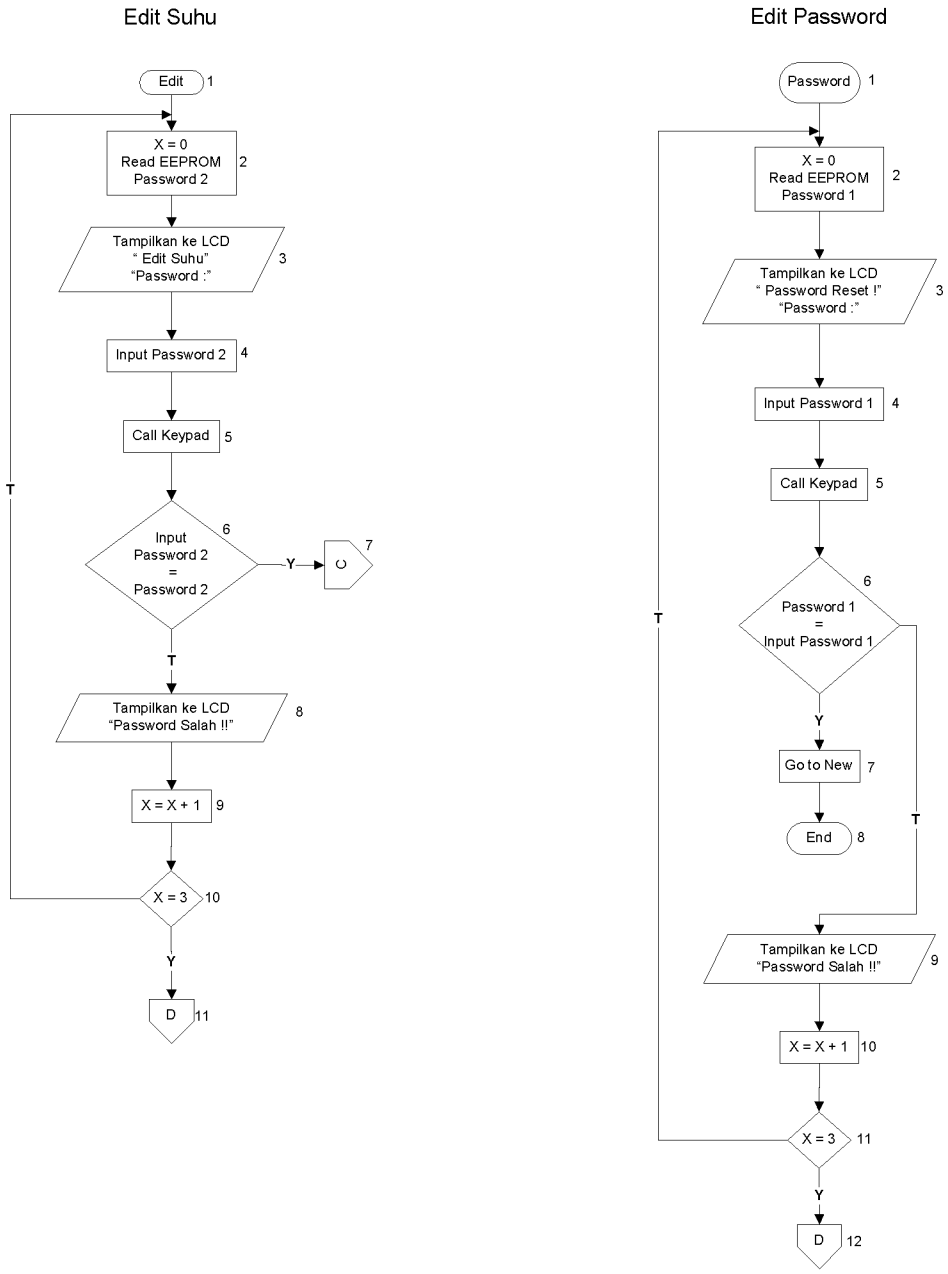


Gambar 3.14. *Flowchart* pengirim.

