

**MIKROKONTROLER 80C31 SEBAGAI PEMONITOR
DAN PENGENDALI SUHU TUNGKU LISTRIK
25°C - 999°C SERTA MENGIRIM DATA
TEMPERATUR KE IBM PC**

TUGAS AKHIR

Disusun oleh :

Nama : Yves Azella

N.R.P. : 011860106

N.I.R.M. : 873206710250394



**JUSUSAN TEKNIK ELEKTRO
FFKULTAS TEKNOLOGI INDUSTRI
INTITUT TEKNOLOGI INDONESIA
SERPONG
1997**

**MIKROKONTROLER 80C31 SEBAGAI PEMONITOR
DAN PENGENDALI SUHU TUNGKU LISTRIK
25°C - 999°C SERTA MENGIRIM DATA
TEMPERATUR KE IBM PC**

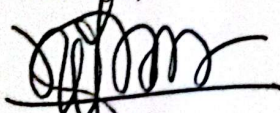
TUGAS AKHIR

Disusun oleh :

Yves Azella
(011860106)
(873206710250394)

Tugas Akhir ini telah diterima dan disahkan
untuk memenuhi persyaratan akademis
dalam menempuh Ujian Sarjana
Jurusan Elektro, Fakultas Teknologi Industri
Institut Teknologi Indonesia
Serpong

Disetujui,



Ir. Willem Pate, MSC.
Dosen Pembimbing

mengetahui,



Ir. Imam Sugandi
Ketua Jurusan elektro

ABSTRAK

Bagian utama dari alat ukur temperatur yang dibuat adalah mikrokontroler 80C31 buatan Intel. Fungsi utama 80C31 adalah sebagai pemonitor dan pengontrol temperatur tungku listrik. Nilai temperatur di tampilkan pada 7 segmen LED 3 digit. Jangkauan ukur temperatur 25°C - 999°C menggunakan sensor termokopel chomel - alumel type K.

Mikrokontroler 80C31 mengirim data serial ke IBM - PC secara "*on line*" dan menampilkan pada layar monitor berupa grafik (tegangan vs frekuensi, temperatur vs tegangan). Alat ukur yang dibuat ditera dengan alat ukur suhu buatan OMRON, dengan kesalahan maksimum 2°C.

KATA PENGANTAR

Puji dan Syukur penulis panjatkan kepada Tuhan Yang Maha Esa untuk rahmat dan karunia-Nya yang diberikan kepada penulis, sehingga penulis dapat menyelesaikan Tugas Akhir ini yang berjudul "*IBM PC sebagai penerima data dan pembuat grafik suhu 25°C - 999°C yang berasal dari MIKROKONTROLER 80C31*".

Tugas Akhir ini diperlukan sebagai syarat yang harus dipenuhi dalam Program Strata I di Institut Teknologi Indonesia, Serpong - Tangerang.

Pembuatan Tugas Akhir ini tidak lepas dari kerjasama berbagai pihak baik berupa bimbingan, saran maupun kritiknya yang baik dan bermanfaat bagi penulis dalam upaya penyusunan.

Pada kesempatan ini penulis ingin menyampaikan terima kasih yang sebesar-besarnya kepada Bapak Ir. Willem Pate. MSC selaku pembimbing Tugas Akhir yang telah memberikan bimbingan semangat dan pengarahan selama pembuatan dan penulisan Tugas Akhir ini. Sehubungan dengan hal tersebut, penulis menyampaikan rasa terimakasih yang sebesar-besarnya kepada :

1. Bapak Ir. Imam Sugandi, sebagai ketua jurusan Teknik Elektro,
Fakultas Teknologi Industri, Institut Teknologi Indonesia.

2. Ibu DR. Ir. Fatimah Z.P, sebagai koordinator Tugas Akhir pada jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Indonesia.
3. Papa, Mama tercinta, kakak Ir. Gestiviane Erlanda dan adik Okti Rimahdian yang selalu mendoakan dan memberikan dorongan moril maupun materiil selama ini.
4. Bapak Kusnandar, yang telah memberikan banyak bantuan dalam pemahaman aplikasi praktis rangkaian elektronika.
5. Ir. Eka Putra Astajaya yang telah banyak membantu menyediakan fasilitas perangkat keras dan perangkat lunak serta saran dan pemikiran.
6. Serta kepada teman-teman, Ir. Leonardo Tobing, Ir. Yayan Kurniawan, Ir. Afrit Cipto Sulaksono, Ir.Hendra, Ir. Arif, Ir. Teuku Fahrizal, Dra. Meliana, Ricardo Tobing S.Kom., Rita SH, Herman Nurya Kusuma S.Kom., Arief Erdiyanto Sony, Ir. Heriman dan rekan - rekan mahasiswa Elektro ITI lainnya, yang telah memberi dukungan, masukan dan tukar pendapat dalam menyelesaikan penyusunan tugas akhir ini.
7. Bapak Drs. A. Tamudji dan ibu, Bapak S.P.L. Tobing dan ibu yang telah memberikan bantuan secara moril kepada penulis hingga tugas akhir ini selesai.
8. Staf karyawan jurusan Teknik Elektro yang telah memberikan bantuannya dalam pengurusan tugas akhir ini hingga selesai.

Penulis menyadari bahwa Tugas Akhir ini tidak lepas dari berbagai kekurangan,

oleh karena itu penulis akan berterimakasih bila ada saran maupun kritik yang berguna bagi tercapainya Tugas Akhir ini dengan baik.

Akhir kata penulis berharap semoga tugas akhir ini dapat bermanfaat.

Jakarta, 1 April 1997

Penulis

Yves Azella
011860106

DAFTAR ISI

	Halaman
ABSTRAK	i
KATA PENGANTAR	ii
DAFTAR ISI	iv
DAFTAR GAMBAR	vii
DAFTAR TABEL	x
DAFTAR LAMPIRAN	xi
BAB I. PENDAHULUAN	1
1.1 Latar Belakang	1
1.2 Tujuan	2
1.3 Permasalahan	2

	Halaman
1.4 Batasan Permasalahan	3
1.5 Sistematika Penulisan	3
BAB II TEORI PENDUKUNG RANCANGAN SISTEM	6
2.1. Termokopel	6
2.1.1. Efek Seebach	7
2.1.2. Efek Peltier	8
2.1.3. Efek Thomson	8
2.2. Keluarga Mikrokontroler MCS-51	9
2.2.1. Organisasi memori	11
2.2.2. Special Function Register (SFR)	13
2.2.3. Pencacah dan pewaktu Mikrokontroler	14
2.2.4. Port I/O	16
2.3. Komunikasi serial	17
BAB III PERANCANGAN PERANGKAT KERAS	18
3.1. Transduser Termokopel	19
3.2. Termokopel terkalibrasi	19
3.2.1. Penguat operasi	20
3.2.2. Tegangan referensi LM 329.	21
3.2.3. Kalibrasi rangkaian.	22

	Halaman
3.3. Konverter tegangan ke frekuensi	23
3.4. Sistem pengontrol Mikrokontroler 8031	27
3.4.1. Mikrokontroler 80C31	28
3.4.2. IC Latch 74HCT373	30
3.4.3. EPROM	32
3.4.4. IC Register geser 74LS164	32
3.5. Layar peraga	34
3.6. Papan kunci	35
3.7. Saklar kendali daya AC	36
3.7.1. Triak	38
3.7.2. IC 4011	37
3.7.3. IC 4066	38
3.7.4. IC 4050	38
BAB IV PERANCANGAN PERANGKAT LUNAK	39
4.1. Mode pemantauan	40
4.1.1. Inisialisasi mode pemantauan .	41
4.1.2. Sampling data	41
4.1.3. Subrutin konversi data ke BCD	43
4.1.4. Subrutin konversi BCD ke 7 segmen	43
4.1.5. Subrutin tampilan layar peraga	44

	Halaman
4.2. Program mode kendali maksimum dan minimum	44
4.2.1. Input papan kunci	46
4.2.2. Aksi kendali ON/OFF	47
4.3. Program mode kendali pewaktu	48
4.4. Master Terminal Unit (MTU)	50
BAB V UJI COBA SISTEM	51
5.1. Pengukuran termokopel termometer terkalibrasi	51
5.2. Pengukuran tegangan ke frekuensi uji coba serial	54
5.3. Perbandingan hasil pemantauan	56

KESIMPULAN

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR GAMBAR

	Halaman
Gambar 2.1. Efek Seebeck	7
Gambar 2.2. Diagram Blok Keluarga Mikrokontroler MCS-51	10
Gambar 2.3. Daerah Memori Data Internal	12
Gambar 2.4. Off-Chip Oscillator	14
Gambar 2.5. Diagram Pewaktu Urutan Eksekusi	15
Gambar 3.1. Diagram Blok Pemantau dan Pengendali Temperatur	19
Gambar 3.2. Rangkaian Termokopel Termometer Terkalibrasi	20
Gambar 3.3. Rangkaian Konverter Tegangan ke Frekuensi	23
Gambar 3.4. Diagram Blok yang Disederhanakan dari LM331	24
Gambar 3.5. Grafik Keluaran Akibat Pengisian dan Pengosongan Kapasitor	26
Gambar 3.6. Sistem Mikrokontroler dan Layar Peraga	28
Gambar 3.7. Rangkaian Logika dan Operasi Tunggal 74HCT373	32
Gambar 3.8. Rangkaian Logika IC 74LS164 Untuk Tipe SIPO	33
Gambar 3.9. Rangkaian Papan Kunci	35
Gambar 3.10. Rangkaian Saklar Kendali Daya AC	36
Gambar 4.1. Diagram Alir Program Mode Pemantauan	40
Gambar 4.2. Diagram Alir Program Mode Kendali Maksimum Minimum	45
Gambar 4.3. Diagram Alir Program Mode Kendali Pewaktu	49
Gambar 5.1a. Grafik Pengukuran Tegangan Termokopel Berbanding Temperatur	53

	Halaman
Gambar 5.1b. Grafik Pengukuran Tegangan Output (TTk) Berbanding Temperatur	53
Gambar 5.2a. Grafik Pengukuran Frekuensi Output (v/f) Berbanding Tegangan Output (TTK)	55
Gambar 5.2b. Grafik Tampilan 7 - Segmen Berbanding Tegangan Output Output (ttk)	55
Gambar 5.3. Grafik Perbandingan Besar Temperatur	57

DAFTAR TABEL

	Halaman
Tabel 2.1. Keluarga Mikrokontroler MCS -51	9
Tabel 2.2. Special Function Register (SFR)	13
Tabel 2.3. Fungsi Khusus Port 1 dan Port 3	16
Tabel 3.1. Tabel Kebenaran Operasi Tunggal 74HCT373	31
Tabel 3.2. Tabel Kebenaran IC Register Geser 74HCT164	34
Tabel 5.1. Pengukuran Termometer Termokopel Terkalibrasi	52
Tabel 5.2. Pengukuran Konverter Tegangan ke Frekuensi	54
Tabel 5.3. Hasil perbandingan Pemantauan Temperatur	56

DAFTAR LAMPIRAN

- LAMPIRAN A.** FOTO ALAT PEMANTAU DAN PENGENDALI
TEMPERATUR 25 ° C - 999 ° C BERBASIS
MIKROKONTROLER - 80C31
- LAMPIRAN B** DIAGRAM ALIR PROGRAM TEMPERATUR DAN
PENGENDALI TEMPERATUR 25 ° C - 999 ° C
BERBASIS MIKROKONTROLER - 80C31

BAB I

PENDAHULUAN

1.1. Latar Belakang.

Perkembangan teknologi industri dan elektronika di Indonesia akhir-akhir ini sangat pesat. Gejala tersebut terlihat juga pada laboratorium penelitian dan pengembangan industri keramik. Di laboratorium ini dilakukan penelitian-penelitian untuk mengetahui sifat-sifat dasar dan karakteristik yang penting dari bahan keramik baru. Penelitian juga dilakukan untuk mencari metoda uji yang baru dan pembuatan uji yang baru dan pembuatan peralatan uji yang sesuai. Salah satu yang paling penting adalah mencari metoda uji baru untuk pemanasan bahan keramik dan alat pemanasan (*heater*).

Pemanasan dilakukan untuk mengetahui sifat suatu bahan keramik pada teperatur tertentu. Pemanasan dilakukan juga untuk mengetahui suhu pengolahan optimumnya.

Daerah temperatur yang dianalisa sampai 1000° C. Hal ini membutuhkan pemanas (*heater*) suhu tinggi yang terkendali.

Berdasarkan hal di atas, telah dirancang dan dibuat kendali temperatur yang dapat beroperasi sampai temperatur 999°C dengan pengendalian *On-Off*. Pengendalian temperatur mengendalikan pemanas (*heater*) pada daerah harga temperatur yang telah ditentukan. Pengendalian temperatur memiliki kemampuan menyimpan 4 buah harga temperatur dan lama waktu pengendalian pada masing-masing harga temperatur tersebut. Berdasarkan spesifikasi di atas penggunaan pengendalian temperatur ini melebar melampaui tujuan awalnya. Pengendalian temperatur ini dapat dipakai untuk bermacam-macam aplikasi asalkan sesuai dengan batas-batas kemampuannya.

1.2 . Tujuan.

Merancang dan membuat alat pemantau dan pengendali temperatur dengan jangkah ukur 25°C - 999°C berbasis keluarga mikrokontroler *MCS-51 (80C31)*. Pengendalian temperatur mempunyai 2 mode. Mode pertama, dapat menyimpan sampai 4 buah harga temperatur dengan lama waktu pengendaliannya. Mode kedua, pengendalian temperatur pada daerah harga maksimum dan minimum tertentu.

Dengan menggunakan *RS 232* data dikirim secara serial ke IBM PC dengan menggunakan COM1 dan ditampilkan ke layar monitor dalam bentuk grafik dengan menggunakan perangkat lunak Visual Basic

1.3 . Permasalahan.

Memilih transduser temperatur yang sesuai. Merancang rangkaian penguat tegangan transduser temperatur dan menambahkan harga tegangan yang sesuai dengan temperatur

ruang Keluaran rangkaian ini diharapkan linier sebesar $10 \text{ mV}^{\circ}\text{C}$. Merancang tegangan konverter ke frekuensi yang linier. Keluaran pulsa rangkaian konverter tegangan ke frekuensi merupakan masukan rangkaian sistem mikrokontroler 80C31.

Sistem kontroler 80C31 memberikan keluaran harga temperatur di 7-segmen dan pengendalian temperatur sesuai masukan papan kunci (*keypad*). Saklar pengendali daya AC memutuskan atau menghubungkan pemanas (*heater*) dengan tegangan jala-jala sehingga terjadi pengendalian *On-Off*.

1.4 . Batasan masalah.

Pembahasan teori secara singkat, hanya pada bagian yang langsung berhubungan dengan cara kerja rangkaian secara garis besar. Pembahasan komponen hanya berdasarkan kegunaannya pada rangkaian. Perangkat lunak yang diperlukan pada sistem mikrokontroler dibahas berdasarkan kesatuan instruksi untuk suatu tujuan tertentu. Hasil pengukuran dan kalibrasi alat dilakukan tidak pada tiap derajat temperatur, tetapi pada beberapa besaran temperatur yang dianggap dapat mewakili penunjukkan kelinieran hasil pengukuran dan kalibrasi.

1.5. Sistematika penulisan.

Sistematika penulisan tugas akhir ini terdiri dari 5 (lima) bab, kesimpulan dan beberapa lampiran.

BAB I : berisikan pendahuluan yang menjelaskan tentang latar belakang, Tujuan dan Sistematika penulisan.

BAB II : pembahasan tentang bermacam jenis transduser yang umum dipakai seperti

termokopel, resistansi platinum dan pertemuan semikonduktor. Membahas keluarga mikrokontroler MCS-51, pembahasan meliputi spesifikasi umum anggota keluarganya, organisasi memori, register fungsi khusus yang dimiliki, osilator dan pewaktu, dan port input-outputnya.

BAB III berisikan perangkat keras pemantau dan pengendali temperatur.

Pembahasan meliputi rangkaian termokopel termometer terkalibrasi, rangkaian konverter tegangan ke frekuensi, rangkaian sistem mikrokontroler, rangkaian papan kunci (*keypad*) dan rangkaian saklar kendali daya AC untuk pengendalian *on - off* pemanas (*heater*).

BAB IV : membahas perangkat lunak. Pembahasan meliputi kumpulan instruksi yang membentuk pengoperasian tertentu dari pemantauan temperatur, pengendalian temperatur maksimum minimum dan pengendalian temperatur dengan waktu.

BAB V : membahas hasil pengukuran dan kalibrasi keluaran tegangan termokopel, keluaran termokopel termometer terkalibrasi, keluaran frekuensi konverter tegangan ke frekuensi, tampilan harga temperatur di *7-segmen* dan data aksi pengendalian sesuai dengan masukan dari papan kunci (*keypad*). Besaran temperatur hasil pemantau alat pemantau dan pengendali temperatur ini juga dapat dibandingkan dengan besaran temperatur hasil pemantauan alat pengendali temperatur buatan OMRON sebagai acuan kalibrasi.

KESIMPULAN merupakan penutup makalah. Di sini penulis mencoba menarik kesimpulan mengenai hal-hal penting yang terdapat pada alat pemantau dan pengendali temperatur. Kesimpulan ini juga didapat berdasarkan hasil pengujian dan kalibrasi.

BAB II

TEORI PENDUKUNG RANCANGAN SISTEM

2.1. TERMOKOPEL

Termokopel adalah tansduser temperatur yang dibuat dari pertemuan dua logam yang berlainan jenis. Gaya gerak listrik (*GGL*) yang dibangkitkan sebanding dengan perbedaan temperatur antara kedua pertemuan itu. Pertemuan antara kedua logam yang dikenakan panas dinamakan pertemuan panas (*Hot Junction*) sedang ujung kedua logam pada sisi yang lain dinamakan pertemuan dingin (*Cold Junction*). Gambar 2.1

Termokopel

Tegangan pada ujung pertemuan dingin akan bertambah dengan makin besarnya perbedaan temperatur antara titik pertemuan panas dan titik pertemuan dingin, sehingga terjadi gejala termolistrik yang terdiri dari :

- Efek Seeback.

- Efek Peltier.
- Efek Thomson.

2.1.1 Efek Seebeck.

Dalam suatu rangkaian tertutup yang terdiri dari dua logam yang berbeda dan membentuk dua pertemuan yang berbeda pada temperatur yang berbeda, maka akan ada arus listrik mengalir di dalam rangkaian tersebut. Besarnya arus listrik ini tergantung pada jenis logam dari masing-masing konduktor dan beda temperatur antara kedua pertemuan. Besarnya tegangan yang ditimbulkan adalah ^[6]:

$$E = \mu_{s12} \times \Delta t \quad \dots\dots\dots (2.1)$$

Dimana :

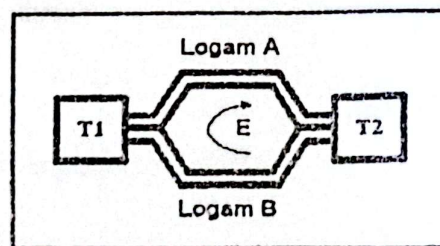
E = Tegangan yang ditimbulkan (*Volt*).

μ_{s12} = Koefisien Seebeck ($\mu V/^{\circ}C$).

Δt = Selisih temperatur titik pertemuan panas.

($T1^{\circ}C$) dan dingin ($T2^{\circ}C$)

Logam A dikatakan positif terhadap logam B bila arah arus pada titik pertemuan dingin ($T2$) berasal dari logam B menuju logam A, seperti pada gambar 2.1.



Gambar 2.1. Efek Seebeck.

2.1.2. Efek Peltier.

Bila arus mengalir pada dua logam berbeda yang membentuk dua titik pertemuan, maka pada kedua titik pertemuan tersebut akan terjadi penyerapan atau pelepasan panas yang tergantung pada arah arusnya. Panas yang diserap atau dilepaskan pada suatu titik hubung besarnya diberikan oleh persamaan^[6]:

$$Q = \mu_{p12} \times I \quad \dots\dots\dots(2.2)$$

Dimana :

Q = Jumlah panas yang diserap (*kalori/det*)

μ_{p12} = Koefisien Peltier (*kalori/det ampere*)

I = Arus yang mengalir (*ampere*)

Koefisien Peltier berharga positif bila panas dilepaskan saat arus mengalir dari logam B ke logam A pada titikpertemuan dingin (T_2).

2.1.3. Efek Thomson.

Bila arus listrik mengalir pada suatu konduktor yang homogen, akan terjadi penyerapan atau pelepasan panas yang berbeda pada tiap titik dari konduktor dan membentuk suatu gradien temperatur. Panas yang diserap atau yang dilepaskan adalah^[6]:

$$Q = \mu_T \cdot I \cdot dt/dx \quad \dots\dots\dots(2.3)$$

dimana :

Q = Jumlah panas yang diserap (*kalori det*)

μ_T = Koefisien Thomson (*kalori cm/det ampere °C*)

I = Arus yang mengalir (*ampere*)

dt/dx = Gradien temperatur ($^{\circ}\text{C cm}$)

t = Temperatur ($^{\circ}\text{C}$)

x = Jarak (*cm*)

2.2. KELUARGA MIKROKONTROLER MCS-51.

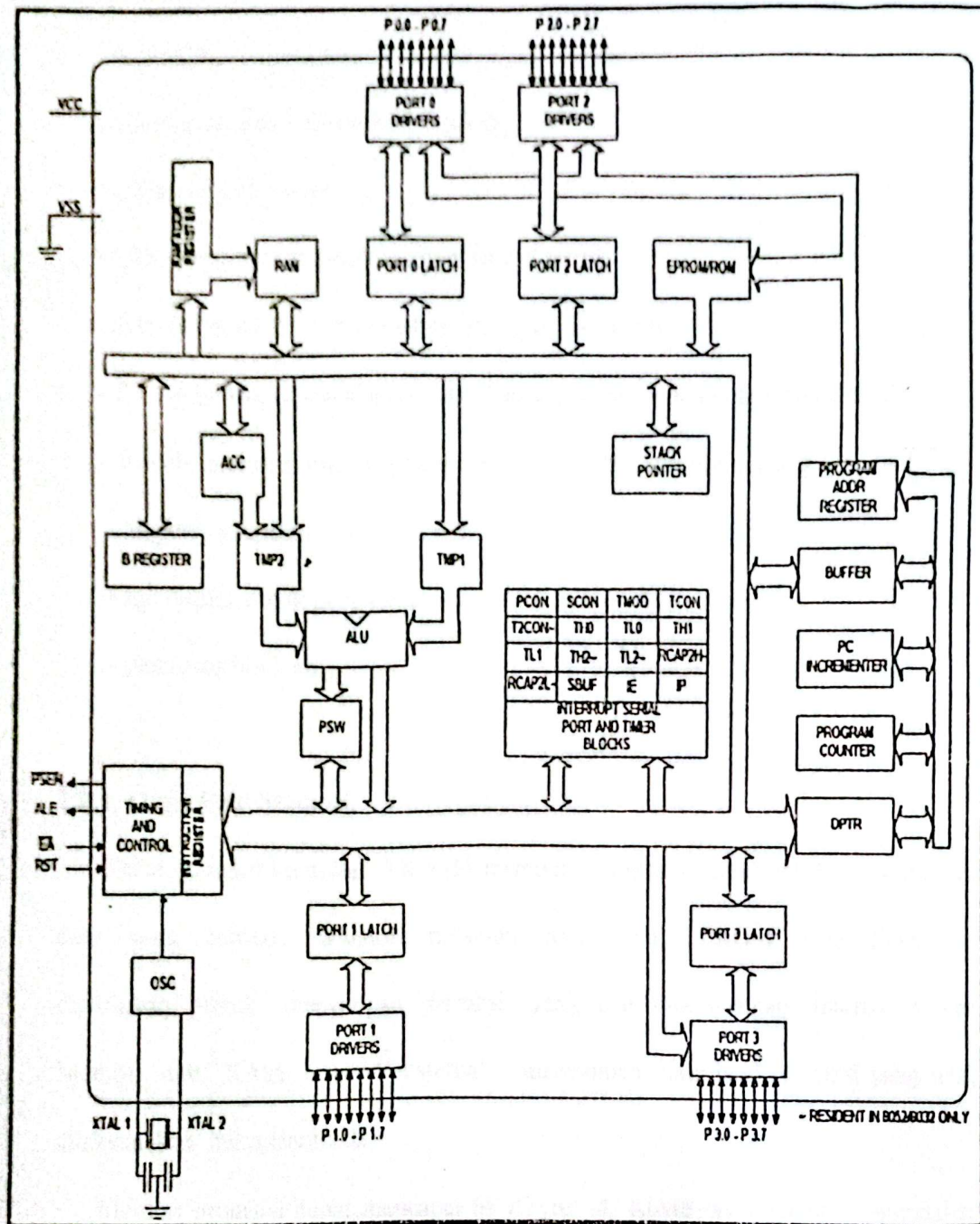
Dewasa ini mikrokontroler banyak dipakai pada peralatan elektronika yang memerlukan sistem pemrosesan dan pengendalian yang handal dengan hanya satu serpih tunggal (*single chip*). Salah satu mikrokontroler yang umum dan luas pemakaiannya adalah keluarga mikrokontroler MCS-51.

Perbedaan keluarga mikrokontroler MCS-51 antara satu dengan yang lainnya yang umum seperti tabel 2.1

MIKROKONTROLER	TEKNOLOGI	MEMORI PROGRAM INTERNAL	MEMORI DATA INTERNAL
8051	HMOS	4KByte ROM	128 byte
8031	HMOS	TIDAK ADA	128 byte
8751H	HMOS1	4KByte EPROM	128 byte
80C51	CHMOS	4KByte ROM	128 byte
80C31	CHMOS	TIDAK ADA	128 byte
8052	HMOS	8KByte ROM	256 byte
8032	HMOS	TIDAK ADA	256 byte

Tabel 2.1 Keluarga Mikrokontroler MCS-51.

Diagram blok yang umum dari keluarga mikrokontroler MCS-51 seperti gambar 2.2.



Gambar 2.2. Diagram blok keluarga mikrokontroler MCS-51

Berdasarkan diagram blok diatas diketahui karakteristik utama keluarga mikrokontroler MCS-51 adalah :

- 8-bit CPU
- on-chip oscillator dan sirkit pewaktu.
- 32 port input/output
- 64K ruang alamat untuk memori data eksternal.
- 64K ruang alamat untuk memori program eksternal.
- 2 buah pewaktu/pencacah (*timer/counter*) 16-bit, 3 buah pada 8032/8052.
- 5 buah sumber interupsi, 6 buah pada 8032/8052, masing-masing dengan 2 tingkat prioritas.
- full duplex untuk port serial.
- pemroses boolean.

