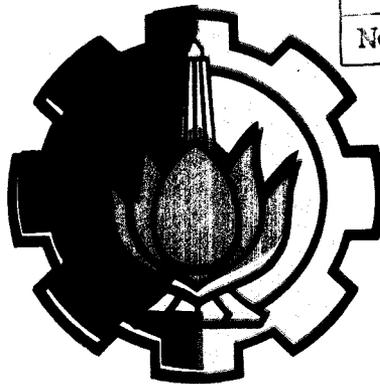


3100096007258

**PEMANTAUAN PEMAKAIAN AIR PADA METER
AIR PELANGGAN PDAM YANG DIKOMUNIKASIKAN
KE IBM PC SEBAGAI SENTRAL PEMANTAU
MELALUI JALUR BAND VHF**

PERPUSTAKAAN ITS	
Tgl. Terima	04 MAY 1994
Terima Dari	TA
No. Agenda Crp.	2003/B

RSE
621.8916
R02
P-1
1994



Oleh :

Mokhamad Abdul Rozaq
NRP. 2882201115

**JURUSAN TEKNIK ELEKTRO
FAKULTAS TEKNOLOGI INDUSTRI
INSTITUT TEKNOLOGI SEPULUH NOPEMBER
SURABAYA
1994**

**PEMANTAUAN PEMAKAIAN AIR PADA METER
AIR PELANGGAN PDAM YANG DIKOMUNIKASIKAN
KE IBM PC SEBAGAI SENTRAL PEMANTAU
MELALUI JALUR BAND VHF**

TUGAS AKHIR

**Diajukan Guna Memenuhi Sebagian Persyaratan Untuk
Memperoleh Gelar Sarjana Teknik Elektro
Pada
Bidang Studi Elektronika
Jurusan Teknik Elektro
Fakultas Teknologi Industri
Institut Teknologi Sepuluh Nopember
Surabaya**

**Mengetahui / Menyetujui
Dosen Pembimbing**



Ir. Karyadi, Msc.

**SURABAYA
PEBRUARI, 1994**

ABSTRAK

Selama ini untuk mendapatkan data pemakaian air pelanggan PDAM diperlukan petugas pencatat meter air. Dengan adanya kemajuan di bidang elektronika dan telekomunikasi proses untuk mendapatkan data meter air pelanggan dapat digantikan oleh perangkat elektronik.

Sistem pemantauan meter air dengan peralatan elektronik ini terdiri dari unit sentral pemantau dan unit meter air pelanggan. Unit sentral bertugas mengambil data dari unit meter air pelanggan dengan menggunakan transmisi pada band VHF. Unit meter air pelanggan mencatat dan mendisplaykan jumlah volume air yang telah disalurkan serta mengirimkan data volume air saat ada sinyal panggilan dari unit sentral.

Unit meter air pelanggan diperlengkapi kode alamat atau password. Untuk mengakses unit meter air, unit pemantau harus mengirimkan kode alamat sesuai dengan meter air yang dituju. Pada keadaan normal unit meter air adalah siap menerima data (ready). Bila meter air menerima kode alamat yang sesuai dengan kode alamat yang dimiliki maka dengan segera meter air mengirimkan data volume air, setelah data dikirimkan unit meter air pelanggan akan berada kembali pada kondisi siap menerima data. Sedangkan jika alamat yang dikirimkan tidak cocok maka unit meter air tidak memberi tanggapan apapun. Dengan cara pengkodean alamat ini unit pemantau dapat mengakses banyak meter air secara bergantian hanya dengan menggunakan satu jalur frekuensi.

KATA PENGANTAR

Atas berkat rahmat Allah yang maha pengasih dan penyayang, maka penulis berhasil menyelesaikan Tugas Akhir yang berjudul :

PEMANTAUAN PEMAKAIAN AIR PADA METER AIR PELANGGAN PDAM YANG DIKOMUNIKASIKAN KE IBM PC SEBAGAI SENTRAL PEMANTAU MELALUI JALUR BAND VHF

Tugas Akhir ini merupakan salah satu syarat yang harus ditempuh untuk meraih gelar kesarjanaan di Jurusan Teknik Elektro, Fakultas Teknologi Industri, Institut Teknologi Sepuluh Nopember Surabaya.

Dalam mengerjakan Tugas Akhir ini penulis banyak mendapat bantuan dan bimbingan dari berbagai pihak. Pada kesempatan ini penulis menyampaikan ucapan terima kasih kepada :

- Ir. Karyadi, Msc., selaku dosen pembimbing yang telah banyak memberikan bimbingan dan pengarahan kepada penulis dalam perencanaan dan pembuatan alat serta penyusunan naskah Tugas Akhir ini.
- Ir. Soetikno, selaku Koordinator Bidang Studi Elektronika Jurusan Teknik Elektro, Fakultas Teknologi Industri ITS yang telah memberikan persetujuan kepada penulis untuk melaksanakan Tugas Akhir ini.



- Ir. Murdi Asmoroadji, selaku dosen wali yang telah banyak memberikan sumbangan pemikiran.
- Dr. Ir. Moch. Salehudin, M.Eng.Sc., selaku Ketua Jurusan Teknik Elektro, Fakultas Teknologi Industri ITS, yang telah memberikan persetujuannya kepada penulis untuk melaksanakan Tugas Akhir ini.
- Seluruh staf pengajar dan administrasi Jurusan Teknik Elektro FTI ITS, yang telah membantu kelancaran pelaksanaan Tugas Akhir ini.
- Seluruh rekan-rekan mahasiswa Jurusan Teknik Elektro FTI ITS dan semua pihak yang telah turut membantu baik secara langsung maupun tidak langsung

Akhir kata, penulis berharap semoga segala sesuatu yang telah dihasilkan dalam pembuatan Tugas Akhir ini dapat bermanfaat bagi kemajuan ilmu pengetahuan dan kesejahteraan umat manusia.

Surabaya, Pebruari 1994

Penulis

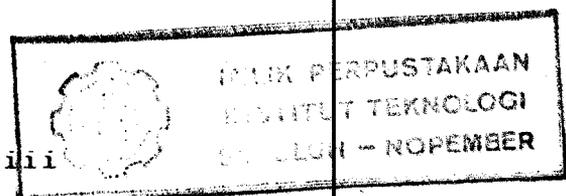
DAFTAR ISI

	<i>Halaman</i>
JUDUL	i
LEMBAR PERSETUJUAN	ii
ABSTRAK	iii
KATA PENGANTAR	iv
DAFTAR ISI	vi
DAFTAR GAMBAR	ix
DAFTAR TABEL	xi
BAB I PENDAHULUAN	1
I.1. LATAR BELAKANG	1
I.2. PERMASALAHAN	2
I.3. PEMBatasan MASALAH	3
I.3. TUJUAN	3
I.4. METODOLOGI	4
I.5. SISTEMATIKA PENULISAN	4
I.5. RELEVANSI	5
BAB II TEORI PENUNJANG	6
II.1. MIKROKONTROLER 8031	6
II.1.1 ARSITEKTUR 8031	7
II.1.2. KONFIGURASI DAN FUNGSI DARI PIN-PIN 8031	9
II.1.3. PERANGKAT KERAS CPU	12
II.1.4. TIMER / COUNTER	16



II.1.5. SISTEM INTERUPSI	17
II.1.6. PORT SERIAL	18
II.2. ROM DAN RAM	20
II.3. ALIRAN AIR	21
II.3.1. PERSAMAAAN KONTINUITAS	21
II.3.2. SENSOR ALIRAN	22
II.4. KOMUNIKASI SERIAL PADA IBM PC	28
II.5. SPEKTRUM GELOMBANG RADIO	30
II.6. MODULASI	32
II.7. MODULASI FREKUENSI (FM)	33
II.7.1. DIRECT FM	38
II.7.2. INDIRECT FM	40
II.7.3. DEMODULASI SINYAL FM	40
II.8. XR-2211 FSK DEMODULATOR	42
II.9. XR-2206 FSK MODULATOR	44
BAB III. PERENCANAAN	46
III.1. PERENCANAAN PERANGKAT KERAS	46
III.1.1. BLOK DIAGRAM	46
III.1.2. RANGKAIAN CLOCK	49
III.1.3. RANGKAIAN RESET	49
III.1.4. RANGKAIAN BUFFER	50
III.1.5. RANGKAIAN DECODER	52
III.1.6. RANGKAIAN DISPLAY	53
III.1.7. RANGKAIAN MEMORI PROGRAM	54
III.1.8. RANGKAIAN SENSOR ALIRAN AIR	55
III.1.9. RANGKAIAN SERIAL INTERFACE RS-232-C	59

III.1.1.	RANGKAIAN ALARM60
III.1.11.	RANGKAIAN MODULATOR FSK61
III.1.12.	RANGKAIAN DEMODULATOR FSK62
III.1.13.	RANGKAIAN PEMANCAR (Tx)64
III.1.14.	RANGKAIAN PENERIMA (RX)67
III.1.15.	RANGKAIAN PEMILIH PEMANCAR / PENERIMA69
III.1.16.	SWITCH PEMILIH70
III.2.	PERENCANAAN PERANGKAT LUNAK71
III.2.1.	PERANGKAT LUNAK UNIT SENTRAL PENANTAU71
III.2.2.	PERANGKAT LUNAK 803173
BAB IV	PENGUJIAN DAN PENGUKURAN ALAT79
IV.1.	PENGUJIAN PROGRAM KOMUNIKASI79
IV.2.	PENGUJIAN RANGKAIAN MODULATOR DAN DEMODULATOR FSK	.80
IV.3.	PENGUJIAN RANGKAIAN PEMANCAR DAN PENERIMA81
IV.4.	PENGUJIAN RANGKAIAN SENSOR82
IV.5.	PENGUJIAN SISTEM SECARA KESELURUHAN82
IV.6.	PENGUKURAN ALIRAN AIR83
BAB V	PENUTUP86
V.1.	KESIMPULAN86
V.2.	SARAN-SARAM87
DAFTAR PUSTAKA	88
LAMPIRAN	89



DAFTAR GAMBAR

	<i>Halaman</i>
2.1 BLOK DIAGRAM STRUKTUR INTERNAL 8031	8
2.2 KONFIGURASI PIN-PIN 8031	11
2.3. GARIS ALIRAN ARUS	22
2.4. SENSOR ALIRAN PADA METER AIR	23
2.5. PIRINGAN SIRIP	24
2.6. MODULASI FREKUENSI	34
2.7. PENGARUH NOISE TERHADAP SINYAL	37
2.8. DIRECT FM MODULATOR DENGAN DIODA VARACTOR	39
2.9. SLOPE DETECTOR	41
2.10 KARAKTERISTIK SLOPE DETEKTOR	42
2.11 DIAGRAM BLOK XR-2211	43
2.12 DIAGRAM BLOK XR-2206	45
3.1 DIAGRAM BLOK SISTEM	47
3.2 RANGKAIAN RESET, CLOCK, DAN BUFFER	50
3.3 DECODER ALAMAT 8031	52
3.4 PEMETAAN MEMORI EXTERNAL 8031	52
3.5 RANGKAIAN DISPLAY	53
3.6 MEMORI PROGRAM	55
3.7 SENSOR VOLUME AIR	58
3.8 RANGKAIAN SERIAL INTERFACE	59
3.9. RANGKAIAN ALARM	60
3.10 RANGKAIAN MODULATOR FSK	61
3.11 RANGKAIAN DEMODULATOR FSK	63

3.12	RANGKAIAN PEMANCAR	65
3.13	RANGKAIAN EKIVALEN OSILATOR	66
3.14	RANGKAIAN PENERIMA	68
3.15	RANGKAIAN PEMILIH Rx / Tx	69
3.16	FLOW CHART PROGRAM UTAMA PADA IBM PC	77
3.17	FLOW CHART PROGRAM PADA 8031	78

DAFTAR TABEL

	<i>Halaman</i>
2.1 ALAMAT AWAL DARI INTERRUPT SERVICE ROUTINE	18
4.1 HASIL PENGUKURAN FREKUENSI MODULATOR FSK	80
4.2 PENGUKURAN VOLUME AIR UNTUK SETIAP 5 LITER AIR ...	84

BAB I

PENDAHULUAN

I.1. LATAR BELAKANG

Dengan semakin meningkatnya jumlah penduduk dalam suatu perkotaan akan diikuti dengan kebutuhan air bersih. Ini dikarenakan air bersih merupakan kebutuhan yang sangat vital bagi manusia. Di kota-kota besar pada umumnya air tanah sudah tidak layak untuk dipakai. Untuk itu diperlukan jaringan pendistribusian air ke rumah-rumah penduduk. Jaringan pendistribusian air minum ini di Indonesia untuk setiap daerah tingkat II dikelola oleh Perusahaan Daerah Air Minum atau PDAM.

Selama ini untuk mendapatkan data pemakaian air pelanggan diperlukan petugas pencatat meter air. Petugas ini mendatangi rumah pelanggan satu persatu untuk mencatat besar pemakaian air oleh pelanggan. Proses pencatatan yang dilakukan oleh petugas pencatat ini memerlukan waktu yang cukup lama. Ini disebabkan karena para petugas harus mendatangi rumah pelanggan. Semakin banyak petugas pencatat maka semakin cepat pula waktu yang diperlukan untuk mencatat seluruh pelanggan PDAM tetapi ini adalah kurang efisien. Pencatatan data meter air oleh petugas ini juga tidak terbebas dari kesalahan. Kesalahan pencatatan ini dikarenakan adanya unsur kesengajaan atau pun kelengahan petugas sebagai manusia.

Adanya kemajuan dibidang elektronika dan telekomunikasi menyebabkan proses pengambilan data dari jarak jauh tanpa pengawasan langsung oleh manusia telah berkembang pesat. Proses pengambilan data tersebut dikenal sebagai remote sensing atau telemetry. Data yang diambil kemudian dapat disimpan atau pun diproses lebih lanjut. Jika data tersebut ditransmisikan melalui kabel serat optik proses ini disebut fiber-optic telemetry sedangkan bila ditransmisikan lewat radio dikenal sebagai radio telemetry.

Proses monitoring pemakaian air pelanggan PDAM yang selama ini dilakukan petugas pencatat dapat digantikan oleh sistem radio telemetry. Dalam sistem telemetry, data pemakaian air minum dapat dipantau dari jarak jauh setiap saat dengan cepat. Cepatnya perolehan data pemakaian air ini akan membawa banyak keuntungan diantaranya informasi mengenai distribusi air, aktifitas pelanggan dalam memakai air minum, kebocoran pipa serta usaha pencurian air dari pipa distribusi dapat diketahui secara cepat. Khusus untuk kebocoran pipa dan usaha pencurian dapat diketahui dengan cara melihat data selisih antara air yang disalurkan pada pipa utama dengan total air yang disalurkan melalui meter air.

I.2. PERMASALAHAN

Di Indonesia khususnya pengamatan meter air pelanggan PDAM masih dilakukan oleh manusia. Permasalahan dalam Tugas Akhir ini adalah bagaimana melakukan perencanaan dan pembuatan alat untuk menggantikan peranan manusia dalam

pencatatan meter air pelanggan setiap bulannya.

I.3. PEMBATASAN MASALAH

Untuk mewujudkan sistem pemantau yang melibatkan puluhan ribu pelanggan, diperlukan banyak pertimbangan diantaranya bandwidth yang dipakai, alokasi frekuensi, hardware, kehandalan peralatan, serta biaya yang harus dikeluarkan. Pertimbangan tersebut perlu dilakukan untuk mengoptimalkan sistem yang dirancang. Pada Tugas Akhir ini dibahas sistem pemantau meter air yang hanya menggunakan satu unit sentral pemantau. Unit sentral direncanakan dapat memantau 1000 unit pelanggan pada satu jalur frekuensi. Unit meter air pelanggan dan unit sentral masing-masing berjumlah satu unit. Unit pemancar dan penerima dibuat bekerja pada frekuensi 110 MHz dengan daya output kecil.

I.4. TUJUAN

Tujuan dari Tugas Akhir ini adalah merencanakan dan membuat sistem pemantauan meter air. Untuk merealisasikan sistem pemantauan meter ini diperlukan unit sentral pemantau dan unit meter air pelanggan. Unit sentral bertugas mengambil data dari meter air pelanggan. Unit meter air pelanggan mendisplaykan besar volume air yang telah disalurkan dan mengirimkan data volume air saat ada sinyal panggilan dari unit sentral.

I.5. METODOLOGI

Pertama yang dilakukan adalah studi literatur mengenai segala sesuatu yang berhubungan dengan Tugas Akhir ini, terutama mempelajari fungsi dan cara kerja mikrokontroler 8031, Aliran fluida, Komunikasi data serial pada IBM PC, spektrum gelombang radio, modulasi frekuensi, Modulator FSK (IC XR-2206), Demodulator FSK (IC XR-2211), serta pemancar dan penerima FM.

Dalam perencanaan alat dilakukan perencanaan secara hardware dan software. Perencanaan hardware dengan mempelajari dan memahami komponen yang dipakai pada perangkat keras. Perencanaan software dilakukan dengan mempelajari fasilitas software mikrokontroler 8031 dalam menjalankan hardware dan kemampuan mengatur input-output (I/O). Selain itu juga dipelajari software Pascal dalam mengirimkan dan menerima data ke komputer IBM PC. Sesudah itu pembuatan peralatan secara software dan hardware dilaksanakan. Langkah selanjutnya adalah melakukan Pengujian dan pengukuran alat. Sesudah semua berakhir maka dilakukan penulisan naskah Tugas Akhir.

I.6. SISTEMATIKA PENULISAN

Sistematika pembahasan pada tugas akhir ini dibagi dalam beberapa bab sebagai berikut :

- Bab 1 : Pendahuluan, bab ini membahas tentang latar belakang, permasalahan, pembatasan masalah, tujuan, metodologi, sistematika penulisan, dan

relevansi

- Bab II : Teori Penunjang, bab ini membahas tentang IC mikrokontroler 8031, Aliran fluida, Komunikasi data serial pada IBM PC, spektrum gelombang radio, modulasi frekuensi, Modulator FSK (IC XR-2206), dan Demodulator FSK (IC XR-2211).
- Bab III : Perencanaan, bab ini membahas tentang teknik perencanaan perangkat keras dan perangkat lunak.
- Bab IV : Pengujian dan Pengukuran Alat, bab ini membahas tentang pengujian unjuk kerja peralatan.
- Bab V : Penutup, bab ini berisi kesimpulan dan saran dari Tugas Akhir ini.

I.7. RELEVANSI

Dengan dibuatnya model sistem pemantauan meter air ini diharapkan dapat menggantikan peran petugas pencatat meter air. Sehingga data pemakaian air oleh pelanggan PDAM dapat diperoleh dengan cepat tanpa petugas pencatat meter air datang ke rumah-rumah pelanggan.