Tabel 3.3. Keterangan *flowchart* pengirim.

No.	Keterangan
1	Mulai.
2	Deklarasi variabel.
3	Deklarasi variabel.
4	B sebagai variabel untuk menyimpan nilai input *keypad*
5	*Setting* password ke *EEPROM*.
6	Kode reset *password* = 165.
7	*Input password* untuk akses alat.
8	Panggil label *keypad.*
9	Label C.
10	Tampilkan ke *LCD* Input SUHU Suhu Atas : Suhu Bawah.:
11	Masukan nilai *set point* batas suhu atas dan batas suhu bawah.
12	Panggil label *keypad.*
13	Tampilkan ke *LCD* Proses Tekan (*) Proses Batal (#).
14	Panggil label *keypad.*
15	Label D.
16	Membaca *password* 1 dan 2 di *EEPROM*.
17	Ke label A.
18	Label A.
19	Tampilkan ke *LCD* SA : SB : Temp : °C Hum : %.
20	Label C.
21	Panggil label sensor.
22	Tampilkan nilai suhu incubator ke *LCD*.
23	Cek, apakah data Temp lebih besar sama dengan SA? Jika Ya, kipas1 *ON*, kipas2 *ON*, lampu *OFF* Jika Tidak, cek apakah data Temp lebih kecil sama dengan SB?
24	Kipas1 *ON*, kipas2 *ON*, lampu *OFF*.
25	Cek, apakah data Temp lebih kecil sama dengan SB? Jika Ya, kipas1 *ON*, kipas2 *OFF*, lampu *ON* Jika Tidak, panggil label *keypad*.
26	Kipas1 *ON*, kipas2*OFF*, lampu *ON*.
27	Panggil label *keypad.*
28	Ke label B.
29	Label B.
30	Cek, apakah tombol yang ditekan tanda (*)? Jika Ya, lompat ke label *password* Jika Tidak, Cek apakah tombol yang ditekan tanda (#)?
31	Lompat ke label edit.
32	Cek, apakah yang ditekan tanda (#)? Jika Ya, lompat ke label *password* Jika Tidak ke label C.
33	Lompat ke label *password*.
34	Lompat ke label C.

Berikut ini adalah gambar 3.15. *flowchart* edit suhu dan *password*.



Gambar 3.15. *Flowchart* edit suhu dan *password*.