2.2.1. Organisasi Memori.

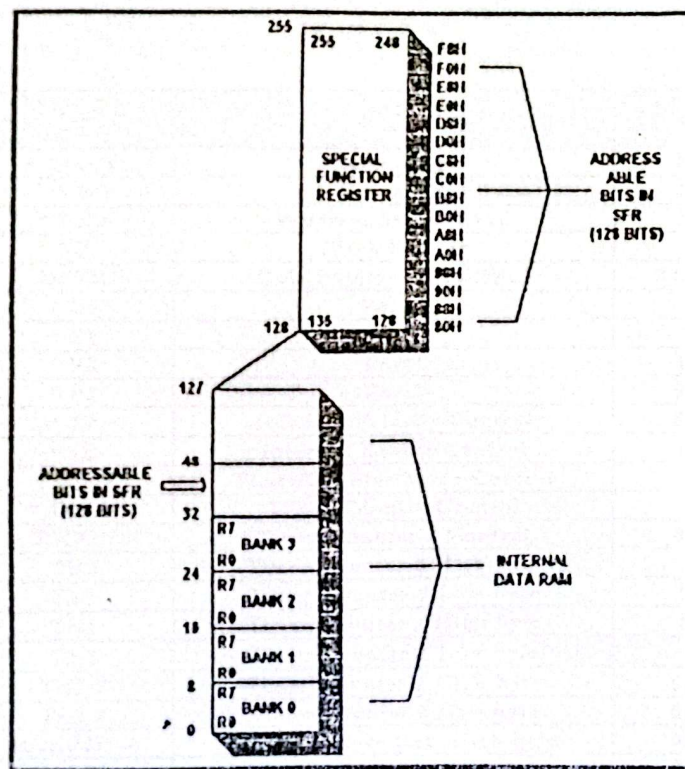
Keluarga mikrokontroler MCS-51 mempunyai lokasi memori untuk program dan data yang berbeda. Memori program ROM atau EPROM internal/eksternal diperlukan untuk menyimpan instruksi yang akan dilaksanakan mikrokontroler. Memori data RAM internal/eksternal menyimpan data hasil instruksi yang telah dilaksanakan mikrokontroler.

Memori program dapat mencapai 64 Kbyte. 4 Kbyte awal dapat menggunakan ROM atau EPROM yang tersedia di dalam mikrokontroler untuk keluarga mikrokontroler MCS-51 yang berakhiran 51, dan 8 Kbyte untuk yang berakhiran 52.

Pada pemakaian memori program eksternal/EPROM, terminal EA (*External Access*), pin 31, harus dibuat logika rendah secara permanen. Apabila terminal EA

dibuat berlogika tinggi, maka eksternal memori program akan dimulai dari 4 Kbyte atau Kbyte memori program internal.

Mikrokontroler mempunyai kemampuan membaca dan menulis data dari dan ke memori data hingga ukuran maksimum 64 Kbyte eksternal. Memori data internal terdiri dari RAM data internal berukuran 128 byte untuk keluarga mikrokontroler MCS-51 yang berakhir 31 dan 51, 256 byte untuk yang berakhir 32 dan 52, dan SFR (*Special Function Register*). Konfigurasi memori data internal dari 128 RAM data internal dan SFR seperti gambar 2.3.



Gambar 2.3. Daerah memori data internal.

RAM data internal juga digunakan oleh fungsi umum mikrokontroler R0-R7, dan penyimpanan data tumpukan (*stack*) untuk manajemen subrutin. Besar byte yang bebas dipakai oleh pemrograman biasanya lebih kecil dari 128 byte, sedangkan daerah antara 20H dan 2Fh dari memori data internal, dinamakan daerah pengalamatan bit (*bit-addressable range*), yang berguna untuk instruksi manipulasi bit.

2.2.2. Special Function Register (SFR).

Tabel SFR (*Special Function Register*) yang berisi simbol, nama dan alamat apada memori data internal seperti tabel 2.2.

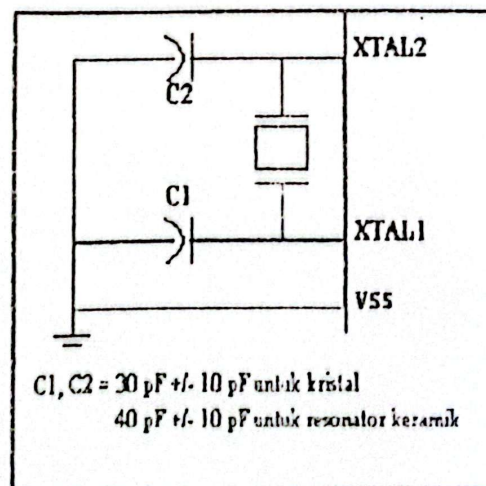
SIMBOL	NAMA	ALAMAT
* A/ACC	Akumulator	0E0H
* B	B Register	0F0H
* PSW	Program Status Word	0D0H
SP	Stack Pointer	81H
DPTR	Data Pointer (DPH-DPL)	82H-83H
* P0	Port 0	80H
* P1	Port 1	90H
* P2	Port 2	0A0H
* P3	Port 3	0B0H
* IP	Interrupt Priority Control	0B8H
* IE	Interrupt Enable Control	0A8H
TMOD	Timer/Counter Mode Control	89H
* TCON	Timer/Counter Control	88H
* T2CON #	Timer/Counter 2 Control	0C8H
TH0	Timer/Counter 0 High byte	8CH
TL0	Timer/Counter 0 Low byte	8AH
TH1	Timer/Counter 1 High byte	8DH
TL1	Timer/Counter 1 Low byte	8BH
TH2 #	Timer/Counter 2 High byte	0CDH
TL2 #	Timer/Counter 2 Low byte	0CCH
RCAP2H #	T/C 2 Capture Reg. High byte	0CBH
RCAP2L #	T/C 2 Capture Reg. Low byte	0CAH
* SCON	Serial Control	98H
SBUF	Serial Data Buffer	99H
PCON	Power Control	87H

Tabel 2.2. Special Function Register.

SFR (*Special Function Register*) adalah register-register untuk fungsi-fungsi khusus yang berada di memori data internal dengan alamat lebih besar dari 07FH. Register-register ini diperlukan untuk pemakaian akses ke register dengan instruksi sedikit mungkin. Masing-masing SFR mempunyai alamat tertentu. Register A (*Accumulator*), sebagai register utama, mampu melakukan operasi logika dan aritmatika, mempunyai alamat internal 0E0H. Register B, PSW (*Program Status Word*) port, stack, data pointer, serial buffer, timer register, capture register dan control register, mempunyai alamat internal tertentu dan juga merupakan bagian dari SFR.

2.2.3. Pencacah dan Pewaktu Mikrokontroler.

XTAL1 dan XTAL2 adalah masukan dan keluaran on-chip oscillator dari mikrokontroler yang dapat di konfigurasi dengan komponen luar (*off-chip oscillator*) sebagai pewaktu seperti pada gambar 2.4. Untuk pemakaian off-chip oscillator, dapat digunakan (*quartz*) atau ceramic resonator.

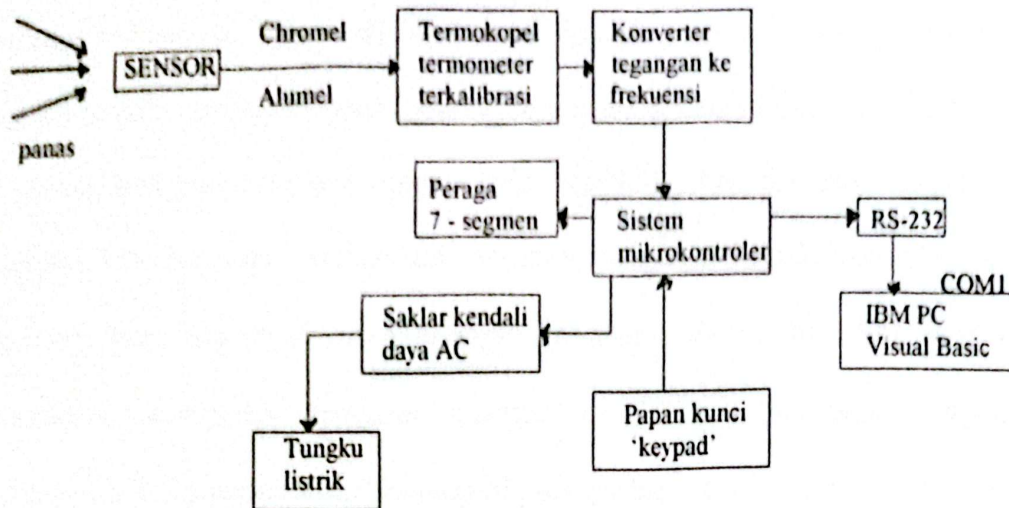


Gambar 2.4. Off-Chip Oscillator.

BAB III

PERANCANGAN PERANGKAT KERAS

Pada prinsipnya, dilihat dari perangkat keras diagram blok pemantau dan pengendali temperatur dengan jangkah ukur 25°C - 999°C menggunakan termokopel sebagai transdusernya. Rangkaian penguat terkalibrasi, konverter tegangan ke frekuensi, mikrokontroler sebagai pengolah masukan frekuensi dan papan kunci, saklar kendali daya AC, dan sebagai transmitter data serial yang menggunakan RS 232 untuk dikirim ke komputer pribadi (*PC Kompatibel*).



Gambar 3.1. Diagram blok kendali temperatur.

3.1. Transduser Termokopel

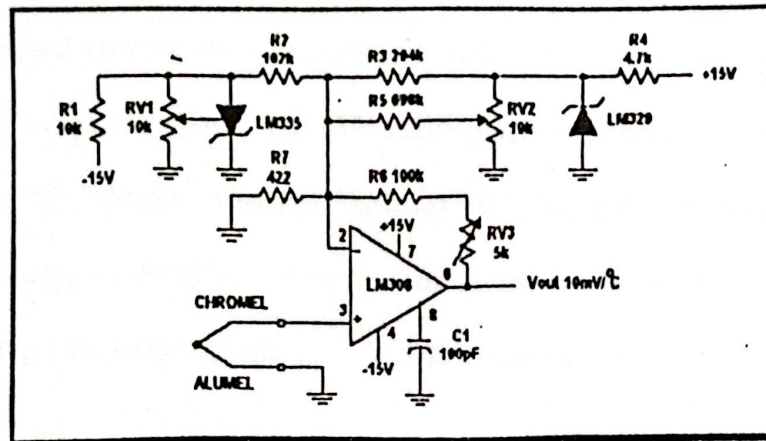
Transduser yang digunakan adalah termokopel tipe K(*Chromel-Alumel*) yang dapat bekerja dengan baik antara temperatur -200°C - 1370°C . Termokopel merubah besaran temperatur/panas menjadi besaran tegangan. Koefisiensi Seebach untuk termokopel tipe K adalah $40,8 \mu\text{V}/^{\circ}\text{C}$. Ini berarti setiap kenaikan suhu 1°C menjadi kenaikan tegangan pada pertemuan dingin termokopel sebesar $40,8 \mu\text{V}$. Besar tegangan yang dihasilkan termokopel ini sangat kecil dan belum mencakup informasi temperatur ruangan. Solusi terpecahkan dengan menghubungkan termokopel kerangkaian termokopel termometer terkalibrasi.

3.2. Termokopel Termometer Terkalibrasi.

Rangkaian termokopel termometer terkalibrasi diperlukan untuk mendapatkan penguatan tegangan dan konversi yang linier dari perubahan temperatur ke tegangan. Besar temperatur ruangan juga diperhitungkan. Instrumen dapat memberikan informasi temperatur ruangan pada saat termokopel belum membaca

informasi temperatur. Bila termokopel sudah dikenakan panas, hasil pengamatan adalah jumlah dari temperatur ruangan dan temperatur pengamatan termokopel.

Rangkaian ini terdiri dari penguat diferensial LM 308, pengamat temperatur ruangan LM 335, dan kompensasi tegangan dengan frekuensi zener LM 329. Kalibrasi yang baik memberikan tegangan keluaran $V_{out} \approx 10 \text{ mV}/^{\circ}\text{C}$ sebagai masukan kerangkaian konverter tegangan ke frekuensi. Rangkaian lengkap termokopel termometer terkalibrasi seperti pada gambar 3.2.



Gambar 3.2 . Rangkaian termokopel termometer terkalibrasi.

3.2.1. Sensor Temperatur Ruangan LM 335

LM 335 digunakan untuk mendapatkan informasi temperatur ruang, sehingga pemantauan temperatur ini memberikan hasil yang merupakan jumlah temperatur ruang dengan temperatur panas yang dikenakan ke termokopel. Hal ini merupakan

salah satu kelebihan instrumen dimana besar temperatur ruang sudah diperhitungkan dan besaran yang dihasilkan mempunyai referensi temperatur ruangan $\approx 25^{\circ}\text{C}$.

LM 335 beroperasi dari -40°C sampai $+100^{\circ}\text{C}$, dan mempunyai keluaran yang linier. Beroperasi seperti terminal zener, LM 335 akan menghasilkan tegangan breakdown yang proporsional dengan temperatur absolut dan menghasilkan keluaran $+10\text{ mV/^{\circ}K}$ untuk penerapan catu positif. Tidak seperti sensor lainnya, LM 335 bila sudah dikalibrasi pada temperatur 25°C hanya menghasilkan kesalahan dibawah 1°C pada keseluruhan daerah operasinya.

Potensiometer yang dihubungkan ke lengan adjust LM 335 akan menghasilkan suatu nilai kalibrasi yang membuat keakuratan seluruh pengukuran temperatur pada daerah operasi LM 335. Keluaran LM 335 sudah proporsional dengan temperatur absolut ($10\text{ mV/^{\circ}K}$), dengan tegangan keluarga 0 Volt pada temperatur 0°K ($-273,15^{\circ}\text{C}$). Dengan demikian, hanya mencari kesesuaian tegangan dengan temperatur ruang saat kalibrasi, yaitu 2,982 V pada temperatur 25°C .

3.2.2. Tegangan Referensi LM 329

LM 329 adalah referensi zener 6,9 V yang digunakan untuk kompensasi kelebihan tegangan yang dihasilkan oleh sensor temperatur ruang LM 335. Diketahui bahwa tegangan yang dihasilkan sensor temperatur LM 335 adalah 2.982 mV pada suhu 25°C . Diharapkan konversi linier keluaran adalah $10\text{ mV/^{\circ}C}$. Jadi untuk temperatur 25°C tegangan seharusnya 250 mV. Kesalahan ini diperbaiki dengan menggunakan LM 329.

3.2.3. Kalibrasi Rangkaian

Setelah komponen-komponen utama dari rangkaian termokopel termometer terkalibrasi diketahui, dapat dijelaskan kerja rangkaian dengan penguatan yang diperlukan dan dikalibrasi sehingga didapat tegangan keluaran rangkaian $V_{out} \approx 10 \text{ mV/}^{\circ}\text{C}$.

Tegangan yang dihasilkan termokopel dihubungkan ke masukan tak membalik adalah $40,8 \mu\text{V/}^{\circ}\text{C}$. Untuk mendapatkan $V_{out} \approx 10 \text{ mV/}^{\circ}\text{C}$ diperlukan penguatan pada masukan tak membalik $\approx 245,7$ kali. Penguatan yang diinginkan didapat dengan mengatur tahanan variabel RV3.

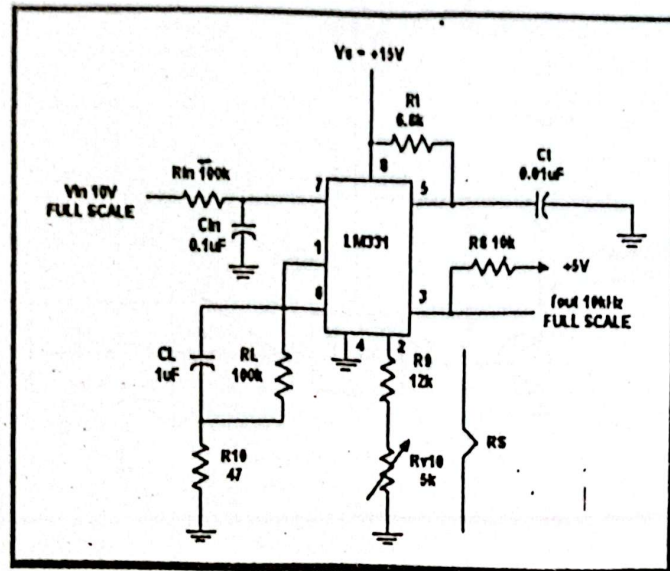
LM 335 yang dicatu dengan tegangan negatif dihubungkan pada masukan membalik sehingga di dapat penambahan tegangan keluaran yang sesuai dengan temperatur ruang sebesar $2,982 \text{ V}$. Penguatan tidak perlukan diberikan pada masukan ini karena keluaran LM 335 sudah linier dan sesuai dengan keluaran rangkaian termokopel termometer terkalibrasi yang dikehendaki.

Kesalahan tegangan referensi hasil penambahan tegangan LM 335 sebesar $2,732 \text{ V}$ diperbaiki oleh LM 329 yang dicatu tegangan positif dan diupayakan masukan membalik.

Tegangan keluaran LM 329 stabil sebesar $6,9 \text{ V}$. Untuk mendapat pengurangan sebesar $2,732 \text{ V}$ dibutuhkan penguatan sekitar $0,5$ kali pada masukannya dengan sedikit pengaturan tegangan keluaran LM 329 menggunakan resistor variabel, sehingga menghasilkan $V_{out} \approx 10 \text{ mV/}^{\circ}\text{C}$ yang sesuai dengan pembacaan sensor temperatur ruang dan tempertur yang dikenakan pada termokopel.

3.3. Konverter Tegangan ke Frekuensi.

Gambar konverter tegangan ke frekuensi dengan menggunakan IC LM331 seperti gambar 3.3.

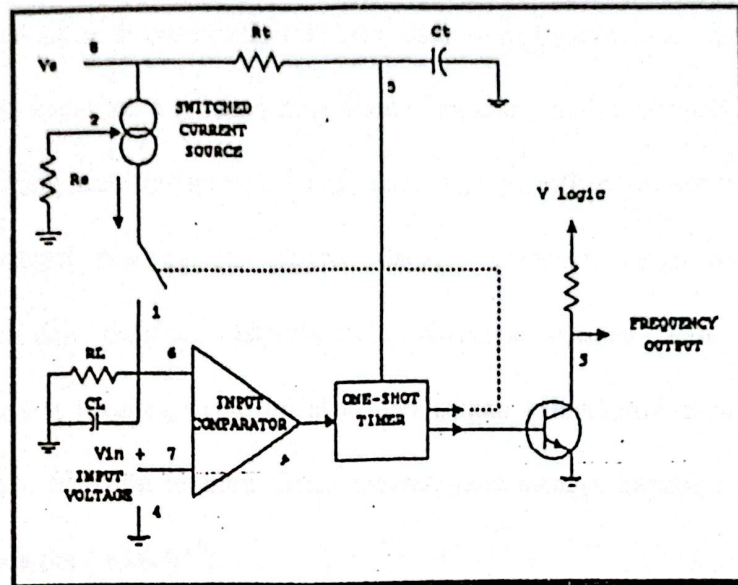


Gambar 3.3. Rangkaian tegangan ke frekwensi.

Komponen utama rangkaian konverter tegangan ke frekuensi adalah LM 331 yang mempunyai kemampuan lebar skala frekuensi penuh $f_{out} = 10$ kHz pada tegangan masukan skala penuh $V_{in} = 10$ V dengan kelinieran 0,03 %. Keluaran dari komponen ini dapat mengemudikan 3 buah TTL (*transistor-transistorlogic*).

LM 331 dirancang untuk operasi yang akurat dari konverter tegangan ke frekuensi atau konverter frekuensi ke tegangan. Diagram blok yang disederhanakan dari LM 331 terdiri dari saklar sumber arus, cermin arus, masukan komparator, dan sebuah monostabil (*one-shot*) yang merupakan salah satu bagian penting dari pewaktu.

Diagram blok yang disederhanakan dari konverter tegangan ke frekuensi LM 331 seperti gambar 3.4.



Gambar 3.4. Diagram blok yang disederhanakan dari LM 331

Ketika masukan komparator mendeteksi tegangan V_1 pada pin7 lebih besar dari tegangan V_x pada pin 6, monostabil diset yang mengakibatkan aktifnya keluaran transistor dan saklar saklar sumber arus. Ketika tegangan pin 5 naik sampai $2/3 V_{cc}$ dalam waktu $1.1 R_t C_t$, monostabil akan diset tidak aktif.

Selama perioda tidak aktif $t_1 = 1.1 R_t C_t$, arus mengalir dari saklar sumber arus sebesar $i_s = 1.90V/R_t$, yang memberikan pengisian kapasitor C_t . Hal ini secara normal akan mengakibatkan tegangan V_x lebih besar dari tegangan V_1 . Pada akhir perioda pewaktu, monostabil akan direset, sehingga saklar sumber arus tidak aktif lagi.

Sekarang tidak ada arus mengalir dari pin 1, dan kapasitor C_t akan mengosongkan

tegangannya melalui R_L sampai tegangan V_x jatuh dibawah tegangan V_1 . Selanjutnya komparator akan kembali mengaktifkan monostabil dan siklus selanjutnya terjadi sama seperti diatas.

Dari keterangan di atas terlihat bahwa akan terjadi pengulangan pengisian dan pengosongan kapasitor C_L sebagai masukan komparator untuk mengaktifkan set atau reset monostabil. Keluaran monostabil ini akan mengakibatkan transistor berada pada keadaan saturasi atau cut-off dengan lama waktunya sesuai dengan waktu pengosongan dan pengisian kapasitor C_L . Keadaan saturasi atau cut-off dari transistor yang berulang-ulang ini akan menjadikan keluarannya berupa sederetan pulsa dengan frekuensi tertentu. Besar muatan yang mengisi kapasitor C_L saat arus mengalir pada pin 1 adalah ^[4]:

$$Q_{in} = i_s \times t_1 = \frac{1,90 \text{ V}}{R_s} \times (1,1 \times R_L C_L)$$

Besar muatan ini akan menyebabkan tegangan pada kapasitor C_L akan naik sebesar ΔV . Setelah waktu pengisian kapasitor C_L terlampaui, kapasitor akan mengosongkan kenaikan tegangan tersebut sampai besar tegangan V_x jatuh pada level tegangan masukan V_1 . Besar muatan yang dilepaskan saat pengosongan tegangan ΔV adalah :

$$Q_{out} = I_L \times t_2 = \frac{V_x}{R_L} \times t_2$$

Dapat ditunjukkan jika besar tegangan $\Delta V \ll V_1$, dan $R_1 C_1 \ll R_L C_L$, maka besar tegangan $V_x \approx V_1$ dan $t_1 \ll t_2$. Sehingga :

$$Q_{out} \approx \frac{V_1}{R_L} \times t_2$$

Frekwensi keluaran (f_{out}) besarnya bergantung dari t_2 yang berhubungan langsung dengan tegangan masukan V_1 dan dapat dicari berdasarkan $Q_{in} = Q_{out}$.

Jadi ^[4] :

$$\frac{1,90 \text{ V}}{R_s} \times (1,1 \times R_1 C_1) \approx \frac{V_1}{R_L} t_2, \quad \frac{1}{t_2} \approx \frac{V_1 R_s}{2,09 \text{ V } R_L} \frac{1}{R_1 C_1}$$

$$f_{out} \approx \frac{V_1 R_s}{2,09 \text{ V } R_L} \frac{1}{R_1 C_1} \dots\dots\dots (3.4)$$

Untuk : $R_s = 14,212 \text{ Kohm}$

$R_L = 100 \text{ Kohm}$

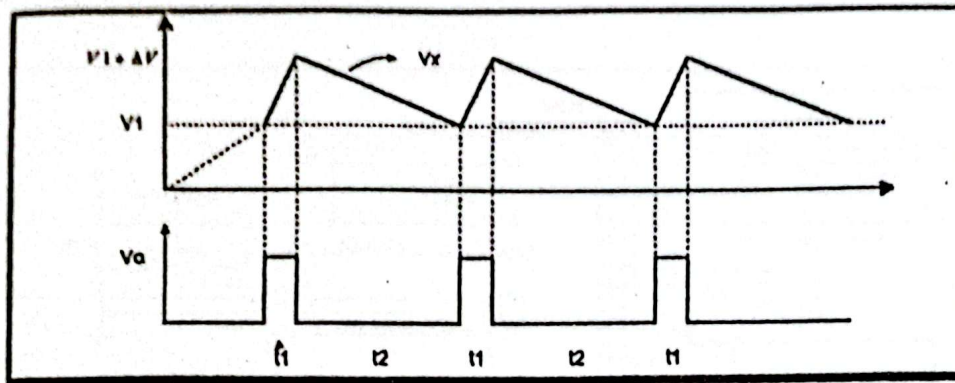
$R_T = 6,8 \text{ Kohm}$

$C_T = 10 \text{ nf}$

maka, ^[4] $f_{out} \approx V_1 \dots\dots\dots (3.5)$

Frekuensi keluaran f_{out} sebanding dengan tegangan masukan V_1 dan mempunyai besar skala maksimum 10 kHz untuk tegangan skala maksimum sebesar

10 Volt. Grafik f_{out} akibat pengisian dan pengosongan kapasitor C_L seperti gambar 3.5. [7]



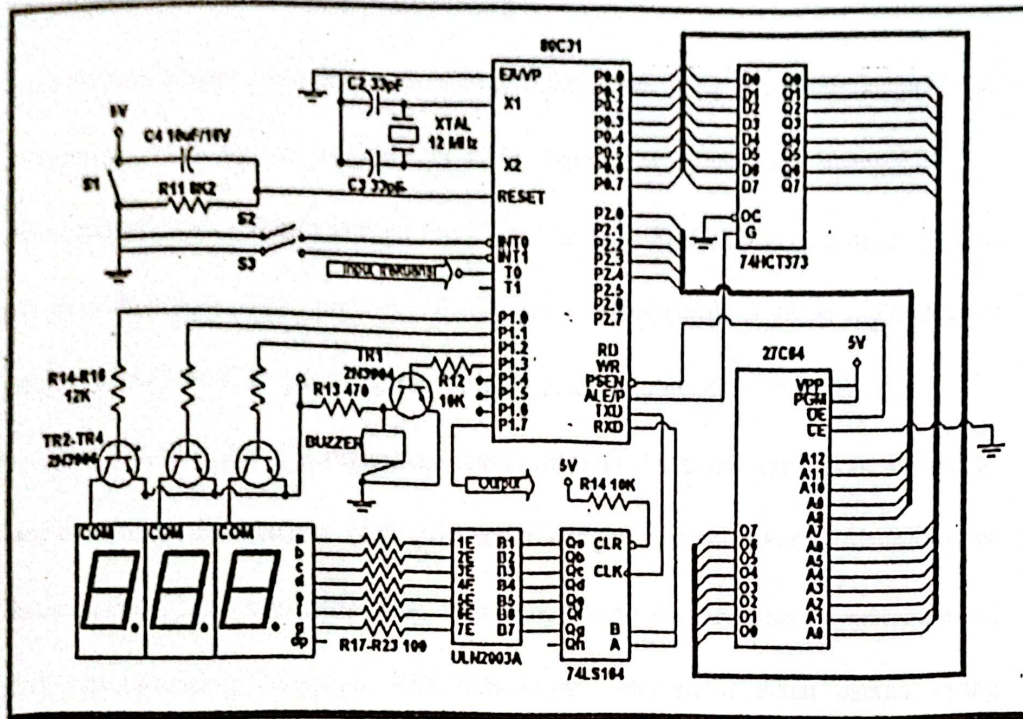
Grafik 3.5. Grafik keluaran akibat pengisian dan pengosongan kapasitor C_L

3.4. Sistem Mikrokontroler

Sistem mikrokontroler menerima masukan deretan pulsa frekuensi tertentu sesuai besar temperatur yang dipantau dan masukan papan kunci (*keypad*). Komponen utama bagian ini adalah mikrokontroler 80C31 sebagai unit pemroses masukan dan keluaran yang bekerja dengan frekuensi 12 MHz. Mikrokontroler dapat bekerja dengan baik di bantu oleh penahan (*latch*) alamat atau data menggunakan IC 74HCT373, dan tempat penyimpanan instruksi/program berupa EPROM 27C64.