BAB II

TEORI PENUNJANG

II.1. MIKROKONTROLER 8031¹⁾

Mikrokontroler 8031 merupakan salah satu anggota keluarga MCS-51. Anggota keluarga MCS-51 lainnya adalah mikrokontroler 8051 dan mikrokontroler 8751. Keluarga MCS-51 memiliki konfigurasi pin-pin, pewaktuan (timing), dan karakteristik listrik yang sama. Perbedaan utamanya adalah dalam hal memori program internalnya. Mikrokontroler 8751 memiliki 4 KByte EPROM (*Erasable Programmable Read Only Memory*). Mikrokontroler 8051 memiliki 4 KByte ROM (*Read Only Memory*) yang telah diisi program sesuai kehendak pemakai. Sedangkan mikrokontroler 8031 tidak memiliki memori program dalam, mikrokontroler 8031 hanya bisa bekerja dengan menggunakan program eksternal (memori program luar).

Keluarga MCS-51 mampu mengakses 64 KByte memori program eksternal dan 64 KByte memori data eksternal. MCS-51 juga mempunyai 32 jalur input/output (I/O) receive-buffered, serial I/O dua arah, Timer atau counter, serta fasilitas interupt.

Dalam tugas akhir ini yang digunakan adalah mikrokontroler 8031 maka hanya segala sesuatu yang mengenai mikrokontroler 8031 saja yang akan dibahas.

1). -----, *Embedded Controller Handbook*, Vol. II, Intel Corp., Santa Clara California USA, 1986, hal. 7-1 s. d 7.2

II.1.1. ARSITEKTUR 8031²⁾

Blok diagram struktur internal dari mikrokontroler 8031 ditunjukkan pada gambar 2.1. Secara umum struktur atau organisasi internal mikrokontroler 8031 adalah sebagai berikut :

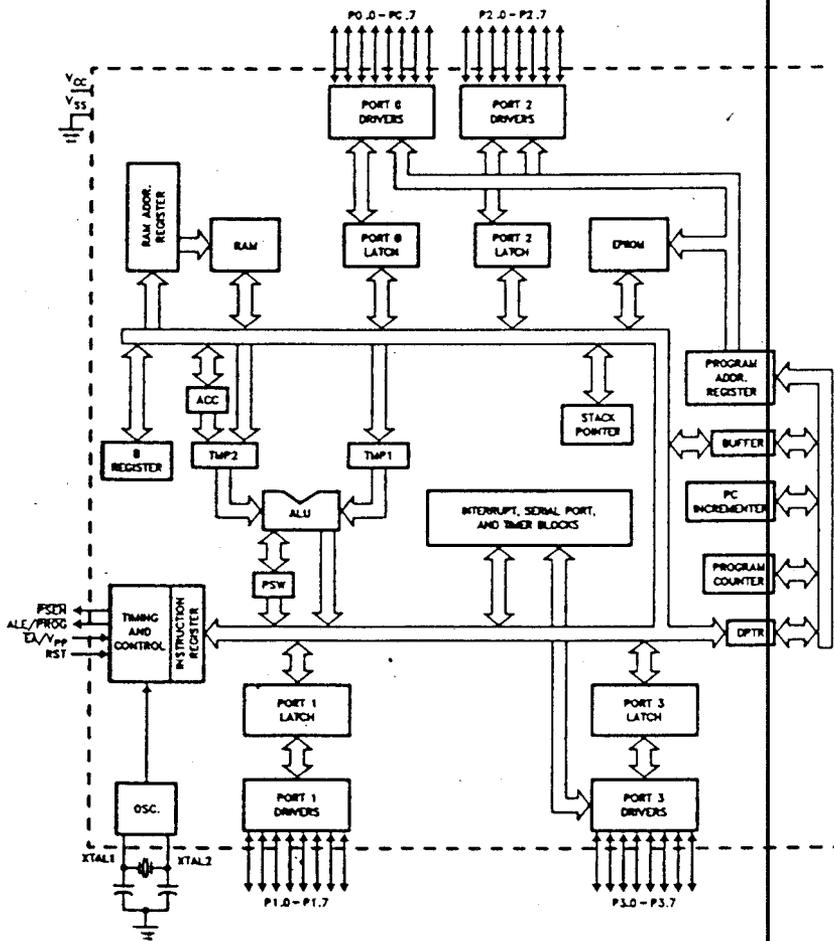
- Unit pengolah utama (CPU) 8 bit
- Kemampuan untuk proses boolean (single-bit logic)
- 64 Kbyte ruang pengalamatan untuk memori data luar
- 64 Kbyte ruang pengalamatan untuk memori program luar
- 128 byte data RAM internal
- 32 jalur port bidirectional, setiap port dapat dialamati
- 2 buah Timer / Counter 16 bit
- Full duplex UART
- 5 jalur interupsi dengan 2 tingkat prioritas yang dapat diprogram
- Clock on-chip osilator

Memori data internal besarnya adalah 256 byte selanjutnya dibagi menjadi dua yaitu 128 byte RAM data internal dan 128 byte register fungsi khusus (SFR = Special Function Registers). RAM data internal mempunyai empat buah register bank dimana setiap bank terdiri dari delapan buah register, 128 bit yang bisa dialamati (128 addressable bit), dan stack. Lokasi stack ditentukan oleh penunjuk stack (Stack

2). Wharton, John, An Introduction to the Intel MCS-51 Single-Chip Microcomputer Family, Intel Application Note(AP-69), Intel Corp., Santa Clara USA, 1980, hal 4 s.d. 13

Pointer) 8 bit.

Keluarga MCS-51 mempunyai alamat memori data dan memori program secara terpisah. Oleh karena itu CPU dapat mengakses sampai 64 Kbyte memori program dan 64 Kbyte memori data.



GAMBAR 2.1³⁾

BLOK DIAGRAM STRUKTUR INTERNAL 8031

3). Ibid, hal 5

Dari 128 byte bawah yang digunakan sebagai RAM internal, 32 byte bagian bawah dikelompokkan menjadi 4 buah bank register dan setiap bank terdiri dari 8 register. Program dapat mengakses register-register ini dengan instruksi pada R0 sampai R7, pemilihan bank melalui Program Status Word (PSW). Di atas keempat bank tersebut terdapat 16 byte atau 128 bit yang dapat dialamati perbit (128 addressable bit). Untuk stack hanya dapat disimpan dalam RAM data internal. Lokasi stack ditentukan oleh penunjuk stack (Stack Pointer) 8 bit.

Semua register, kecuali program counter dan kedelapan register bank terdapat dalam ruang alamat Register Fungsi Khusus (SFR). Register-register ini adalah register akumulator, register B, PSW, Stack Pointer, Data Pointer (DPTR), I/O Port, Register-register untuk sistem interupsi, timer dan saluran serial. Lokasi 128 bit pada SFR dapat dialamati sebagai bit atau byte.

II.1.2. KONFIGURASI DAN FUNGSI DARI PIN-PIN 8031⁴⁾

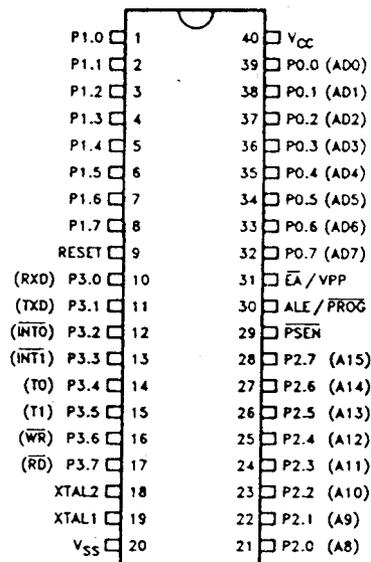
Gambar 2.2 memperlihatkan konfigurasi dari kaki-kaki 8031. Sedangkan fungsinya adalah sebagai berikut :

- Vss : Pin ini dihubungkan dengan ground dari sumber tegangan rangkaian.

4). Volume II, Op. Cit., hal 7-31 s.d. 7-33

- Vcc : Pin ini dihubungkan dengan sumber sumber tegangan 5 volt.
- Port 0
Port 0 merupakan port I/O 8 bit dua arah. Port ini juga digunakan sebagai multiplek bus alamat rendah dan bus data selama pengaksesan ke memori luar.
- Port 1
Port 1 merupakan port I/O 8 bit dua arah. Dapat bekerja untuk operasi byte maupun bit, tergantung dari pengaturan perangkat lunak.
- Port 2
Port 2 merupakan port I/O 8 bit dua arah. Port ini juga digunakan untuk mengeluarkan alamat tinggi selama pengaksesan ke memori luar.
- Port 3
Port 3 merupakan port I/O 8 bit dua arah. Port 3 memiliki fungsi-fungsi khusus sebagai berikut :
 - RxD (P3.0). Port serial sebagai penerima masukan data tak sinkron (asynchronous) atau masukan / keluaran data sinkron (synchronous)
 - TxD (P3.1). Port serial sebagai pengirim keluaran data tak sinkron atau keluaran clock sinkron
 - INT0 (P3.2). Masukan interupt 0 atau masukan kontrol gerbang untuk counter 0
 - INT1 (P3.3). Masukan interupt 1 atau masukan kontrol gerbang untuk counter 1

- T0 (P3.4). Masukan untuk counter 0
- T1 (P3.5). Masukan untuk counter 1
- WR (P3.6). Sinyal kontrol tulis untuk mengunci byte data dari Port 0 ke dalam Memori Data Luar.
- RD (P3.7). Sinyal kontrol baca untuk menghubungkan Memori data luar ke Port

GAMBAR 2.2⁵⁾

KONFIGURASI PIN-PIN 8031

- RST/VPD

Perubahan tegangan dari rendah ke tinggi pada kaki ini (kira-kira 3V) akan mereset 8031.

- ALE/PROG

ALE (Address Latch Enable) menghasilkan keluaran yang digunakan untuk mengunci alamat ke memori luar selama

5). Wharton, John, Op. cit., hal 1

operasi yang normal. Pada 87C51 digunakan sebagai masukan pulsa penerimaan program selama pemrograman EPROM.

- PSEN

Keluaran PSEN (Program Store Enable) adalah sinyal kontrol yang menghubungkan memori program eksternal dengan bus selama operasi normal.

- EA/VDD

Untuk 8031 kaki ini harus dihubungkan dengan ground agar dapat menjalankan instruksi dari Memori Program Luar. Sedangkan untuk 8051 dan 8751 bila kaki ini dihubungkan dengan taraf tinggi (high level) akan menjalankan instruksi dari ROM atau EPROM dalam. Kaki ini juga digunakan untuk menerima tegangan pemrograman EPROM sebesar 21 V.

- XTAL1

Masukan ke amplifier osilator berpenguatan tinggi. Kaki ini dihubungkan dengan kristal atau sumber osilator dari luar.

- XTAL2

Keluaran dari penguat osilator. Kaki ini dihubungkan dengan kristal bila menggunakan sumber osilator dari dalam.

II.1.3. PERANGKAT KERAS CPU[Ⓞ]

Fungsi dari tiap-tiap blok pada diagram blok yang terdapat pada gambar 2.1 adalah sebagai berikut :

Ⓞ. Ibid., hal. 5 s.d. 8

- Dekoder instruksi

Setiap instruksi program diterjemahkan oleh dekoder instruksi. Bagian ini akan membangkitkan sinyal yang mengontrol fungsi dari setiap bagian di dalam CPU.

- Program Counter

Program Counter (PC) 16 bit berfungsi untuk mengontrol urutan instruksi yang akan dijalankan.

- Memori Program Dalam

8051/8751 memiliki 4 kilo byte memori program yang dalam chip. Sedangkan 8031 tidak memiliki memori program didalam serpih.

- RAM Data Dalam

Pada 128 byte RAM Data Dalam terdapat :

- Bank Register

Terdapat empat buah bank register di dalam RAM Data dalam, yang setiap bank terdapat register R0 sampai R7.

- 128 Addressable Bit

Terdapat dalam 16 byte yang berlokasi pada alamat 20H samapai 2FH di dalam RAM Dalam.

- Stack

Stack dapat ditempatkan dimana saja pada RAM Data Dalam. Banyaknya stack dapat mencapai 128 byte.

- Register fungsi Khusus (SFR)

Register Fungsi khusus adalah sebagai berikut :

- ACC

Register A digunakan sebagai akumulator. Untuk

instruksi-instruksi yang menunjuk ke akumulator cukup disebut sebagai A, tetapi register itu sendiri diistilahkan ACC.

- Register B

Register B digunakan bersama register A untuk instruksi perkalian dan pembagian.

- Register Program Status Word (PSW)

Carry (CY), auxiliary carry (AC), user flag 0 (F0), pemilih bank register (RS0 dan RS1), overflow (OV) dan parity flag (P) terdapat dalam register Program Status Word. Flag CY, AC, dan OV pada umumnya menandakan keadaan dari operasi aritmatika yang terakhir. Flag P merupakan parity dari register A. Flag carry juga digunakan sebagai akumulator Boolean untuk operasi bit. F0 adalah flag serba guna (general purpose).

- DPTR

Data Pointer (DPTR) adalah register 16-bit, yang terdiri dari penunjuk Data Tinggi (data Pointer High) dan Penunjuk data Rendah (Data Pointer Low). Fungsinya adalah untuk memegang alamat 16-bit sebagai pointer ke ROM dan RAM luar.

- Port 0, 1, 2, dan 3

Empat buah port tersebut mempunyai 32 jalur I/O untuk berhubungan dengan perangkat keras lainnya di luar. Semua port bisa dialamati secara byte maupun bit. Port 0 (P0) dan port 2 (P2) bisa digunakan untuk menambah

Jumlah memori luar. Port 3 (P3) memiliki sinyal kontrol khusus seperti sinyal baca dan sinyal tulis. Port 1 (P1) digunakan untuk I/O.

- Register Interrupt Priority

Register Interrupt Priority memiliki bit-bit kontrol untuk mengaktifkan interrupt pada taraf yang diinginkan.

- Register Interrupt Enable

Register Interrupt Enable memiliki bit-bit kontrol untuk mengaktifkan kelima sumber interrupt.

- Register Timer / Counter

Bit-bit yang terdapat pada bit ini digunakan untuk memilih pewaktu / pencacah yang akan bekerja.

- Rangkaian osilator dan timing

Rangkaian osilator yang terdapat di dalam chip adalah rangkaian parallel anti-resonant dengan batas frekuensi mulai dari 1,2 MHz sampai 12 MHz. Frekuensi tersebut dibagi 12 sehingga menghasilkan siklus instruksi minimum $1 \mu\text{s}$ bila menggunakan kristal 12 MHz. Kaki XTAL2 adalah keluaran dari amplifier berpenguatan tinggi, sedangkan XTAL1 sebagai masukan. Kristal dihubungkan dengan kaki XTAL1 dan XTAL2 menghasilkan umpan balik dan penggeseran fasa yang dibutuhkan untuk berosilasi. Jika XTAL1 dihubungkan dengan sumber frekuensi dari luar, XTAL2 tidak perlu dihubungkan ke kristal.

II.1.4. TIMER / COUNTER⁷⁾

Mikrokontroler 8031 mempunyai dua buah register 16 bit yang dapat digunakan sebagai timer / counter. Register register tersebut adalah TH0 (timer/counter 0 byte tinggi), TLO (timer/counter 0 byte rendah), TH1 (timer/counter 1 byte tinggi), TLO (timer/counter 1 byte rendah). Setiap timer/counter dikontrol oleh bit dalam register TMOD untuk memilih fungsi sebagai timer atau counter. Terdapat empat buah mode timer / counter yang dapat dipilih yaitu :

- Mode 0

Menghasilkan timer/counter 8 bit dengan prescaler pembagi 32. Register TH1 atau TH0 sebagai timer/counter, register TL1 atau TLO bit 4-0 sebagai prescaler, sedangkan bit 5-7 diabaikan.

- Mode 1

Menghasilkan timer/counter 16 bit. TH1 dan TL1 atau TH0 dan TLO di-kaskade.

- Mode 2

Menghasilkan timer/counter 8 bit yang autoreload. Register TH1 atau TH0 berisi bilangan yang akan di-reload ke register TL1 atau TLO setiap kali overflow.

- Mode 3

Timer 0 dibagi menjadi dua buah timer/counter 8 bit yang terpisah. Timer 1 tidak berjalan.

7). Ibid., hal 7-10 s. d. 7-12

II.1.5. SISTEM INTERUPSI⁸⁾

Mikrokontroler 8051 mempunyai lima sumber interrupt, yang masing-masing dapat dikontrolkan oleh perangkat lunak, yaitu dengan mengatur kondisi logika bit-bit pada register IE. Dan juga dapat diprogram pada salah satu dari dua tingkat prioritas. Lima sumber interrupt itu adalah :

INT 0

Interupsi dari luar (external request) pada pin P3.2INT 1

INT 1

Interupsi dari luar pada pin INT 1 (kaki P3.3)

Timer 0

Overflow dari register timer 0 akan mengaktifkan flag permintaan interrupt (interrupt request flag) TF0.

Timer 1

Overflow dari register timer 1 akan mengaktifkan flag permintaan interrupt TF1.

Port Serial

Flag TI akan aktif bila pengiriman satu frame serial telah lengkap dan flag RI akan aktif bila penerimaan satu frame serial telah lengkap.

8). Ibid., Hal 7-23 s.d. 7-25

TABEL 2.1⁹⁾

ALAMAT AWAL DARI INTERRUPT SERVICE ROUTINE

Sumber Interrupt	Alamat awal
External interrupt 0 (INT0)	0003H
Timer/Counter 0 (T0)	000BH
External Interrupt 1 (INT1)	0013h
Timer/Counter 1 (T1)	001BH
Port Serial	0023H

Setiap sumber interrupt dapat di-enable atau di-disable secara tersendiri dengan mengubah bit set atau clear pada register Interrupt Enable (IE). Semua sumber interrupt juga dapat di-enable atau di-disable secara keseluruhan.

II.1.6. PORT SERIAL¹⁰⁾

Mikrokontroler 8031 mempunyai port serial di dalam serpih. Port serial ini dapat digunakan untuk komunikasi data serial, baik yang menggunakan hubungan half duplex maupun full duplex, atau untuk menambah port I/O dengan menghubungkan ke register geser (shift register). Komunikasi serial dilakukan melalui register SBUF. Pengiriman data

9). Ibid., hal. 7-25

10). Ibid., hal 7-13 s. d. 7-14

serial dilakukan dengan menuliskan data yang akan dikirim ke SBUF, dan penerimaan data serial dilakukan dengan membaca data dari register SBUF.

Port Serial ini dapat dioperasikan ke dalam empat mode, yaitu :

- Mode 0 (synchronous)

Data serial 8 bit dikirim dan diterima melalui RxD, dengan bit terendah (LSB) yang pertama, dan TxD mengeluarkan clock penggeser (shift clock). Baud rate tetap $1/12$ dari frekuensi osilator.

- Mode 1 (asynchronous)

Data 10 bit dikirimkan melalui TxD atau diterima melalui RxD dengan urutan : satu start bit, 8 bit data (LSB yang pertama) dan satu stop bit. Pada penerimaan data, stop bit akan mengisi RBB pada register SCON. Baud rate dapat diubah-ubah (variabel), ditentukan oleh isi bit SMOD pada register PCON dan kecepatan overflow timer 1.

- Mode 2 (asynchronous)

Data 11 bit dikirimkan melalui TxD atau diterima melalui TxD dengan urutan : satu start bit, 8 bit data (LSB yang pertama), satu bit data dapat diprogram (bit ke sembilan), dan satu stop bit. Pada pengiriman data, bit data ke sembilan (TBB) dapat dipilih '0' atau '1'. TBB dapat digunakan sebagai parity bit. Pada penerimaan data, bit data kesembilan akan mengisi RBB pada register SCON. Baud rate ditentukan oleh isi bit SMOD jika berlogika 0 bekerja

pada $1/32$ dan jika berlogika 1 pada $1/64$ dari frekuensi osilator.