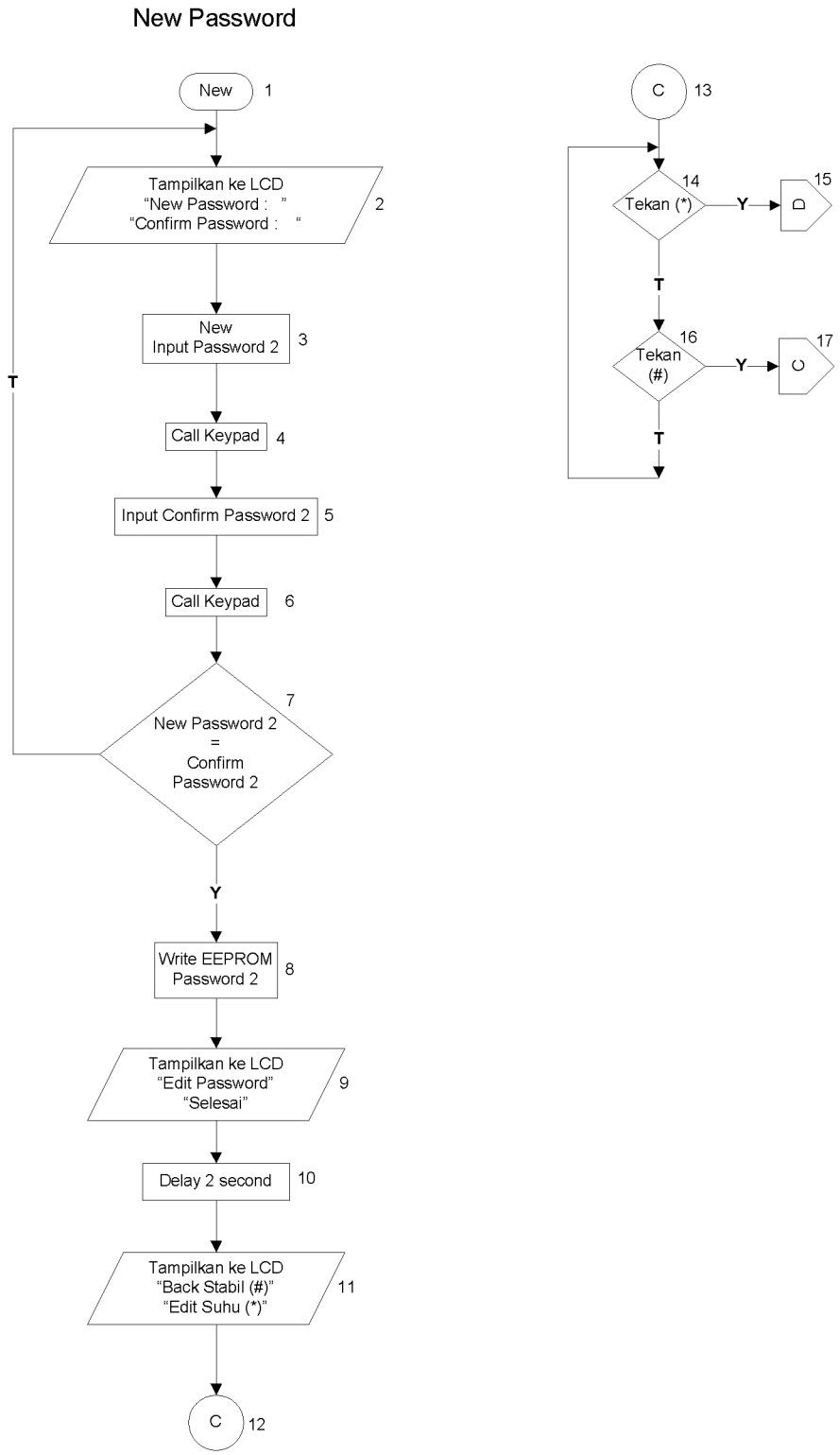
Tabel 3.4. Keterangan *flowchart e*dit suhu.

No.	Keterangan
1	Label edit.
2	X diberi data bernilai 0. Membaca *password2* di *EEPROM.*
3	Tampilkan ke *LCD* Edit Suhu Password :
4	Masukan *password2.*
5	Panggil label *keypad.*
6	Cek, apakah *password2* yang ditekan sama dengan *password2* di *EEPROM* Jika Ya, kelabel D dihalaman berikutnya Jika Tidak, tampilkan ke *LCD* Passwor Salah !!.
7	Ke label C dihalaman sebelumnya.
8	Tampilkan ke *LCD* Password Salah !!
9	Nilai X + 1, lalu simpan hasilnya di X.
10	Cek, apakah X bernilai 3. Jika Ya, ke label D dihalaman sebelumnya. Jika Tidak, kembali membaca *password2* di *EEPROM.*
11	Ke label D dihalamanya sebelumnya.

Tabel 3.5. Keterangan *flowchart* edit *password.*

No.	Keterangan
1	Label *password.*
2	X diberi data bernilai 0. Membaca *password1* di *EEPROM.*
3	Tampilkan ke *LCD* Password Reset? Password :
4	Masukan *password1*.
5	Panggil label *keypad.*
6	Cek, apakah *password1* yang ditekan sama dengan *password1* di *EEPROM*? Jika Ya, lompat ke label *new*. Jika Tidak, tampilkan ke *LCD* Password Salah !!
7	Lompat ke label *new*.
8	Selesai.
9	Tampilkan ke *LCD* Password Salah !!
10	Nilai X + 1, lalu simpan hasilnya ke X.
11	Cek, apakah X bernilai 3? Jika Ya, ke label D dihalaman sebelumnya Jika Tidak, kembali membaca *password1* di *EEPROM.*
12	Ke label D dihalaman sebelumnya.