Setelah data masukan di proses, hasil yang didapat dari sistem mikrokontroler ini berupa data biner untuk diberikan ke bagian rangkaian peraga 7-segmen sebagai pemberi informasi besar temperatur dalam bilangan desimal yang dapat dimengerti. Keluaran juga diberikan ke rangkaian saklar kendali daya AC untuk pengendali

temperatur.



Gambar 3.6. Sistem mikrokontroler dan layar peraga.^[3]

3.4.1. Mikrokontroler 80C31

Mikrokontroler 80C31 adalah salah satu anggota keluarga mikrokontroler MCS-51 yang dibuat dengan teknologi CMOS (*Complementary Metal Oxide Silicon*) dengan karakteristik berupa 32 jalur masukan/keluaran. RAM sebesar 128 bytes, 64 K

ruang memori program eksternal, 63 K ruang memori data eksternal. Dua buah pewaktu/pencacah 16 bit, 5 buah sumber interupsi masing-masing dengan 2 tingkat prioritas. Dilengkapi dengan full duplex port serial, on-chip osilator dan sirkuit pewaktu. Sebagai tambahan, secara perangkat lunak mikrokontroler 80C31 dapat berada pada keadaan Idle atau Power Down.

Masukan berupa deretan pulsa dengan frekuensi yang sebanding dengan besar temperatur dihubungkan ke port 3.4/TO. Perangkat lunak akan mengolah dan memberikan hasil olahan berupa data biner ke port 3.0/RXD secara serial dengan pewaktu diberikan oleh port 3.1/TXD, yang membentuk aplikasi register geser (*shift register*) serial ke paralel pada port serial mikrokontroler.

Port 1.0, 1.1, dan 1.2 digunakan untuk memilih *7-segmen* yang akan aktif pada saat data biner dikeluarkan oleh port serial, dengan memanfaatkan keadaan saturasi dan/atau *cut-off* dari transistor yang terhubung ke anoda bersama (*common anoda*) dari masing-masing *7-segmen*. Bila data besar temperatur diluar daerah yang diperbolehkan, port 1.4 akan melempar sinyal untuk mengaktifkan *buzzer* dengan membuat transistor pada keadaan *cut-off*.

Port 1.0 sampai port 1.6 dipergunakan juga untuk matrik papan kunci yang akan berukuran 4 baris dan 3 kolom. Port 1.0 - port 1.3 terhubung ke jalur baris papan kunci, sedangkan port 1.4 - port 1.6 terhubung ke jalur kolomnya.

Bila saklar S1 ditekan akan mengaktifkan port reset mikrokontroler sehingga instrumen berkerja pada mode pemantauan temperatur. Bila saklar S2 ditekan akan menghubungkan port 3.2/INT0 ke *ground* sehingga instrumen bekerja pada mode

kendali temperatur maksimum minimum. Bila saklar S3 ditekan akan menghubungkan port 3.3/INT1 ke *ground* sehingga instrumen bekerja pada mode kendali dengan pewaktu.

Mikrokontroler 80C31 tidak mempunyai EPROM internal, sehingga untuk menyimpan instruksi/program diperlukan EPROM eksternal. Untuk mengakses EPROM ini, mikrokontroler menggunakan port 0.0 - port 0.7 sebagai jalur data D0-D7 dan jalur alamat A0-A7 secara multipleks. Port 1.1 - port 1.7 dipakai sebagai jalur alamat tinggi A8 - A15. Dengan demikian jumlah memori yang dapat diakses adalah 64 Kbyte.

Untuk memberitahu mikrokontroler agar dapat mengakses EPROM eksternal dari lokasi 00000H, port EA (External Access) harus bernilai rendah. Bila tidak mikrokontroler akan mengakses program dari memori internal sampai Program Counter berisi alamat yang lebih besar dari 0FFFH. Isyarat pembaca instruksi program eksternal dikeluarkan oleh mikrokontroler melalui port PSEN (*Program Store Enable*).

Sinyal untuk membedakan latch alamat atau data diberikan oleh mikrokontroler melalui port ALE (*Address Latch Enable*) untuk menahan alamat yang menggunakan byte rendah selama akses ke memori program eksternal. Hal ini diperlukan karena adanya multipleks penggunaan jalur alamat dan data pada port 0.0 - port 0.7.

3.4.2 . IC Latch 74HCT373

IC latch 74HCT373 merupakan bentuk rangkaian logika keadaan berupa tipe D

oktal (*tri-state octal D-type*). IC ini mempunyai kemampuan ingatan masukan yang disimpan/ ditahan sampai isinya dihapus atau diisi ulang oleh bit informasi yang baru pada detak pewaktu tertentu.

IC 74HCT373 dirakit dalam 20-pin DIP (*Dual In-line Package*) dengan 8 jalur data masukan berlabel D0-D7, dan 8 jalur berlabel Q0-Q7. Sebagai tambahan dari dua pin untuk power, terdapat masukan berlabel G sebagai pewaktu detak dan masukan berlabel OC sebagai tanda diperoleh terjadinya hubungan dengan keluaran.

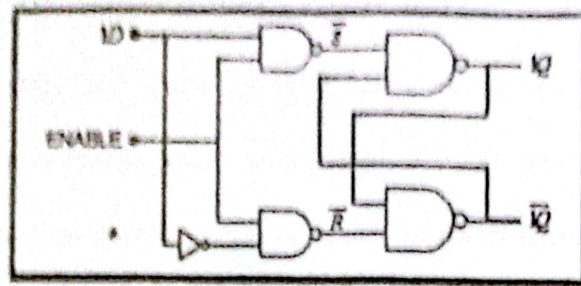
Pada saat masukan G bernilai tinggi, keluaran Q akan mengikuti besaran data pada masukan data sama seperti kasus flip-flop tipe D. Pada saat masukan G bernilai rendah, tidak ada perubahan. Hal ini menyatakan bahwa perubahan pada keluaran hanya bisa terjadi ada perubahan pada masukan data saat G bernilai tinggi.

Pin berlabel OC bila bernilai rendah akan mengizinkan keluaran berhubungan dengan rangkaian lain. Bila bernilai tinggi, semua keluaran akan berada pada keadaan impedansi tinggi tanpa tergantung oleh harga masukan.

Rangkaian logika untuk operasi data tunggal dan tabel kebenarannya seperti tabel dan gambar di bawah ini.

Masukan			Keluaran
OC	G	D	Q
1	X	X	Z
0	0	X	NC
0	1	0	0
0	1	1	1

Tabel 3.1. Tabel kebenaran operasi tunggal 74HCT373.^[3]



Gambar 3.7. Rangkaian logika operasi tunggal 74HCT373.^[5]

3.4.3. Erasable Programmable Read Only Memory (EPROM)

EPROM adalah suatu rangkaian terpadu yang berfungsi untuk menyimpan instruksi/program yang diperlukan oleh mikrokontroler, sehingga mikrokontroler dapat bekerja dengan baik. Program dapat disimpan di EPROM menggunakan EPROM programmer, dan dihapus dengan sinar ultra violet.

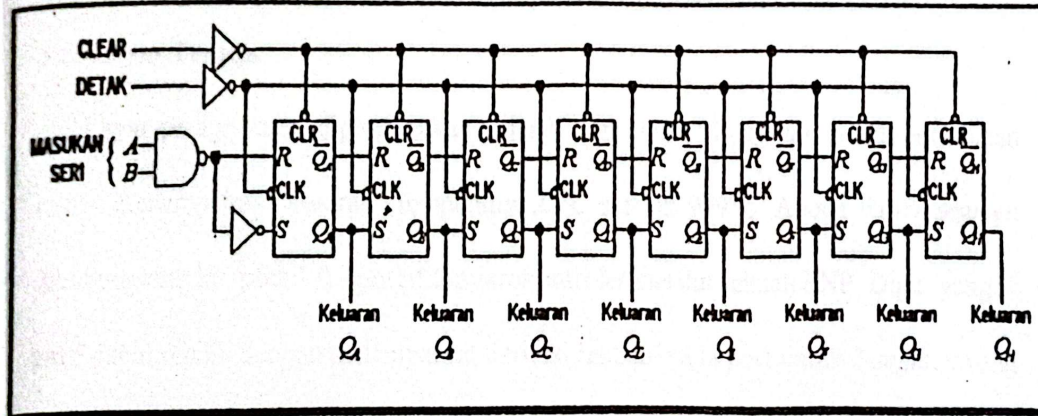
EPROM yang digunakan adalah 27C64 dengan kemampuan menyimpan program sebesar 8 Kbyte. Pemilihan tipe CHMOS hanya untuk kesesuaian penggunaannya dengan tipe mikrokontroler 80C31 dan IC latch 74HCT373.

3.4.4. IC Register Geser 74LS164

Register geser adalah suatu rangkaian yang menggunakan flip-flop yang saling dihubungkan secara seri, sehingga setiap bit yang disimpan di keluaran Q digeser ke flip-flop berikutnya. Pergeseraan bit ini terjadi pada setiap pulsa clock. Pulsa-pulsa clock tersebut dikirim kesemua flip-flop dalam register, sehingga operasinya berjalan secara sinkron.

IC register 74LS164 digunakan sebagai register masukan serial, keluaran paralel SIPO (Serial In Parallel Out). Pada tipe SIPO, data disajikan satu bit pada suatu saat lalu digeser masuk pada setiap pulsa clock. Sesudah perangkat pulsa clock lengkap, register menjadi penuh dan isinya dapat dibaca di terminal keluaran atau dikeluarkan melalui seperangkat saluran paralel. Sebagai tambahan untuk IC 74LS164, pin berlabel CLR harus ber kondisi tinggi agar dapat aktif pada setiap ada pulsa clock.

IC register geser 74LS164 menerima masukan data serial dari pin RXD mikrokontroler, sedangkan clock terhubung ke kaki TXD mikrokontroler, keluaran paralel IC register geser ini dihubungkan ke rangkaian layar peraga untuk ditampilkan.



Gambar 3.8. Rangkaian logika IC74LS164 untuk tipe SIPO^[5]

Tabel kebenaran IC 74LS164 dapat dilihat pada tabel 3.2 di bawah ini.

MASUKAN				KELUARAN							
t_n				t_{n+1}							
CLR	CLK	A	B	Q _A	Q _B	Q _C	Q _D	Q _E	Q _F	Q _G	Q _H
0	X	X	X	0	0	0	0	0	0	0	0
1	0	X	X	Tidak berubah							
1	1	1	1	1	Geser						
1	1	0	X	0	Geser						
1	1	X	0	0	Geser						

Tabel 3.2 Tabel kebenaran IC register geser 74LS164.^[3]

3.5. Layar Peraga

Layar peraga yang dipergunakan adalah tiga buah *7-segmen* anoda bersamaan untuk menampilkan besaran temperatur 25°C sampai 999°C. Anoda tiap *7-segmen* dihubungkan ke port 1.0 - port 1.2 mikrokontroler melalui sebuah PNP. Digit yang aktif dapat dipilih dengan melempar bit bernilai rendah pada port untuk *7-segmen* yang diinginkan.

Untuk menyalakan segmen tiap digit, dipergunakan transistor paralel dalam suatu kemasan IC tipe ULN2003A. Kaki basis IC ini dihubungkan ke keluaran IC register geser 74LS164, dan kaki emiternya terhubung ke tiap - tiap segmen dari *7-segmen*.

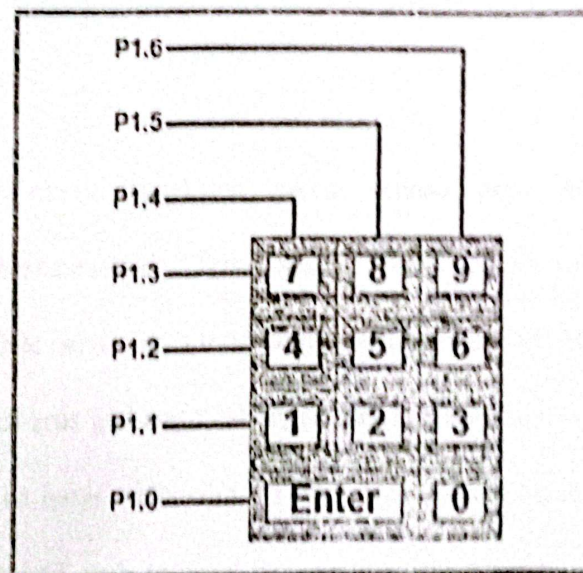
3.6. Papan kunci.

Konfigurasi saklar yang umum pada papan kunci (*keypad*) adalah bentuk matrik dari baris dan kolom. Pada instrumen ini papan kunci yang digunakan adalah jenis kontak mekanis dengan konfigurasi matrik 4 x 3 yaitu 4 baris dan 3 kolom. Tiap jalur dari baris dan kolom tersebut terhubung ke port 1.0 sampai port 1.6 mikrokontroler. Konfigurasi papan kunci seperti gambar 3.10

Kolom papan kunci berperan sebagai pemberi logika 0 secara bergantian terus menerus. Pada suatu saat hanya satu kolom berlogika 0, dua kolom lainnya berlogika 1. Baris papan kunci saat awal diberi logika 1.

Suatu baris papan kunci akan menerima nilai kembali logika 0 bila ada penekanan tombol papan kunci saat kolom yang berhubungan berada pada logika 0.

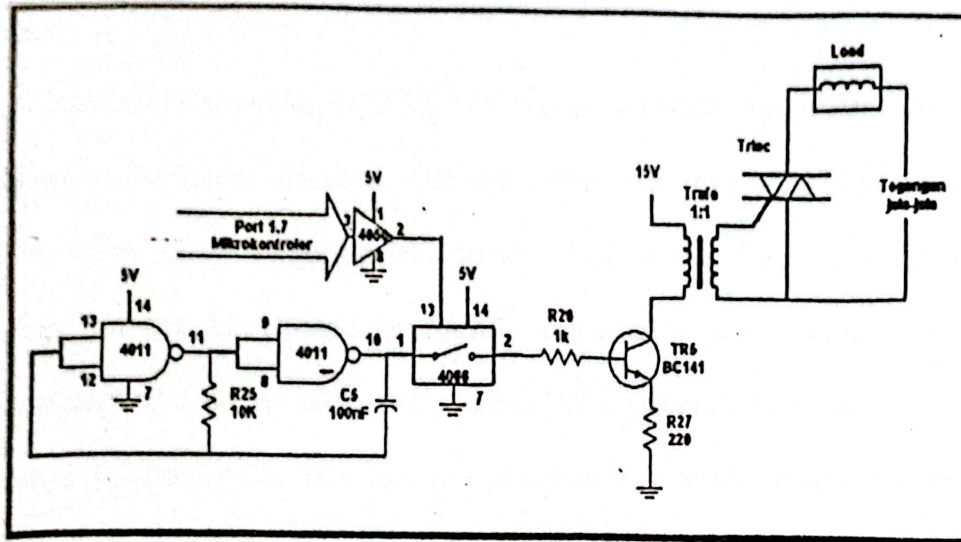
Perangkat lunak akan memberikan solusi tombol apa yang telah ditekan.



Gambar 3.9. Rangkaian papan kunci

3.7. Saklar Kendali Daya AC

Rangkaian saklar kendali daya AC menerima masukan dari port 1.7 mikrokontroler. Logika 1 dari port membuat rangkaian mensuplai daya AC ke beban (*heater*). Logika 0 dari port membuat rangkaian memutuskan suplai daya AC ke beban.



Gambar 3.10 Rangkaian saklar kendali daya AC

3.7.1. Triak

Komponen utama rangkaian saklar kendali daya AC adalah piranti semikonduktor bernama triak. Triak adalah semikonduktor yang akan berkonduksi dalam dua arah bila suatu arus tertentu melewati gerbangnya. Setelah berada dalam keadaan konduksi arus gerbang tidak diperlukan lagi sampai arus yang melewati arah konduksi turun ke batas minimum. Triak dapat dipacu kembali oleh pulsa positif maupun pulsa negatif pada terminal gerbang tunggalnya.

Triak memberikan solusi yang baik untuk pengendalian daya AC. Triak memiliki beberapa keunggulan dibandingkan dengan saklar mekanik konvensional atau relay konvensional. Triak terhindar dari bouncing ataupun kelelahan bahan seperti halnya saklar mekanik. Dibandingkan dengan relay konvensional triak memberikan aksi pensaklaran yang jauh lebih cepat, sehingga memberikan pengendalian yang lebih akurat.

Triak dapat dipacu oleh sinyal DC, AC atau pulsa dengan daya rendah. Sirkuit pemacu dapat berupa transistor, UJT (*unijunction transistor*) saklar bilateral atau dioda 4 layer. Untuk mengisolasi sirkuit pemacu daya rendah dengan sirkuit suplai beban tegangan AC dipergunakan isolasi dengan transformator arus. Untuk pemacuan dengan menggunakan transistor atau UJT diperlukan pulsa dengan frekuensi tertentu. Frekuensi pulsa diatur agar saat pemacuan triak berada sedekat mungkin dengan titik nol dari sinusoidal suplai beban tegangan AC. Zero point switching ini diperlukan untuk menghindari terjadinya noise yang tinggi (*electromagnetic interference*).

3.7.2. IC 4011

IC 4011 adalah rangkain terintegrasi dari 4 buah gerbang NAND dalam satu kemasan. Masing-masing gerbang NAND memiliki 2 input. Untuk membuat generator pulsa hanya dibutuhkan 2 gerbang NAND. Pulsa dengan frekuensi 1 KHz didapat dengan menghubungkan tahanan 10 Kohm, kapasitor 100 nF dan 2 buah gerbang NAND seperti gambar 2.11. Secara umum pemacuan triak dengan frekuensi 1 KHz sudah mendekati pemacuan zero point switching.

3.7.3 . IC 4066

IC 4066 adalah rangkaian terintegrasi 4 buah saklar bilateral dalam satu kemasan. Masing-masing saklar ini memiliki jalur kendali yang akan menghubungkan jalur masukan dengan jalur keluaran. IC ini dirancang untuk dapat mentransmisikan sinyal pulsa sampai 10 MHz.

IC ini dibutuhkan untuk mengendalikan pulsa yang akan memacu triak berdasarkan hasil keluaran port 1.7 mikrokontroler yang terhubung ke jalur kendali. Logika 1 dari port 1.7 mikrokontroler akan menyebabkan pulsa pada masukan IC ini diteruskan ke keluarannya. Pulsa ini akan memacu triak setelah melewati isolasi transformator arus yang bekerja setelah transistor berada pada keadaan saturasi.

3.7.4 IC 4050

IC 4050 adalah rangkaian terintegrasi 6 buah penyangga tidak membalik (noninverting buffer) dalam satu kemasan. IC ini umumnya digunakan sebagai inverter COS/MOS ke DTL/TTL dan dapat mengemudikan dua buah beban DTL/TTL. Pada rangkaian saklar kendali daya AC, IC ini digunakan untuk penyangga level tegangan port 1.7 mikrokontroler 80C31 agar sesuai dengan level tegangan saklar bilateral dari IC 4066.

BAB IV

PERANCANGAN PERANGKAT LUNAK

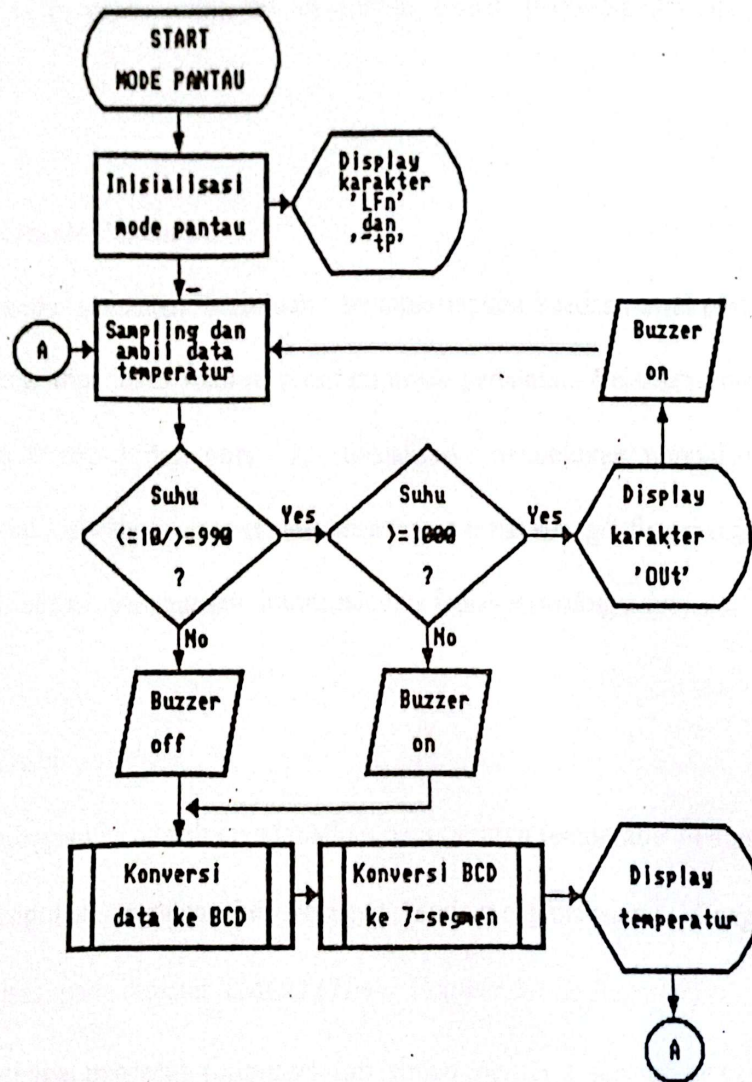
Perangkat lunak (*software*) dalam bentuk program diperlukan untuk membantu perangkat keras (*hardware*) agar dapat berfungsi dengan benar. Perangkat lunak mengambil informasi dari perangkat keras, menganalisa, mengolah, memanipulasikan memberikan informasi hasilnya ke perangkat keras untuk tugas tertentu. Program dibangun menggunakan bahasa assembly khusus kode mesin untuk keluarga mikrokontroler *MCS-51* dan disimpan di *EPROM* setelah di assemble dengan menggunakan macro assembler V4.01d.

Secara garis besar program dibagi menjadi tiga bagian. Pertama, program untuk mode pemantauan. Kedua, program untuk mode pengendalian maksimum minimum. Ketiga, program untuk pengendalian dengan pewaktu. Program mode pengendalian

maksimum minimum dan mode pengendalian dengan pewaktu menggunakan sebagai besar program mode pemantau ditambah program input papan kunci dan aksi pengendalian yang sesuai.

4.1. MODE PEMANTAU

Diagram alir (flowchart) program mode pemantau seperti pada gambar 4.1



Gambar 4.1. Diagram aliran program mode pemantauan.

Program mode pemantauan bertujuan mendapatkan data temperatur format heksadesimal yang sesuai dengan frekuensi yang diberikan rangkaian konverter tegangan ke frekuensi. Menganalisa data, mengkonversi data ke format BCD (*Binary Coded Decimal*), mengkonversi BCD ke 7-segmen. Akhirnya akan menampilkan data tersebut pada 7-segmen. Analisa data memberikan peringatan buzzer bila temperatur di bawah atau diatas 999°C. Program mode ini diaktifkan akibat penekanan tombol power on atau tombol reset.

4.1.1. Inisialisasi Mode Pemantau.

Inisialisasi mode pemantau bertujuan mempersiapkan keadaan awal port, register dan variabel yang diperlukan seluruh program mode pemantau. Pelaksanaannya berupa inisialisasi logika 0 dan 1 dari port 1, inisialisasi penunjukan tumpukan (*Stack Pointer/SP*), inisialisasi timer/counter, inisialisasi port serial sebagai fungsi register geser (*shifregister*), inisialisasi penggunaan interupsi dan jenisnya (*falling edge*).

4.1.2. Sampling Data

Fungsi utama bagian ini adalah mendapatkan data besaran temperatur berupa frekuensi dari rangkaian pengubah tegangan ke frekuensi dan menganalisanya. Langkah awal melakukan inisialisasi pada register TMOD (*Timer Counter Mode Register*) timer/counter 0 sebagai pencacah (*counter*) dan timer/counter 1 sebagai pewaktu (*timer*) yang menentukan lama waktu cacah.

Data temperatur yang telah diubah ke frekuensi dan merupakan masukan sistem mikrokontroler adalah 0-10KHz dengan perbedaan temperatur 0-1000. Nilai register pencacah mikrokontroler akan bertambah satu dengan terdeteksinya falling edge yang terjadi pada pin T0 dari port 3. Dengan demikian penambahan satu itu sama dengan besar satu perioda dari sinyal masukan. Dengan perbandingan ini, diperlukan waktu cacah tertentu sehingga register pencacah dapat mempunyai nilai yang sesuai dengan besar frekuensi masukan.

Berdasarkan hal diatas, waktu cacah yang diperlukan adalah:^[6]

$$T = \frac{\text{besar cacah}}{(F_{\text{maks}} - F_{\text{min}})} = \frac{1000}{10 \text{ KHz}} = 100\text{ms} \quad \dots\dots\dots(4.1)$$

Besar waktu cacah ini di inisialisasi ke register pewaktu, sehingga saat waktu cacah tercapai mikrokontroler berhenti mencacah dan menyimpan hasil cacah ke RAM data internal. Caranya dengan menggunakan interupsi apabila waktu berulang (*roll-over*) register pewaktu terjadi. Roll-over terjadi bila isi register pewaktu bergulir dari nilai maksimum FFFFH (65535D) ke 0000H dengan detak penambahan satu sesuai dengan siklus mesin/detik, yaitu 1µs untuk pemakaian 12 MHz osilator.

Berdasarkan penjelasan diatas, register akan roll-over setiap 65,536 ms apabila diisi nilai awal 0000H. Untuk menutupi kekurangan waktu cacah, setelah interupsi akibat roll-over register pewaktu diisi lagi dengan nilai awal yang baru agar terjadi roll-over cedula untuk waktu cacah 100 ms.

Kekurangan waktu cacah pertama adalah :

$$100 \text{ ms} - 65,536 \text{ ms} = 34,464 \text{ ms} = 34464 \mu\text{s}$$

jadi nilai awal untuk roll-over kedua adalah :

$$65536 \mu\text{s} - 34464 \mu\text{s} = 31072 \mu\text{s} = 7961\text{H}$$

Setelah waktu pencacah di lalui dan cacah dihentikan, data disimpan di RAM memori internal dengan alamat tertentu untuk dianalisa dan di tampilkan pada layar peraga 7-segmen. Bila temperatur terlalu dingin atau terlalu panas ($10\ 990^{\circ}\text{C}$), buzzer akan memberikan peringatan. Bila temperatur lebih besar dari 999°C , diluar kemampuan 3 buah 7-segmen, selain peringatan buzzer karakter 'OUT' juga ditampilkan.