- Mode 3 (asynchronous)

Mode 3 sama dengan mode 2 kecuali pada mode 3 baud rate dapat diubah-ubah (variabel), ditentukan oleh isi bit SMOD dan kecepatan overflow timer 1.

2.2. ROM DAN RAM

Memory jenis ROM (Read Only Memory) bersifat non volatile, artinya data yang telah disimpan tidak akan hilang ketika hubungan power supply ke ROM diputuskan. Ada beberapa jenis ROM, tetapi yang paling banyak digunakan sekarang adalah jenis EPROM (Erasable Programable Read Only Memory). EPROM memiliki jalur address yang sesuai dengan kapasitas memorinya. Sebagai contoh, EPROM 2764 memiliki kapasitas 8 KByte ($8192 \times 8 = 2^{13}$) mempunyai 13 jalur address, yaitu A0 - A12. Pada EPROM terdapat pin \overline{CS} (Chip Select) yang berfungsi untuk mengaktifkan rangkaian buffer input/output internal dari EPROM. Pada EPROM ini juga terdapat pin \overline{OE} (Output Enable) yang berfungsi untuk mengaktifkan data output dari EPROM.

Pin PGM di-set low pada saat EPROM sedang diprogram dan high pada operasi pembacaan (read). Pada pin Vpp diberi tegangan pemrograman yang sesuai ketika waktu pemrograman.

RAM digunakan untuk menyimpan data sementara waktu. Data pada RAM akan hilang bila supply tegangan terputus sehingga

RAM digolongkan jenis volatile memory. Pada RAM data dapat dibaca maupun ditulis. Ada dua jenis RAM yaitu RAM statis dan RAM dinamis. RAM dinamis memerlukan siklus penyegaran memori sedangkan RAM statis tidak memerlukan. Pada RAM selain terdapat pin \overline{CS} dan \overline{OE} seperti halnya EPROM, juga terdapat \overline{WE} (Write Enable). \overline{WE} yang aktif low ini merupakan sinyal kontrol pada proses penulisan RAM.

II.3. ALIRAN AIR

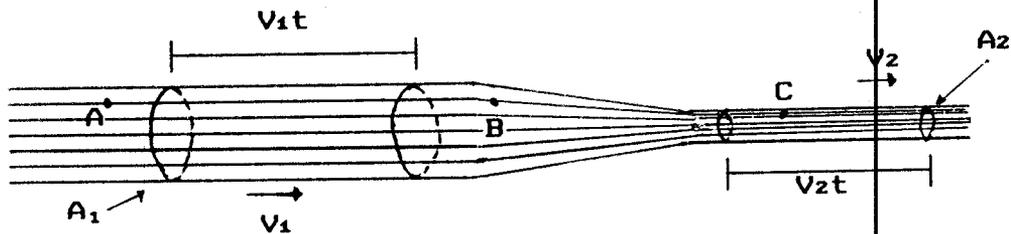
II.3.1. PERSAMAAN KONTINUITAS

Gambar 2.3 melukiskan suatu bagian pipa yang mana air mengalir dari kiri ke kanan. Seperti terlihat pada gambar partikel partikel yang lewat titik A selalu melewati titik B dan titik C. Garis yang menghubungkan ketiga titik tersebut disebut garis arus atau streamline.

Bila luas penampang pipa berlainan, maka besarnya kecepatan partikel pada setiap titik juga berlainan. Tetapi kecepatan partikel-partikel pada saat melewati titik A akan sama besarnya. Demikian juga saat melewati titik B dan C.

Gambar 2.3 menunjukkan suatu aliran air dalam suatu pipa yang mempunyai penampang berbeda. Jika A_1 adalah luas penampang pada titik 1, dan V_1 kecepataannya, maka dalam t detik, partikel yang berada pada titik 1 akan berpindah sejauh $V_1 t$ dan volume yang lewat penampang A_1 sebanyak $A_1 V_1 t$. Volume air yang lewat penampang A_1 persatuan waktu adalah $A_1 V_1$. Maka dengan cara yang sama, volume air yang lewat

penampang A_2 persatuan waktu adalah A_2V_2 .



GAMBAR 2.3
GARIS ALIRAN ARUS

Besar volume air yang lewat penampang A_1 dan A_2 persatuan waktu adalah sama besar sehingga diperoleh :

$$Q = A_1V_1 = A_2V_2 \quad 11)$$

Besaran Q dinamakan debit, yang mempunyai satuan m^3/dt . Persamaan $A_1V_1 = A_2V_2$ dikenal sebagai persamaan kontinuitas. Konsekuensi dari hubungan di atas adalah bahwa kecepatan akan membesar jika luas penampang mengecil, demikian sebaliknya.

II.3.2. SENSOR ALIRAN¹²⁾

Pada gambar 2.4 ditunjukkan pengukuran jumlah volume air yang dialirkan pipa dengan menggunakan sensor photo-transistor. Kecepatan putar baling-baling sebanding dengan kecepatan aliran air. Hal ini secara matematis dapat

11). Dosen-dosen Fisika-ITS, Diktat Fisika, FMIPA-ITS, 1988, hal 179.

12). Douglas V. Hal, Microprocessors and Interfacing: Programming and Hardware, McGraw-Hill Book Company, Singapore, 1987, hal. 442

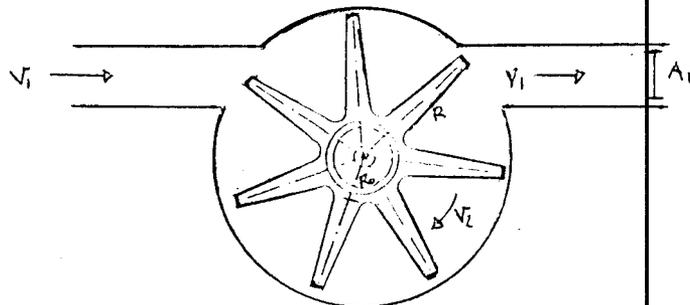
dituliskan sebagai berikut :

$$w = k.f$$

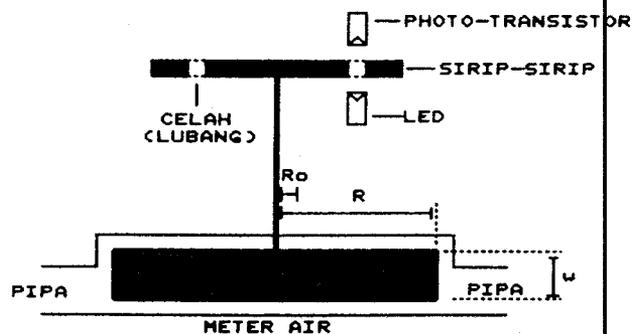
dimana w : kecepatan (m/det)

k : konstanta (m/pulsa)

f : jumlah pulsa persatuan waktu (pulsa/det)



(a)



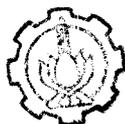
(b)

GAMBAR 2.4

SENSOR ALIRAN PADA METER AIR

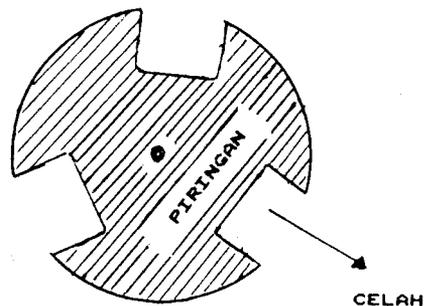
(a) PANDANGAN ATAS

(b) PANDANGAN SAMPING



MILIK PERPUSTAKAAN
INSTITUT TEKNOLOGI
SEPULUH - NOPEMBER

Seperti nampak pada gambar letak photo-transistor dan LED dipasang saling berhadapan. Diantara photo-transistor dan LED dipasang sirip-sirip yang memiliki 3 celah. Dengan memakai photo-transistor, LED, dan sirip-sirip yang terpasang pada sumbu baling-baling, maka akan diperoleh pulsa setiap terjadi pemotongan sinar pada basis photo-transistor oleh sirip yang berputar. Kecepatan aliran dalam pipa sinkron dengan putaran sirip sehingga pulsa yang diperoleh juga sinkron dengan kecepatan air yang mengalir. Informasi pulsa cacahan dapat juga dipakai untuk menentukan besar volume air yang mengalir. Setiap pulsa cacahan yang terjadi berkaitan dengan penambahan jumlah volume. Dengan menghitung jumlah total pulsa yang terjadi maka dapat diketahui volume total air yang telah melewati meter air.



GAMBAR 2.5
PIRINGAN SIRIP

Jumlah total volume air yang telah dialirkan adalah berbanding lurus dengan jumlah pulsa cacahan, sehingga dapat dituliskan sebagai berikut :

Volume $\sim c$ dimana c : jumlah pulsa cacahan

$$\text{Volume} = k.c \text{ (m}^3\text{)} \quad (2.1)$$

dimana k : konstanta (m^3)

Jadi untuk mengetahui volume air yang melewati meter air, harga k (konstanta) harus diketahui. Harga k dapat diketahui dari hasil perhitungan atau pengkalibrasian meter air.

Gambar 2.4 memperlihatkan aliran air pada meter air. Menurut hukum kontinuitas debit air dalam pipa adalah sama dengan debit air pada rongga meter air.

$$Q_1 = Q_2$$

$$A_1V_1 = A_2V_2$$

dimana

Q_1 : debit air pada pipa (m/det)

Q_2 : debit air pada rongga baling-baling (m/det)

A_1 : luas penampang pipa (m^2)

A_2 : luas penampang baling-baling (m^2)

V_1 : kecepatan air pada pipa (m/det)

V_2 : kecepatan air pada rongga baling-baling (m/det)

Penurunan rumus volume air yang melewati meter air pada kecepatan konstan adalah sebagai berikut :

$$\begin{aligned} Q_v &= Q_1 t && \text{dimana } Q_v : \text{volume} \\ &= A_1 V_1 t && \text{dimana } t : \text{waktu (det)} \\ &= A_2 V_2 t && \text{dimana } A_1 V_1 = A_2 V_2 \end{aligned} \quad (2.2)$$

Aliran air pada rongga baling-baling memiliki kecepatan yang sesuai dengan jari-jari baling-baling. Semakin jauh jarak partikel air dari titik tengah baling-baling makin besar kecepatannya demikian pula sebaliknya. Kecepatan elemen air seluas ∂A_z pada jarak r dari titik pusat rongga meter air memenuhi persamaan :

$$\partial Q_v = V_z \cdot t \cdot \partial A_z \quad \text{dimana } \partial Q_v : \text{Volume pada luasan selebar } \partial A_z$$

$$\partial Q_v = \frac{2\pi}{T} \cdot r \cdot t \cdot \partial A_z \quad \text{dimana } T : \text{periode baling-baling melakukan tiga cacahan (satu putaran).}$$

Untuk satu cacahan (1/3 putaran) diperoleh :

$$\begin{aligned} \partial Q_v &= \frac{2\pi r}{3T} \cdot t \cdot \partial A_z \\ &= \frac{2\pi r}{3T} \cdot t \cdot w \cdot \partial r \quad \text{dimana } w : \text{lebar baling-baling} \\ &= (2/3) \cdot \pi \cdot r \cdot w \cdot (t/T) \cdot \partial r \end{aligned}$$

karena $t/T = c = \text{jumlah cacahan}$ maka

$$\partial Q_v = (2/3) \cdot \pi \cdot r \cdot w \cdot c \cdot \partial r$$

Untuk luasan baling-baling selebar A dengan jari-jari luar R dan jari-jari dalam R_0 maka :

$$\begin{aligned} Q_v &= \int_{R_0}^R (2/3) \cdot \pi \cdot w \cdot r \cdot c \cdot \partial r \\ &= (2/3) \cdot \pi \cdot w \cdot c \int_{R_0}^R r \partial r \\ &= (2/3) \cdot \pi \cdot w \cdot (1/2) \cdot (R - R_0)^2 \\ &= \pi (R - R_0)^2 c / 3 \end{aligned}$$

Sesuai dengan persamaan 2.1 harga k diperoleh :

$$k = \pi (R - R_0)^2 w / 3$$

Untuk melakukan perhitungan kecepatan aliran air yang tidak konstan harus diketahui terlebih dahulu fungsi aliran air. Pada kecepatan aliran tidak konstan harga konstanta k adalah tetap. Karena jumlah pulsa adalah sebanding dengan jumlah volume maka dengan diketahuinya harga k , volume air yang melewati meter air dapat diketahui.

Untuk kecepatan yang tidak konstan maka harga T akan selalu berubah-ubah. Untuk melakukan perhitungan volume maka dilakukan pemecahan waktu (t) berdasarkan kecepatan. Kecepatan air yang mengalir pada meter air merupakan fungsi diskrit (tidak kontinyu). Jadi untuk melakukan perhitungan volume air didasarkan pada kecepatan pada saat tertentu. Misalnya dalam jangka waktu t terjadi 3 kali perubahan kecepatan maka :

$$t = t_1 + t_2 + t_3$$

$$t_1 = \text{jangka waktu selama kecepatan } V_1$$

$$t_2 = \text{jangka waktu selama kecepatan } V_2$$

$$t_3 = \text{jangka waktu selama kecepatan } V_3$$

$$\text{Volume} = (\pi(R-R_0)^2 w/3) \cdot \left(\frac{t_1}{T_1} + \frac{t_2}{T_2} + \frac{t_3}{T_3} \right)$$

$$= (\pi(R-R_0)^2 w/3) \cdot (c_1 + c_2 + c_3)$$

$$= (\pi(R-R_0)^2 w/3) \cdot c \quad \text{dimana } c : \text{jumlah total cacahan}$$

Karena harga $T \ll t$,

$T_1 \ll t_1$, $T_2 \ll t_2$, dan $T_3 \ll t_3$ maka persamaan umum,

$\text{Volume} = \pi(R-R_0)^2 w c / 3 = kc$ dapat dipakai untuk menghitung volume air yang dilewatkan meter air.

II.4. KOMUNIKASI DATA SERIAL PADA IBM PC

Komunikasi data dapat digolongkan menjadi tiga bentuk yaitu¹⁹⁾:

1. Simplex : Transmisi data berlangsung satu arah.
2. Half duplex : transmisi data berlangsung dua arah dalam waktu yang berbeda.
3. Full Duplex : transmisi data berlangsung dua arah dalam waktu yang sama.

Berdasarkan bentuk sinyal yang dikirim, transmisi dibagi menjadi dua, yaitu transmisi analog dan transmisi digital. Transmisi analog adalah transmisi sinyal secara kontinyu, seperti sinyal bunyi atau suara. Transmisi analog sangat peka terhadap noise dan distorsi. Transmisi digital adalah transmisi sinyal yang berupa aliran pulsa yang berlogika low dan high.

Dilihat dari jumlah bit yang yang ditransmisikan, transmisi data dibagi dua yaitu transmisi data paralel dan transmisi data serial. Dalam suatu sistem mikrokomputer, transmisi data selalu dilaksanakan secara paralel. Karena hal tersebut merupakan cara yang tercepat yang masih dapat dilakukan. Namun untuk transmisi jarak jauh, komunikasi data secara paralel akan membutuhkan banyak kabel, sehingga cara ini tidak lagi efisien. Oleh karena itu, transmisi data dengan jarak jauh data yang akan dikirimkan diubah dari bentuk paralel menjadi serial, sehingga data tersebut dapat

19). Ibid., hal 321.

dikirimkan dengan hanya melalui sepasang kabel. Data serial yang diterima kemudian diubah kembali ke dalam bentuk paralel.

Hal yang penting dari komunikasi serial adalah frekuensi dari bit-bit yang dikirim. Frekuensi ini disebut baud rate. Baud rate didefinisikan sebagai bit-bit yang dikirim tiap detik melalui satu jalur serial. Baud rate standard adalah 110, 150, 300, 1200, 2400, 4800, dan 9600 bit/s. Ada dua macam sistem komunikasi serial yaitu :

1. Sistem serial sinkron : data yang dikirimkan disinkronkan dengan bit-bit protokol.
2. Sistem serial asinkron : data yang dikirimkan selalu ditambahkan dengan bit-bit pengontrol start bit, stop bit, atau ditambahkan bit parity.

Komunikasi serial pada komputer IBM PC mengikuti standart EIA RS-232. Standart ini dikembangkan oleh Electronic Industries Association. Standart RS-232 mengatur interfacing antara komputer dengan modem (modulator demodulator). Komputer dalam hal ini disebut *Data Terminal Equipment (DTE)*, dan modem disebut *Data Communication Equipment (DCE)*. Standart RS-232 menetapkan fungsi 25 pin penghubung untuk komunikasi serial, level tegangan, rise time dan fall time, kecepatan bit maksimum serta kapasitansi antara kawat penghubung.

Pada IBM PC ke-25 pin yang disarankan standart RS-232 tidak dipakai semua. Sesuai dengan standart RS-232 konektor

pada port serial IBM PC adalah male sedangkan pada DCE haruslah female. Konektor yang digunakan adalah DB-25P male dan DB-25S female. RS-232 menggunakan standart negatif, jadi tingkat logika satu berada dalam daerah tegangan -3 sampai -15 Volt (-25 Volt tanpa beban), dan tingkat logika 0 berada dalam daerah 3 sampai 15 Volt (25 Volt tanpa beban). Kapasitansi maksimum dari kabel penghubung yang diperbolehkan adalah 2500 pF. Sedangkan resistansi output dari sistem ditetapkan tidak boleh kurang dari 300 Ohm.

II.5. SPEKTRUM GELOMBANG RADIO¹⁴⁾

Untuk memudahkan pengenalannya maka spektrum gelombang radio digolongkan menurut sifat-sifat yang dimilikinya. Spektrum gelombang radio digolongkan sebagai berikut :

1. VLF (*Very Low Frequency*)

VLF mempunyai batas frekuensi dari 3 sampai 30 KHz dan panjang gelombang 100 km sampai 10 km. Getaran radio pada frekuensi ini bersifat ground wave dan sangat tidak efisien untuk dipancarkan. Yang dimaksud ground wave yaitu getaran yang dirambatkan oleh permukaan bumi. Sifat gelombang groundwave adalah perambatannya lurus dan sukar untuk dibiaskan, hanya mempunyai polaritas vertikal, dan polaritas vertikal langsung diserap oleh bumi.

14). Gary M. Miller, Modern Electronic
Prentice Hall Inc., New Jersey, 1980, hal. 3

2. LF (*Low Frequency*)

LF mempunyai batas frekuensi dari 30 kHz sampai dengan 300 kHz dengan panjang gelombang dari 10 km sampai dengan 1 km. Gelombang ini juga bersifat ground wave.

3. MF (*Medium Frequency*)

MF mempunyai batas frekuensi dari 300 kHz sampai dengan 3000 kHz dan panjang gelombang dari 1 km sampai 100 m. MF merupakan kombinasi antara ground wave dan sky wave. Sky wave akan dominan bila berada di atas 2 MHz. Gelombang ini banyak digunakan untuk pemancar yang bersifat lokal. Sky wave adalah getaran radio yang perambatannya disebabkan oleh pantulan ionosfer dan permukaan bumi.