Berikut ini adalah gambar 3.16. *flowchart new password.*

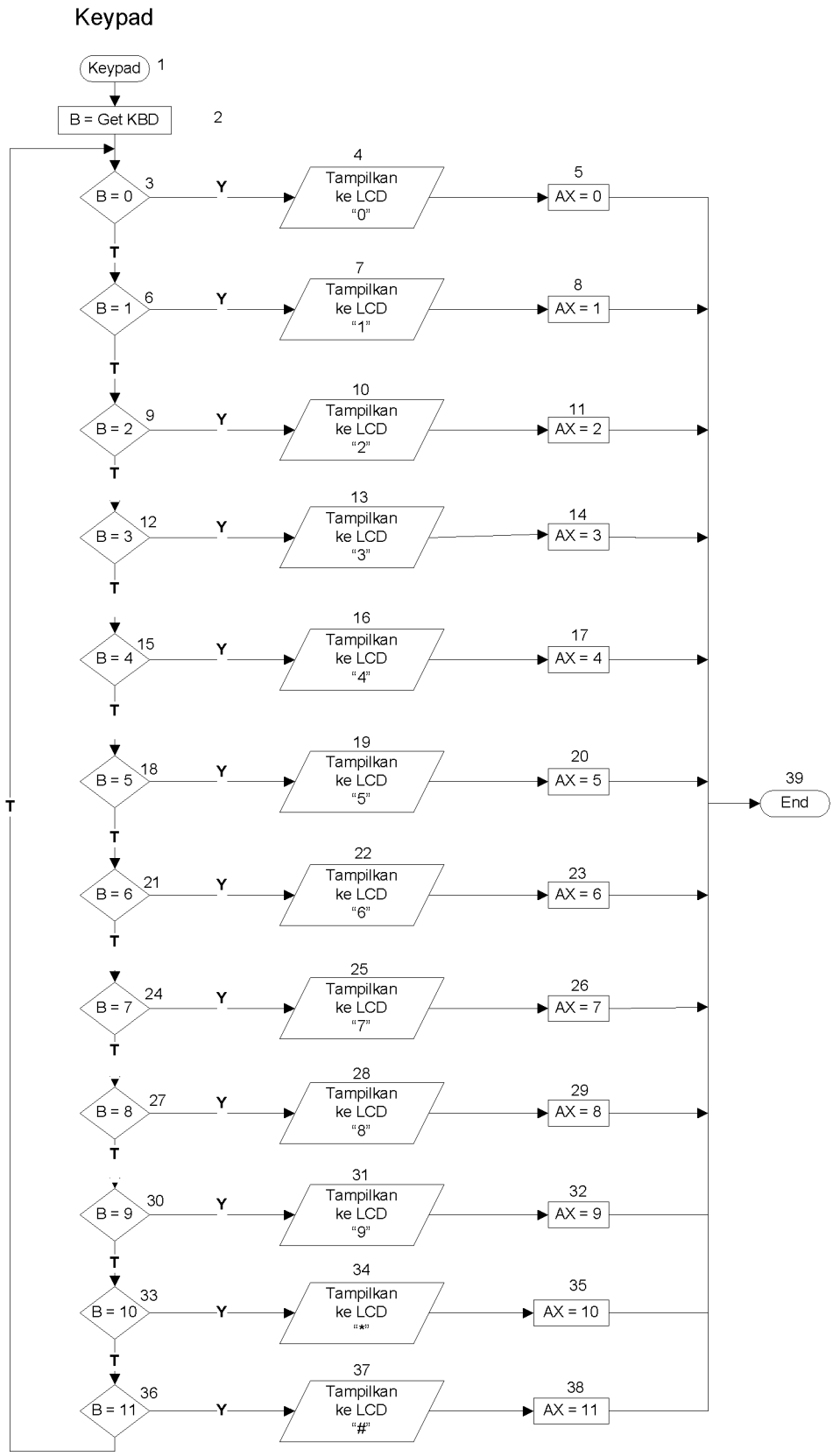


Gambar 3.16. *Flowchart new password.*

Tabel 3.6. Keterangan *flowchart new password.*

No.	Keterangan
1	Label New.
2	Tampilkan ke *LCD* New Password : Confirm Password :
3	Masukan *password2* yang baru.
4	Panggil label *keypad.*
5	Masukan *password2* yang sudah disetujui.
6	Panggil label *keypad.*
7	Cek, apakah *password2* yang baru sama dengan *password2* yang sudah di setujui? Jika Ya, simpan *password2* di *EEPROM.* Jika Tidak, tampilkan ke *LCD* New Password : Confirm Password :
8	Simpan *password2* di *EEPROM.*
9	Tampilkan ke *LCD* Edit Password Selesai
10	Tunggu selama 2 detik.
11	Tampilkan ke *LCD* Back Stabil (#) Edit Suhu (*)
12	Ke label C.
13	Label C.
14	Cek, apakah yang ditekan tanda (*)? Jika Ya, ke label D dihalaman sebelumnya. Jika Tidak, cek apakah yang ditekan tanda (#)?
15	Ke label D dihalaman sebelumnya.
16	Cek apakah yang ditekan tanda (#)? Jika Ya, ke label C dihalaman sebelumnya. Jika Tidak, cek apakah yang ditekan tanda (*)
17	Ke label C sebelumnya.