4.1.3. Subrutin Konversi Data Ke BCD.

Setelah dianalisa, data temperatur diubah ke dalam bentukan desimal yang dikodekan biner (*Binary Code Decimal/BCD*). Besaran data dengan format heksadesimal/biner (0-f) diubah ke dalam bentuk desimal dengan format (0-9) dalam besaran berupa satuan, puluhan dan ratusan. Hasil konversi disimpan dalam tiga buah alamat RAM internal dengan masing - masing alamat berisi sebuah bilangan desimal.

4.1.4. Subrutin Tampilan Konverssi BCD Ke 7-Segmen.

Setelah data dalam bentuk BCD didapat, program kemudian memanggil subrutin konversi BCD ke kode 7-segmen. Hasil konversi tahap ini berupa tiga byte kode 7-segmen yang masing - masing akan ditampilkan oleh layar peraga. Nilai - nilai hasil konversi BCD ke 7-segmen disesuaikan dengan penyalaaan segmen yang dikehendaki dari

7-segmen. Jadi jika kita ingin menampilkan bilangan '8' pada *7-segmen*, maka konversi BCD ke *7-segmen* harus bernilai '11111100B' dan bila huruf 'L' ingin kita tampilkan, maka nilainya adalah '00011100B'.

4.1.5. Subrutin Tampilan Layar Peraga.

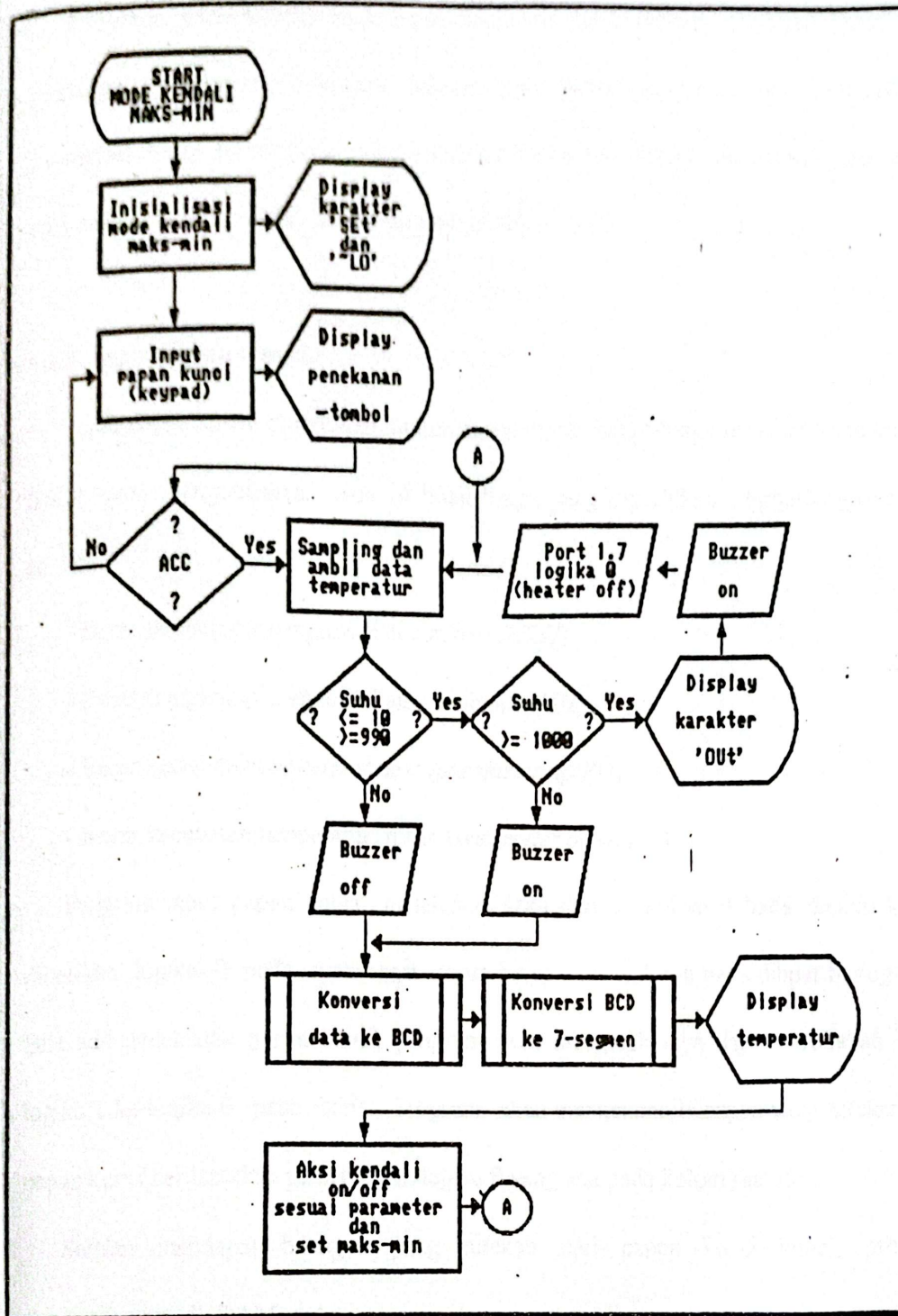
Pada subrutin inilah program melempar keluar melalui port tranmisi serial tiga data hasil konversi BCD ke *7-segmen*. Karena *7-segmen* mempunyai data yang ingin ditampilkan dengan nilai yang berbeda, maka diperlukan koordinasi antara data yang ingin ditampilkan dengan salah satu dari ketiga *7-segmen* yang di aktifkan. Tiap *7-segmen* untuk data tertentu diaktifkan selama 5 ms. Setelah itu data selanjutnya dilempar oleh port serial terakhir yang akan ditampilkan pada *7-segmen* yang lainya begitu juga halnya untuk data terakhir yang akan ditampilkan. Dengan cara ini kita melihat tiga *7-segmen* memberikan hasil pemantauan temperatur secara bersamaan, walau sebenarnya tiap *7-segmen* menyala bergantian pada saat data diberikan.

Agar tampilan dapat diamati dengan baik, maka pengubah data tampilan dari ketiga *7-segmen* dan pembacaan data dari hasil pemantauan yang baru hanya dilakukan setiap 1 detik. Setelah 1 detik, program kembali ke bagian program sampling untuk membaca data baru.

4.2. PROGRAM MODE KENDALI MAKSIMUM MINIMUM

Diagram alir program kendali maksimum minimum secara garis besar seperti gambar

4.2



Gambar 4.2. Diagram alir program mode kendali maksimum minimum

Program mode kendali maksimum minimum menggunakan sebagian besar mode pemantauan. Program ditambah bagian input papan kunci dan aksi pengendalian. Inisialisasi mode ini membuat mikrokontroler hanya melakukan pembacaan instruksi dari program mode kendali maksimum minimum.

4.2.1. Input Papan Kunci.

Input papan kunci diperlukan untuk menyimpan harga-harga masukan yang berguna untuk aksi pengendalian. Ada 4 buah harga yang diperlukan. Harga-harga tersebut adalah :

- Besar temperatur pengendalian minimum ($^{\circ}LO$)
- Besar temperatur pengendalian maksimum ($^{\circ}HI$)
- Besar kecepatan temperatur naik (*parameter up/PU*)
- Besar kecepatan temperatur turun (*parameter down/PD*)

Program input papan kunci melakukan scan dari 3 kolom 4 baris. Setiap kolom diberikan logika 0 pada suatu saat secara bergantian. Seluruh baris dibuat berlogika 1. Bila ada penekanan papan kunci, program akan mengetahuinya dari perubahan harga logika 1 ke logika 0 pada baris. Program akan mengetahui bilangan yang ditekan dari papan kunci berdasarkan pemeriksaan logika 0 yang ada pada kolom saat itu.

Setelah mendapat bilangan yang ditekan dari papan kunci, program menyimpannya di RAM data internal dan menampilkannya pada 7-segmen. Setelah seluruh masukan yang berguna untuk pengendalian terpenuhi, program akan

menampilkan karakter 'ACC' pada 7-segmen. Program selesai melakukan input papan kunci.

Setelah selesai pada bagian input papan kunci, program melaksanakan seluruh mode pemantauan dari sampling sampai tampilan layar peraga untuk harga temperatur yang terdeteksi. Program lalu masuk sebagai pengendalian.

4.2.2. Aksi Kendali On/Off.

Program aksi kendali on/off memberikan logika 1 atau logika 0 pada port 1.7 mikrokontroler. Logika 1 menghubungkan beban (*heater*) ke tegangan jala-jala. Logika 0 memutuskannya. Hal ini membuat heater dalam keadaan on atau off. Dengan cara demikian besar panas yang dihasilkan heater dapat dikendalikan.

Tanpa memberikan nilai parameter up dan/atau parameter down, heater akan on terus menerus sampai harga temperatur maksimum ($^{\circ}HI$) yang diinginkan tercapai. Kemudian heater akan off terus menerus sampai harga temperatur minimum ($^{\circ}LO$) yang diinginkan tercapai.

Pada kebanyakan heater, keadaan diatas akan memberikan lonjakan harga yang sangat tinggi diatas besaran yang diinginkan (*over shoot*). Untuk menghindari hal ini digunakan suatu variabel tertentu yang menahan laju kecepatan naik temperatur dari $^{\circ}C$ diatas minimum sampai maksimum ($^{\circ}LO$). Kedua harga tersebut adalah *parameter up* (Pu°) dan *parameter down* (Pd°).

Parameter *up* dengan satuan detik akan membuat heater on setelah sepuluh detik *off* saat temperatur kurang 10°C dari maksimum (*%H*). Setelah waktu parameter *up* terlampaui, heater kembali *off* selama sepuluh detik.

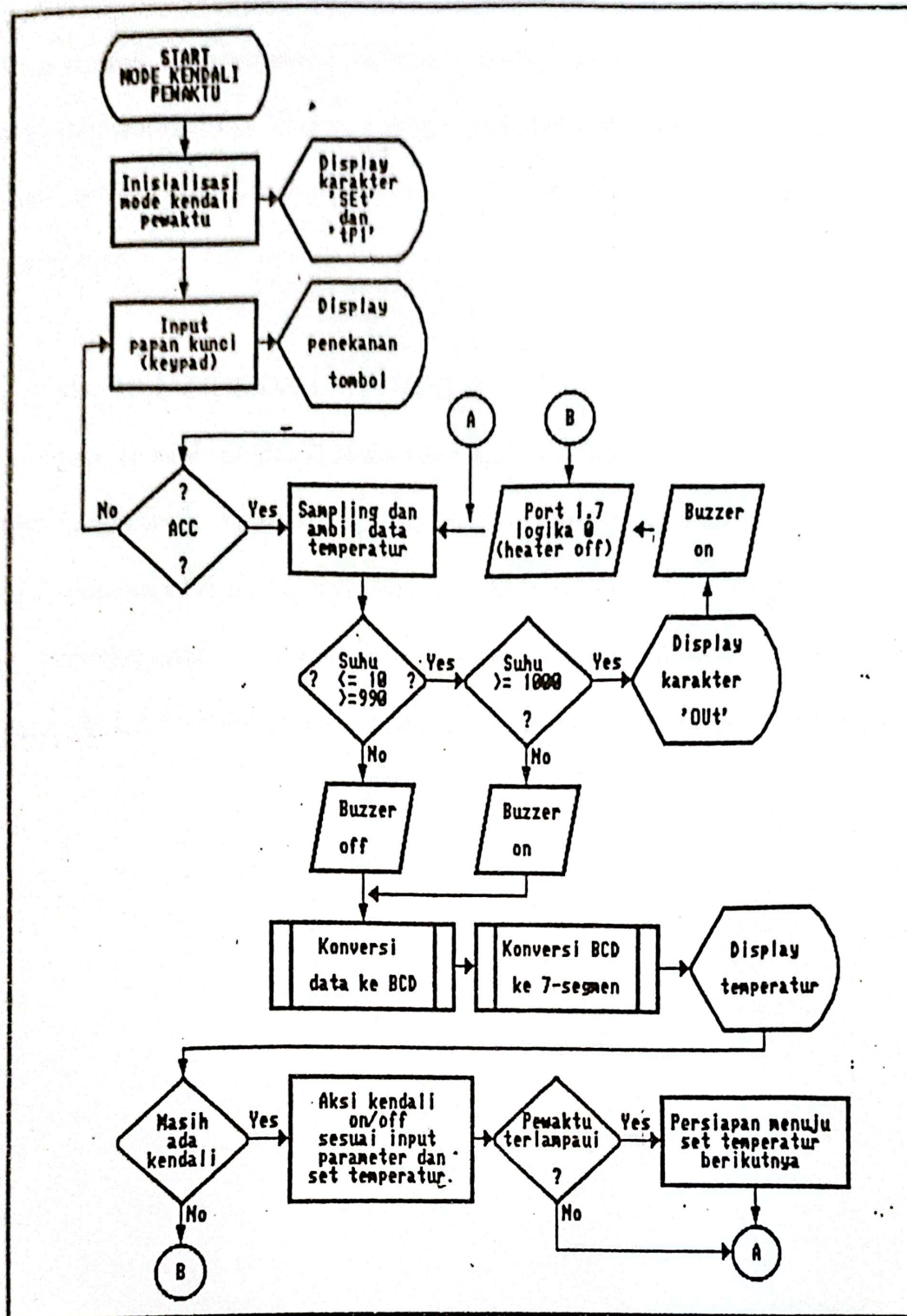
Parameter *down* dengan satuan detik akan membuat heater on setelah sepuluh detik *off* saat temperatur lebih 10 °C dari minimum (*%L*). Setelah waktu parameter *down* terlampaui, heater kembali *off* selama sepuluh detik.

4.3. PROGRAM MODE KENDALI PEWAKTU

Program mode kendali dengan pewaktu mempunyai aliran program yang hampir sama dengan program mode kendali maksimum minimum. Perbedaan sedikit terletak pada inisialisasi untuk penggunaan mode kendali pewaktu, variabel masukan dari papan kunci dan aksi pengendalian.

Pada mode ini dapat di simpan 4 besaran temperatur dengan waktunya masing-masing. Keempat besaran temperatur itu berlabel tp1 sampai tp4. Pewaktu berlabel ts1 sampai ts4 dengan satuan menit. Program akan menahan harga temperatur, misalnya tp1, selama waktu ts1. Kemudian program akan mencari harga temperatur selanjutnya, misalnya tp2. Program akan menahan harga temperatur tersebut selama waktu ts2. Demikian seterusnya.

Penggunaan secara maksimum dapat menyimpan 4 besaran temperatur dan pewaktunya. Bila ingin mengurangi penggunaannya dapat dilakukan dengan memberikan nilai '000' pada harga temperatur yang akan dimasukkan dari papan kunci. program



Gambar 4.3. Diagram alir program mode kendali pewaktu.

akan mencari harga temperatur berikutnya untuk ditahan sesuai dengan pewaktunya. Bila seluruh harga-harga temperatur dengan pewaktunya selesai dikendalikan, program akan membuat logika 0 pada port 1.7 (*heater off*). Kemudian program masuk ke mode pemantauan.

4.4. MASTER TERMINAL UNIT (MTU)

Master terminal unit (*MTU*) pengendalian jarak jauh dan merupakan pengolah data lebih lanjut yang menggunakan komputer pribadi IBM PC kompatibel. Komunikasi data serial dengan Remote Terminal Unit (*RTU*) dilakukan melalui gerbang *RS-232*.

Perangkat lunak yang digunakan pada Master Terminal Unit (*MTU*) adalah Visual Basic Ver. 3.0. Pemilihan perangkat lunak ini didasari atas bentuk tampilan yang bagus.

BAB V

UJI COBA SISTEM

Setelah seluruh perangkat keras dan perangkat lunak selesai dibuat, dilakukan uji coba. Pengujian berupa pengukuran dan kalibrasi dari rangkaian termokopel termometer terkalibrasi dan rangkaian konverter tegangan ke frekuensi.

Perbandingan dilakukan dengan menggunakan kendali temperatur buatan OMRON.

5.1 PENGUKURAN TERMOKOPEL TERMOMETER TERKALIBRASI

Rangkaian termokopel termometer terkalibrasi menerima masukan dari termokopel. Tegangan termokopel sesuai dengan temperatur yang mengenainya dan memiliki perbandingan sesuai dengan koefisien Seebeck sebesar $40,8 \text{ mV/}^{\circ}\text{C}$. IC M335 memberikan tambahan tegangan sesuai dengan temperatur ruang. Tegangan keluaran yang diinginkan dari rangkaian termokopel termometer terkalibrasi adalah $0 \text{ mV/}^{\circ}\text{C}$.

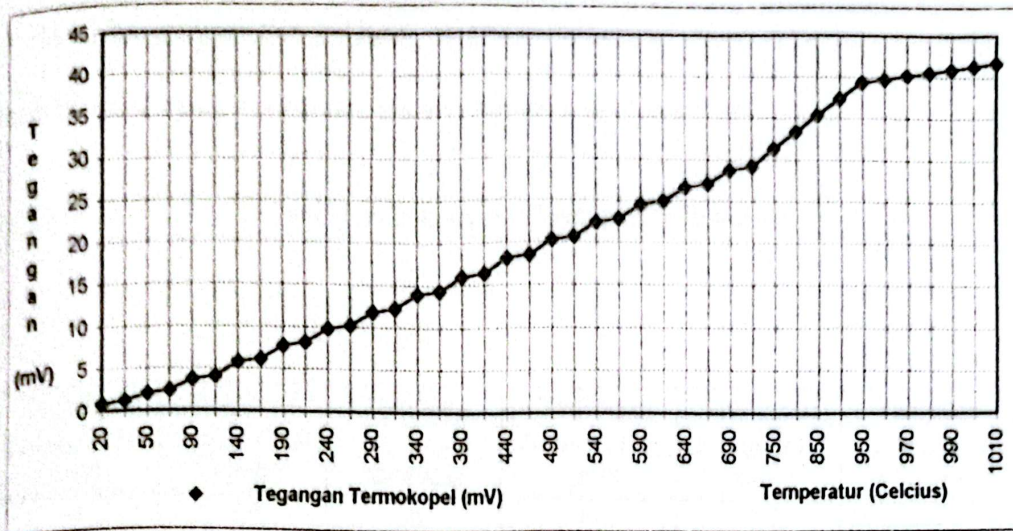
Hasil pengukuran dan kalibrasi pada temperatur ruang akan memperlihatkan hubungan yang linier dari temperatur, tegangan termokopel dan tegangan keluaran rangkaian termokopel termometer terkalibrasi.

Hasil pengukuran dan kalibrasi termokopel termometer terkalibrasi (TTK) pada temperatur ruang (25°C) seperti pada tabel 5.1.

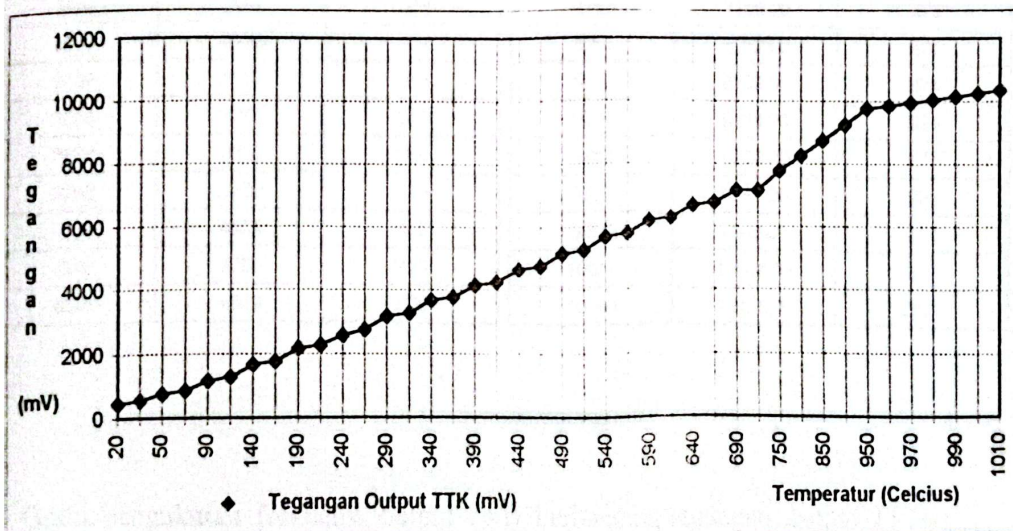
Tabel 5.1 Pengukuran termokopel termometer terkalibrasi

Temperatur Termokopel (°C)	Tegangan Termokopel (mV)	Tegangan Output (TTK) (mV)	Temperatur Termokopel (°C)	Tegangan Termokopel (mV)	Tegangan Output (TTK) (mV)
10	0.40	347	490	20.22	5160
20	0.80	449	500	20.65	5258
30	1.20	546	540	22.35	5661
50	2.02	752	550	22.78	5759
60	2.43	855	590	24.49	6161
90	3.68	1157	600	24.91	6257
100	4.10	1259	640	26.61	6662
140	5.73	1652	650	27.03	6759
150	6.13	1748	690	28.72	7153
190	7.73	2157	700	29.14	7144
200	8.13	2254	750	31.23	7763
240	9.75	2547	800	33.30	8261
250	10.16	2744	850	35.34	8747
290	11.80	3159	900	37.36	9241
300	12.21	3258	950	39.35	9757
340	13.88	3663	960	39.75	9855
350	14.29	3760	970	40.14	9950
390	15.98	4151	980	40.53	10047
400	16.40	4248	990	40.92	10144
440	18.09	4645	1000	41.31	10251
450	18.51	4749	1010	41.70	10356

Grafik pengukuran termokopel berbanding temperatur dan output (TTK) berbanding temperatur seperti gambar 5.1a dan 5.1b.



Gambar 5.1a Grafik pengukuran tegangan termokopel berbanding temperatur.



Gambar 5.1b Grafik pengukuran tegangan output (TTK) berbanding temperatur.

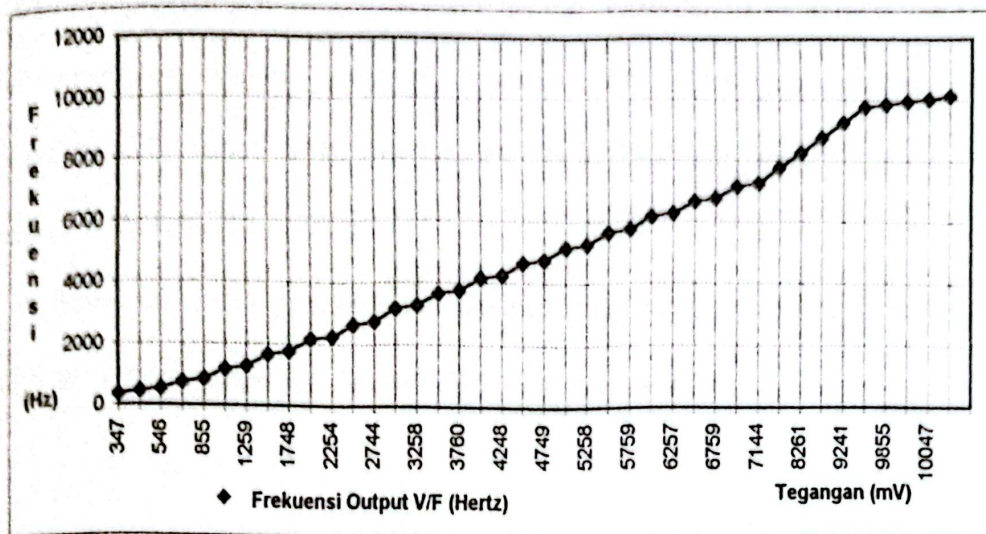
5.2 PENGUKURAN KONVERTER TEGANGAN KE FREKUENSI

Hasil pengukuran dan kalibrasi rangkaian konverter tegangan ke frekuensi (v/f) serta tampilan harga temperatur pada 7-segmen seperti tabel 5.2.

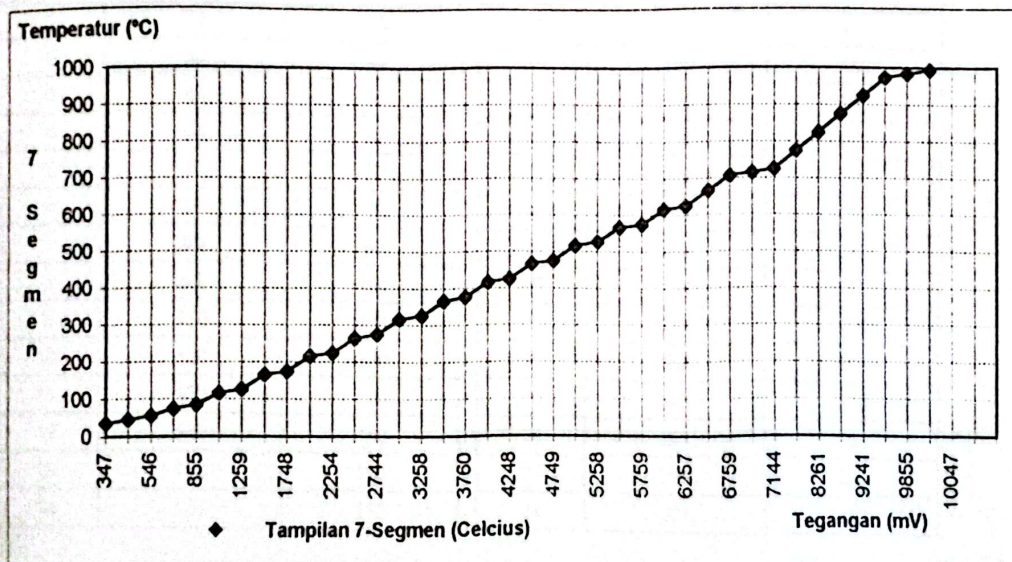
Tabel 5.2. Pengukuran tegangan ke frekuensi

Tegangan Output (TTK) (mV)	Frekuensi Output (TTK) (Hertz)	Tampilan 7-Segmen (Celcius)	Tegangan Output (TTK) (mV)	Frekuensi Output (TTK) (Hertz)	Tampilan 7-Segmen (Celcius)
347	350	35	4749	4747	475
449	451	45	5160	5158.00	516
546	549	55	5258	5255.00	526
752	755	75	5661	5654.00	565
855	853	85	5759	5752.00	575
1157	1160	116	6161	6153.00	615
1259	1262	126	6257	6250.00	625
1652	1650	165	6662	6653.00	665
1748	1744	174	6759	6752.00	705
2157	2155	216	7153	7144.00	714
2254	2252	225	7144	7237.00	724
2647	2641	264	7763	7755.00	775
2744	2739	274	8261	8252.00	825
3159	3163	316	8747	8739.00	874
3258	3263	326	9241	9236.00	924
3663	3662	366	9757	9750.00	975
3760	3759	376	9855	9847.00	985
4151	4152	415	9950	9943.00	994
4248	4250	425	10047	10036.00	OUT
4645	4649	465	10144	10136.00	OUT

Grafik pengukuran frekuensi output (v/f) berbanding tegangan output (TTK) dan tampilan 7-segmen berbanding tegangan output (TTK) seperti gambar 5.2a dan 5.2b.