4. HF (*High Frequency*)

HF mempunyai batas frekuensi dari 3 MHz sampai 30 MHz dan panjang gelombang dari 100 m sampai dengan 10 m. Perambatan HF adalah tidak stabil serta mudah dibiaskan tergantung pada kondisi cuaca.

5. VHF (*Very High Frequency*)

VHF mempunyai batas frekuensi dari 30 MHz sampai dengan 300 MHz, dan panjang gelombang dari 10 m sampai dengan 1 m. VHF bersifat space wave dengan pancaran yang lurus seperti sinar. Banyak diaplikasikan untuk komunikasi lokal. Gelombang yang bersifat space wave pancarannya lurus dan bisa dihalangi oleh bangunan bertingkat atau gunung.

6. UHF (*Ultra High Frequency*)

Batas frekuensinya dari 300 MHz sampai dengan 3000 MHz serta batas gelombang dari 1 m sampai dengan 10 cm. Banyak diaplikasikan pada multiplek radio link, komunikasi satelit dan radar.

7. SHF (*Super High Frequency*)

Mempunyai batas frekuensi 3 GHz sampai dengan 30 GHz dan batas gelombang 10 cm sampai dengan 1 cm. Merupakan gelombang space wave.

II.6. MODULASI

Semua jenis bentuk informasi harus diubah terlebih dahulu ke bentuk listrik agar dapat dibawa oleh gelombang radio. Cara menumpangkan informasi ke gelombang radio ini dinamakan modulasi. Memodulasi berarti melakukan perubahan terhadap parameter dari suatu gelombang pembawa (*carrier*) yang memiliki frekuensi tinggi dengan menggunakan sinyal informasi yang memiliki frekuensi lebih rendah.

Berdasarkan tipe modulasi maka sistem komunikasi dibedakan menjadi tiga jenis yaitu :

1. Sistem komunikasi analog

Sistem komunikasi analog mengenal tiga bentuk modulasi yaitu Amplitudo Modulation (AM), Frequency modulation (FM), dan phase modulation (PM).

2. Sistem komunikasi pulsa

Pada sistem ini dikenal tiga bentuk modulasi yaitu Pulse Amplitudo Modulation (PAM), Pulse Width Modulation (PWM), dan Pulse Position Modulation (PPM). Sistem komunikasi pulsa mengubah sinyal analog menjadi bentuk pulsa-pulsa sesuai dengan bentuk modulasi yang digunakan.

3. Sistem komunikasi digital

Untuk komunikasi digital dikenal banyak bentuk modulasi diantaranya adalah Amplitudo Shift Keying (ASK), Frequency Shift Keying (FSK), Phase Shift Keying (PSK), sedangkan modulasi digital yang lebih kompleks diantaranya Pulse Code Modulation (PCM), Quadrature Amplitudo Modulation (QAM), serta Delta Modulation.

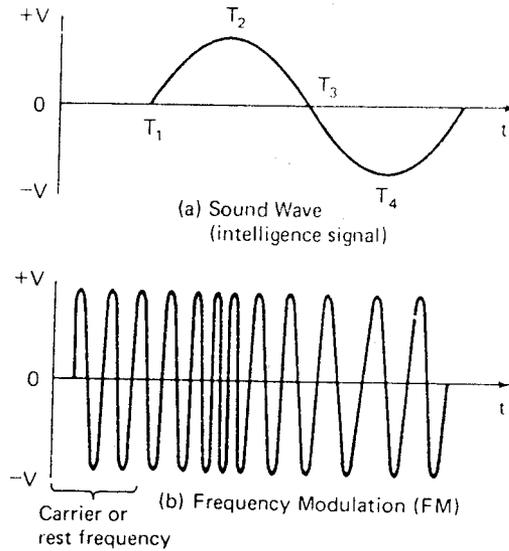
Kegunaan modulasi adalah :

- Memudahkan radiasi
- Untuk keperluan multiplexing dari beberapa kanal informasi
- Agar sinyal informasi dapat dikirimkan pada banyak kanal
- Mengurangi noise dan interferensi pada sinyal informasi

II.7. MODULASI FREKUENSI (FM)

Modulasi frekuensi merupakan modulasi dimana frekuensi gelombang pembawa dimodulasi oleh sinyal pemodulasi. Dengan demikian frekuensi gelombang pembawa berubah-ubah sebanding dengan dengan sinyal pemodulasi. Gambar 2.6 menunjukkan fase dan frekuensi dari sebuah sinyal pemodulasi yang dimodulasi frekuensi pada

sinyal pembawa.



GAMBAR 2.6¹⁵⁾

MODULASI FREKUENSI

Persamaan sudut sesaat dari gelombang termodulasi frekuensi adalah sebagai berikut :¹⁶⁾

$$M(t) = V_c \cos (\omega_c t + \theta(t))$$

dimana, $\theta(t) = \frac{K_1 V_a}{\omega_a} \cos \omega_a t$

K_1 adalah konstanta sensitivitas deviasi (rad/sec volt).

Didefinisikan suatu konstanta baru yaitu : deviasi

frekuensi : $\Delta F = K_1 \cdot V_a$

maka fase dari sinyal FM adalah :

$$\theta(t) = \frac{\Delta F}{F_a} \cos \omega_a t$$

dimana : $m = \Delta F/F_a$ adalah indeks modulasi dari sinyal FM yaitu perbandingan deviasi frekuensi dengan frekuensi

15). Ibid., hal 151.

16). Tomasi, W., FUNDAMENTAL OF ELECTRONICS COMMUNICATIONS SYSTEMS, Prentice Hall, Englewood Cliffs, New Jersey, 1980, Hal. 264

pemodulasi. Dengan demikian didapatkan persamaan fase dari sinyal FM :

$$\theta(t) = m \cos \omega_a t$$

Dan persamaan sinyal FM lengkapnya adalah :

$$M(t) = V_c \cos (\omega_c t + m \cos \omega_a t)$$

Menurut Carson bandwidth dari persamaan sinyal FM dapat dibagi menjadi 2 jenis berdasarkan bilangan indeks modulasinya (m), yaitu : minimum dan maksimum bandwidth. Sedangkan indeks modulasi dibagi menjadi 2 golongan, yaitu :

- rendah, untuk $m < 0,2$
- tinggi, untuk $m > 0,2$

Untuk indeks modulasi yang rendah, bandwidthnya :

$$B = 2 F_a$$

Biasa disebut dengan *Narrow Band FM* (NBFM).

Dan untuk indeks modulasi yang tinggi, bandwidthnya :

$$B = 2(\Delta F)$$

Biasa disebut dengan *Wide Band FM* (WBFM)

Sedangkan menurut Bessel Bandwidthnya adalah :

$$B = 2(n \times F_a)$$

dimana : n = jumlah sideband

F_a = frekuensi sinyal pemodulasi tertinggi

Pada akhirnya Carson merumuskan Bandwidth untuk sistem FM tanpa memperhatikan indeks modulasi, rumus ini biasa dikenal dengan Hukum Carson :¹⁴⁾

$$B = 2(\Delta F + F_a)$$

14). Ibid, hal. 155

dimana : ΔF = frekuensi deviasi

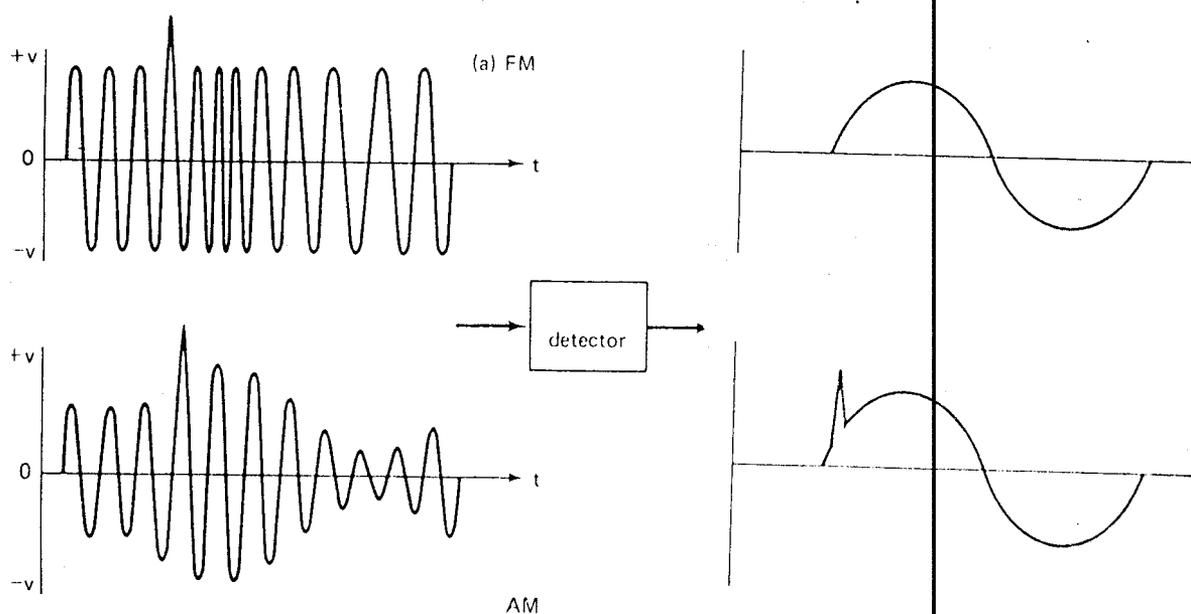
F_a = frekuensi sinyal pemodulasi tertinggi

Rumus yang diberikan Carson ini akan memberikan hasil Bandwidth yang lebih kecil dibandingkan dengan rumus yang diberikan oleh Bessel.

Adanya noise dapat merubah amplitudo gelombang pembawa. Pada modulasi frekuensi adanya perubahan amplitudo yang diakibatkan oleh noise tidak menimbulkan persoalan. Perubahan amplitudo tersebut tidak akan terdeteksi pada proses demodulasi frekuensi. Proses demodulasi frekuensi memakai acuan deviasi frekuensi pada gelombang pembawa. Gambar 2.7 menunjukkan pengaruh noise pada proses modulasi frekuensi dibandingkan modulasi amplitudo.

Dari hasil pengukuran didapatkan bahwa noise pada modulasi amplitudo yang diakibatkan oleh noise adalah sekitar 2,5 - 3 db lebih besar dibandingkan dengan noise yang ditemui pada modulasi frekuensi, sehingga pada modulasi frekuensi diperlukan daya pancar yang lebih kecil untuk mendapatkan S/N yang sama.¹⁷⁾

17). Henry Sterk & Frans B. Tuteur, Modern Electronic Communication Theory and System, Prentice Hall, New Jersey, 1980, Hal 313.



GAMBAR 2.7¹⁸⁾

PENGARUH NOISE TERHADAP SINYAL

Modulasi frekuensi secara luas digunakan pada :

1. Radio broadcasting komersial dengan lebar kanal 200 KHz dari frekuensi 88 sampai dengan 108 MHz.
2. Televisi dengan lebar band audio 50 KHz pada 54 sampai 88 MHz, 174 samapai 216 MHz, dan 470 sampai 890 MHz.
3. Narrow-band untuk keperluan pelayanan umum pada kanal 108 sampai 174 MHz serta frekuensi di atas 890 MHz.
4. Narrow-band untuk amatir radio pada kanal 29,6 MHz, 52 sampai 53 MHz, 146 sampai 147,5 MHz, 440 sampai 450 MHz, serta frekuensi di atas 890 MHz.

Modulasi frekuensi tidak dapat digunakan pada frekuensi di bawah 30 MHz. Hal ini dikarenakan adanya distorsi fase yang

18). Gary M. Miller, Op. Cit., hal 160.

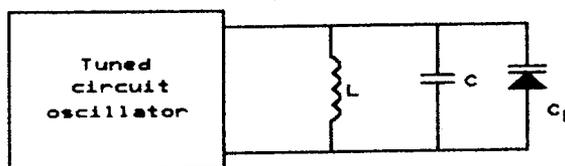
diakibatkan oleh inosfer bumi pada frekuensi di bawah 30 MHz. Pada frekuensi di atas 30 MHz sinyal FM ditransmisikan secara *line-of-sight* dan tidak terpengaruh oleh inosfer. Alasan tersebut yang membatasi transmisi dari sinyal yang bermodulasi frekuensi tidak dapat ditransmisikan pada jarak lebih dari 60 km tanpa menggunakan stasiun relay.

Untuk membangkitkan sistem modulasi FM dikenal dua teknik pembangkitan, yaitu direct FM dan Indirect FM.:

2.7.1 DIRECT FM

Teknik ini memanfaatkan perubahan sinyal pemodulasi untuk mengubah-ubah frekuensi sinyal pembawa (*carrier*). Untuk setiap perubahan amplitudo sinyal pemodulasi akan langsung mengakibatkan perubahan pada frekuensi sinyal pembawa. Salah satu metoda pembangkitan secara langsung adalah dengan menggunakan dioda varactor.

Modulator jenis dioda varaktor memanfaatkan sifat dari bias balik (*reverse bias*) pada junction dioda, untuk setiap perubahan amplitudo sinyal pemodulasi yang masuk akan mengakibatkan perubahan kapasitansi pada diode tersebut. Akibat perubahan itu maka rangkaian kapasitansi pada rangkaian LC akan berubah pula. Sehingga frekuensi yang dihasilkan akan ikut berubah seiring dengan perubahan pada amplitudo sinyal pemodulasi.

GAMBAR 2.8¹⁹⁾

DIRECT FM MODULATOR DENGAN DIODE VARACTOR

Untuk menentukan frekuensi tengah dari modulator ini dipakai persamaan :

$$F_o = \frac{1}{2 \pi \sqrt{L(C+C_D)}}$$

dimana :

F_o : frekuensi tengah

L : induktansi induktor

C : kapasitansi kapasitor

C_D : kapasitansi dioda varaktor

Sedangkan jika dimasukkan sinyal pemodulasi, maka :

$$F_c = \frac{1}{2 \pi \sqrt{L(C+\Delta C)}}$$

Dari kedua persamaan itu, dapat diturunkan suatu persamaan

¹⁹⁾. Ibid., hal. 168.

untuk mencari besarnya deviasi frekuensi, yaitu :

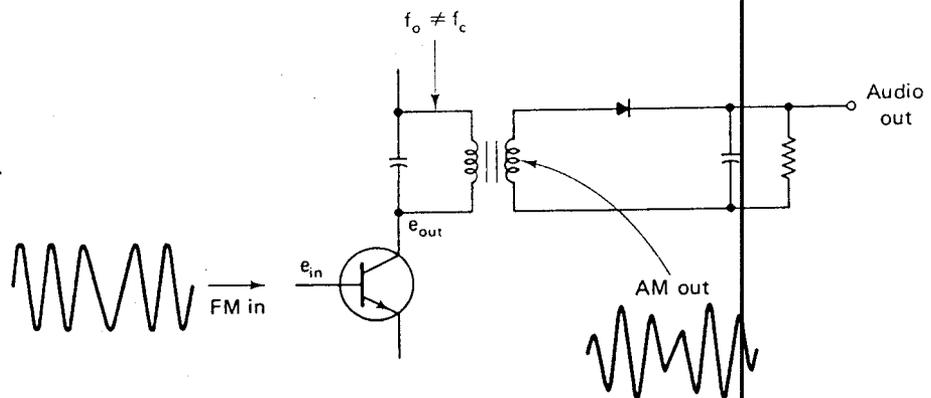
$$\Delta F = F_o - F_m$$

2.7.2 INDIRECT FM

Teknik ini tidak langsung mengubah frekuensi pembawa (*carrier*) untuk setiap perubahan amplitudo sinyal pemodulasi. Setiap kali perubahan amplitudo pada sinyal pemodulasi akan mengakibatkan perubahan fase pada sinyal pembawa, sehingga membentuk modulasi fase. Frekuensi dari sinyal pembawa tersebut ekuivalen dengan integral dari sinyal pemodulasi.

II.7.3. DEMODULASI SINYAL FM

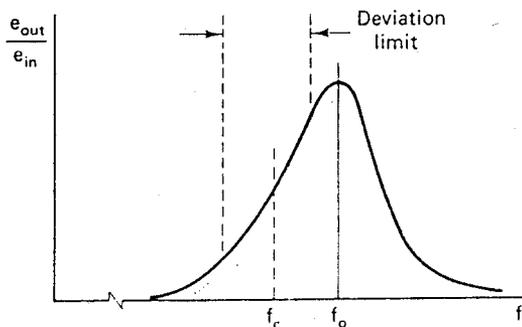
Demodulasi sinyal modulasi frekuensi diperlukan untuk menghasilkan sinyal informasi dari sinyal termodulasi frekuensi. Demodulator ini disebut juga sebagai diskriminator. Salah satu jenis diskriminator adalah slope detector. Slope detector dibentuk dari rangkaian LC tank. Gambar 2.9 menunjukkan bagian utama dari slope detector. Karakteristik slope detektor yang ditunjukkan pada gambar 2.10.



GAMBAR 2.9²⁰⁾
SLOPE DETECTOR

Prinsip kerja dari slope detektor adalah mengkonversikan besaran frekuensi menjadi besaran tegangan. Rangkaian LC tank ini beresonansi pada frekuensi resonansi f_0 . Intermediate Frequency (IF) diusahakan jatuh pada daerah yang linier pada gambar 2.8. Frekuensi tengah IF (f_c) dirancang terletak pada daerah yang linier. Hal ini bertujuan agar perubahan frekuensi input berbanding lurus dengan tegangan output yang dihasilkan.

20). Ibid., hal. 197

GAMBAR 2.10²¹⁾

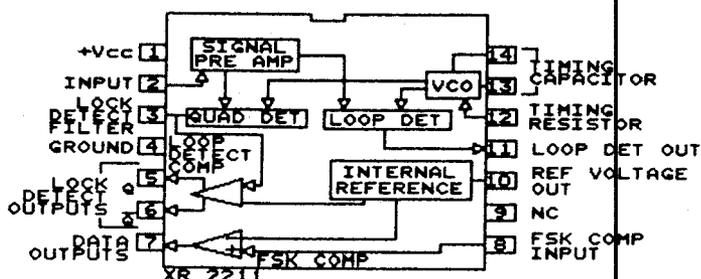
KARAKTERISTIK SLOPE DETEKTOR

II.8. XR-2211 FSK DEMODULATOR²²⁾

XR-2211 adalah phase locked loop (PLL) didisain khususnya untuk penerapan komunikasi data. XR-2211 sangat cocok diaplikasikan sebagai modem. XR-2211 merupakan pasangan dari XR-2211. Jika XR-2206 dipakai sebagai modulator FSK maka XR-2211 bekerja sebaliknya yaitu sebagai demodulator. XR-2211 dapat dioperasikan pada tegangan 4,5 - 20 V dan mempunyai jangkauan frekuensi 0,01 sampai dengan 300 KHz. Komponen eksternal dari XR-2211 dipakai untuk menentukan frekuensi tengah, bandwidth, serta delay output.