Berikut ini adalah gambar 3.17. *flowchart keypad.*



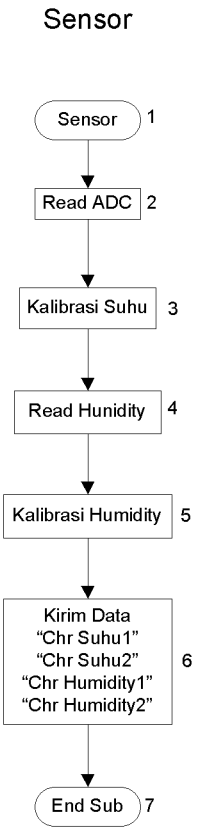
Gambar 3.17. *Flowchart keypad.*

Tabel 3.7. Keterangan *flowchart keypad.*

No.	Keterangan
1	Label *keypad*.
2	B sebagai variabel untuk menyimpan nilai input *keypad.*
3	Cek, apakah B bernilai 0? Jika Ya, tampilkan ke *LCD* 0 Jika Tidak, cek apakah B bernilai 1.
4	Tampilkan ke *LCD* angka 0.
5	AX bernilai 0.
6	Cek, apakah B bernilai 1? Jika Ya, tampilkan ke *LCD* 1 Jika Tidak, cek apakah B bernilai 2.
7	Tampilkan ke *LCD* angka 1.
8	AX bernilai 1.
9	Cek, apakah B bernilai 2? Jika Ya, tampilkan ke *LCD* 2 Jika Tidak, cek apakah B bernilai 3.
10	Tampilkan ke *LCD* angka 2.
11	AX bernilai 2.
12	Cek, apakah B bernilai 3? Jika Ya, tampilkan ke *LCD* 3 Jika Tidak, cek apakah B bernilai 4.
13	Tampilkan ke *LCD* angka 3.
14	AX bernilai 3.
15	Cek, apakah B bernilai 4? Jika Ya, tampilkan ke *LCD* 4 Jika Tidak, cek apakah B bernilai 5.
16	Tampilkan ke *LCD* angka 4.
17	AX bernilai 4.
18	Cek, apakah B bernilai 5? Jika Ya, tampilkan ke *LCD* 5 Jika Tidak, cek apakah B bernilai 6.
19	Tampilkan ke *LCD* angka 5.
20	AX bernilai 5.
21	Cek, apakah B bernilai 6? Jika Ya, tampilkan ke *LCD* 6 Jika Tidak, cek apakah B bernilai 7.
22	Tampilkan ke *LCD* angka 6.
23	AX bernilai 6.
24	Cek, apakah B bernilai 7? Jika Ya, tampilkan ke *LCD* 7 Jika Tidak, cek apakah B bernilai 8.
25	Tampilkan ke *LCD* angka 7.
26	AX bernilai 7.
27	Cek, apakah B bernilai 8? Jika Ya, tampilkan ke *LCD* 8 Jika Tidak, cek apakah B bernilai 9.
28	Tampilkan ke *LCD* angka 8.
29	AX bernilai 8.
30	Cek, apakah B bernilai 9? Jika Ya, tampilkan ke *LCD* 9 Jika Tidak, cek apakah B bernilai 10.
31	Tampilkan ke *LCD* angka 9.
32	AX bernilai 9.

33	Cek, apakah B bernilai 10? Jika Ya, tampilkan ke *LCD* * Jika Tidak, cek apakah B bernilai 11.
34	Tampilkan ke *LCD* tanda *.
35	AX bernilai 10.
36	Cek, apakah B bernilai 11? Jika Ya, tampilkan ke *LCD* # Jika Tidak, cek apakah B bernilai 0.
37	Tampilkan ke *LCD* tanda #.
38	AX bernilai 11.
39	Selesai.

Berikut ini adalah gambar 3.18. *flowchart* sensor.