Gambar 5.2a. Grafik pengukuran frekuensi output (v/f) berbanding tegangan output (TTK).



Gambar 5.2b. Grafik tampilan 7-segmen berbanding tegangan output (TTK).

5.3. PERBANDINGAN HASIL PEMANTAUAN

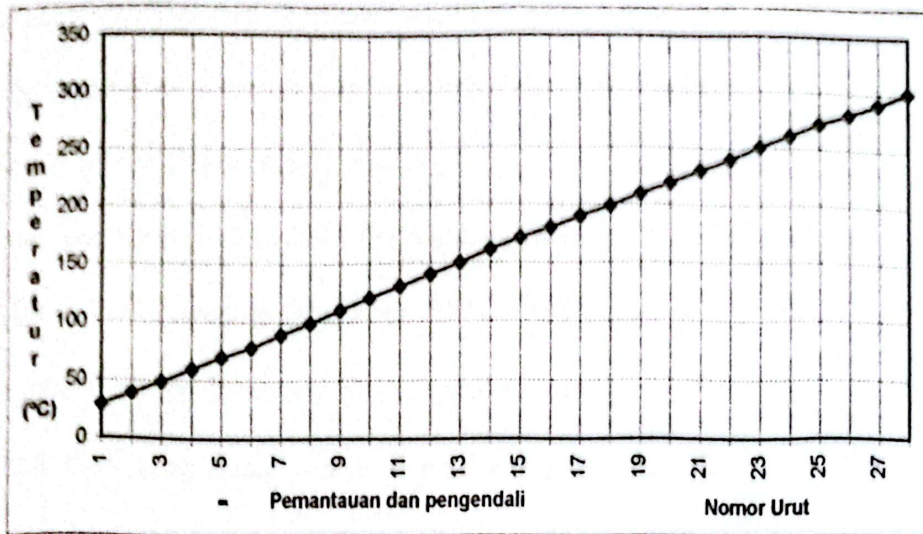
Perbandingan hasil pemantauan alat pemantau dan pengendali temperatur dengan hasil pemantauan kendali temperatur buatan OMRON dilakukan dari besar temperatur 25°C sampai 999°C. Pembacaan dan perbandingan besar temperatur dilakukan setiap kenaikan 10°C.

Hasil pembacaan dan perbandingan alat pemantau dan pengendali temperatur dengan alat kendali temperatur buatan OMRON seperti tabel 5.3.

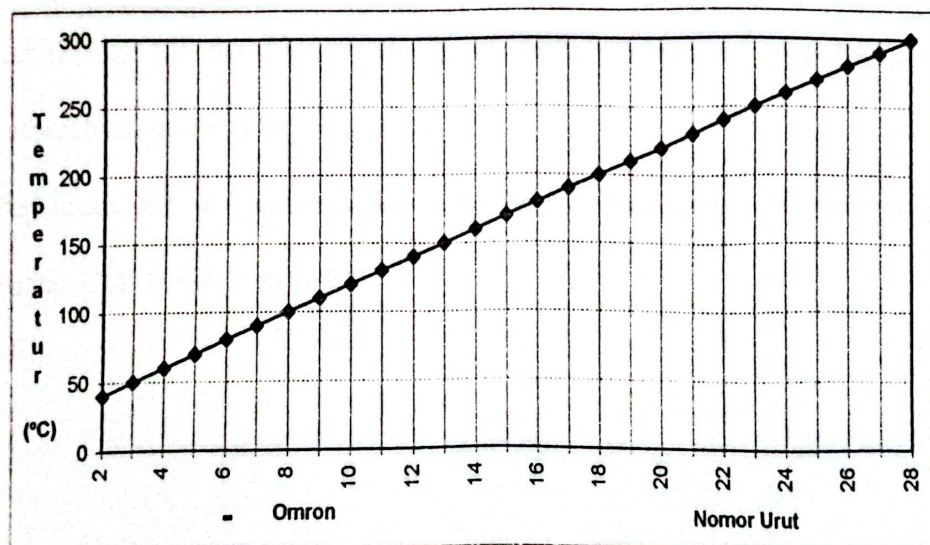
Tabel 5.3. Hasil perbandingan pemantauan temperatur.

No.	Pemantau dan Pengendali (Celcius)	OMRON (Celcius)	No.	Pemantau dan Pengendali (Celcius)	OMRON (Celcius)
1	29	30	15	171	170
2	39	40	16	180	180
3	49	50	17	190	190
4	59	60	18	198	200
5	70	70	19	209	210
6	78	80	20	219	220
7	89	90	21	230	230
8	99	100	22	240	240
9	110	110	23	251	250
10	120	120	24	261	260
11	130	130	25	272	270
12	140	140	26	281	280
13	150	150	27	291	290
14	161	160	28	301	300

Grafik perbandingan besar temperatur dari alat pemantau dan pengendali temperatur dengan alat kendali temperatur buatan OMRON seperti gambar 5.3.



Gambar 5.3.a. Grafik perbandingan besar temperatur dari alat pemantau dan pengendali.



Gambar 5.3.b. Grafik perbandingan temperatur dari alat kendali temperatur buatan OMRON.

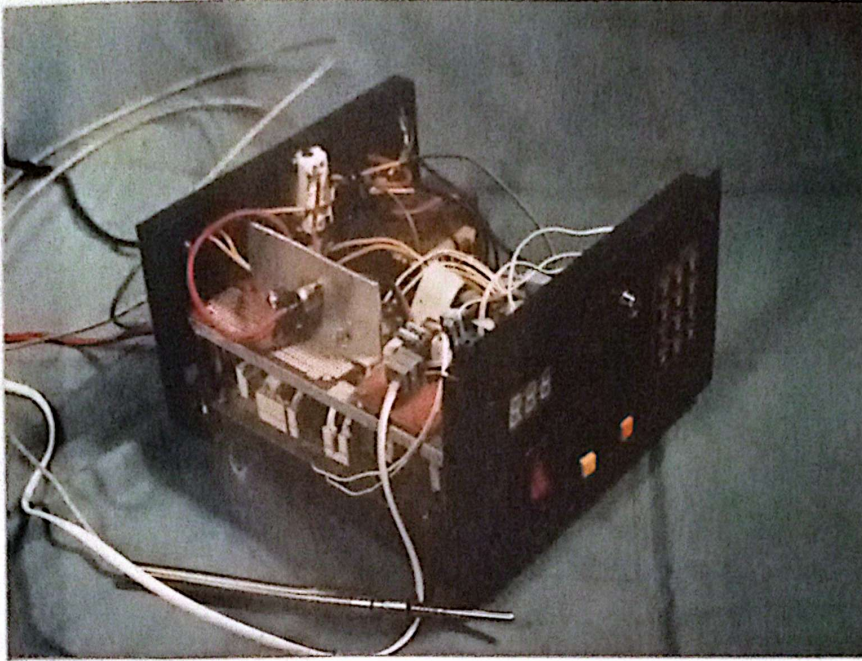
KESIMPULAN

1. Waktu sampling yang diberikan mikrokontroler sebesar 100 ms menghasilkan pembacaan data frekuensi yang benar untuk ditampilkan pada layar peraga 3 buah seven segmen sesuai harga temperatur.
2. Dari grafik gambar 5.1a dan 5.1b pengukuran tegangan terhadap temperatur linier $25^{\circ}\text{C} - 720^{\circ}\text{C}$, demikian pula linier $720^{\circ}\text{C} - 950^{\circ}\text{C}$ dan $950^{\circ}\text{C} - 1010^{\circ}\text{C}$. Ini disebabkan respon komponen yang digunakan kurang linier^[10].
3. Dari hasil pengukuran dan alat pembanding peneraan rangkaian konverter tegangan ke frekuensi (vf), grafik pengukuran frekuensi keluaran (v/f), berbanding lurus tegangan keluaran (TTK). terlihat dari tegangan^[1] 347 mV - 7144 mV terjadi kenaikan linier, demikian pula dari^[2] 7144mV - 9241 mV dan^[3] 9241 mV - 10144 mV tegangan keluaran linier terhadap frekuensi., grafik [1] dan [2] dan [3] tidak sejenis. Hal ini disebabkan komponen yang digunakan respon tegangan terhadap frekuensi tidak linier.
4. Perbedaan hasil pembacaan alat ukur suhu yang di buat dengan alat ukur suhu buatan OMRON maksimum 2°C .

DAFTAR PUSTAKA

- [1] BOYET, Howard, The 8051 Programming, Interfacing, Applications, Microprocessor Training Inc., U.S.A, 1990
- [2] Intel Corporation, Embedded Controller Handbook Vol 1, Santa lara, California, 1988
- [3] Elektor Electronics, 8051/8032 Assembler Course, November 1992
- [4] RODEN, Martin S., Electronic Design, The Benjamin Cummings Publishing, Inc., 1991
- [5] LEE, Samuel C., Digital Circuits and Logic Design, Prentice Hall, Inc., Englewood Cliffs, U.S.A, 1976
- [6] SINCLAIR, Ian Robertson, Beginer's Guide to Digital Electronics, Newnees Technical Book, 1980
- [7] LOVEDAY, George, Essential Electronics, Pitman Publishing Company, 1986
- [8] WOOLLARD, Barry, Practical Electronics, Mc Graw Hill Book Company (UK) Limited 1988
- [9] Data Sheet Integrated Circuit.
- [10] Tugas Akhir ITI , Eka Putra Astajaya, th 1995

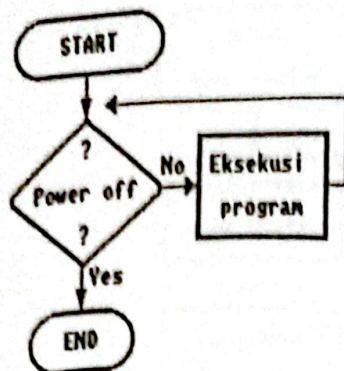
LAMPIRAN A



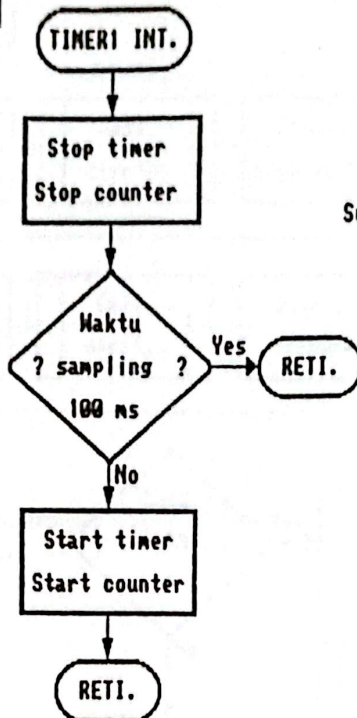
**FOTO ALAT PEMANTAU DAN PENGENDALI TEMPERATUR 25°C - 999°C
BERBASIS MIKROKONTROLER - 80C31**

LAMPIRAN B

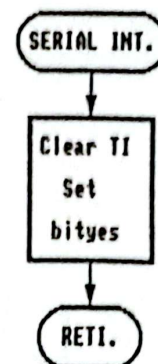
**DIAGRAM ALIR PROGRAM TEMPERATUR DAN PENGENDALI
TEMPERATUR 25°C - 999°C BERBASIS MIKROKONTROLER - 80C31**