21). Gary M. Miller, Loc. cit

22). _____, EXAR Data Sheet, 750 Palomal Avenue, 1982



GAMBAR 2.11

DIAGRAM BLOK XR-2211

Bagian utama PLL yang terdapat di dalam XR-2211 terdiri dari input preamplifier, phase detector, Voltage Controlled Oscilator (VCO). Preamplifier dipakai sebagai limiter sekaligus juga dipakai sebagai penguat. Sinyal yang berada di atas 2 mV rms dikuatkan pada level yang konstan. Phase detector menghasilkan selisih dari frekuensi input dan frekuensi output VCO. Perbedaan fase dari phase detector dipakai untuk mengemudikan VCO. VCO di dalam XR-2211 bekerja berdasarkan jumlah arus yang mengalir pada R_o dan R_1 . R_o dipakai untuk menentukan frekuensi free-running dari VCO. Sedangkan R_1 merupakan salah satu komponen pembentuk filter LPF.

Tegangan referensi pada pin 10 secara internal sudah dibias pada level tegangan tertentu, yaitu :

$$V_R = V_{cc}/2 - 0,65 \text{ Volt}$$

Pin 10 harus dihubungkan dengan kapasitor 0,1 μ F agar XR-2211

dapat beroperasi dengan baik. Output Loop Phase Detector (Pin 11) memiliki impedansi yang tinggi. PLL loop filter dibentuk oleh komponen R_1 dan C_1 yang dihubungkan pada Pin 11. Dengan tidak adanya sinyal input atau tidak adanya sinyal error maka tegangan DC pada pin 11 adalah menyamai tegangan tegangan pada pin 10 (V_R). Puncak ayunan tegangan pada output phase detector adalah sama dengan $\pm V_R$. Frekuensi free-running dari VCO ditentukan oleh R_0 dan R_0 . Frekuensi free-running dari VCO adalah : $f_0 = \frac{1}{R_0 C_0} \text{ Hz}^{23)}$

C_0 terletak antara pin 13 dan 14

R_0 terletak antara pin 12 dan ground

XR-2211 tidak memiliki terminal output VCO, VCO output secara internal telah terhubung pada bagian phase detector. Untuk keperluan pengaturan Free-running VCO dapat dilakukan dengan menghubungkan Pin 2 dengan pin 10. Pengukuran frekuensi free-running dilakukan pada pin 3 dengan memutuskan juga sinyal input yang masuk.

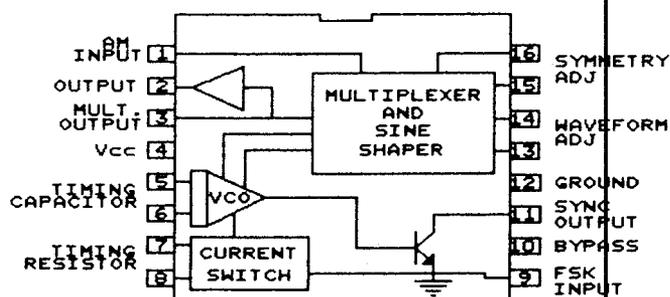
II.9. XR-2206 FSK MODULATOR²⁴⁾

XR-2206 merupakan sebuah generator fungsi yang dibangun dari rangkaian terintegrasi monolit dan mampu membangkitkan bentuk gelombang sinus, segi empat, dan gigi gergaji dengan kualitas tinggi dalam hal kemantapan dan kecermatannya.

23). Loc. cit.

24). Wasito suyono, PT Elex Media Komputindo Jakarta, 1992, Hal. 60.

Bentuk gelombang outputnya dapat dimodulasi amplitudo atau frekuensi. Frekuensi kerjanya berada pada jangkauan 0,01 sampai dengan 1 MHz. XR-2206 cocok sekali diaplikasikan sebagai modem. Bila XR-2206 diaplikasikan sebagai modem maka XR-2206 bertindak sebagai modulator FSK.



GAMBAR 2.12

DIAGRAM BLOK XR-2206

XR-2206 dapat dibagi menjadi empat blok fungsi yaitu osilator VCO, pembangkit gelombang sinus, penguat penyangga, dan perangkat saklar-saklar arus. Saklar arus ini digunakan untuk memilih resistor timing sesuai dengan logika pada input FSK.

BAB III PERENCANAAN ALAT

Perencanaan pembuatan tugas akhir ini terdiri dari dua bagian, yaitu perencanaan perangkat keras dan perangkat lunak. Perencanaan perangkat keras akan dibahas antara lain blok diagram, display, transduser, modulator FSK, demodulator FSK, pemancar, dan penerima. Sedangkan untuk perencanaan perangkat lunak dibahas mengenai diagram alir (flow chart) dan uraian cara kerja programnya.

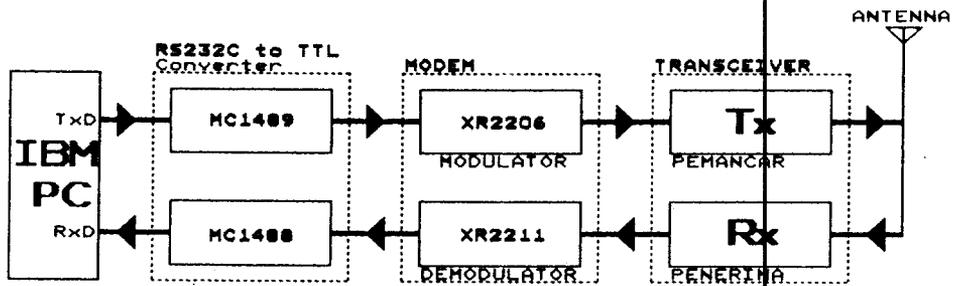
III.1. PERENCANAAN PERANGKAT KERAS

Perangkat keras dalam tugas akhir ini terdiri dari dua bagian utama yaitu perangkat sentral pemantau dan perangkat meter air pelanggan.

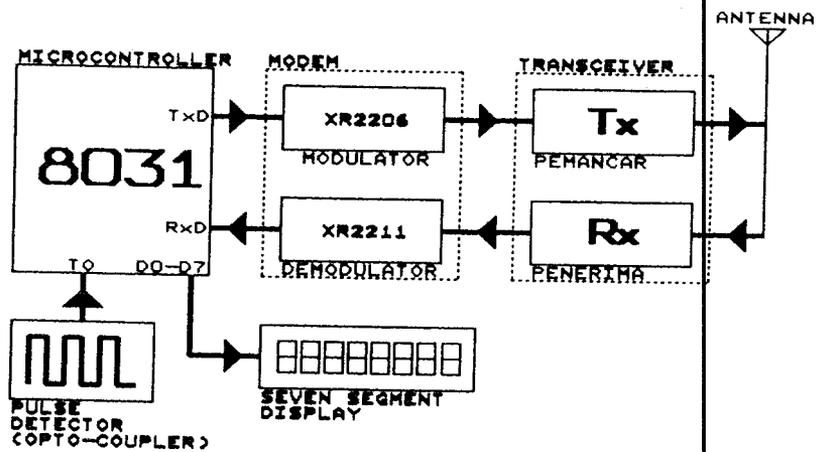
III.1.1 BLOK DIAGRAM

Blok diagram perangkat keras yang dibuat ditunjukkan seperti pada gambar 2.1. Blok rangkaian unit sentral pemantau terdapat juga pada unit meter air pelanggan, kecuali pada blok rangkaian serial interface RS-232-C.

Unit sentral berfungsi untuk memantau data volume air pada unit meter air pelanggan. Unit meter air pelanggan PDAM mencatat banyaknya volume air yang dilewatkan serta mengirimkan data ke unit sentral pemantau. Unit pelanggan mengirimkan data hanya bila menerima kode alamat yang sesuai dengan yang dimilikinya.



(a)



(b)

GAMBAR 3.1

DIAGRAM BLOK SISTEM

(a) UNIT SENTRAL PEMANTAU

(b) UNIT METER AIR PELANGGAN

IBM PC pada unit Sentral mengirimkan kode alamat dalam bentuk data digital yang dikirimkan secara serial. Melalui rangkaian serial interface RS-232-C data digital tersebut kemudian diubah ke level TTL. Data digital itu selanjutnya

dijadikan sinyal analog FSK dengan menggunakan IC XR-2206. Sinyal analog FSK dimodulasi sinyal pembawa VHF dan kemudian dipancarkan ke udara.

Mikrokontroler 8031 yang merupakan komponen utama dalam meter air selalu memeriksa kode alamat yang diterimanya. Bila ada sinyal dengan byte lengkap masuk pada kaki RxD maka mikrokontroler melakukan mengerjakan rutin program interupsi serial. Program interupsi serial ini berguna untuk memeriksa kode alamat. Mikrokontroler memerintahkan pengiriman data pemakaian air bila ternyata kode alamat yang diterima bersesuaian dengan kode yang dimilikinya.

Untuk mencatat volume air yang dipakai oleh pelanggan digunakan sensor photo-transistor. Photo-transistor dan LED diletakkan saling berhadapan. Photo-transistor akan mengeluarkan satu pulsa cacahan bila cahaya yang masuk photo-transistor dari LED kemudian dihalangi oleh sirip. Besar volume air yang mengalir adalah sebanding dengan jumlah cacahan yang dihasilkan oleh photo-transistor. Banyaknya cacahan yang dihasilkan oleh block rangkaian sensor diinputkan pada kaki interrupt timer T0.

Pada meter air dipasang peraga seven segment yang dipakai untuk mendisplaykan besar pemakaian air yang telah dilakukan pelanggan.

III.1.2. RANGKAIAN CLOCK

Rangkaian clock digunakan untuk menjalankan instruksi-instruksi pada mikrokontroler 8031. Frekuensi clock yang dapat diterima oleh mikrokontroler 8031 berada pada jangkauan 1,2 sampai dengan 12 MHz. Dengan frekuensi 12 MHz ini maka mikrokontroler 8031 memiliki siklus instruksi minimum sebesar $1\mu s$.

III.1.3. RANGKAIAN RESET

Untuk mereset mikrokontroler, kaki Reset/Vpd dijaga pada keadaan berlogika tinggi sedikitnya selama 24 periode osilator. Pada rangkaian ini frekuensi osilatornya adalah 12 MHz, sehingga periodanya adalah sekitar 83 ns. Jadi untuk mereset mikrokontroler, kaki Reset/Vpd harus dijaga pada keadaan tinggi paling sedikit selama

$$= 24 \times T_{\text{osilator}} = 24 \times 83 \text{ ns} = 1992 \text{ ns}$$

Untuk memenuhi kriteria itu maka rangkaian RC harus memenuhi persamaan : $V_{\text{reset}} = V_{\text{cc}} \cdot e^{-(t/RC)}$

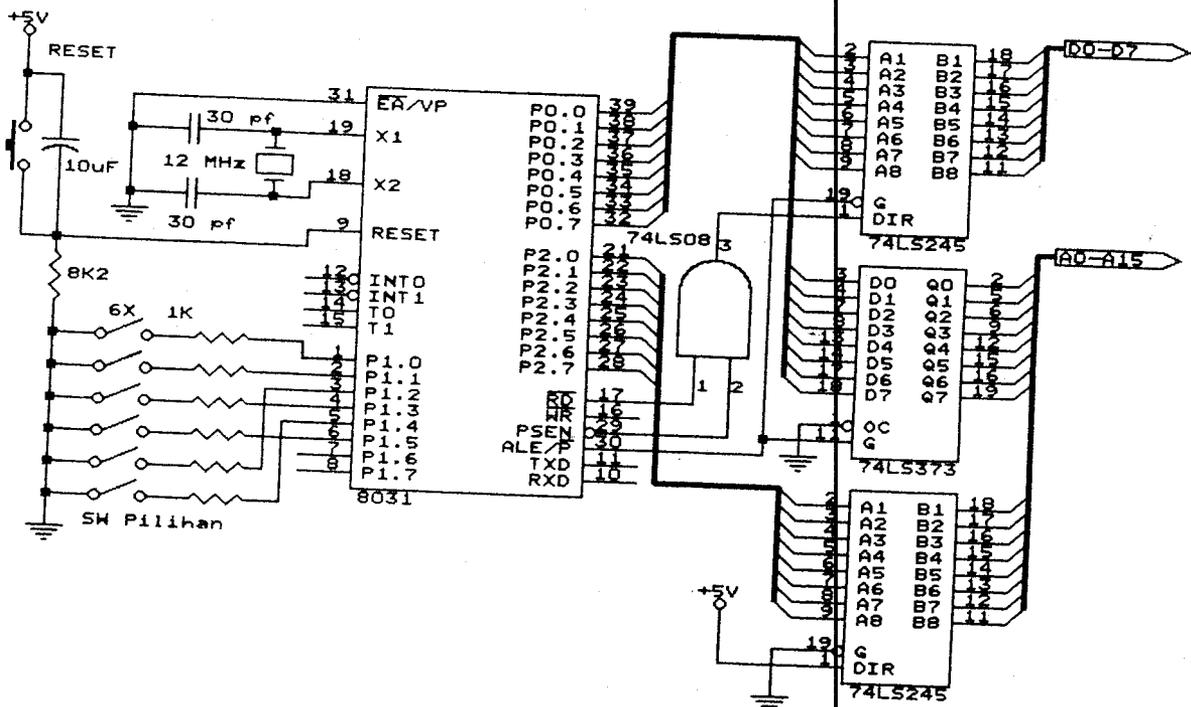
Dengan memakai harga $R = 8,2 \text{ K}\Omega$ dan t adalah penurunan tegangan dari 5 Volt ke 3 Volt maka penggunaan kapasitor sebesar $10 \mu F$ cukup memadai.

Pada saat saklar ditekan, kapasitor membuang muatannya. Tegangan reset naik mendekati tegangan sumber dan pada saat saklar reset dilepas tegangan reset turun secara ekponensial menuju ke ground. Setelah reset dilakukan, mikrokontroler menghentikan eksekusi instruksi dan tidak aktif. semua register direset ke keadaan awalnya (initial), dan operasi

normal dilakukan lagi mulai dari lokasi program 0000H.

III.1.4. RANGKAIAN BUFFER

Rangkaian buffer dan rangkaian multiplek data-alamat ditunjukkan pada gambar 3.2. Rangkaian multiplek data-alamat diperlukan karena bus data dan bus alamat rendah dari 8031 menjadi satu pada Port 0. Untuk keperluan ini digunakan IC 74LS373 dan IC 74LS245.



GAMBAR 3.2

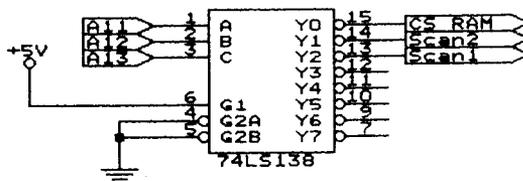
RANGKAIAN RESET, CLOCK, DAN BUFFER

IC 74LS373 berfungsi untuk menampung 8 bit alamat rendah (A0-A7). Kaki kontrol outputnya (\overline{OC} , kaki no 1) selalu pada kondisi rendah (dihubungkan ke ground), sedangkan kaki enable-nya (G, pin nomer 11) dihubungkan ke kaki ALE 8031 (pin nomer 30). Pada saat 8031 mengakses memori eksternal, ALE aktif, sehingga mengaktifkan enable 74LS373 dan me-latch bus alamat rendah ke sistem bus data rangkaian. IC 74LS245 fungsinya menampung 8 bit data (D0-D7). Kaki enable-nya (G, pin nomer 11) dihubungkan ke kaki ALE 8031, bila ALE aktif maka 74LS245 tidak berfungsi ketika ALE berlogika low maka 74LS245 bekerja untuk melewatkan data bus yang mana arah transfer data busnya tergantung kaki DIR (pin nomer 1). Apabila DIR berlogika high maka data dari mikrokontroler di transfer ke device luar sedangkan bila DIR berlogika rendah maka data dari device luar di-transfer ke mikrokontroler. Device dalam hal ini berupa RAM, ROM, serta IC TTL yang lainnya.

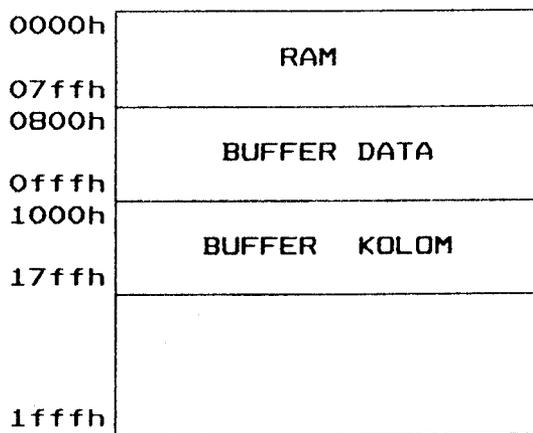
Port 2 berfungsi sebagai bus alamat tinggi, port 2 di-buffer dengan 74LS245. Pufferan ini diperlukan untuk menghindari pembebanan yang berlebihan. Pada Port 2 ini dihubungkan dengan lebih dari satu beban setara TTL (Port 2 maksimum men-drive 1 beban setara IC TTL). Pin enable-nya (G, kaki nomer 19) dihubungkan ke ground, dan pin kontrol arahnya (DIR) diberi kondisi high.

III.1.5. RANGKAIAN DECODER

Rangkaian decoder alamat mikrokontroler 8031 diperlihatkan pada gambar 3.3.

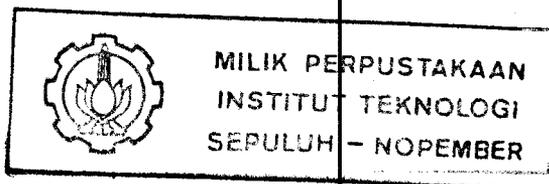


GAMBAR 3.3
DECODER ALAMAT 8031



GAMBAR 3.4
PEMETAAN MEMORI EXSTERNAL 8031

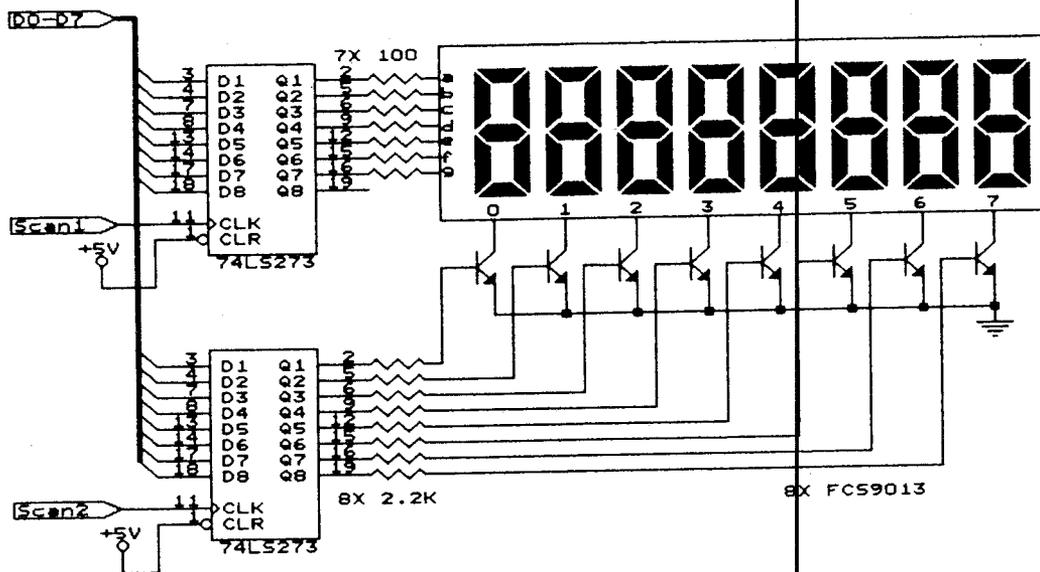
Rangkaian decoder ini digunakan untuk memilih salah satu device yang harus aktif. Alamat 0 - 7ffh ditempati oleh RAM, alamat 800h - fffh ditempati buffer data scanning display



seven segment, dan alamat 1000h - 17ffh ditempati buffer kolom display seven segment.