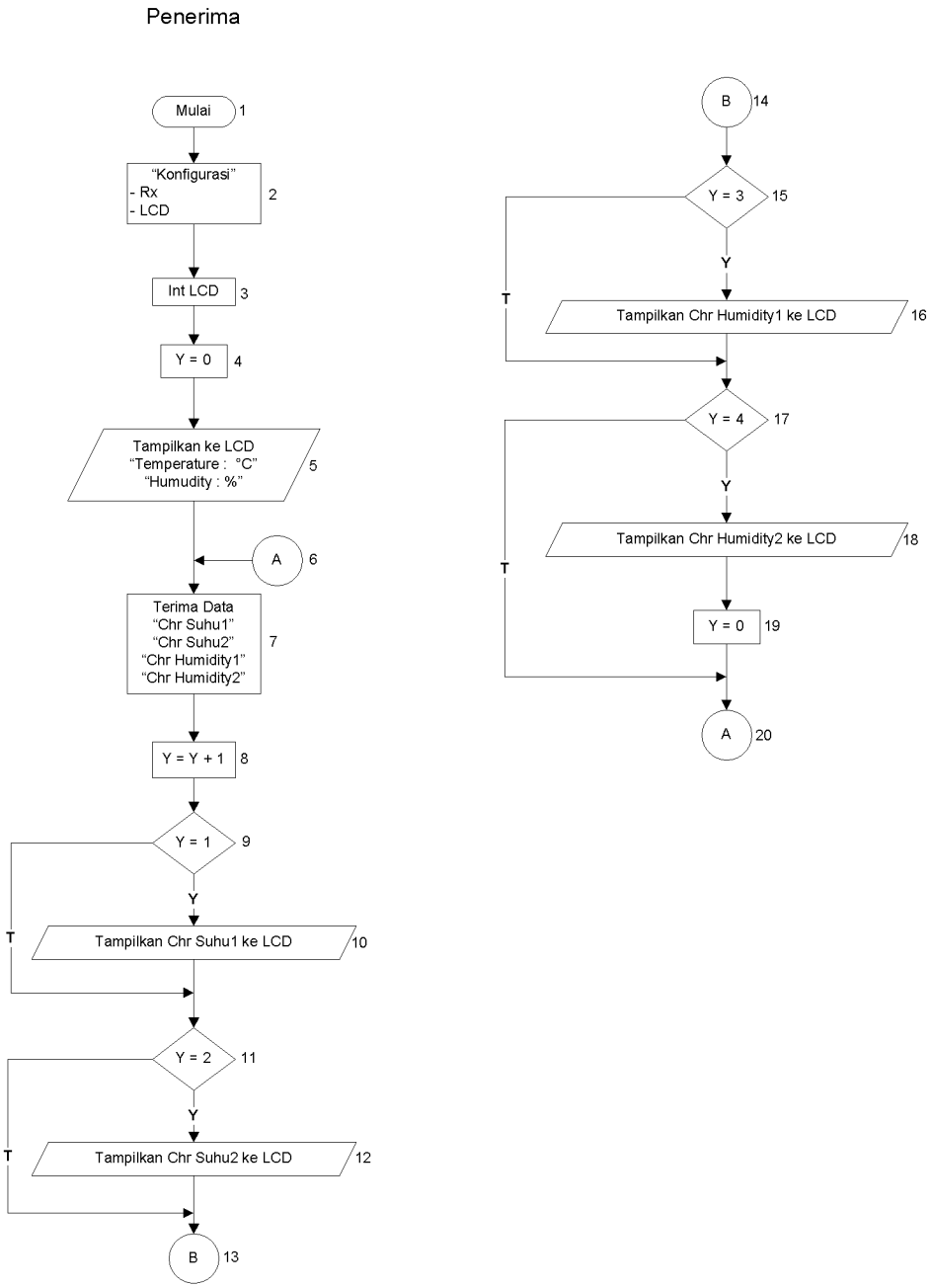


Gambar 3.18. *Flowchart* sensor.

Tabel 3.8. Keterangan *flowchart* sensor.

No.	Keterangan
1	Label sensor.
2	Membaca *ADC*.
3	Kalibrasi sensor suhu.
4	Membaca kelembaban.
5	Kalibrasi sensor kelembaban.
6	Mengirim data suhu1, suhu2, kelembaban1, kelembaban2.
7	Akhir sub.

Berikut ini adalah gambar 3.19. *flowchart* penerima.



Gambar 3.19. *Flowchart* penerima.

Tabel 3.9. Keterangan *flowchart* penerima.

No.	Keterangan
1	Mulai.
2	Konfigurasi *Input/Output* alat.
3	Inisialisasi LCD.
4	Y duberi data bernilai 0.
5	Tampilkan ke *LCD* Temperature : °C" Humidity : %.
6	Label A.
7	Menerimadata suhu1, suhu2, selembaban1, kelembaban2.
8	Nilai Y + 1, lalu simpan hasilnya di Y.
9	Cek, apakah Y bernilai 1 ? Jika Ya, tampilkan *Chr* Suhu1 ke *LCD*. Jika Tidak, cek, apakah Y bernilai 2.
10	Tampilkan data Suhu1 ke *LDC*.
11	Cek, apakah Y bernilai 2 ? Jika Ya, tampilkan ke *LCD* Chr Suhu2. Jika Tidak, ke label B.
12	Tampilkan data ke *LCD* Suhu2.
13	Ke label B.
14	Label B.
15	Cek, apakah Y bernilai 3 ? Jika Ya, tampilkan ke *LCD Chr Humidity1*. Jika Tidak, cek, apakah Y bernilai 4.
16	Tampilkan data ke *LCD* Kelembaban1
17	Cek, apakah Y bernilai 4 ? Jika Ya, tampilkan ke *LCD Chr Humidity2*. Jika Tidak, ke label A.
18	Tampilkan data ke *LCD* Kelembaban2.
19	Y bernilai 0
20	Ke label A