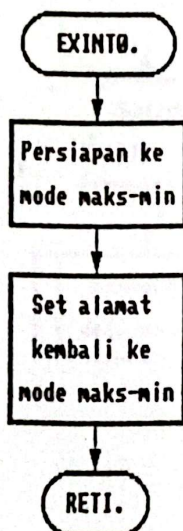
Program power on/off



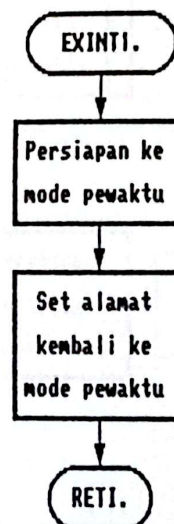
Subrutin interupsi timer 1



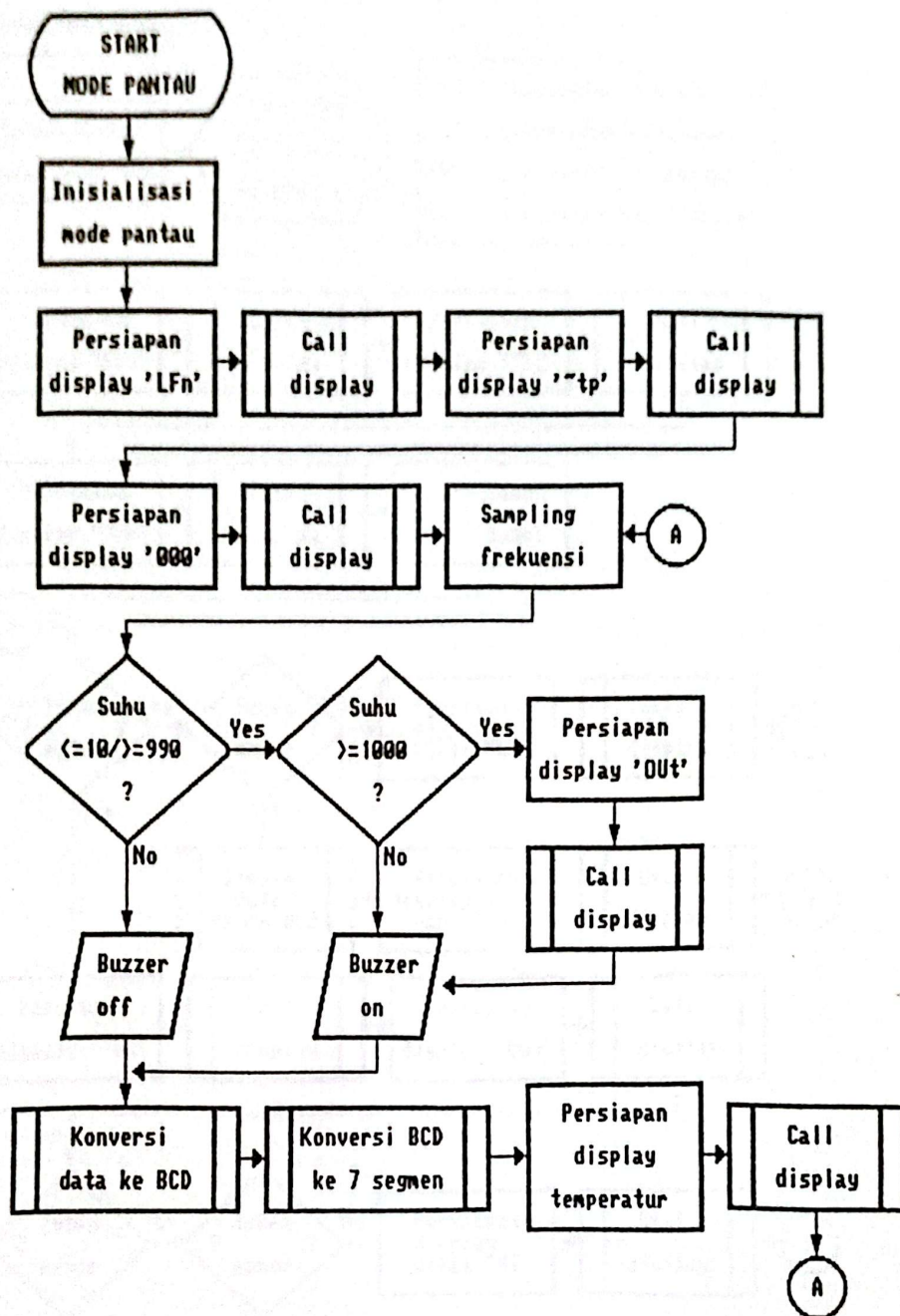
Subrutin interupsi serial



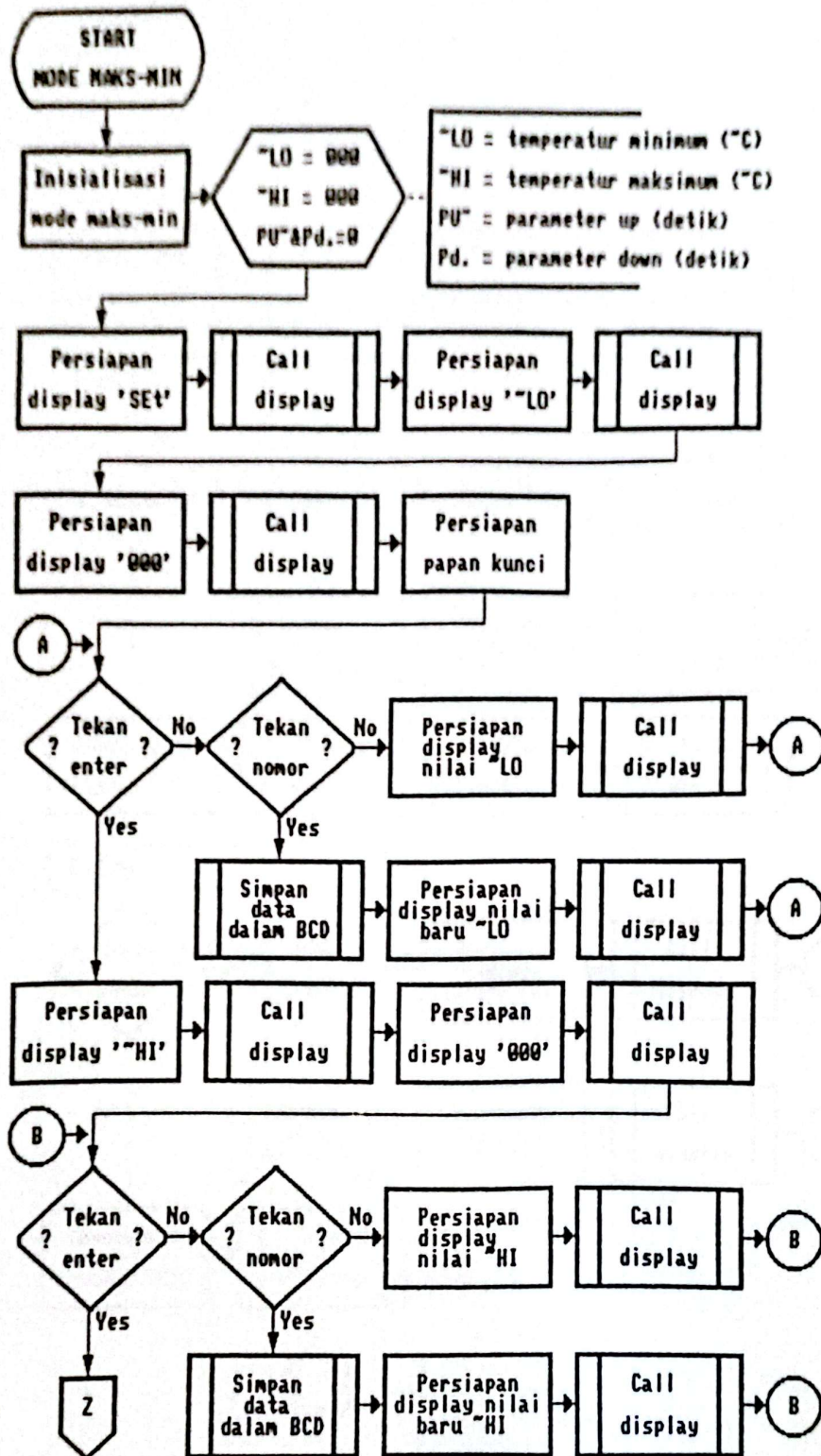
Subrutin interupsi eksternal 0



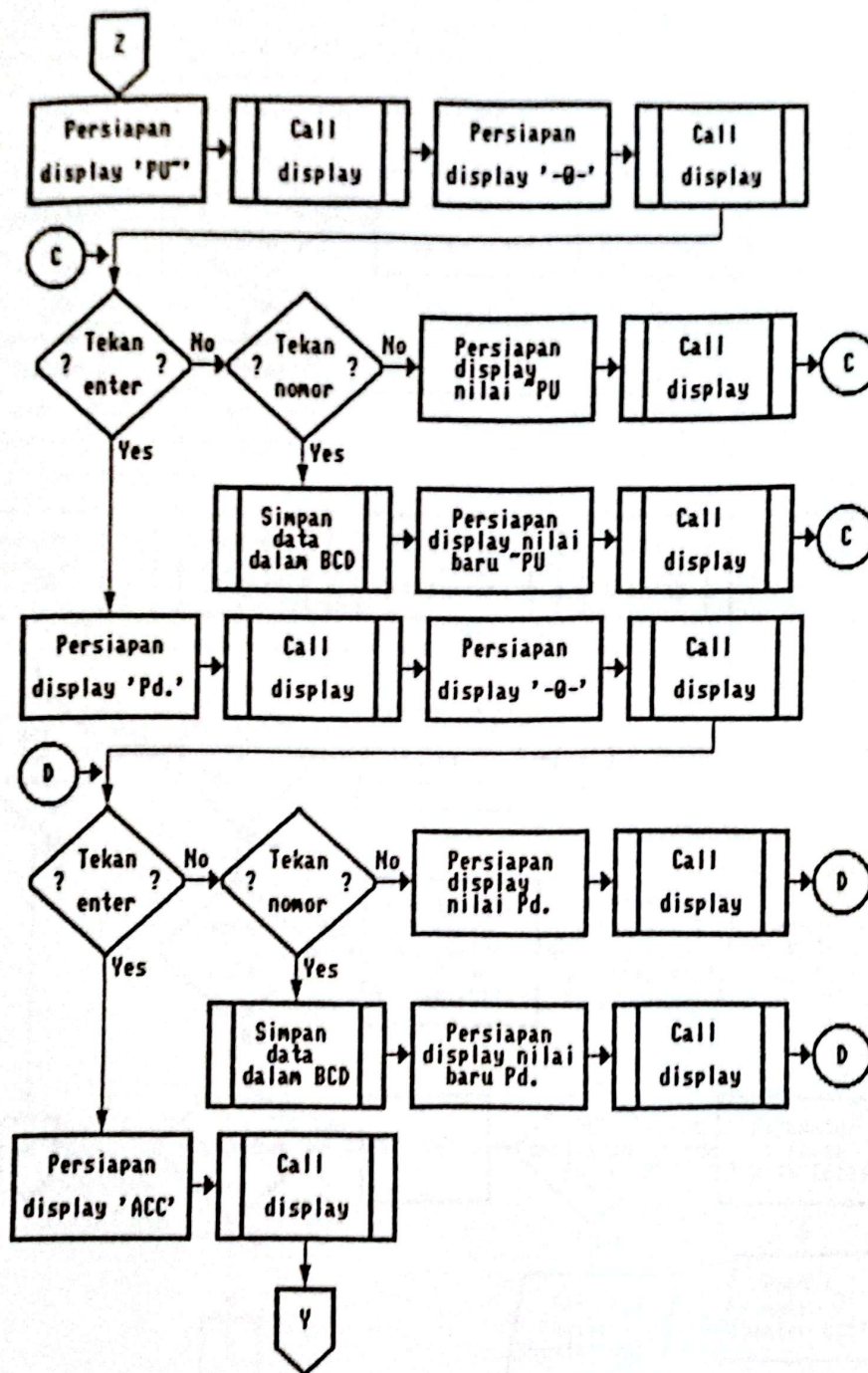
Subrutin interupsi eksternal 1

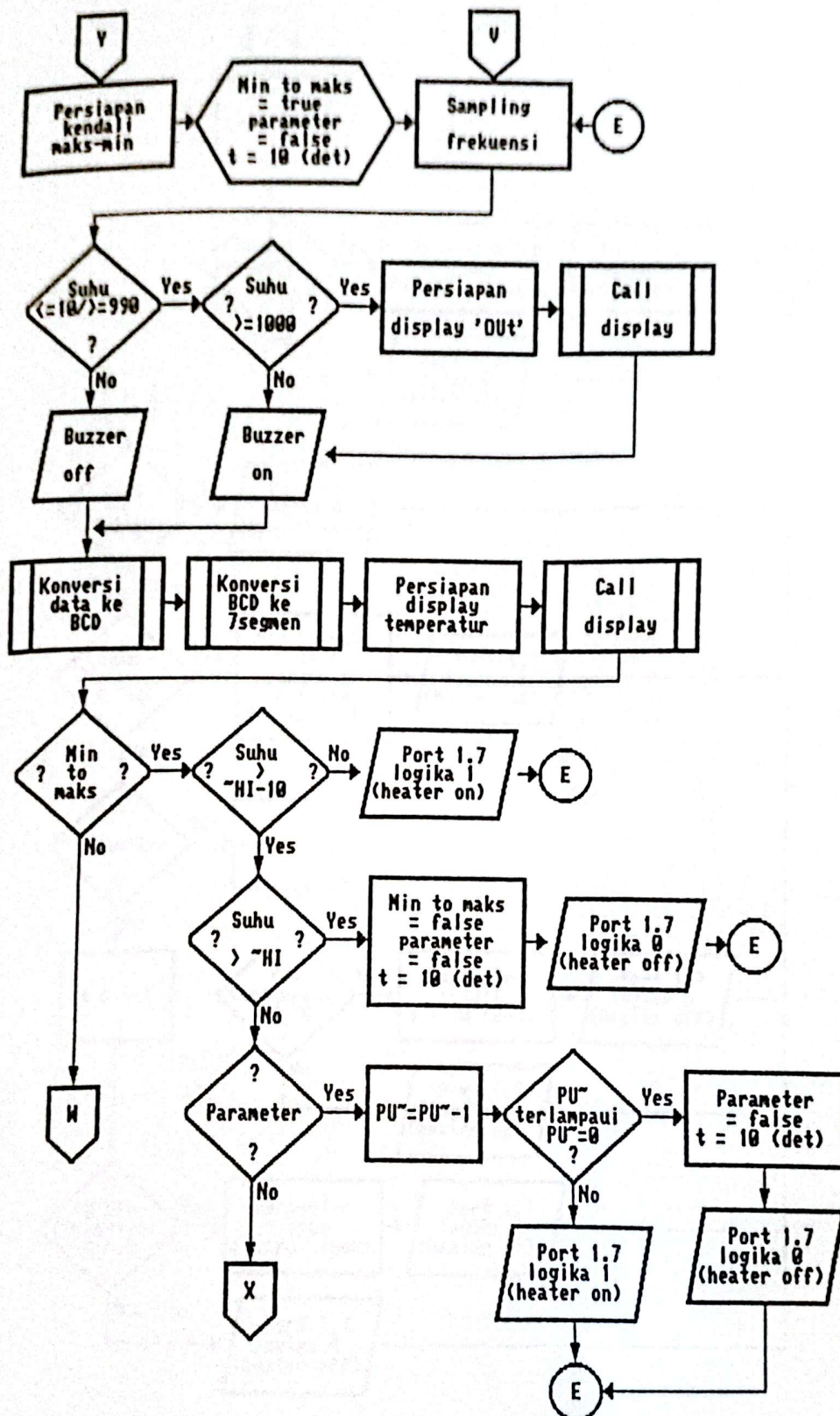


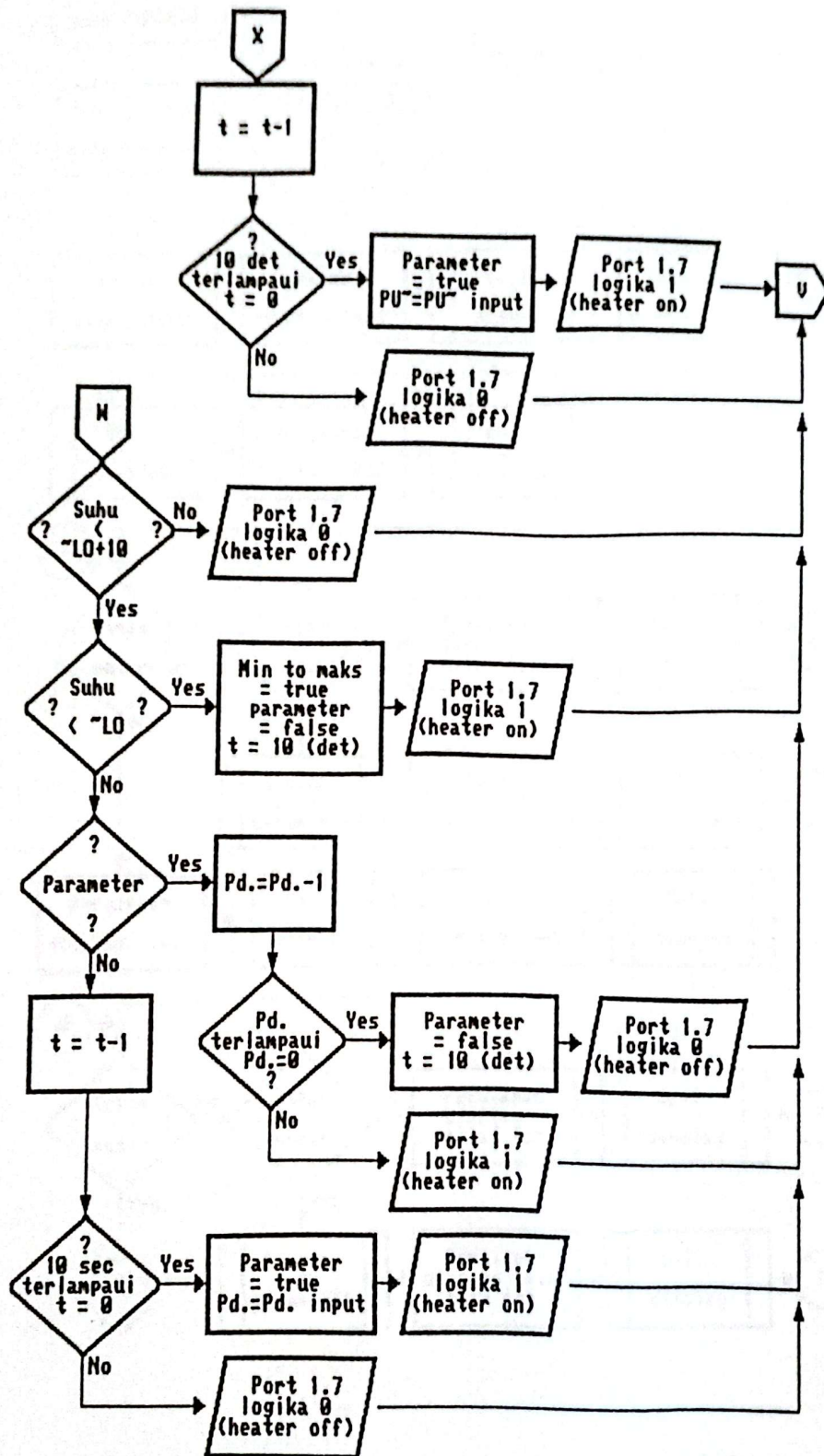
Program mode pemantauan

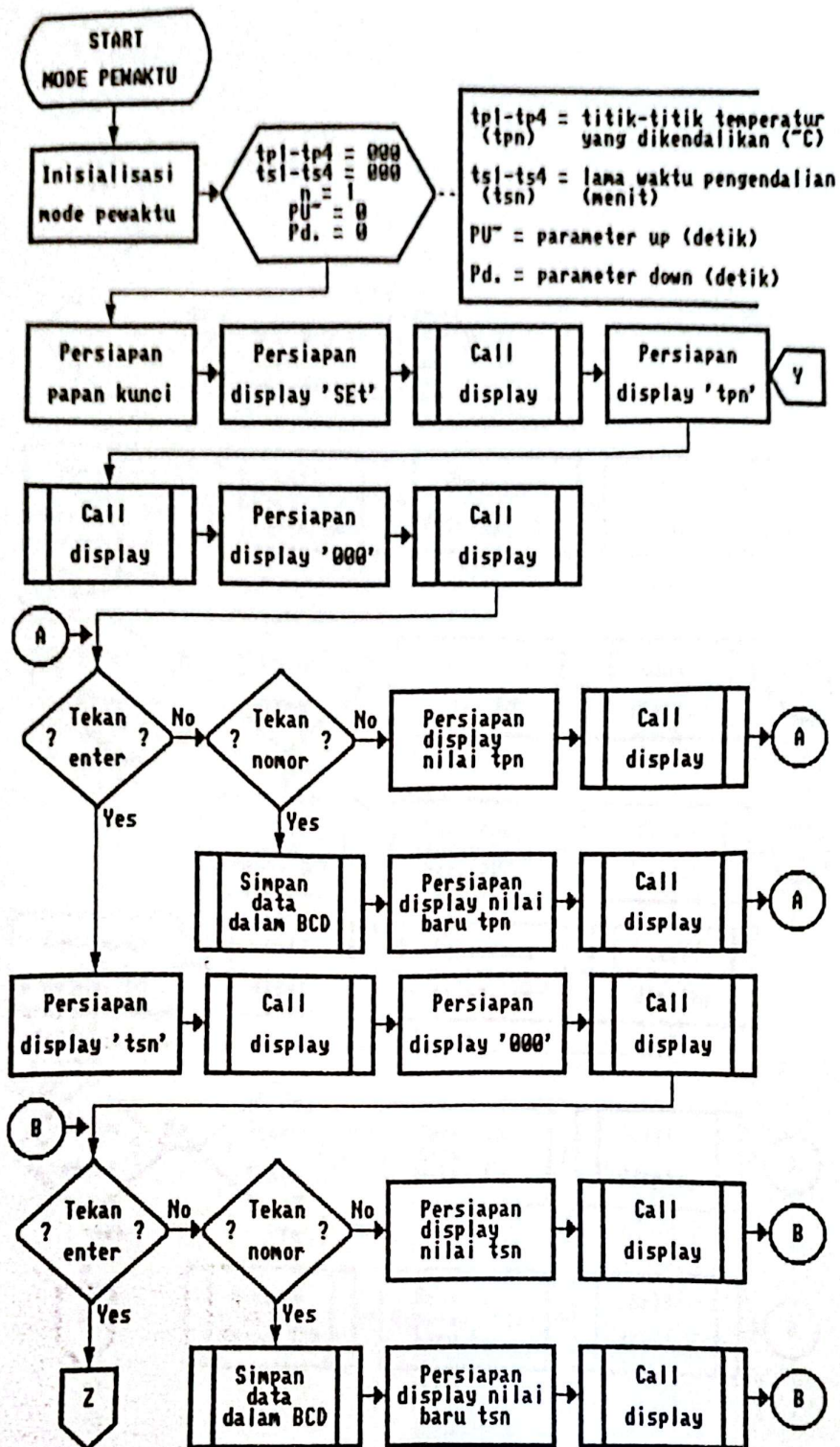


Pengendalian maksimum - minimum

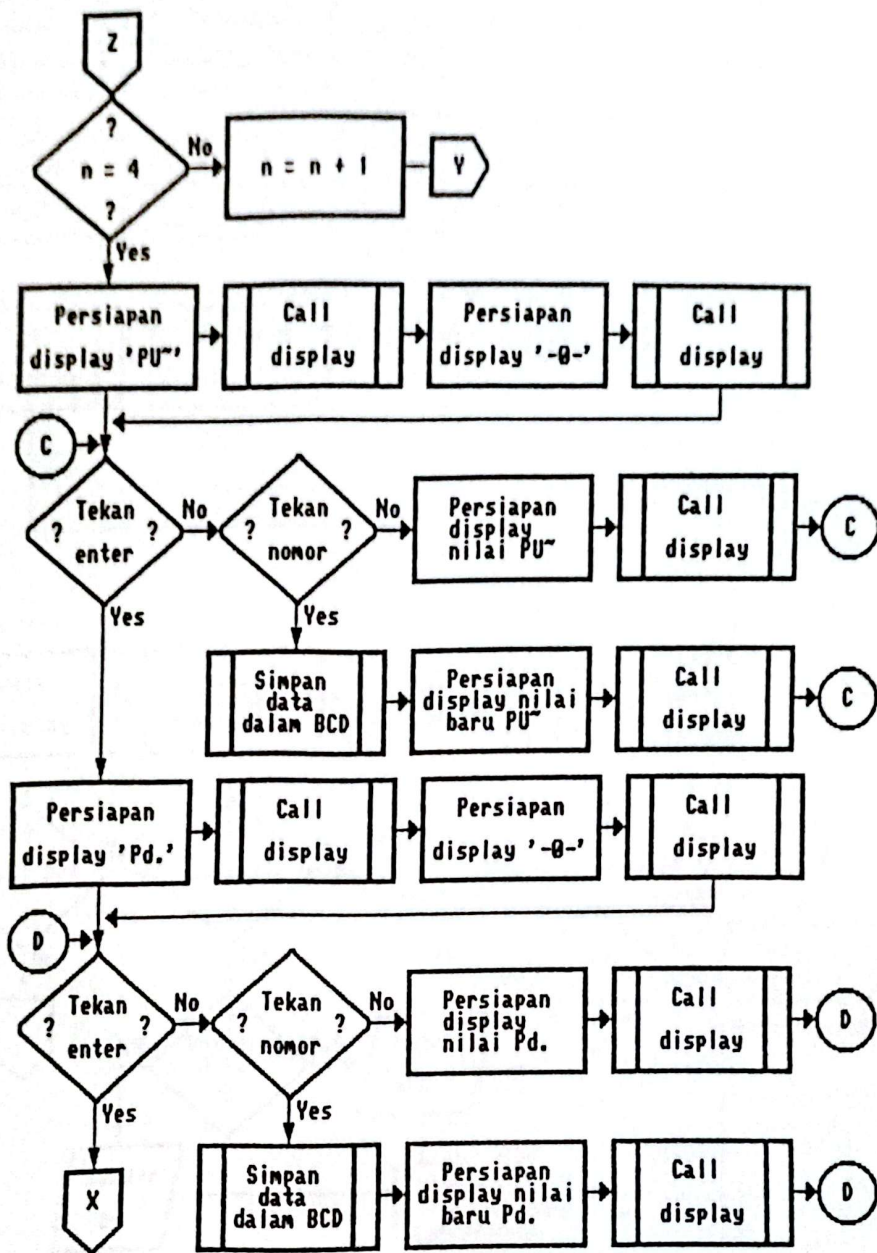


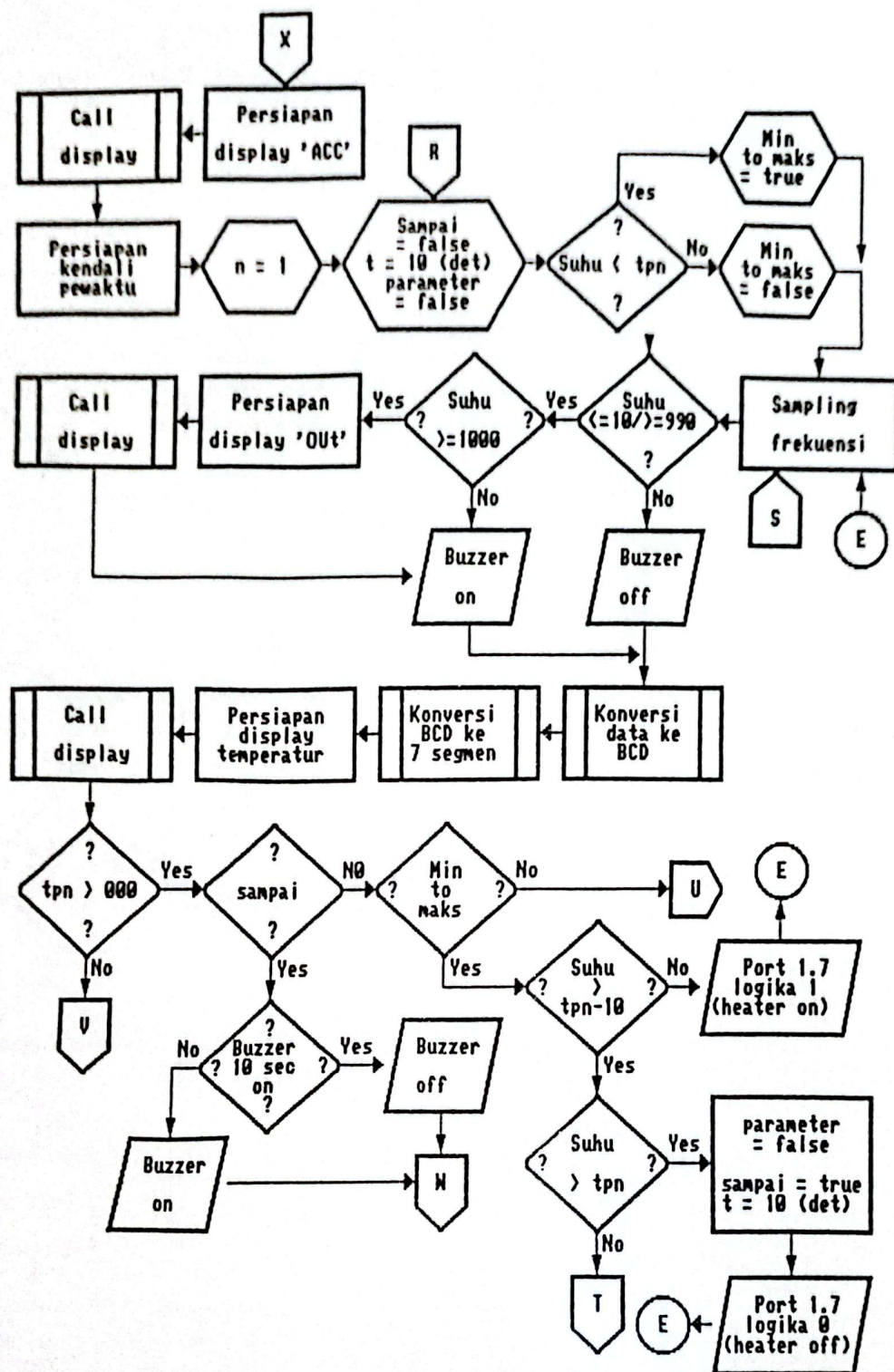


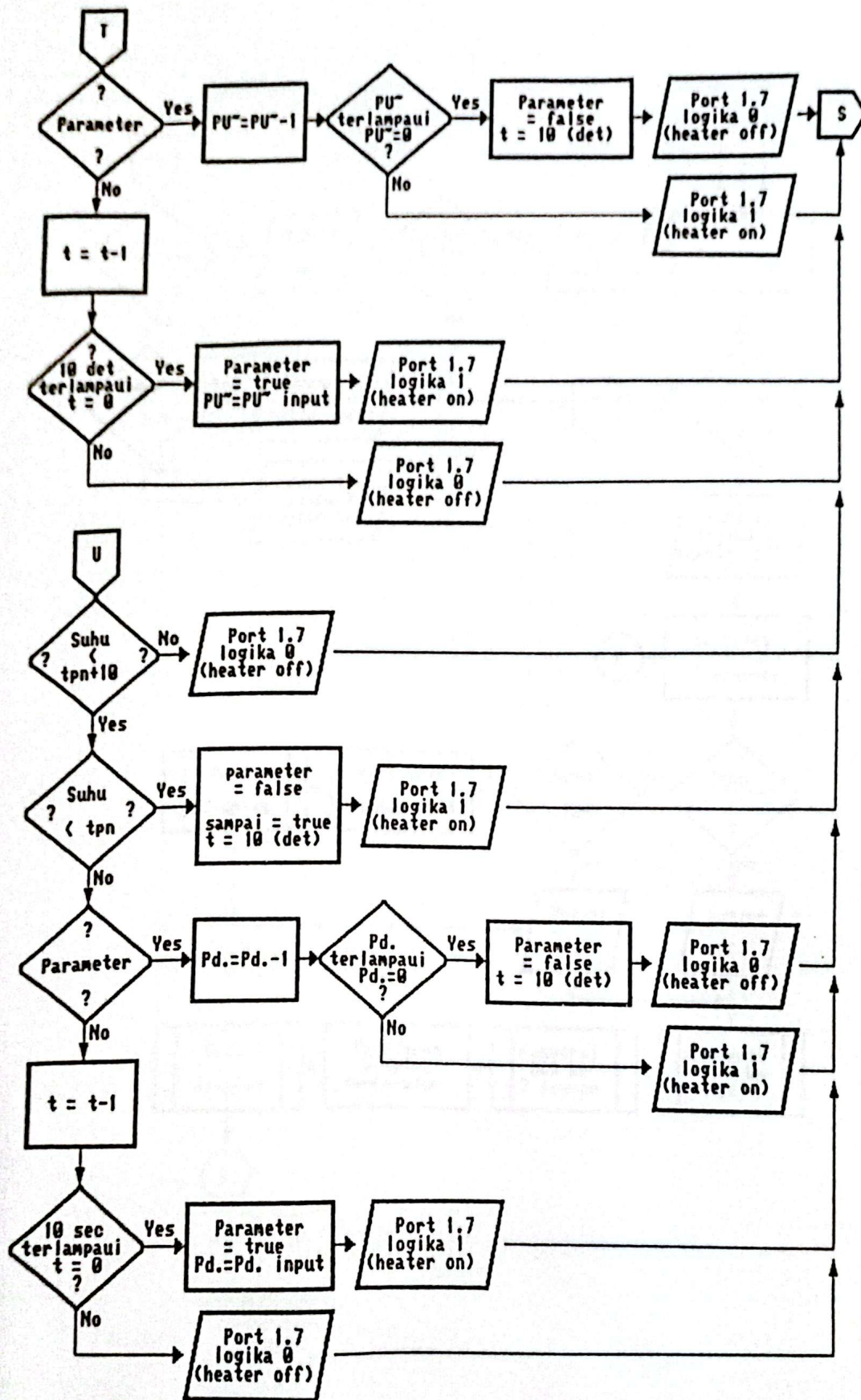


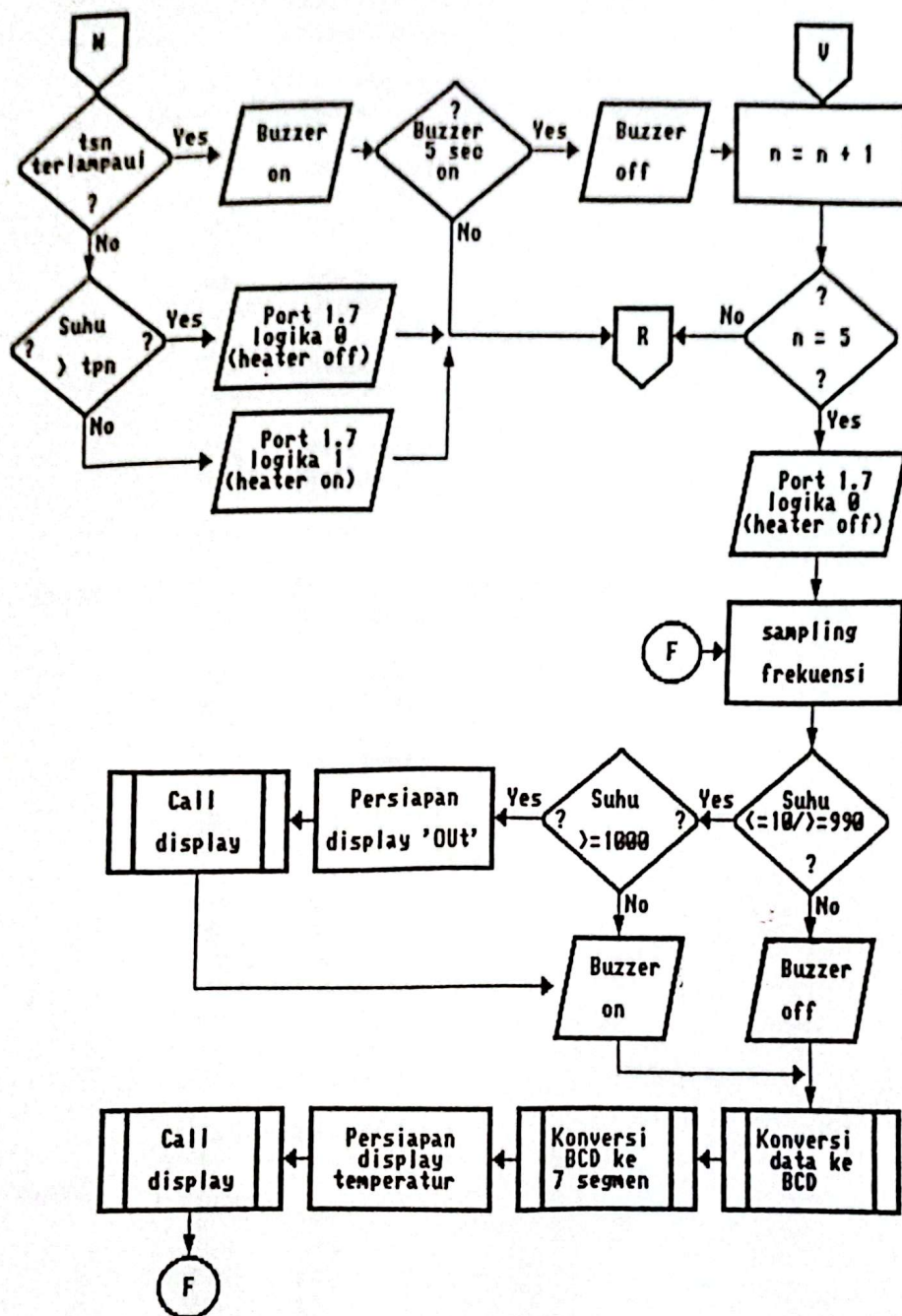


Pengendalian dengan pewaktu










```

;*****\
;*****/
;
;NAMA PROGRAM : SERIAL INTERFACE TEST PROGRAM
;                BAHASA MESIN MCS-80C31
;PEMROGRAM    : YVES AZELLA
;NIRM         : 011860106
;UNIVERSITAS  : INSTITUT TEKNOLOGI INDONESIA
;FAKULTAS     : FTI/ELEKTRO
;
;*****\
;*****/
;
EXINT0      EQU    03H
EXINT1      EQU    13H
RESET       EQU    00H
;
;                ORG    RESET
;                LJMP   START
;
;                ORG    EXINT0
;                LJMP   EX0INT
;
;                ORG    EXINT1
;                LJMP   EX1INT
;
;                ORG    30H
;
START:      MOV     SCON,#01000010B
            MOV     TMOD,#00100000B
;
            MOV     SP,#40H
            MOV     PSW,#00H
;
            MOV     TH1,#0E6H
;
            SETB    IT1
            SETB    IT0
;
            SETB    EA
            SETB    EX0
            SETB    EX1
;
            MOV     R5,#00H
            MOV     R6,#00H
            MOV     R7,#00H
;
            LCALL   DELAY
;
SEND:      NOP
            SETB    TR1
            CLR     EX0
            CLR     EX1
            CLR     F0
;
;            LCALL   DELAY1
;            NOP
;            MOV     SBUF,R7
;
OUTCHAR1:  JNB     TI,OUTCHAR1
            CLR     TI
;
;            LCALL   DELAY1
;            NOP

```

```

MOV     SBUF,R6
;
OUTCHAR2: JNB     TI,OUTCHAR2
          CLR     TI
;
          LCALL   DELAY1
          NOP
          MOV     SBUF,R5
;
OUTCHAR3: JNB     TI,OUTCHAR3
          CLR     TI
;
          SETB    EX0
          SETB    EX1
;
LOOP:    NOP
          JNB     F0,LOOP
;
          NOP
          INC     R5
          CJNE    R5,#0AH,LANJUT
          MOV     R5,#00H
          INC     R6
          CJNE    R6,#0AH,LANJUT
          MOV     R6,#00H
          INC     R7
          CJNE    R7,#0AH,LANJUT
          MOV     R7,#00H
          MOV     R6,#00H
          MOV     R5,#00H
;
LANJUT:  NOP
          LJMP    SEND
;
EX0INT:  CLR     TR1
          SETB    F0
          LCALL   DELAY
          RETI
;
EX1INT:  CLR     TR1
          SETB    F0
          LCALL   DELAY
          RETI
;
DELAY:   MOV     R1,#04H
LOP1:    MOV     R2,#FFH
LOP2:    MOV     R3,#FFH
LOP3:    DJNZ    R3,LOP3
          DJNZ    R2,LOP2
          DJNZ    R1,LOP1
          RET
;
DELAY1:  MOV     R1,#0AH
LOPP1:   MOV     R2,#FFH
LOPP2:   DJNZ    R2,LOPP2
          DJNZ    R1,LOPP1
          RET
;
END

```



```

;*****\
;*****/
;
;NAMA PROGRAM : KENDALI TEMPERATUR BERDASARKAN
;                BAHASA MESIN MCS-51
;PEMROGRAM    : YVES AZELLA
;NIRM         : 011860106
;UNIVERSITAS  : INSTITUT TEKNOLOGI INDONESIA
;FAKULTAS     : FTI/ELEKTRO
;
;*****\
;*****/
;
;*****
;    EKVIVALENSI ALAMAT PROGRAM DAN DATA MEMORI  *
;*****
;
BCD1      EQU 54H
BCD2      EQU 55H
BCD3      EQU 56H
;-----
BISELEC   EQU 57H
;-----
BITBURN   EQU 00H
BITBUZZ   EQU 01H
BITCURR   EQU 02H
BITENDU   EQU 03H
BITENDD   EQU 04H
BITLIMT   EQU 05H
BITOVR    EQU 06H
BITTAG    EQU 07H
BITTAGA   EQU 08H
BITTAGU   EQU 09H
BITTAGD   EQU 0AH
BITYES    EQU 0BH
BITZSTO   EQU 0CH
BITZSTP   EQU 0DH
;-----
COLOM0    EQU 30H
COLOM1    EQU 31H
COLOM2    EQU 32H
;-----
DIGIT1    EQU 24H
DIGIT2    EQU 25H
DIGIT3    EQU 26H
;-----
ENDCOL    EQU 2FH
;-----
EXTINT0    EQU 03H
EXTINT1    EQU 13H
;-----
KEYCURR    EQU 33H
;-----
KEYLOW1    EQU 27H
KEYLOW2    EQU 28H
KEYLOW3    EQU 29H
;-----
KEYHIG1    EQU 2AH
KEYHIG2    EQU 2BH
KEYHIG3    EQU 2CH