III.1.6. RANGKAIAN DISPLAY

Rangkaian display dipakai untuk menampilkan jumlah volume air yang telah disalurkan. Rangkaian display menggunakan seven segment yang penyalan dilakukan secara scanning agar perangkat keras menjadi sederhana dibanding penyalan dilakukan secara statis. Dalam sistem scanning ini tidak semua seven segment menyala secara bersamaan, dalam waktu tertentu hanya dapat dinyalakan satu segment saja.



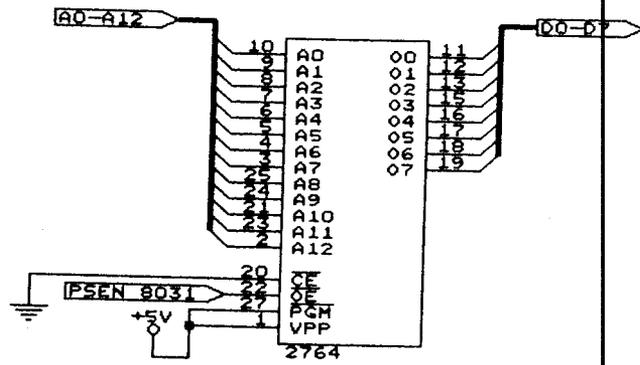
GAMBAR 3.5
RANGKAIAN DISPLAY

Untuk membuat efek semua seven segment kelihatan menyala maka dilakukan penyalan seven segment secara berturutan secara cepat sehingga mata tidak dapat mengikuti perubahan tersebut. Dalam proses scanning ini berarti bahwa suatu seven segment menyala 1/bag waktu yang diperlukan untuk menyalakan keseluruhan seven segment.

Pada gambar 3.5 memperlihatkan rangkaian display yang terdiri dari 8 buah seven segment common katoda. Kedelapan seven segment ini dikendalikan oleh IC 74LS273. IC 74LS273 merupakan jenis IC Data flip-flop. IC dipakai untuk menyimpan data yang didisplaykan ke seven segment.

III.1.7 RANGKAIAN MEMORI PROGRAM

Semua program yang berguna menjalankan instruksi-instruksi mikrokontroler disimpan dalam 8 KByte EPROM 2764. Kaki -OE(Output Enable) dari EPROM dihubungkan dengan sinyal -PSEN 8031 yang mengakibatkan IC ini baru aktif bila 8031 membaca memory program luar. Karena hanya digunakan satu ROM untuk semua program maka tidak diperlukan decoder alamat untuk memory program.



GAMBAR 3.6
MEMORI PROGRAM

RAM dipakai untuk menyimpan data yang sifatnya variabel. Data tampilan pada seven segment diambilkan data yang tersimpan pada RAM. Dalam perencanaan dipakai RAM sebesar 2 KByte. Untuk membaca dan menulis data pada ram digunakan sinyal kontrol -WR dan -RD.

III.1.8. RANGKAIAN SENSOR ALIRAN AIR

Fungsi utama dari rangkaian sensor aliran air adalah menghasilkan pulsa yang besarnya sebanding dengan aliran air yang telah melewati. Dalam rangkaian dipakai sensor dari jenis photo-transistor. Photo-transistor dan LED yang dipasang saling berhadapan. Cahaya yang masuk ke LED dapat dihalangi oleh sirip-sirip pada sumbu kincir. Dengan menggunakan photo-transistor, LED, dan sirip-sirip yang dipasang pada sumbu kincir, maka akan diperoleh pulsa setiap terjadi pemotongan sinar pada basis photo-transistor oleh

sirip yang ikut berputar.

Dari hasil pengukuran dimensi meter air didapatkan :

$R : 2.55 \text{ cm}$ dan $w : 1,47 \text{ cm}$.

Dengan memakai persamaan yang ada pada teori penunjang didapatkan :

$$\text{Volume} = \pi(R-R_0)^2 wc/3 \quad (\text{cm}^3)$$

atau dapat dituliskan :

$$\text{Volume} = k.c \text{ (cm}^3\text{)} \quad \text{dimana } k = \pi(R-R_0)^2 wc/3 \text{ (cm}^3\text{)}$$

R : jari-jari luar baling-baling (cm)

R_0 : jari-jari dalam baling-baling (cm)

w : lebar baling-baling (cm)

c : jumlah cacahan

$$\begin{aligned} \text{Volume} &= \pi.(2,55-0,15)^2.(1,47).c/3 \text{ (cm}^3\text{)} \\ &= 8.88 . c \text{ (cm}^3\text{)} \end{aligned}$$

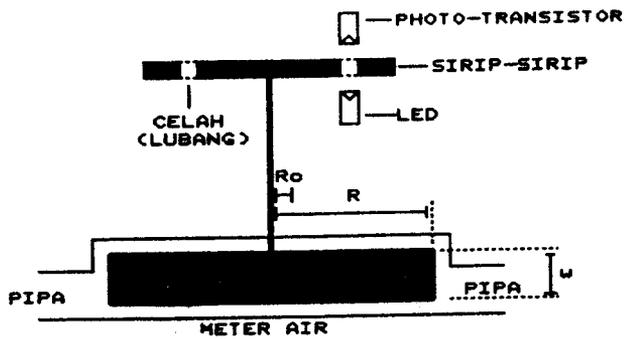
Dari hasil pengkalibrasian meter air didapatkan harga $k = 8,85$. Pengkalibrasian dilakukan dengan mengalirkan air 5 liter pada meter air. Dengan diketahuinya jumlah cacahan yang terjadi untuk aliran 5 liter air maka harga k dapat diketahui. Untuk menghitung k dalam hal ini dapat dipakai persamaan :

$$k = \frac{\text{Volume}}{c}$$

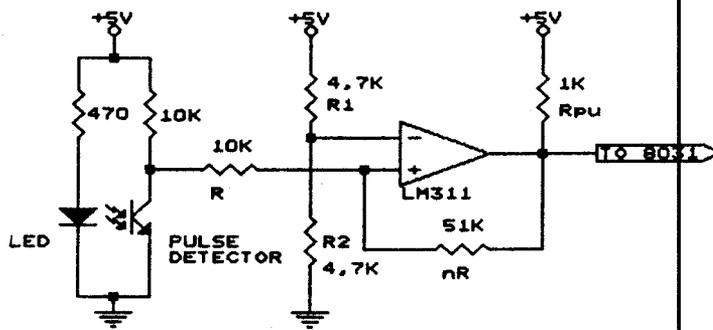
dimana c : jumlah cacahan untuk setiap 5 liter

Volume : volume yang dialirkan (5 liter)

Harga k yang Perhitungan k secara teoritis mendekati hasil



(a)



(b)

GAMBAR 3.7

SENSOR VOLUME AIR

(a) PELETAKAN SENSOR

(b) RANGKAIAN SENSOR

4. Menentukan harga n

$$n = \frac{+V_{sat} - (-V_{sat})}{V_{UT} - V_{LT}} = \frac{5 - (0)}{3 - 2} = 5$$

5. Dipilih harga R = 10kΩ maka nR = 51KΩ

6. Menentukan Vref

$$V_{ref} = \frac{V_{UT} - V_{LT}}{2} = \frac{3 + 2}{2} = 2,5 \text{ Volt}$$

Dipilih harga R₁ = 4,7KΩ maka

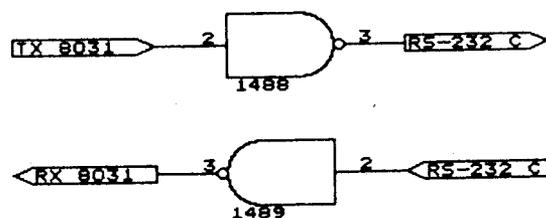
$$R_2 = \frac{R_2 \cdot V_{CC}}{V_{ref}} - R_2 = \frac{4700 \cdot 5}{2,5} - 4700 = 4700 \Omega$$

Tegangan referensi adalah sebesar 2,5 Volt dari hasil

pembagian tegangan oleh R_1 dan R_2 .

III.1.9. RANGKAIAN SERIAL INTERFACE RS-232-C

Sebelum data serial diterima oleh IBM PC, data tersebut harus diubah terlebih dahulu ke dalam standart level TTL dan sebaliknya sebelum data dari IBM PC ditransmisikan, harus diubah terlebih dahulu ke level RS-232-C. Untuk mengubah level RS-232-C menjadi standart TTL, dalam perencannaan ini digunakan MC1489 dan untuk mengubah level TTL ke level RS-232-C digunakan MC1488. Level TTL logika '0' didefinisikan pada level 0 - 0,4 Volt, logika '1' didefinisikan pada level 2,4 - 5 Volt. Pada standart level RS-232-C logika '0' didefinisikan pada tegangan (+3V) - (+15V), sedang logika '1' didefinisikan pada level (-15V) - (-3V). Rangkaian pengubah level tersebut diperlihatkan pada gambar 3.8.

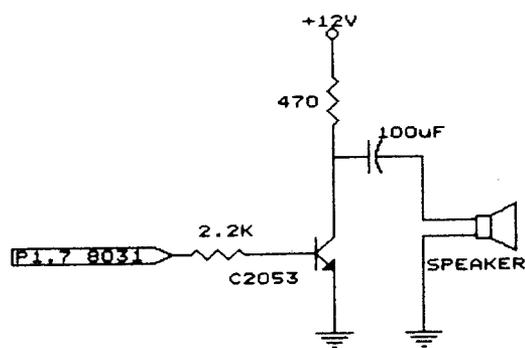


GAMBAR 3.8

RANGKAIAN SERIAL INTERFACE

III.1.10. RANGKAIAN ALARM

Rangkaian alarm dipakai untuk memberitahu pelanggan bahwa ada proses pengambilan data meter air dan pelanggan diharapkan segera membayar ke lokel PDAM. Rangkaian alarm diperlihatkan pada gambar 3.9. Transistor type 2 SC 2053 dipakai untuk mengemudikan speaker. Arus output port P1.7 microcontroller 8031 terlalu rendah untuk dipakai mengemudikan speaker secara langsung.



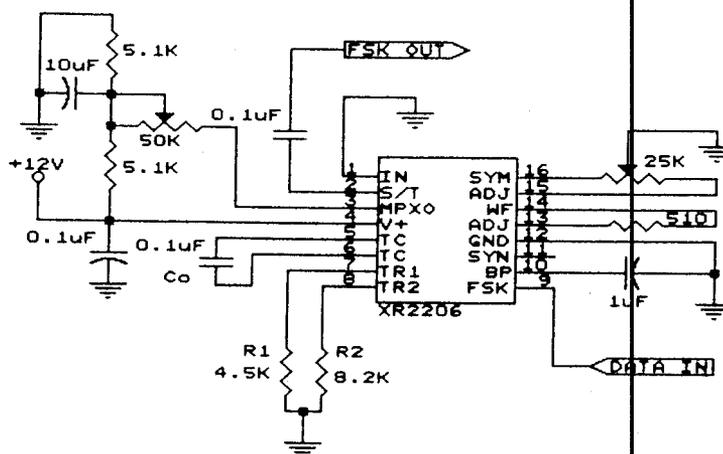
GAMBAR 3.9

RANGKAIAN ALARM

Untuk menggetarkan membran louspeaker diperlukan sinyal berupa pulsa-pulsa listrik pada input speaker. Pulsa-pulsa ini dikirimkan dari port P1.7 berupa data satu bit yang dibuat berosilasi dari low ke high secara terus menerus. Frekuensi osilasi akan menentukan frekuensi dari nada yang dihasilkan oleh speaker. Nada yang dikeluarkan speaker dapat diatur dengan mengatur perioda pulsa yang dikirimkan.

III.1.11. RANGKAIAN MODULATOR FSK

Rangkaian modulator FSK menggunakan komponen utama IC modulator XR-2206 seperti diperlihatkan pada gambar 3.10. Rangkaian modulator berfungsi untuk mengubah data digital menjadi sinyal analog. Tujuan perubahan data digital menjadi sinyal analog ini adalah agar data digital dapat ditransmisikan oleh perangkat pemancar (Tx). Perangkat pemancar dirancang hanya dapat menstranmisikan sinyal analog. Keluaran dari rangkaian modulator adalah sinyal analog yang termodulasi frekuensi digital atau disebut juga sebagai sinyal Frequency Shift Keying (FSK).



GAMBAR 3.10

RANGKAIAN MODULATOR FSK

Frekuensi output XR-2206 (pin 2) ditentukan oleh C_0 , R_1 , dan R_2 . Frekuensi space (logika '0') dirancang pada 2200 Hz dan frekuensi mark pada 1200 Hz. Pada sinyal digital

berlogika low saklar internal XR-2206 mengaktifkan R_1 sedangkan pada logika high akan mengaktifkan R_2 . Frekuensi output modulator ditentukan oleh persamaan berikut :

$$f_1 = \frac{1}{R_1 C_0}$$

$$f_2 = \frac{1}{R_2 C_0}$$

Pada perencanaan dipilih $C_0 = 0,1 \mu f$ sehingga didapatkan harga resistor untuk $f_1 = 2200 \text{ Hz}$ dan $f_2 = 1200 \text{ Hz}$ adalah $R_2 = 8.3333 \text{ K}\Omega$ dan $R_1 = 4.5456 \text{ K}\Omega$. Harga resistor hasil perhitungan disesuaikan dengan harga yang ada dipasaran, diperoleh $R_1 = 4.5 \text{ K}\Omega$ dan $R_2 = 8.2 \text{ K}\Omega$. Besar frekuensi dengan menggunakan $R_1 = 4.5 \text{ K}\Omega$ dan $R_2 = 8.2 \text{ K}\Omega$ adalah $f_1 = 2222 \text{ Hz}$ dan $f_2 = 1219 \text{ Hz}$.

III.1.12. RANGKAIAN DEMODULATOR FSK

Rangkaian demodulator FSK menggunakan komponen utama IC modulator XR-2211 seperti diperlihatkan pada gambar 3.11. Rangkaian demodulator berguna mengembalikan sinyal analog (FSK) menjadi sinyal digital. Agar komunikasi dapat berjalan dengan baik maka baud rate modul modulator FSK dan demodulator FSK dibuat sama. Dalam perencanaan ini dipakai 1200 bps. Baud rate sebesar 1200 merupakan baud rate tertinggi yang disarankan oleh pembuat IC XR-2211. Bila modul demodulator dapat bekerja baik pada baud rate 1200 bps maka modul ini dapat bekerja pula pada baud rate dibawah 1200 bps. Dengan adanya pertimbangan kecepatan data maka dalam perencanaan ini dipilih baud rate 1200 bps.

berikut :

1. Besar frekuensi mark space f_1 dan f_2 disesuaikan dengan frekuensi modulator yaitu $f_1 = 1219$ Hz dan $f_2 = 2222$ Hz. Dihitung harga frekuensi tengah dari f_1 dan f_2 .

$$f_o = (f_1 + f_2) / 2 = 1720 \text{ Hz}$$

2. Harga R_o dipilih sebesar 20K sedang harga yang diizinkan untuk R_o adalah 10K - 100K Ohm. Untuk keperluan pengaturan dipakai resistor tetap 18K Ω diseri dengan resistor variabel 5K Ω .

3. Perhitungan harga C_o

$$C_o = 1 / (R_o f_o) \quad \text{dimana } R_o = 20K\Omega \text{ dan } f_o = 1720 \text{ Hz}$$

$$C_o = 0,029 \mu\text{F} \text{ disesuaikan dengan harga yang ada } C_o = 0,027 \mu\text{f.}$$

4. Perhitungan R_1

$$R_1 = R_o (f_o / (f_1 - f_2))$$

$$R_1 = 34 \text{ K}\Omega \quad \text{dipakai } R_1 = 33 \text{ K}\Omega$$

5. Perhitungan C_1

$$C_1 = C_o / 4 \quad \text{dipakai } C_1 = 0,01 \mu\text{f}$$

6. Perhitungan C_F

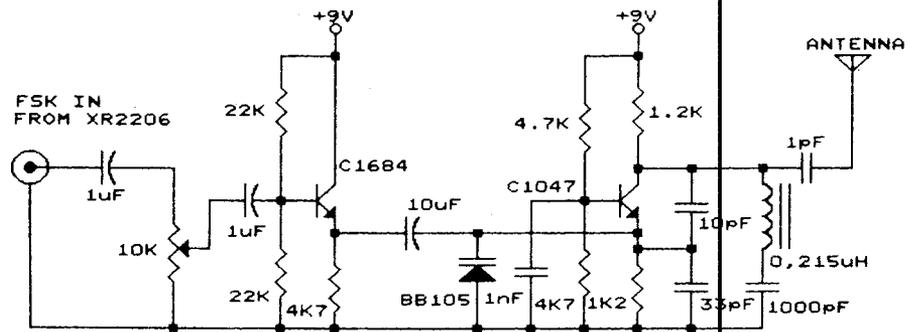
$$C_F = 3 / (\text{Baud rate})$$

$$= 3 / 1200 = 0,0025 \mu\text{f} \quad \text{dipakai } C_F = 0,0027 \mu\text{f}$$

III.1.13. RANGKAIAN PEMANCAR (Tx)

Rangkaian pemancar dirancang bekerja pada frekuensi 112 MHz. Rangkaian pemancar memakai osilator dari jenis claff. Rangkaian pemancar bertugas memodulasikan sinyal informasi

FSK ke sinyal pembawa VHF. Dengan memodulasikan sinyal informasi FSK ke jalur band VHF maka sinyal informasi FSK dapat ditransmisikan ke tempat lain melalui media udara. Jangkauan pancaran tergantung pada daya output pemancar.



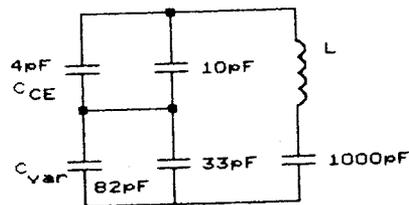
GAMBAR 3.12

RANGKAIAN PEMANCAR

Rangkaian pemancar diperlihatkan pada gambar 3.12. Input pemancar merupakan sinyal audio FSK. Sebelum sinyal informasi FSK yang merupakan sinyal audio diumpankan ke bagian osilator, sinyal audio FSK harus dilewatkan pada rangkaian buffer yang dibentuk oleh transistor C1684. Rangkaian osilator yang dibentuk oleh transistor C1047 menggunakan common base. Configurasi common base ini memiliki impedansi yang cukup rendah. Untuk menyesuaikan impedansi antara output modulator FSK (XR-2206) dengan bagian osilator ini diperlukan rangkaian penyesuai impedansi atau buffer.