```

```

;-----
KEYPARU EQU 2DH
KEYFARD EQU 2EH
;-----
LMTOUT0 EQU 0E7H
LMTOUT1 EQU 03H
LMTOVR0 EQU 0DEH
LMTOVR1 EQU 03H
LMTUNDO EQU 00AH
LMTUND1 EQU 00H
;-----
REGMODE EQU 34H
;-----
RESET EQU 00H
;-----
SAMHIGH EQU 35H
SAMLOW EQU 36H
;-----
SERIAL EQU 23H
;-----
TMRINT1 EQU 1BH
;-----
TP1HLD1 EQU 37H
TP1HLD2 EQU 38H
TP1HLD3 EQU 39H
;-----
TP2HLD1 EQU 3AH
TP2HLD2 EQU 3BH
TP2HLD3 EQU 3CH
;-----
TP3HLD1 EQU 3DH
TP3HLD2 EQU 3EH
TP3HLD3 EQU 3FH
;-----
TP4HLD1 EQU 40H
TP4HLD2 EQU 41H
TP4HLD3 EQU 42H
;-----
TM1HLD1 EQU 43H
TM1HLD2 EQU 44H
TM1HLD3 EQU 45H
;-----
TM2HLD1 EQU 46H
TM2HLD2 EQU 47H
TM2HLD3 EQU 48H
;-----
TM3HLD1 EQU 49H
TM3HLD2 EQU 4AH
TM3HLD3 EQU 4BH
;-----
TM4HLD1 EQU 4CH
TM4HLD2 EQU 4DH
TM4HLD3 EQU 4EH
;-----
TPTMEND EQU 4FH
;-----
TRSECND EQU 50H
TRMINIT EQU 51H
TRHOURS EQU 52H
;-----

```



```

ZREGTIM EQU 53H
;-----
;
;          ORG     RESET
;          LJMP    INITAWL
;
;          ORG     EXTINT0
;          LJMP    EXT0INT
;
;          ORG     TMRINT1
;          LJMP    TC1INTR
;
;          ORG     SERIAL
;          LJMP    TRINTR
;
;*****
;      INISIALISASI AWAL PERANGKAT KERAS      *
;*****
;
;          ORG     30H
INITAWL:  MOV     SP,#60H
;          MOV     P1,#01111111B
;          MOV     SCON,#00H
;          MOV     TCON,#00000001B
;          MOV     IE,#10000001B
;
;          CLR     BITBURN
;          CLR     BITBUZZ
;          CLR     BITCURR
;          CLR     BITLIMT
;          CLR     BITOVR
;
;          SETB    PT1
;
;          JB      F0,TOHOLD
;          JC      TOKEY
;          JB      OV,PANTAU
;
;          LJMP    LOGOLFN
;
TOHOLD:   NOP
;          CLR     F0
;          LJMP    KEYHOLD
;
TOKEY:    NOP
;          CLR     C
;          LJMP    KEYPAD
;
;*****
;      PROGRAM LOGO LFN      *
;*****
;
LOGOLFN:  MOV     BISELEC,#00H
;
;          MOV     DIGIT1,#00011100B
;          MOV     DIGIT2,#10001110B
;          MOV     DIGIT3,#11101100B
;
;          LCALL   DISPLAY
;          LJMP    LOGOLFN

```

```

;.....
; PROGRAM MODE PEMANTAUAN
;.....
PANTAU:          CLR      OV
;
MOV      DIGIT1,#11000110B
MOV      DIGIT2,#00011110B
MOV      DIGIT3,#11001110B
;
SAMPLIN:         NOP
SETB     ET1
CLR      BITOVR
;
MOV      TMOD,#15H
MOV      TLO,#00H
MOV      TH0,#00H
MOV      TL1,#00H
MOV      TH1,#00H
;
ORL      TCON,#50H
;
LCALL    DISPLAY
LCALL    DISPLAY
LCALL    DISPLAY
;
;*****
; PEMERIKSA HASIL SAMPLING DAN TAMPILKAN *
;*****
GETSAM:          ANL      TCON,#0AFH
CLR      ET1
;
MOV      SAMLOW,TLO
MOV      SAMHIGH,TH0
;
CLR      C
;
MOV      A,SAMLOW
SUBB     A,#LMTUNDO
MOV      A,SAMHIGH
SUBB     A,#LMTUND1
JC       UNDRANG
;
MOV      A,#LMTOUT0
SUBB     A,SAMLOW
MOV      A,#LMTOUT1
SUBB     A,SAMHIGH
JC       OUTRANG
;
MOV      A,#LMTOVR0
SUBB     A,SAMLOW
MOV      A,#LMTOVR1
SUBB     A,SAMHIGH
JC       OVERANG
;
CLR      BITBUZZ
SETB     P1.3

```



```

ACTION:      NOP
             LCALL  HEX2BCD
             NOP
             LCALL  CONVERT
;
             LJMP   SAMPLIN
;
UNDRANG:     CLR    P1.3
             SETB   BITBUZZ
             LJMP   ACTION
;
OVERANG:     CLR    P1.3
             SETB   BITBUZZ
             LJMP   ACTION
;
OUTRANG:     CLR    P1.3
             SETB   BITBUZZ
;
             MOV    DIGIT1,#11111100B
             MOV    DIGIT2,#01111100B
             MOV    DIGIT3,#00011110B
;
             LJMP   SAMPLIN
;
;*****
;  RUTIN KONVERSI HEXA KE BINARY CODE DECIMAL  *
;*****
;
HEX2BCD:     NOP
             MOV    R5,#00H
             MOV    R6,#00H
             MOV    A,SAMLOW
             MOV    B,#64H
             DIV    AB
             MOV    R4,A
             MOV    A,#0AH
             XCH    A,B
             DIV    AB
             SWAP   A
             ADD    A,B
             MOV    R3,A
;
             MOV    A,SAMHIGH
             CJNE   A,#03H,TWOBIT
             MOV    R5,#68H
             MOV    R6,#07H
             LJMP   JUMLAH
;
TWOBIT:      NOP
             CJNE   A,#02H,ONEBIT
             MOV    R5,#12H
             MOV    R6,#05H
             LJMP   JUMLAH
;
ONEBIT:      NOP
             CJNE   A,#01H,JUMLAH
             MOV    R5,#56H
             MOV    R6,#02H
;
JUMLAH:      NOP

```

```

CLR      C
MOV      A, R3
ADD      A, R5
DA       A
MOV      BCD3, A
ANL      BCD3, #0FH
SWAP     A
MOV      BCD2, A
ANL      BCD2, #0FH
MOV      A, R4
ADDC     A, R6
DA       A
MOV      BCD1, A
ANL      BCD1, #0FH
;
RET
;
;*****
;RUTIN KONVERSI BINARY CODE DECIMAL KE 7SEGMENT*
;*****
;
CONVERT:  NOP
MOV      DPTR, #CHARGEN
MOV      A, BCD1
MOVC     A, @A+DPTR
MOV      DIGIT1, A
MOV      A, BCD2
MOVC     A, @A+DPTR
MOV      DIGIT2, A
MOV      A, BCD3
MOVC     A, @A+DPTR
MOV      DIGIT3, A
;
RET
;
;*****
;CHARACTER GENERATOR YG MERUPAKAN LOOK-UP TABLE*
;*****
;
CHARGEN:  DB      11111100B      ; '0'
          DB      01100000B      ; '1'
          DB      11011010B
          DB      11110010B
          DB      01100110B
          DB      10110110B      ; '5'
          DB      10111110B
          DB      11100000B
          DB      11111110B
          DB      11110110B      ; '9'
;
;*****
;          RUTIN DELAY TAMPILAN 7-SEGMENT      *
;*****
;
DELAY:    NOP
MOV      R2, #0AH
LOOP2:    MOV      R3, #0FFH
LOOP1:    DJNZ     R3, LOOP1
          DJNZ     R2, LOOP2
;

```



```

RET
;
;*****
; RUTIN TAMPILAN LIGHT EMMITED DIODE 7-SEGMENT *
;*****
;
DISPLAY:    NOP
            MOV     R6, #40H
;
NEXLOOP:    NOP
            MOV     R7, #03H
            MOV     R0, #DIGIT1
            MOV     A, #11111110B
;
NEXTBIT:    NOP
            CLR     EX0
            SETB    ES
            MOV     SBUF, @R0
WAITYES:    JNB     BITYES, WAITYES
            CLR     BITYES
            CLR     ES
            SETB    EX0
;
            JB     BITBURN, BURNOUT
            CLR     A.7
BUZZER:     NOP
            JB     BITBUZZ, BUZZREQ
            SETB    A.3
;
POINDIG:    NOP
            MOV     P1, A
;
            LCALL   DELAY
;
            DJNZ    R7, NOTALL
            LJMP    THATALL
;
BURNOUT:    NOP
            SETB    A.7
            LJMP    BUZZER
;
BUZZREQ:    NOP
            CLR     A.3
            LJMP    POINDIG
;
NOTALL:     NOP
            INC     R0
            RL      A
            SETB    A.0
            LJMP    NEXTBIT
;
THATALL:    NOP
            DJNZ    R6, NEXLOOP
;
            RET
;
;*****
; PROGRAM MODE KENDALI DENGAN PENAHANAN WAKTU *
;*****

```

```

;
KEYHOLD:    NOP
;
            MOV     REGMODE,#00H
            MOV     KEYCURR,#00H
;
            MOV     KEYPARU,#00H
            MOV     KEYPARD,#00H
;
            MOV     TRSECND,#00H
            MOV     TRMINIT,#00H
            MOV     TRHOURE,#00H
;
            MOV     R1,#TP1HLD1
MAKENOL:    MOV     @R1,#00H
            INC     R1
            CJNE    R1,#TPTMEND,MAKENOL
;
            MOV     COLOM0,#01101111B
            MOV     COLOM1,#01011111B
            MOV     COLOM2,#00111111B
;
            MOV     DIGIT1,#10110110B
            MOV     DIGIT2,#00011110B
            MOV     DIGIT3,#11011010B
;
            LCALL   DISPLAY
            LCALL   DISPLAY
;
            MOV     DIGIT1,#00011110B
            MOV     DIGIT2,#11001110B
            MOV     DIGIT3,#01100000B
;
            LCALL   DISPLAY
            LCALL   DISPLAY
;
            MOV     DIGIT1,#11111100B
            MOV     DIGIT2,#11111100B
            MOV     DIGIT3,#11111100B
;
            LCALL   DISPLAY
;
            MOV     R1,#COLOM2
            MOV     P1,@R1
;
            LCALL   DEBOUNC
;
KEYIN:      MOV     R3,#00H
            MOV     A,P1
;
KEYSCAN:    JNB     A.0,CLOSKEY
;
ROWNEXT:    INC     R3
            RR      A
            CJNE    R3,#04H,KEYSCAN
;
            DEC     R1
            CJNE    R1,#ENDCOL,COLNEXT
            LJMP    COLALL
;

```



```

CLOSKEY:    PUSH    ACC
            MOV     A,R1
            ANL     A,#0FH
            SWAP    A
            ORL     A,R3
            MOV     KEYCURR,A
            POP     ACC
            LJMP    PREDIKS

;
COLALL:     LCALL   DISPKEY
            MOV     R1,#COLOM2

;
COLNEXT:    MOV     P1,@R1
            LCALL   DEBOUNC
            LJMP    KEYIN

;
;*****
;      RUTIN ANALISA HASIL PENEKANAN KEYPAD      *
;*****
;
PREDIKS:    NOP
            MOV     A,KEYCURR
            CJNE    A,#23H,EIGHT8
            MOV     A,#09H
            MOV     R4,#11110110B
            LCALL   BCDSAVE
            LJMP    COLALL

;
EIGHT8:     MOV     A,KEYCURR
            CJNE    A,#13H,SEVEN7
            MOV     A,#08H
            MOV     R4,#11111110B
            LCALL   BCDSAVE
            LJMP    COLALL

;
SEVEN7:     MOV     A,KEYCURR
            CJNE    A,#03H,SIX6
            MOV     A,#07H
            MOV     R4,#11100000B
            LCALL   BCDSAVE
            LJMP    COLALL

;
SIX6:       MOV     A,KEYCURR
            CJNE    A,#22H,FIVE5
            MOV     A,#06H
            MOV     R4,#10111110B
            LCALL   BCDSAVE
            LJMP    COLALL

;
FIVE5:      MOV     A,KEYCURR
            CJNE    A,#12H,FOUR4
            MOV     A,#05H
            MOV     R4,#10110110B
            LCALL   BCDSAVE
            LJMP    COLALL

;
FOUR4:      MOV     A,KEYCURR
            CJNE    A,#02H,THREE3
            MOV     A,#04H
            MOV     R4,#01100110B

```

```

        LCALL  BCDSAVE
        LJMP   COLALL

;
THREE3:  MOV    A,KEYCURR
        CJNE   A,#21H,TWO2
        MOV    A,#03H
        MOV    R4,#11110010B
        LCALL  BCDSAVE
        LJMP   COLALL

;
TWO2:    MOV    A,KEYCURR
        CJNE   A,#11H,ONE1
        MOV    A,#02H
        MOV    R4,#11011010B
        LCALL  BCDSAVE
        LJMP   COLALL

;
ONE1:    MOV    A,KEYCURR
        CJNE   A,#01H,ZERO0
        MOV    A,#01H
        MOV    R4,#01100000B
        LCALL  BCDSAVE
        LJMP   COLALL

;
ZERO0:   MOV    A,KEYCURR
        CJNE   A,#10H,ENTERX
        MOV    A,#00H
        MOV    R4,#11111100B
        LCALL  BCDSAVE
        LJMP   COLALL

;
ENTERX:  INC     REGMODE
        PUSH   ACC
        MOV    A,REGMODE
        CJNE   A,#01H,SHOHL2
        POP    ACC

;
SHOTIM1: MOV    DIGIT1,#00011110B
        MOV    DIGIT2,#10110110B
        MOV    DIGIT3,#01100000B

;
        LCALL  DISPLAY
        LCALL  DISPLAY

;
        MOV    DIGIT1,#11111100B
        MOV    DIGIT2,#11111100B
        MOV    DIGIT3,#11111100B

;
        LJMP   COLALL

;
SHOHL2:  CJNE   A,#02H,SHOTIM2
        POP    ACC

;
        MOV    TM1HLD1,TRHOUR
        MOV    TM1HLD2,TRMINIT
        MOV    TM1HLD3,TRSECND

;
        MOV    DIGIT1,#00011110B
        MOV    DIGIT2,#11001110B
        MOV    DIGIT3,#11011010B

```



```

;          LCALL  DISPLAY
;          LCALL  DISPLAY

;          MOV     DIGIT1,#11111100B
;          MOV     DIGIT2,#11111100B
;          MOV     DIGIT3,#11111100B

;          LJMP    COLALL

; SHOTIM2:      CJNE  A,#03H,SHOHL3
;              POP   ACC

;          MOV     DIGIT1,#00011110B
;          MOV     DIGIT2,#10110110B
;          MOV     DIGIT3,#11011010B

;          LCALL  DISPLAY
;          LCALL  DISPLAY

;          MOV     DIGIT1,#11111100B
;          MOV     DIGIT2,#11111100B
;          MOV     DIGIT3,#11111100B

;          LJMP    COLALL

; SHOHL3:      CJNE  A,#04H,SHOTIM3
;              POP   ACC

;          MOV     TM2HLD1,TRHOUR
;          MOV     TM2HLD2,TRMINIT
;          MOV     TM2HLD3,TRSECND

;          MOV     DIGIT1,#00011110B
;          MOV     DIGIT2,#11001110B
;          MOV     DIGIT3,#11110010B

;          LCALL  DISPLAY
;          LCALL  DISPLAY

;          MOV     DIGIT1,#11111100B
;          MOV     DIGIT2,#11111100B
;          MOV     DIGIT3,#11111100B

;          LJMP    COLALL

; SHOTIM3:      CJNE  A,#05H,SHOHL4
;              POP   ACC

;          MOV     DIGIT1,#00011110B
;          MOV     DIGIT2,#10110110B
;          MOV     DIGIT3,#11110010B

;          LCALL  DISPLAY
;          LCALL  DISPLAY

;          MOV     DIGIT1,#11111100B
;          MOV     DIGIT2,#11111100B
;          MOV     DIGIT3,#11111100B

```

```

        LJMP    COLALL
;
;SHOHL4:    CJNE    A, #06H, SHOTIM4
            POP     ACC
;
            MOV     TM3HLD1, TRHOUR
            MOV     TM3HLD2, TRMINIT
            MOV     TM3HLD3, TRSECND
;
            MOV     DIGIT1, #00011110B
            MOV     DIGIT2, #11001110B
            MOV     DIGIT3, #01100110B
;
            LCALL   DISPLAY
            LCALL   DISPLAY
;
            MOV     DIGIT1, #11111100B
            MOV     DIGIT2, #11111100B
            MOV     DIGIT3, #11111100B
;
            LJMP    COLALL
;
;SHOTIM4:    CJNE    A, #07H, HLDPARU
            POP     ACC
;
            MOV     DIGIT1, #00011110B
            MOV     DIGIT2, #10110110B
            MOV     DIGIT3, #01100110B
;
            LCALL   DISPLAY
            LCALL   DISPLAY
;
            MOV     DIGIT1, #11111100B
            MOV     DIGIT2, #11111100B
            MOV     DIGIT3, #11111100B
;
            LJMP    COLALL
;
;HLDPARU:    CJNE    A, #08H, HLDPARD
            POP     ACC
;
            MOV     TM4HLD1, TRHOUR
            MOV     TM4HLD2, TRMINIT
            MOV     TM4HLD3, TRSECND
;
            MOV     DIGIT1, #11001110B
            MOV     DIGIT2, #01111100B
            MOV     DIGIT3, #11000110B
;
            LCALL   DISPLAY
            LCALL   DISPLAY
;
            MOV     DIGIT1, #00000010B
            MOV     DIGIT2, #11111100B
            MOV     DIGIT3, #00000010B
;
            LJMP    COLALL
;
;HLDPARD:    CJNE    A, #09H, LEDACC
            POP     ACC

```



```

;      MOV      DIGIT1,#11001110B
;      MOV      DIGIT2,#01111010B
;      MOV      DIGIT3,#00111010B
;
;      LCALL    DISPLAY
;      LCALL    DISPLAY
;
;      MOV      DIGIT1,#00000010B
;      MOV      DIGIT2,#11111100B
;      MOV      DIGIT3,#00000010B
;
;      LJMP     COLALL
;
; LEDACC:      POP      ACC
;
;      MOV      DIGIT1,#11101110B
;      MOV      DIGIT2,#10011100B
;      MOV      DIGIT3,#10011100B
;
;      LCALL    DISPLAY
;      LCALL    DISPLAY
;
;      LJMP     HLDCTRL
;
; *****
;      RUTIN MENYIMPAN NILAI PENEKANAN KEYPAD      *
; *****
;
; BCDSAVE:     PUSH     ACC
;              MOV      A,REGMODE
;
; SAVHLD1:     CJNE     A,#00H,SAVTIM1
;              POP      ACC
;              MOV      TP1HLD1,TP1HLD2
;              MOV      TP1HLD2,TP1HLD3
;              MOV      TP1HLD3,A
;              MOV      DIGIT1,DIGIT2
;              MOV      DIGIT2,DIGIT3
;              MOV      DIGIT3,R4
;
;              RET
;
; SAVTIM1:     CJNE     A,#01H,SAVHLD2
;              POP      ACC
;              MOV      TM1HLD1,TM1HLD2
;              MOV      TM1HLD2,TM1HLD3
;              MOV      TM1HLD3,A
;              MOV      DIGIT1,DIGIT2
;              MOV      DIGIT2,DIGIT3
;              MOV      DIGIT3,R4
;
;              MOV      TRSECND,#3CH
;
;              CLR      C
;              MOV      A,TM1HLD1
;              MOV      B,#64H
;              MUL      AB
;              MOV      TRMINIT,A
;              MOV      TRHOUR,B

```

```

;
CLR      C
MOV      A, TM1HLD2
MOV      B, #0AH
MUL      AB
CLR      C
ADD      A, TM1HLD3
ADD      A, TRMINIT
MOV      TRMINIT, A

;
MOV      A, TRHOURE
ADDC     A, #00H
MOV      TRHOURE, A

;
RET

;
SAVHLD2: CJNE   A, #02H, SAVTIM2
POP      ACC
MOV      TP2HLD1, TP2HLD2
MOV      TP2HLD2, TP2HLD3
MOV      TP2HLD3, A
MOV      DIGIT1, DIGIT2
MOV      DIGIT2, DIGIT3
MOV      DIGIT3, R4

;
RET

;
SAVTIM2: CJNE   A, #03H, SAVHLD3
POP      ACC
MOV      TM2HLD1, TM2HLD2
MOV      TM2HLD2, TM2HLD3
MOV      TM2HLD3, A
MOV      DIGIT1, DIGIT2
MOV      DIGIT2, DIGIT3
MOV      DIGIT3, R4

;
MOV      TRSECND, #3CH

;
CLR      C
MOV      A, TM2HLD1
MOV      B, #64H
MUL      AB
MOV      TRMINIT, A
MOV      TRHOURE, B

;
CLR      C
MOV      A, TM2HLD2
MOV      B, #0AH
MUL      AB
CLR      C
ADD      A, TM2HLD3
ADD      A, TRMINIT
MOV      TRMINIT, A

;
MOV      A, TRHOURE
ADDC     A, #00H
MOV      TRHOURE, A

;
RET
;

```



```

SAVHLD3:    CJNE    A, #04H, SAVTIM3
            POP     ACC
            MOV     TP3HLD1, TP3HLD2
            MOV     TP3HLD2, TP3HLD3
            MOV     TP3HLD3, A
            MOV     DIGIT1, DIGIT2
            MOV     DIGIT2, DIGIT3
            MOV     DIGIT3, R4
;
            RET
;
SAVTIM3:    CJNE    A, #05H, SAVHLD4
            POP     ACC
            MOV     TM3HLD1, TM3HLD2
            MOV     TM3HLD2, TM3HLD3
            MOV     TM3HLD3, A
            MOV     DIGIT1, DIGIT2
            MOV     DIGIT2, DIGIT3
            MOV     DIGIT3, R4
;
            MOV     TRSECND, #3CH
;
            CLR     C
            MOV     A, TM3HLD1
            MOV     B, #64H
            MUL     AB
            MOV     TRMINIT, A
            MOV     TRHOURE, B
;
            CLR     C
            MOV     A, TM3HLD2
            MOV     B, #0AH
            MUL     AB
            CLR     C
            ADD     A, TM3HLD3
            ADD     A, TRMINIT
            MOV     TRMINIT, A
;
            MOV     A, TRHOURE
            ADDC    A, #00H
            MOV     TRHOURE, A
;
            RET
;
SAVHLD4:    CJNE    A, #06H, SAVTIM4
            POP     ACC
            MOV     TP4HLD1, TP4HLD2
            MOV     TP4HLD2, TP4HLD3
            MOV     TP4HLD3, A
            MOV     DIGIT1, DIGIT2
            MOV     DIGIT2, DIGIT3
            MOV     DIGIT3, R4
;
            RET
;
SAVTIM4:    CJNE    A, #07H, SAVHLDU
            POP     ACC
            MOV     TM4HLD1, TM4HLD2
            MOV     TM4HLD2, TM4HLD3
            MOV     TM4HLD3, A

```

```

MOV     DIGIT1,DIGIT2
MOV     DIGIT2,DIGIT3
MOV     DIGIT3,R4

;
MOV     TRSECND,#3CH

;
CLR     C
MOV     A,TM4HLD1
MOV     B,#64H
MOV     AB
MOV     TRMINIT,A
MOV     TRHOURE,B

;
CLR     C
MOV     A,TM4HLD2
MOV     B,#0AH
MOV     AB
CLR     C
ADD     A,TM4HLD3
ADD     A,TRMINIT
MOV     TRMINIT,A

;
MOV     A,TRHOURE
ADDC    A,#00H
MOV     TRHOURE,A

;
RET

;
SAVHLDD: CJNE    A,#08H,SAVHLDD
POP     ACC
MOV     KEYPARU,A
MOV     DIGIT1,#00000010B
MOV     DIGIT2,R4
MOV     DIGIT3,#00000010B

;
RET

;
SAVHLDD: POP     ACC
MOV     KEYPARD,A
MOV     DIGIT1,#00000010B
MOV     DIGIT2,R4
MOV     DIGIT3,#00000010B

;
RET

;
;*****
;      OUTPUT KENDALI DENGAN SET PEWAKTU      *
;*****
;
HLDCTRL: NOP
MOV     R1,#0AH
MOV     REGMODE,#00H
MOV     ZREGTIM,#06H

;
CLR     BITCURR
CLR     BITLIMT
CLR     BITTAG
CLR     BITTAGA
CLR     BITTAGU
CLR     BITTAGD

```



```

CLR      BITENDU
CLR      BITENDD
CLR      BITZSTP

;
SETB     BITZSTO

;
MOV      KEYHIG1,TP1HLD1
MOV      KEYHIG2,TP1HLD2
MOV      KEYHIG3,TP1HLD3
MOV      KEYLOW1,TP1HLD1
MOV      KEYLOW2,TP1HLD2
MOV      KEYLOW3,TP1HLD3

;
MOV      TRSECND,TM1HLD3
MOV      TRMINIT,TM1HLD2
MOV      TRHOURE,TM1HLD1

;
HLD SAMP:  NOP
CLR      BITOVR
SETB     ET1

;
MOV      TMOD,#15H
MOV      TLO,#00H
MOV      TH0,#00H
MOV      TL1,#00H
MOV      TH1,#00H

;
ORL      TCON,#50H

;
LCALL    DISPLAY

;
;*****
;  PEMERIKSA HASIL SAMPLING DAN TAMPILKAN  *
;*****
;
GETHOLD:  ANL      TCON,#0AFH
CLR      ET1

;
MOV      SAMLOW,TLO
MOV      SAMHIGH,TH0

;
CLR      C

;
MOV      A,SAMLOW
SUBB     A,#LMTUNDO
MOV      A,SAMHIGH
SUBB     A,#LMTUND1
JC       UDRANGE

;
MOV      A,#LMTOUT0
SUBB     A,SAMLOW
MOV      A,#LMTOUT1
SUBB     A,SAMHIGH
JC       UTRANGE

;
MOV      A,#LMTOVR0
SUBB     A,SAMLOW
MOV      A,#LMTOVR1
SUBB     A,SAMHIGH
JC       UVRANGE

```

```

;      CLR      BITBUZZ
;      SETB     P1.3

;      LJMP     KEEPON

; UDRANGE:      CLR      P1.3
;               SETB     BITBUZZ
;               LJMP     KEEPON

; UVRANGE:      CLR      P1.3
;               SETB     BITBUZZ
;               LJMP     KEEPON

; UTRANGE:      CLR      P1.3
;               SETB     BITBUZZ
;               CLR      P1.7
;               CLR      BITBURN

;      MOV      DIGIT1,#11111100B
;      MOV      DIGIT2,#01111100B
;      MOV      DIGIT3,#00011110B

;      LJMP     HLDSAMP

; KEEPON:      LCALL    HEX2BCD
;               LCALL    CONVERT

; HOLDOUT:      NOP
;               JB      BITTAG,OKEOKEO
;               JB      BITTAG,NEXTTAG

;      SETB     BITTAGA
;      CLR      C
;      MOV      A,TP1HLD3
;      SUBB     A,BCD3
;      MOV      A,TP1HLD2
;      SUBB     A,BCD2
;      MOV      A,TP1HLD1
;      SUBB     A,BCD1
;      JC      FIRSTDW

; FIRSTUP:      SETB     BITTAGU
;               CLR      BITTAGD
;               CLR      BITCURR
;               CLR      BITENDU
;               CLR      BITLIMT
;               LJMP     NEXTTAG

; FIRSTDW:      SETB     BITTAGD
;               CLR      BITTAGU
;               CLR      BITCURR
;               CLR      BITENDD
;               SETB     BITLIMT

; NEXTTAG:      NOP
;               JB      BITZSTO,TOTMTAG
;               DEC     ZREGTIM
;               MOV     A,ZREGTIM
;               JZ      STPBUZZ

```



```

        SETB   BITBUZZ
        CLR    P1.3

;TOTMTAG:      NOP
        LCALL  TIMETAG
        NOP
        LJMP   HLDSAMP

;STFBUZZ:      NOP
        SETB   BITZSTO
        CLR    BITBUZZ
        SETB   P1.3
        LJMP   TOTMTAG

;OKEOKEO:      NOP
        JB     BITZSTP,TOUTHLD
        DEC    ZREGTIM
        MOV    A,ZREGTIM
        JZ     BUZZSTP
        SETB   BITBUZZ
        CLR    P1.3

;TOUTHLD:      NOP
        LCALL  OUTHOLD
        NOP
        LJMP   HLDSAMP

;BUZZSTP:      NOP
        SETB   BITZSTP
        CLR    BITBUZZ
        SETB   P1.3
        LJMP   TOUTHLD

;
;*****
;  RUTIN KENDALI TEMPERATUR DENGAN SET WAKTU  *
;*****
;
;OUTHOLD:      NOP
        MOV    A,REGMODE
        CJNE   A,#00H,TOIHLD2
;
        CLR    C
        MOV    A,TP1HLD3
        SUBB   A,#01H
        MOV    A,TP1HLD2
        SUBB   A,#00H
        MOV    A,TP1HLD1
        SUBB   A,#00H
        JC     TOHOLD2
;
        LJMP   IONEHLD
;
;TOIHLD2:      NOP
        LJMP   IHLDTWO
;
;IONEHLD:      NOP
        CLR    C
        MOV    A,TP1HLD3
        SUBB   A,BCD3
        MOV    A,TP1HLD2

```

```

        CLR    BITENDD
        CLR    BITCURR
        SETB   BITLIMT
;
        RET
;
;.....
;
IHLDTWO:    NOP
            MOV    A, REGMODE
            CJNE   A, #01H, TOIHLD3
;
            CLR    C
            MOV    A, TP2HLD3
            SUBB   A, #01H
            MOV    A, TP2HLD2
            SUBB   A, #00H
            MOV    A, TP2HLD1
            SUBB   A, #00H
            JC     TOHOLD3
;
            LJMP   ITWOHLD
;
TOIHLD3:    NOP
            LJMP   IHLDTRI
;
ITWOHLD:    NOP
            CLR    C
            MOV    A, TP2HLD3
            SUBB   A, BCD3
            MOV    A, TP2HLD2
            SUBB   A, BCD2
            MOV    A, TP2HLD1
            SUBB   A, BCD1
            JC     HLDCOL2
;
            MOV    A, TRMINIT
            JZ     TM2MINH
;
TIM2HOT:    NOP
            DEC    TRSECND
            MOV    A, TRSECND
            JZ     TM2SECH
;
            SETB   BITBURN
            SETB   P1.7
;
            RET
;
HLDCOL2:    NOP
            MOV    A, TRMINIT
            JZ     TM2MINC
;
TIM2CLD:    NOP
            DEC    TRSECND
            MOV    A, TRSECND
            JZ     TM2SECC
;
            CLR    BITBURN
            CLR    P1.7

```



```

;      RET
;
; TM2SECH:      NOP
;               MOV      TRSECND, #3CH
;               DEC      TRMINIT
;               MOV      A, TRMINIT
;               JZ       TM2MINH
;
;               SETB     BITBURN
;               SETB     P1.7
;
;      RET
;
; TM2SECC:      NOP
;               MOV      TRSECND, #3CH
;               DEC      TRMINIT
;               MOV      A, TRMINIT
;               JZ       TM2MINC
;
;               CLR      BITBURN
;               CLR      P1.7
;
;      RET
;
; TM2MINH:      NOP
;               MOV      TRSECND, #3CH
;               MOV      TRMINIT, #0FFH
;               MOV      A, TRHOURE
;               JZ       TOHOLD3
;               DEC      TRHOURE
;
;               SETB     BITBURN
;               SETB     P1.7
;
;      RET
;
; TM2MINC:      NOP
;               MOV      TRSECND, #3CH
;               MOV      TRMINIT, #0FFH
;               MOV      A, TRHOURE
;               JZ       TOHOLD3
;               DEC      TRHOURE
;
;               CLR      BITBURN
;               CLR      P1.7
;
;      RET
;
; TOHOLD3:      NOP
;               MOV      R1, #0AH
;               MOV      REGMODE, #02H
;               MOV      ZREGTIM, #06H
;               SETB     P1.3
;               CLR      BITBUZZ
;               CLR      BITTAG
;               CLR      BITTAGU
;               CLR      BITTAGD
;               CLR      BITZSTP
;               CLR      BITZSTO

```

```

SUBB    A,BCD3
MOV     A,TF3HLD2
SUBB    A,BCD2
MOV     A,TF3HLD1
SUBB    A,BCD1
JC      HLDCOL3

;
MOV     A,TRMINIT
JZ      TM3MINH

;
TIM3HOT:    NOP
DEC      TRSECND
MOV     A,TRSECND
JZ      TM3SECH

;
SETB    BITBURN
SETB    P1.7

;
RET

;
HLDCOL3:    NOP
MOV     A,TRMINIT
JZ      TM3MINC

;
TIM3CLD:    NOP
DEC      TRSECND
MOV     A,TRSECND
JZ      TM3SECC

;
CLR     BITBURN
CLR     P1.7

;
RET

;
TM3SECH:    NOP
MOV     TRSECND,#3CH
DEC     TRMINIT
MOV     A,TRMINIT
JZ      TM3MINH

;
SETB    BITBURN
SETB    P1.7

;
RET

;
TM3SECC:    NOP
MOV     TRSECND,#3CH
DEC     TRMINIT
MOV     A,TRMINIT
JZ      TM3MINC

;
CLR     BITBURN
CLR     P1.7

;
RET

;
TM3MINH:    NOP
MOV     TRSECND,#3CH
MOV     TRMINIT,#0FFH
MOV     A,TRHOUR

```



```

JZ      TOHOLD4
DEC     TRHOURE

;
SETB    BITBURN
SETB    P1.7

;
RET

;
TM3MINC: NOP
MOV     TRSECND, #3CH
MOV     TRMINIT, #0FFH
MOV     A, TRHOURE
JZ      TOHOLD4
DEC     TRHOURE

;
CLR     BITBURN
CLR     P1.7

;
RET

;
TOHOLD4: NOP
MOV     R1, #0AH
MOV     REGMODE, #03H
MOV     ZREGTIM, #06H
SETB    P1.3
CLR     BITBUZZ
CLR     BITTAG
CLR     BITTAGU
CLR     BITTAGD
CLR     BITZSTP
CLR     BITZSTO

;
MOV     KEYHIG1, TP4HLD1
MOV     KEYHIG2, TP4HLD2
MOV     KEYHIG3, TP4HLD3
MOV     KEYLOW1, TP4HLD1
MOV     KEYLOW2, TP4HLD2
MOV     KEYLOW3, TP4HLD3

;
MOV     TRSECND, TM4HLD3
MOV     TRMINIT, TM4HLD2
MOV     TRHOURE, TM4HLD1

;
CLR     C
MOV     A, TP4HLD3
SUBB    A, TP3HLD3
MOV     A, TP4HLD2
SUBB    A, TP3HLD2
MOV     A, TP4HLD1
SUBB    A, TP3HLD1
JC      FORTHDW

;
SETB    BITTAGU
CLR     BITTAGD
CLR     BITCURR
CLR     BITENDU
CLR     BITLIMT

;
RET
;

```

```

FORTHWD:      SETB   BITTAGD
               CLR    BITTAGU
               CLR    BITCURR
               CLR    BITENDD
               SETB   BITLIMT
;
               RET
;
*****
;
IHLDFOR:      NOP
               MOV    A, REGMODE
               CJNE   A, #03H, TOENDNO
;
               CLR    C
               MOV    A, TP4HLD3
               SUBB   A, #01H
               MOV    A, TP4HLD2
               SUBB   A, #00H
               MOV    A, TP4HLD1
               SUBB   A, #00H
               JC     CONORNO
;
               LJMP   IFORHLD
;
TOENDNO:      NOP
               LJMP   SECOEND
;
IFORHLD:      NOP
               CLR    C
               MOV    A, TP4HLD3
               SUBB   A, BCD3
               MOV    A, TP4HLD2
               SUBB   A, BCD2
               MOV    A, TP4HLD1
               SUBB   A, BCD1
               JC     HLDCOL4
;
               MOV    A, TRMINIT
               JZ     TM4MINH
;
TIM4HOT:      NOP
               DEC    TRSECND
               MOV    A, TRSECND
               JZ     TM4SECH
;
               SETB   BITBURN
               SETB   P1.7
;
               RET
;
HLDCOL4:      NOP
               MOV    A, TRMINIT
               JZ     TM4MINC
;
TIM4CLD:      NOP
               DEC    TRSECND
               MOV    A, TRSECND
               JZ     TM4SECC
;

```

```

        CLR    BITBURN
        CLR    P1.7

;
        RET

;
TM4SECH:    NOP
        MOV    TRSECND, #3CH
        DEC    TRMINIT
        MOV    A, TRMINIT
        JZ     TM4MINH

;
        SETB   BITBURN
        SETB   P1.7

;
        RET

;
TM4SECC:    NOP
        MOV    TRSECND, #3CH
        DEC    TRMINIT
        MOV    A, TRMINIT
        JZ     TM4MINC

;
        CLR    BITBURN
        CLR    P1.7

;
        RET

;
TM4MINH:    NOP
        MOV    TRSECND, #3CH
        MOV    TRMINIT, #0FFH
        MOV    A, TRHOURE
        JZ     CONORNO
        DEC    TRHOURE

;
        SETB   BITBURN
        SETB   P1.7

;
        RET

;
TM4MINC:    NOP
        MOV    TRSECND, #3CH
        MOV    TRMINIT, #0FFH
        MOV    A, TRHOURE
        JZ     CONORNO
        DEC    TRHOURE

;
        CLR    BITBURN
        CLR    P1.7

;
        RET

;
CONORNO:    NOP
        MOV    REGMODE, #04H
        MOV    ZREGTIM, #06H
        SETB   BITTAG
        CLR    BITZSTP
        CLR    BITZSTO
        CLR    BITBURN
        CLR    P1.7
        SETB   BITBUZZ

```



```

        CLR    P1.3
;
        RET
;
;*****
;
SECOEND:    NOP
;
        MOV    DIGIT1,#11000110B
        MOV    DIGIT2,#00011110B
        MOV    DIGIT3,#11001110B
;
        MOV    A,ZREGTIM
        JZ     ENDDLDC
        SETB   BITBUZZ
        CLR    P1.3
;
        RET
;
ENDHLCDC:  NOP
        CLR    BITBUZZ
        SETB   P1.3
;
        SETB   OV
        CLR    F0
        CLR    C
;
        POP    ACC
        POP    ACC
        CLR    A
        PUSH   ACC
        CLR    A
        PUSH   ACC
;
        RET
;
;*****
; RUTIN Mencari Besar Temperatur Yang Ditahan *
;*****
;
TIMETAG:   NOP
        CLR    C
        MOV    A,KEYLOW3
        SUBB   A,#01H
        MOV    A,KEYLOW2
        SUBB   A,#00H
        MOV    A,KEYLOW1
        SUBB   A,#00H
        JC     TOTAGUD
;
        JB     BITLIMT,RTCNWOD
        LJMP   UPCTRL
;
RTCNWOD:   LJMP   DOWNCTR
;
UPCTRL:    NOP
        JB     BITENDU,OKUPPAR
;
        CLR    C
        MOV    A,SAMLOW

```

```

        ADDC    A, #0AH
        MOV     SAMLOW, A
        MOV     A, SAMHIGH
        ADDC    A, #00H
        MOV     SAMHIGH, A

;
        LCALL  HEX2BCD

;
        CLR     C
        MOV     A, KEYHIG3
        SUBB    A, BCD3
        MOV     A, KEYHIG2
        SUBB    A, BCD2
        MOV     A, KEYHIG1
        SUBB    A, BCD1
        JC      NAIPARA

;
        LJMP   HLDBURN

;
NAIPARA:    NOP
        SETB   BITENDU

;
        MOV     SAMLOW, TL0
        MOV     SAMHIGH, TH0

;
        LCALL  HEX2BCD

;
OKUPPAR:    NOP
        SETB   BITTAGU
        CLR     BITLIMT
        CLR     C
        MOV     A, KEYHIG3
        SUBB    A, BCD3
        MOV     A, KEYHIG2
        SUBB    A, BCD2
        MOV     A, KEYHIG1
        SUBB    A, BCD1
        JC      DOWNINT

;
        JB      BITCURR, DWNFROM

;
        DJNZ    R1, LOOCDLH
        LJMP    BURNINT

;
LOOCDLH:    NOP
        LJMP    HLDCCOOL

;
DWNFROM:    NOP
        DJNZ    R1, NRUBDLH
        LJMP    COOLINT

;
NRUBDLH:    NOP
        LJMP    HLDBURN

;
TOTAGUD:    NOP
        LJMP    TAGUPDW

;
UPINIT:     NOP
        MOV     R1, #0AH
        CLR     BITCURR

```

```

        SETB    BITLIMT
        CLR     C
        MOV     A, BCD3
        SUBB    A, KEYLOW3
        MOV     A, BCD2
        SUBB    A, KEYLOW2
        MOV     A, BCD1
        SUBB    A, KEYLOW1
        JC      UPINIT
;
        JB      BITCURR, UPFROM
;
        DJNZ    R1, HLDCCOOL
        LJMP    BURNINT
;
UPFROM:    NOP
        DJNZ    R1, HLDBURN
        SJMP    COOLINT
        NOP
COOLINT:   NOP
        CLR     BITCURR
        MOV     R1, #0AH
        CLR     BITBURN
HLDCCOOL:  CLR     P1.7
;
        RET
;
BURNINT:   NOP
        SETB    BITCURR
        JB      BITLIMT, DWNBURN
UPBURN:    MOV     R1, KEYPARU
        MOV     A, R1
        JZ      COOLINT
        SJMP    HLDBURN
DWNBURN:   NOP
        MOV     R1, KEYPARU
        MOV     A, R1
        JZ      COOLINT
HLDBURN:   SETB    BITBURN
        SETB    P1.7
;
        RET
;
TAGUPDW:   NOP
        CLR     BITBUZZ
        SETB    P1.3
        SETB    BITTAG
        MOV     ZREGTIM, #0BH
;
        RET
;
;*****
;*****
; PROGRAM MODE PENGENDALIAN MAKSIMUM - MINIMUM *
;*****
;
KEYPAD:    NOP
;
        MOV     REGMODE, #00H
        MOV     KEYCURR, #00H

```



```

      POP    ACC
      LJMP   MANIPUL

; ALLCOL:    LCALL  DISPKEY
      MOV    R1, #COLOM2

; NEXTCOL:   MOV    P1, @R1
      LCALL  DEBOUNC
      LJMP   INKEY

;
; *****
;   RUTIN ANALISA HASIL PENEKANAN KEYPAD   *
; *****
;
; MANIPUL:   MOV    A, KEYCURR
      CJNE   A, #23H, EIGHT
      MOV    A, #09H
      MOV    R4, #11110110B
      LCALL  SAVEBCD
      LJMP   ALLCOL

; EIGHT:     MOV    A, KEYCURR
      CJNE   A, #13H, SEVEN
      MOV    A, #08H
      MOV    R4, #11111110B
      LCALL  SAVEBCD
      LJMP   ALLCOL

; SEVEN:     MOV    A, KEYCURR
      CJNE   A, #03H, SIX
      MOV    A, #07H
      MOV    R4, #11100000B
      LCALL  SAVEBCD
      LJMP   ALLCOL

; SIX:       MOV    A, KEYCURR
      CJNE   A, #22H, FIVE
      MOV    A, #06H
      MOV    R4, #10111110B
      LCALL  SAVEBCD
      LJMP   ALLCOL

; FIVE:      MOV    A, KEYCURR
      CJNE   A, #12H, FOUR
      MOV    A, #05H
      MOV    R4, #10110110B
      LCALL  SAVEBCD
      LJMP   ALLCOL

; FOUR:      MOV    A, KEYCURR
      CJNE   A, #02H, THREE
      MOV    A, #04H
      MOV    R4, #01100110B
      LCALL  SAVEBCD
      LJMP   ALLCOL

; THREE:     MOV    A, KEYCURR
      CJNE   A, #21H, TWO
      MOV    A, #03H
      MOV    R4, #11110010B

```

```

; POP ACC
;
MOV DIGIT1, #11001110B
MOV DIGIT2, #01111010B
MOV DIGIT3, #00111010B
;
LCALL DISPLAY
LCALL DISPLAY
;
MOV DIGIT1, #00000010B
MOV DIGIT2, #11111100B
MOV DIGIT3, #00000010B
;
LJMP ALLCOL
;
SHOWACC: POP ACC
;
MOV DIGIT1, #11101110B
MOV DIGIT2, #10011100B
MOV DIGIT3, #10011100B
;
LCALL DISPLAY
LCALL DISPLAY
;
LJMP CONTROL
;
;*****
; RUTIN MENYIMPAN NILAI PENEKANAN KEYPAD *
;*****
;
SAVEBCD: PUSH ACC
MOV A, REGMODE
;
SAVELOW: CJNE A, #00H, SAVEHIG
POP ACC
MOV KEYLOW1, KEYLOW2
MOV KEYLOW2, KEYLOW3
MOV KEYLOW3, A
MOV DIGIT1, DIGIT2
MOV DIGIT2, DIGIT3
MOV DIGIT3, R4
;
RET
;
SAVEHIG: CJNE A, #01H, SAVPARU
POP ACC
MOV KEYHIG1, KEYHIG2
MOV KEYHIG2, KEYHIG3
MOV KEYHIG3, A
MOV DIGIT1, DIGIT2
MOV DIGIT2, DIGIT3
MOV DIGIT3, R4
;
RET
;
SAVPARU: CJNE A, #02H, SAVPARD
POP ACC
MOV KEYPARU, A
MOV DIGIT1, #00000010B
MOV DIGIT2, R4

```



```

        MOV     DIGIT3,#00000010B
;
        RET
;
SAVPARD:    POP     ACC
        MOV     KEYPARD,A
        MOV     DIGIT1,#00000010B
        MOV     DIGIT2,R4
        MOV     DIGIT3,#00000010B
;
        RET
;
;*****
;      RUTIN TAMPILAN PENEKANAN TOMBOL KEYPAD      *
;*****
;:      MOV     R6,#20H
;
NEXLOP:    MOV     R7,#03H
           MOV     R0,#DIGIT1
           MOV     A,#11111110B
;
NEXBIT:    CLR     EX0
           SETB    ES
           MOV     SBUF,@R0
WATYES:    JNB     BITYES,WATYES
           CLR     BITYES
           CLR     ES
           SETB    EX0
;
           CLR     A.7
           SETB    A.3
           MOV     P1,A
;
           LCALL   DELAY
;
           DJNZ    R7,NOALL
           LJMP    THAALL
;
NOALL:     INC     R0
           RL      A
           SETB    A.0
           LJMP    NEXBIT
;
THAALL:    DJNZ    R6,NEXLOP
;
           RET
;
;*****
;      RUTIN DELAY DEBOUNCE      *
;*****
;
DEBOUNC:   MOV     R2,#2FH
LOOOP2:    MOV     R3,#0FFH
LOOOP1:    DJNZ    R3,LOOOP1
           DJNZ    R2,LOOOP2
;
           RET
;
;*****
;      RUTIN KENDALI OUTPUT SESUAI INPUT KEYPAD      *
;*****

```



```

;*****
;
; CONTROL:      CLR      BITCURR
;               CLR      BITLIMT
;               CLR      BITENDU
;               CLR      BITENDD
;               SETB     BITBURN
;               SETB     P1.7
;
;               MOV      R1, #0AH
;
; CONSAMP:      CLR      BITOVR
;               SETB     ET1
;
;               MOV      TMOD, #15H
;               MOV      TLO, #00H
;               MOV      TH0, #00H
;               MOV      TL1, #00H
;               MOV      TH1, #00H
;
;               ORL      TCON, #50H
;
;               LCALL    DISPLAY
;
;*****
; PEMERIKSA HASIL SAMPLING DAN TAMPILKAN *
;*****
;
; GETSAMP:      ANL      TCON, #0AFH
;               CLR      ET1
;
;               MOV      SAMLOW, TLO
;               MOV      SAMHIGH, TH0
;
;               CLR      C
;
;               MOV      A, SAMLOW
;               SUBB     A, #LMTUNDO
;               MOV      A, SAMHIGH
;               SUBB     A, #LMTUND1
;               JC       UNRANGE
;
;               MOV      A, #LMTOUT0
;               SUBB     A, SAMLOW
;               MOV      A, #LMTOUT1
;               SUBB     A, SAMHIGH
;               JC       OTRANGE
;
;               MOV      A, #LMTOVR0
;               SUBB     A, SAMLOW
;               MOV      A, #LMTOVR1
;               SUBB     A, SAMHIGH
;               JC       OVRANGE
;
;               CLR      BITBUZZ
;               SETB     P1.3
;
;               LJMP     CARRYON
;
; UNRANGE:      CLR      P1.3

```

```

        LCALL DEBOUNC
        LCALL DEBOUNC
        LCALL DEBOUNC
        LCALL DEBOUNC
        LCALL DEBOUNC
        LCALL DEBOUNC
        LCALL DEBOUNC
        LCALL DEBOUNC
        LCALL DEBOUNC

;      INC      BISELEC
      MOV      A,BISELEC
      CJNE     A,#01H,MAYCTRL

;      SETB     OV
      CLR      C
      CLR      F0
      JMP      TOBEGIN

; MAYCTRL:      CJNE     A,#02H,MAYHOLD
      SETB     C
      CLR      OV
      CLR      F0
      JMP      TOBEGIN

; MAYHOLD:      CJNE     A,#03H,MAYLOOP
      SETB     F0
      CLR      OV
      CLR      C
      JMP      TOBEGIN

; MAYLOOP:      MOV      BISELEC,#01H
      SETB     OV
      CLR      C
      CLR      F0

; TOBEGIN:      POP      ACC
      POP      ACC
      CLR      A
      PUSH     ACC
      CLR      A
      PUSH     ACC

;      RETI

; *****
; PROGRAM AKIBAT INTERUPSI TIMER 1 (T1) *
; *****
;
TC1INTR:      NOP
      ANL      TCON,#0AFH
      CPL      BITOVR
      JB       BITOVR,SECTIME

;      RETI

;
SECTIME:      MOV      TL1,#60H
      MOV      TH1,#79H
      ORL      TCON,#50H
;

```



```

;
MOV     SAMLOW,TLO
MOV     SAMHIGH,TH0
;
LCALL   HEX2BCD
;
OKPARDW: CLR     C
MOV     A,BCD3
SUBB    A,KEYLOW3
MOV     A,BCD2
SUBB    A,KEYLOW2
MOV     A,BCD1
SUBB    A,KEYLOW1
JC      INITUP
;
JB      BITCURR, FROMUP
;
DJNZ    R1, COOLING
LJMP    INTBURN
;
FROMUP:  DJNZ    R1, BURNING
SJMP    INTCOOL
NOP
;
INTCOOL: NOP
CLR     BITCURR
MOV     R1,#0AH
COOLING: CLR     BITBURN
CLR     P1.7
LJMP    CONSAMP
;
INTBURN: NOP
SETB    BITCURR
JB      BITLIMT, BURNDWN
BURNUP: MOV     R1,KEYPARU
MOV     A,R1
JZ      INTCOOL
SJMP    BURNING
BURNDWN: NOP
MOV     R1,KEYPARD
MOV     A,R1
JZ      INTCOOL
BURNING: SETB    BITBURN
SETB    P1.7
LJMP    CONSAMP
;
;*****
;  PROGRAM AKIBAT INTERUPSI EKSTERNAL (INT0)  *
;*****
;
EXT0INT: CLR     EA
ANL     TCON,#0AFH
;
SETB    P1.0
SETB    P1.1
SETB    P1.2
;
LCALL   DEBOUNC
LCALL   DEBOUNC
LCALL   DEBOUNC

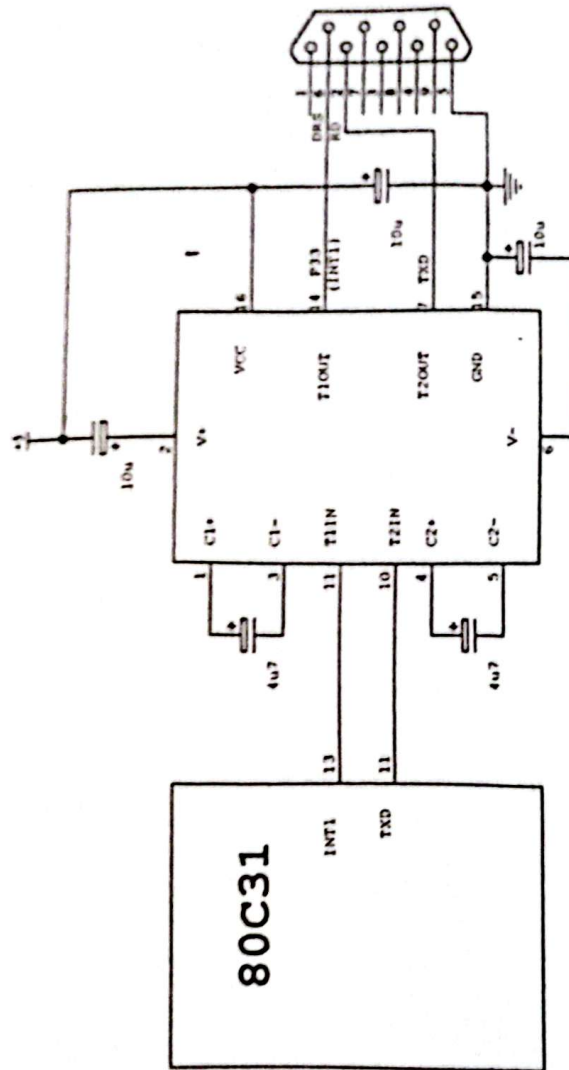
```



```

      RETI
;.....
;PROGRAM AKIBAT INTERUPSI TRANSMISI SERIAL END *
;.....
;
;      CLR      TI
;      SETB     BITYES
;
;      RETI
;.....
;      END      ;      END      ;      END      *
;.....
;      END

```



Title		KANGKAT IM		SERIAL	
Size	Number	Revision		Sheet of	
Ad				Drawn By	
Date	24 Oct 1985			TUES KIRITA	
File	C:\PWS\CHIPS\80C31				