Pembangkitan sinyal modulasi frekuensi (FM) yang

dilakukan pada rangkaian pemancar ini memakai dioda varaktor. Dengan memanfaatkan sifat dari dioda varactor yang dibias balik (reverse bias) maka untuk setiap perubahan amplitudo sinyal pemodulasi yang masuk akan mengakibatkan perubahan kapasitansi pada dioda tersebut. Akibat perubahan kapasitansi pada rangkaian LC akan menghasilkan perubahan frekuensi seiring dengan terjadinya perubahan kapasitansi.



GAMBAR 3.13

RANGKAIAN EKIVALEN OSILATOR

Besar frekuensi osilator ditentukan oleh harga L dan C dari rangkaian. Rangkaian osilator yang merupakan osilator dari jenis Clapp dapat disederhanakan seperti diperlihatkan pada gambar 3.13. Untuk menghitung besar frekuensi dipakai rumus sebagai berikut :

$$f = \frac{1}{2\pi\sqrt{LC}}$$

dimana C merupakan harga kapasitor total dari rangkaian. Dalam rangkaian ini harga C_{total} dan f diketahui sedangkan harga L belum diketahui. Dari perhitungan yang dilakukan didapatkan harga L = 0,162 μ H.

Titik kerja dari rangkaian buffer yang menggunakan transistor Q₁ adalah sebagai berikut :

Sumber tegangan pengganti Thevenin dari rangkaian input adalah 4,5 Volt, sedangkan resistansi penggantinya adalah 11 KOhm.

Pada loop input diperoleh :

$$V_{BB} = I_B \cdot R_B + V_{BE} + I_E \cdot R_E$$

$$4,5 = I_B \times 11 \times 10^3 + 0,6 + (1 + \beta_{dc}) \times I_B \times 4700$$

$$3,9 = 485700 \cdot I_B$$

$$I_B = 8,0296 \mu A$$

$$I_C = \beta_{dc} \cdot I_B = 802,96 \mu A$$

$$I_E = (1 + \beta_{dc}) \cdot I_B = 810,99 \mu A$$

Sedangkan pada loop output :

$$V_{CC} = V_{CE} + I_E \cdot R_E$$

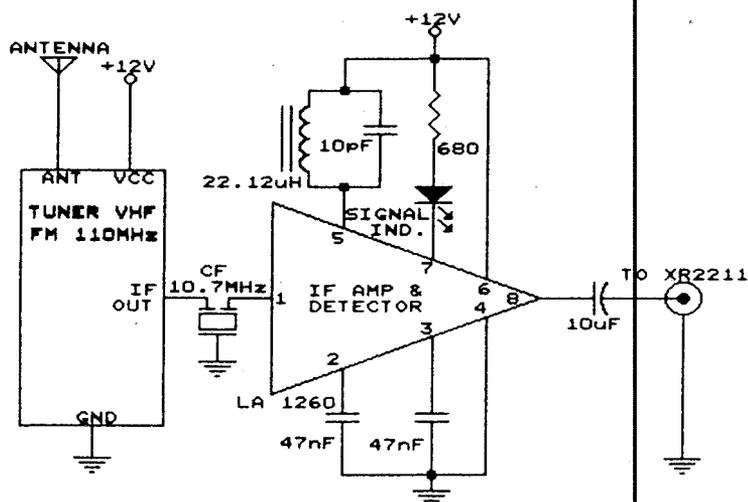
$$9 = V_{CE} + (810,99 \times 10^{-6} \times 4700)$$

$$V_{CE} = 5,1883 \text{ Volt}$$

Jadi titik kerjanya adalah : (5,1883 V; 801,99 μA)

III.1.14. RANGKAIAN PENERIMA (Rx)

Rangkaian penerima dirancang bekerja pada frekuensi 112 MHz sesuai dengan frekuensi kerja dari rangkaian pemancar. Fungsi utama dari rangkaian penerima adalah mendemodulasikan sinyal FSK dari sinyal pembawa VHF. Output sinyal FSK dari rangkaian penerima langsung diumpankan ke rangkaian demodulator FSK.



GAMBAR 3.14

RANGKAIAN PENERIMA

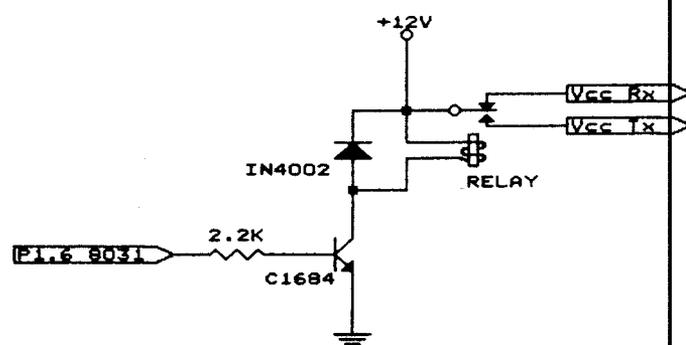
Rangkaian penerima menggunakan tuner VHF FM yang di-set pada frekuensi 112 MHz. Dalam tuner VHF FM sudah tercakup penguat RF, lokal osilator, dan mixer. Sinyal keluaran dari output tuner adalah sinyal yang memiliki selisih frekuensi antara sinyal input VHF dan sinyal osilator. Sinyal selisih tersebut dikenal sebagai sinyal frekuensi menengah (intermediate frequency). Untuk radio penerima FM sinyal frekuensi ini besarnya adalah 10,7 MHz sedangkan untuk radio penerima AM dipakai frekuensi 455 KHz.

Dalam sinyal frekuensi menengah sudah terkandung sinyal informasi FSK. Output Tuner VHF FM perlu dihubungkan dengan keramik filter 10,7 MHz. Keramik filter 10,7 MHz berlaku sebagai band pass filter (BPF) pada frekuensi tengah 10,7 MHz. Frekuensi yang berada jauh diluar frekuensi tengah 10,7

akan diredam. Sebelum sinyal frekuensi menengah di masukkan pada bagian detektor FM (diskriminator FM), sinyal frekuensi menengah yang memiliki amplitudo lemah dikuatkan terlebih dahulu. Rangkaian penerima menggunakan jenis slope detektor. Slope detektor mendemodulasikan sinyal informasi dari sinyal pembawa pada daerah lereng resonansi rangkaian L dan C.

III.1.15. RANGKAIAN PEMILIH PEMANCAR / PENERIMA

Sistem pemantauan meter air yang dirancang hanya bekerja pada satu jalur band frekuensi. Untuk merealisasikan hal ini maka perangkat pemancar dan penerima harus bekerja bergantian. Sistem pemantauan meter ini menggunakan sistem komunikasi half duplex. Dalam sistem komunikasi half duplex informasi disalurkan secara timbal-balik (dua arah) tetapi secara bergantian.



GAMBAR 3.15

RANGKAIAN PEMILIH Rx / Tx

Pergantian perangkat penerima dan pemancar diatur oleh rangkaian pemilih seperti ditunjukkan pada gambar 3.14. Pemilihan antara pemancar dan penerima dilakukan oleh kontak relay. Transistor 2 SC 1684 dipakai mengemudikan relay. Port pengontrol dari unit sentral pemantau adalah pin RTS (Request To Send) sedangkan unit meter air memakai Port P1.6. Dioda yang dipasang paralel dengan relay berfungsi menghilangkan tegangan transient yang dihasilkan oleh mati-hidupnya relay.

III.1.16. SWITCH PEMILIH

Unit meter air pelanggan diperlengkapi dengan switch-switch dengan fungsi khusus. Switch-switch pemilih dihubungkan pada port 1 mikrokontroler 8031. Port 1 tidak perlu dihubungkan dengan resistor pull up sebab port 1 sudah memiliki resistor pull up didalam serpih 8031. Switch pilihan dapat dipakai untuk memilih penunjukkan dalam meter cubic atau liter, kode alamat, dan pemeriksaan komunikasi data.

Meter air yang dibuat direncanakan untuk dapat diakses melalui 2 kode alamat (password) yang berbeda sesuai dengan kondisi switch yang dipilih. Kode alamat tersebut adalah '123777' dan '123000'. Kode alamat itu ditentukan oleh program yang terisi pada EPROM.

Untuk mendapatkan data yang ditunjukkan oleh meter air maka pertama kali unit sentral mengirimkan kode alamat sesuai dengan meter air yang dituju. Meter air yang alamatnya sesuai dengan kode alamat yang dikirimkan segera setelah itu

mengirimkan data debit airnya ke unit sentral. Sedangkan meter air yang alamatnya tidak bersesuaian dengan kode alamat yang dikirimkan tidak memberikan sinyal balasan.

Dengan menggunakan sistim pengkodean alamat ini menghasilkan penghematan yang besar terhadap band frekuensi karena satu jalur frekuensi dapat dipakai untuk memantau banyak meter air pelanggan PDAM. Tanpa sistem pengkodean alamat dapat menyebabkan meter air harus mempunyai frekuensi kerja yang berbeda dengan meter air lainnya.

III.2. PERENCANAAN PERANGKAT LUNAK

Untuk menjalankan keseluruhan perangkat keras diperlukan perangkat lunak yang memberikan perintah pada peralatan. Perangkat lunak yang dibuat terbagi dalam dua bagian utama yaitu :

1. Perangkat lunak unit sentral pemantau
2. Perangkat lunak microcontroller 8031

III.2.1 PERANGKAT LUNAK UNIT SENTRAL PEMANTAU

Perangkat lunak unit sentral pemantau berguna untuk mengendalikan perangkat keras unit sentral. Perangkat lunak ini dibuat dalam bahasa Pascal. Dalam bahasa pascal banyak tersedia fasilitas instruksi untuk mengakses hardware IBM PC. Disamping itu bahasa ini juga cukup memadai dipakai dalam manajemen data.

Tugas utama dari perangkat lunak unit sentral adalah mengambil data dari beberapa meter air. Untuk mendapatkan data yang ada maka perangkat lunak harus mengirimkan kode alamat sesuai dengan dengan meter air yang dituju. Hanya meter air yang alamat sama dengan kode alamat yang terkirim yang segera memberikan respon, sedangkan meter lain yang kode alamatnya tidak sama hanya akan menerima kode alamat tanpa memberikan respon untuk mengirimkan sinyal balasan.

Agar komunikasi dapat berjalan dengan baik antara peralatan sentral dengan 8031 maka kedua baud rate harus memiliki nilai yang sama.

Seperti digambarkan pada diagram flowchart pada awal program dilakukan proses inisialisasi serial interface RS-232. Kode alamat yang dikirimkan dimasukkan dari keyboard Setelah proses pengiriman kode alamat maka program akan melakukan proses penerimaan data volume air yang tercatat di meter air. Data yang diterima merupakan data Volume (pemakaian air oleh pelanggan). Dengan mengalikan Data Volume dengan harga tiap meter cubic didapatkan jumlah tagihan yang harus dibayarkan oleh pelanggan. Jumlah tagihan ini kemudian dikirimkan ke pelanggan sesuai dengan kode alamat pelanggan yang dituju. Data pemakaian air dan jumlah tagihan dapat ditampilkan di layar monitor dan selanjutnya data tersebut dapat disimpan di disket ataupun dicetak ke printer bila diperlukan.

III.2.2 PERANKAT LUNAK 8031

Perangkat lunak 8031 dibuat sesuai dengan bahasa dari tipe mikrokontroler 8031 yaitu assembler 8031. Dengan menggunakan bantuan perangkat lunak *MCS-51 cross assembler*, program assembler yang dibuat diubah ke dalam kode hexadesimal. Selanjutnya dengan menggunakan perangkat lunak HEXBIN kode hexadesimal ini dikonversikan ke kode biner yang merupakan bahasa mesin. Bahasa mesin ini kemudian diisikan ke dalam EPROM.

Pada awal program dilakukan inisialisasi microcontroller dan setting variabel. Pada kondisi awal dijalankan volume yang ditampilkan di bereri nilai nol. Setelah melakukan inisialisasi program melakukan pembacaan alamat sesuai dengan switch address yang dipilih pada port 1. Seperti yang diperlihatkan pada flowchart, program utama hanya bertugas melakukan proses inisialisasi dan menampilkan data volume ke peraga seven segment.

Prosedur interrupt selalu aktif memeriksa setiap data yang masuk pada port serial. Setiap data yang masuk akan dibandingkan dengan kode alamat yang dipakai. Bila kemudian diterima kode alamat yang sesuai maka segera dilakukan proses pengiriman data volume air yang sudah tercatat ke pemancar. Setelah proses pengiriman data dikerjakan, data volume diberi nilai nol.

Prosedur counter berguna melakukan proses perhitungan jumlah volume air yang dialirkan oleh meter air. Perhitungan

besar volume air yang mengalir pada meter air didasarkan pada pulsa cacahan yang dihasilkan dari rangkaian sensor. Semakin besar cacahan maka hasil penjumlahan volume air menjadi semakin besar pula. Jumlah cacahan yang masuk sebanding dengan volume air yang telah dialirkan.

Perangkat lunak microcontroller dapat di golongkan menjadi dua bagian yaitu perangkat lunak komunikasi dan perangkat lunak untuk akuisisi data.

III.2.2.A PERANGKAT LUNAK KOMUNIKASI

Perangkat lunak komunikasi berguna untuk mengadakan jalinan komunikasi dengan peralatan induk serta menyeleksi alamat yang diterimanya untuk disesuaikan dengan alamat yang dimilikinya.

Pada bagian program ini dilakukan inisialisasi pada register-register SFR yang digunakan, yaitu register Interrupt Enable dan register Serial Port Control. Register Interrupt Enable untuk komunikasi serial diaktifkan (di-set). Register Serial Port Control di-set untuk mode 1 receive dan transmit dengan baud rate yang ditentukan oleh timer 1, pada frekuensi kristal osilator 12 MHz. Inisialisasinya adalah sebagai berikut :

```
MOV IE,#10010111B
```

```
MOV IP,#00000000B
```

```
MOV SCON,#01010000B
```

Selanjutnya dilakukan inisialisasi stack pointer, dan

inisialisasi baud rate oleh timer 1.

$$TH1 = 256 - \frac{2^{SMOD} * FreqOSC}{384 * BaudRate}$$

Dengan SMOD = 1, frekuensi osilator 12 MHz dan baud rate 1200 bps diperoleh TH1 = 204.

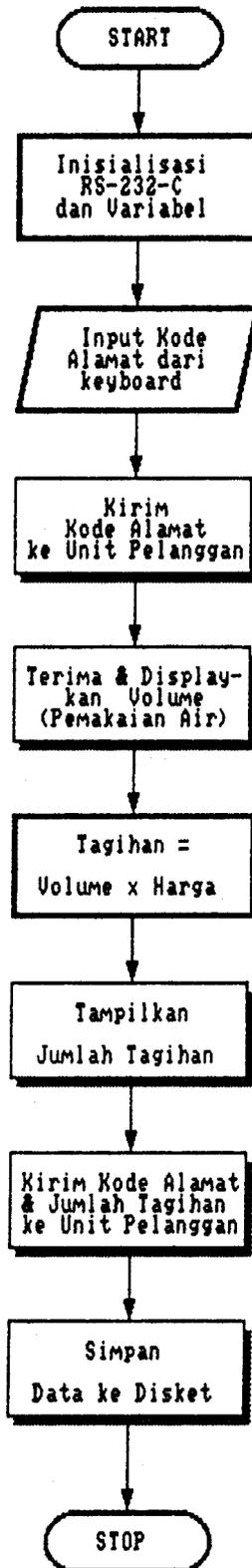
III.2.2.B PERANGKAT LUNAK AKUSISI DATA

Perangkat lunak akusisi data memproses data dari transduser. Perangkat lunak akusisi data dipakai untuk mengambil data dari transduser, mengkonversikan menjadi satuan meter cubic, serta menampilkan pada seven segment.

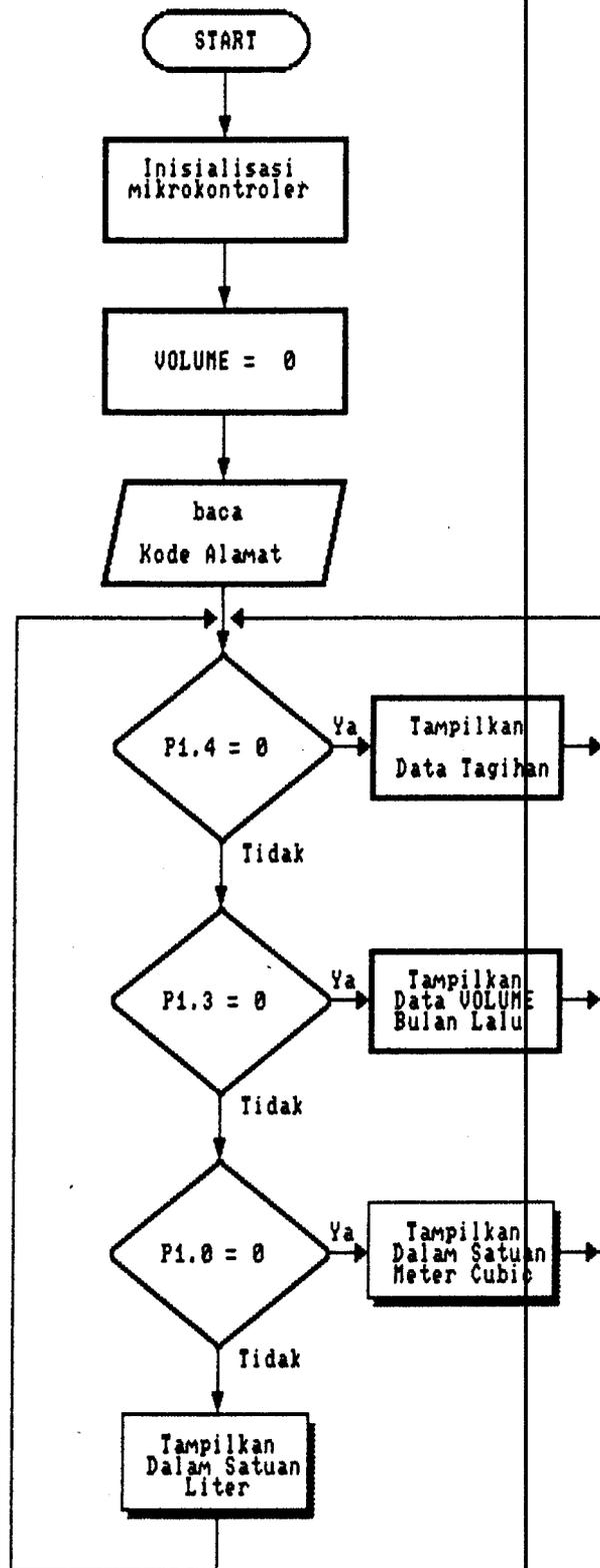
Dari sub bab perencanaan didapatkan persamaan perhitungan volume air yang dialirkan oleh meter air.

$Volume = (8,8889).c \text{ cm}^3$ dimana c = jumlah cacahan
 Dari persamaan di atas dapat diperoleh bahwa jumlah cacahan adalah berbanding lurus dengan volume air yang dialirkan. Dengan mengetahui jumlah pulsa cacahan yang masuk maka jumlah volume air akan diketahui.

Microcontroller 8031 memiliki perangkat counter. Perangkat counter 8031 yang dihubungkan pada output rangkaian sensor akan melakukan proses pencacahan bila ada pulsa yang keluar dari output rangkaian sensor. Satu cacahan mewakili volume sebesar $8,8889 \text{ cm}^3$. Semakin banyak cacahan yang telah terjadi menunjukkan besar volume air yang telah dilewatkan. Perangkat lunak dalam 8031 bertugas menjumlahkan konstanta 8,8889 setiap kali ada cacahan. Instruksi yang dipakai adalah $Volume_total = Volume_total + 8,8899 \text{ cc}$.

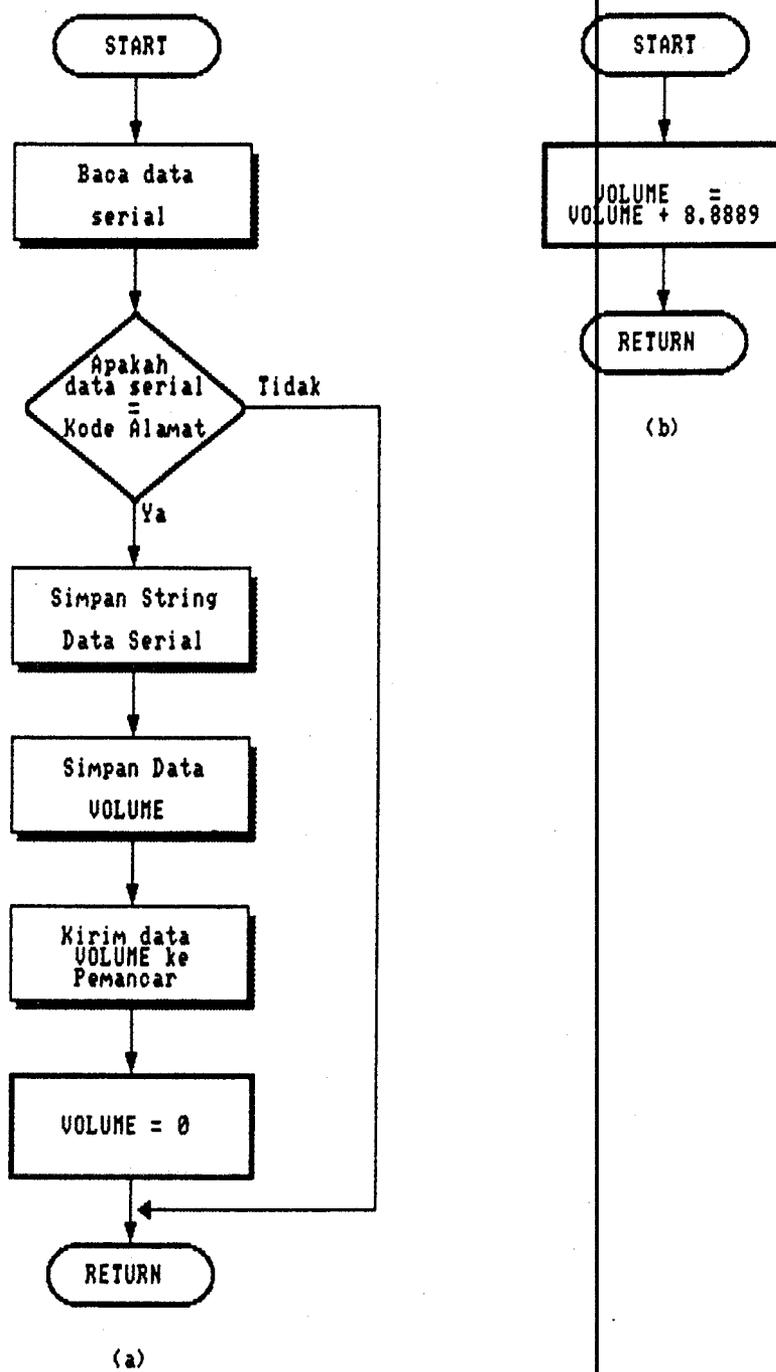


GAMBAR 3.16
DIAGRAM ALIR PROGRAM PADA IBM PC



GAMBAR 3.17

DIAGRAM ALIR PROGRAM UTAMA PADA UNIT PELANGGAN



GAMBAR 3.18

DIAGRAM ALIR INTERRUPT ROUTINE PADA UNIT PELANGGAN

(a) SERIAL INTERRUPT ROUTINE

(b) COUNTER INTERRUPT ROUTINE

BAB IV

PENGUJIAN DAN PENGUKURAN ALAT

Setelah pembuatan alat selesai perlu dilakukan pengujian dan pengukuran alat. Pengujian dan pengukuran alat ini dikerjakan untuk mengetahui apakah alat yang telah dibuat bekerja sesuai dengan yang direncanakan. Sebelum menguji keseluruhan peralatan, terlebih dahulu dilakukan pengujian setiap blok sistem. Pengujian difokuskan pada komunikasi data dari komputer ke meter air serta sebaliknya dari meter air ke komputer melalui jalur band VHF. Pengukuran alat dititikberatkan pada pengukuran volume air yang telah dilewatkan meter air.

IV. 1. PENGUJIAN PROGRAM KOMUNIKASI

Untuk menguji program komunikasi maka data digital pada port Serial IBM PC dihubungkan ke port Serial 8031 melalui rangkaian serial interface. Tujuan hubungan data digital secara langsung ini adalah untuk menghindari adanya kesalahan pada waktu penstransmisian data.

Peralatan meter air serta sentral pemantau diperlengkapi dengan mode pengujian pengiriman dan penerimaan data. Pada mode pengujian data ini, data yang dikirimkan oleh sentral pemantau dapat diamati pada display meter air serta juga sebaliknya data karakter yang dikirimkan oleh meter air dapat diamati pada display layar monitor IBM PC. Dengan

memanfaatkan mode pengujian tersebut maka adanya kesalahan pada proses penstransmisi data yang terjadi dapat diketahui dengan mudah.

Baud rate sebesar 9600 pada hubungan digital secara langsung dapat dicapai tanpa adanya kesalahan data. Baud rate dapat diubah-ubah melalui inisialisasi program serial pada IBM PC dan mikrokontroler 8031.

IV.2. PENGUJIAN RANGKAIAN MODULATOR DAN DEMODULATOR FSK

Sebelum dilakukan pengujian menggunakan data digital, terlebih dahulu dilakukan pengukuran frekuensi yang mewakili keadaan 'high' dan 'low' dengan menggunakan frequency counter merk Leader LDC-823A. Sinyal FSK pada output modulator diamati melalui osiloskop Leader type 1021. Dengan menggunakan osiloskop maka dapat dilakukan pengaturan amplitudo maupun kesimetrian output modulator.

TABEL 4.1

HASIL PENGUKURAN FREKUENSI MODULATOR FSK

Frekuensi	Unit Sentral	Unit Pelanggan
f_1	1164	2184
f_2	1167	2192

Modem yang didisain dalam tugas akhir ini memiliki baud rate sebesar 1200 bps. Pengujian kemampuan modem yang terdiri

dari modulator (XR-2206) dan demodulator (XR-2211) dikerjakan dengan cara menghubungkan output modulator dengan input demodulator secara langsung. Input modulator diumpangkan sinyal digital dari port serial dan output demodulator dihubungkan pula ke port serial IBM PC. Sinyal input dan output modulator serta demodulator diamati juga melalui osiloskop. Baud rate dari sinyal digital diubah-ubah dari 50 bps sampai 9600 bps. Dari hasil pengamatan diketahui bahwa pengiriman dan penerimaan data dapat berjalan dengan baik (tidak ada kesalahan pengiriman dan penerimaan data) pada 1200 bps serta baud rate dibawahnya. Dengan pertimbangan kecepatan pengiriman data maka dalam tugas akhir ini dipakai baud rate sebesar 1200.

IV.3. PENGUJIAN RANGKAIAN PEMANCAR DAN PENERIMA

Peralatan komunikasi yaitu pemancar dan penerima diuji dengan mengumpangkan sinyal audio pada input pemancar. Sinyal audio ini didapatkan dari output modulator FSK. Sinyal output audio diamati lewat osiloskop sedangkan frekuensi osilator pemancar dan penerima dapat dilihat melalui frekuensi counter. Frekuensi osilator pemancar ditala pada 112 MHz. Penalaan dilakukan melalui induktor variabel pada rangkaian osilator pemancar.

Output rangkaian penerima mengeluarkan sinyal audio sesuai dengan sinyal yang masuk pada input pemancar, bila frekuensi kerja pemancar dan penerima adalah sama (match). Agar output pada penerima sama dengan input pada pemancar

perlu dilakukan penalaan frekuensi pada osilator pemancar.

IV.4. PENGUJIAN RANGKAIAN SENSOR

Rangkaian sensor diuji dengan cara menghalangi dan meneruskan sinar yang masuk pada photo-dioda. Output rangkaian sensor akan berlogika 'high' bila photo-dioda tidak mendapat sinar, dan berlogika 'low' bila mendapat sinar. Dengan cara menghalangi sinar dan meneruskan sinar dari LED ke photo-dioda maka pada output rangkaian sensor terbentuk pulsa cacahan. Sekali cahaya menerobos celah keping sirip harus mewakili satu cacahan. Bila didapatkan lebih dari satu cacahan, perlu dilakukan pengaturan rangkaian komparator.

IV.5. PENGUJIAN SISTEM SECARA KESELURUHAN

Unit sentral diletakkan pada jarak 2 meter dari unit meter air pelanggan. Meter air dibuat pada kondisi melakukan penghitungan volume air yang mengalir. Melalui unit sentral meter air dipanggil dengan menggunakan kode alamat yang dimiliki oleh meter air. Setelah proses pengiriman kode alamat oleh unit sentral maka meter air akan segera mengirimkan data volume pada meter air. Data volume ini kemudian diterima oleh unit sentral.

Bila unit meter air pelanggan menerima data kode alamat tidak sesuai dengan kode alamat yang dimilikinya maka meter air tidak akan memberikan tanggapan. Dalam hal ini meter air tidak melakukan pemancaran data balik ke unit sentral.

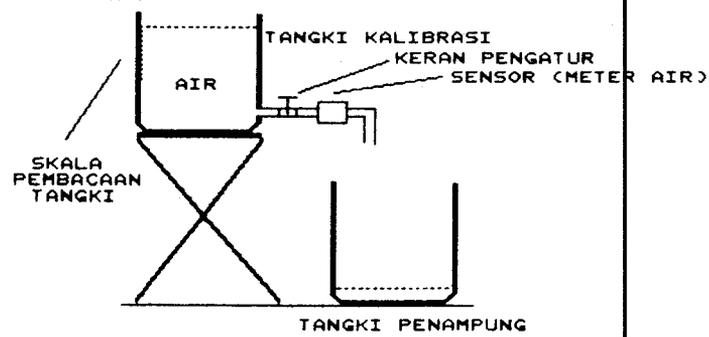
Dari hasil pengamatan yang dilakukan berulang kali dapat

diketahui bahwa komunikasi data melalui jalur band VHF dapat berjalan dengan baik bila frekuensi osilator local dari pemancar dan penerima stabil. Pemancar dan penerima yang dipakai di sini memiliki osilator dari jenis LC (induktor-capasitor). Kestabilan frekuensi osilator jenis LC ini sangat tergantung pada keadaan lingkungan yaitu suhu dan medan magnet.

Jangkauan pemantuan meter air diuji dengan cara menjauhkan letak unit pelanggan dengan letak unit sentral. Dari hasil pengujian jarak maksimal antara unit sentral dan unit pelanggan adalah 9 meter. Pada jarak lebih dari 9 meter maka akan terjadi banyak kesalahan penerimaan data. Jarak jangkauan ini dapat diperjauh dengan menambahkan penguat radio frekuensi (penguat RF) dengan daya yang besar.

IV.6. PENGUKURAN ALIRAN AIR

Pengukuran alat difokuskan pada pengukuran volume air yang dialirkan meter air. Pengukuran dilakukan untuk setiap volume 5 liter, pengukuran untuk setiap 1 m^3 (1000 l) memerlukan waktu yang cukup lama. Gambar 5.1 memperlihatkan proses pengukuran. Tangki penampung 1 diisi dengan air yang telah diketahui volumenya. Besar debit air yang mengalir ke meter air diatur melalui kran pengatur. Tabel 4.1 menunjukkan pengukuran volume air yang dilewatkan meter air untuk setiap 5 liter dengan debit (liter/detik) air yang berbeda-beda.



GAMBAR 5.1

PROSES PENGUKURAN VOLUME AIR

TABEL 4.2

PENGUKURAN VOLUME AIR UNTUK SETIAP 5 LITER AIR

No.	Debit (l/dt)	Pembacaan Meter air (l)
1	0,01	4,995
2	0,01	4,986
3	0,02	5,013
4	0,02	5,022
5	0,03	5,031
6	0,03	5,022
7	0,04	5,040
8	0,04	5,031
9	0,05	5,013
10	0,05	5,004

Dari tabel pengukuran didapatkan bahwa terjadi sedikit perbedaan antara volume air yang dialirkan ke meter air

dengan hasil pembacaan meter air. Perbedaan itu dikarenakan antara lain kekurangtelitian dalam melakukan pengukuran volume air dan juga adanya gaya moment pada sirip-sirip sensor aliran air sehingga sirip-sirip masih berputar meskipun aliran air sudah terhenti.

BAB V

PENUTUP

V.1 KESIMPULAN

Dari pembahasan, perencanaan serta pengujian dan pengukuran alat yang telah dikemukakan pada bab-bab sebelumnya, maka dapat diambil kesimpulan sebagai berikut :

1. Secara teoretis keuntungan penggunaan mikrokontroler 8031 dibandingkan dengan mikroprosesor yang umum seperti Z-80 atau 8088, antara lain rangkaian menjadi lebih sederhana, kecepatan eksekusi yang relatif tinggi, dan memiliki komponen di dalam serpih seperti timer, counter, RAM, serta I/O paralel maupun serial.
2. Ketelitian pembacaan volume air yang melewati meter air tergantung pergerakan kincir air pada bagian sensor, oleh karena itu bagian sensor harus dirancang seteliti mungkin agar tidak merugikan pihak PDAM maupun pihak pelanggan.
3. Satu jalur frekuensi dapat dipakai untuk memantau banyak unit meter air pelanggan, karena dipakainya pengkodean alamat. Banyak unit meter air yang dapat dipantau tergantung pada jarak jangkauan pemancar pada unit sentral dan unit meter air pelanggan.
4. Penggunaan osilator dari jenis rangkain L-C (induktor dan kapasitor) sangat rawan terhadap pengaruh perubahan suhu dan medan magnet disekitar osilator.
5. Jarak jangkauan sistem pemantauan dapat diperjauh dengan

cara memperbesar daya dari pemancar pada unit sentral dan unit meter air pelanggan.

V.2. SARAN-SARAN

Saran-saran yang diharapkan dapat berguna untuk penggunaan atau pengembangan lebih lanjut dari tugas akhir ini adalah sebagai berikut :

1. Penambahan sistem pendeteksian kesalahan terhadap penerimaan data sehingga adanya kesalahan data yang terjadi akibat gangguan selama transmisi data dapat diperbaiki.
2. Osilator jenis L-C pada bagian pemancar dan penerima sebaiknya diganti dari jenis osilator kristal karena osilator kristal memiliki kestabilan frekuensi yang lebih tinggi dibandingkan dengan osilator jenis L-C.
3. Agar penerimaan data menjadi lebih baik maka pada rangkaian demodulator perlu ditambahkan rangkaian band pass filter.
4. Sistem pemantauan meter air pelanggan ini harus dikembangkan lebih lanjut untuk mengatasi kendala kebutuhan power supply, power supply yang harus selalu tersedia terus-menerus, jangkauan pemancaran, serta biaya pembuatan peralatan yang tinggi.

DAFTAR PUSTAKA

-
1. HALL, DOUGLAS, V. Microprocessor and Interfacing, McGraw-Hill Inc, Singapore, 1986.
 2. HAROLD B. KILLEN, Modern Electronics Communication Techniques, Macmilan Publishing, Company, New York, 1980.
 3. MILLER, GARY M., Modern Electronic Communication, Prentice Hall Inc., New Jersey, 1980.
 4. NASHELSKY, LOUIS, Introduction to Digital Computer Technology, Second edition, USA, 1977.
 5. PEATMAN, JOHN B., Design With Microcontrollers, McGraw-Hill Inc, Singapore, 1988.
 6. ROBERT W. FOX, Introduction to Fluid Mechanics, John Willey and Sons Inc., Canada, 1985.
 7. SHANMUGAN, K. SAM, Digital and Analog Communication Systems, John Wiley & Sons, New York, 1979.
 8. TOMASI, W., Fundamental of Electronics Communication System, Prentice Hall, New Jersey, 1980.
 9. STERK, HENRY, Modern Electronic Communication Theory and System, Prentice Hall, New Jersey, 1980.
 10. WHARTON, JOHN, An Introduction to The Intel MCS-51 Single -Chip Microcomputer Family, Intel Corporation, Santa Clara, 1980.
 11. -----, Embedded Controller Handbock, Vol.1, Intel Coorporation, Santa Clara, U.S.A., 1988.
 12. -----, Fisika I, FMIPA-ITS, Surabaya, 1988.
 13. -----, Turbo Pascal 5.50 Reference Guide, Borland International Inc., 1987.

```

{*****}
{
{ Turbo Pascal }
{ Main Program on Sentral Unit (IBM PC AT) }
{ Tugas Akhir Maret 1994 }
{ 288 220 1115 }
{ Copyright (c) 1993, 1994 M. Abdul Rozaq }
{
{*****}

```

```
uses crt,Windows,Graphics;
```

```
var
I : integer;
AddressCode : string[6];
StringRxDot : string; DataRemain : real; StringRx : string;
```

```
procedure WaitRxTest;
var I : longint;
begin
for I := 1 to 60000 do if ((port[LineStatReg] and $01) <> 0) or keypressed then exit;
ErrorInfo(0,'Data not ready'); TimeKey(11,23); PopScr(11,8,68,17,2); end;
```

```
function Receive(LongBCD : byte) : string;
label PassWrd,PassWrd2;
var StringRx : string;
Nibble1,Nibble2 : byte;
begin
Init8051($61);
port[ModemContReg] := $00;
while keypressed do TimeKey(11,23);
Receive := '????????';
PassWrd:; WaitRxTest;
if keypressed then exit;
PassWrd2:; if port[RxBuffer] <> $ff then goto PassWrd;
WaitRxTest;
if keypressed then exit;
if port[RxBuffer] <> $ee then goto PassWrd2;
WaitRxTest;
if keypressed then exit;
if port[RxBuffer] <> $dd then goto PassWrd2;
StringRx := '';
for I := 1 to LongBCD do begin
WaitRxTest;
Nibble1 := port[RxBuffer];
Nibble2 := Nibble1;
StringRx := StringRx + chr((Nibble2 shr 4) + $30);
StringRx := StringRx + chr((Nibble1 and $0f) + $30); end;
Receive := StringRx;
end;
```



```

for H := 1 to SumOfData do
begin
  case H of
    15,29,43,57,71,85,99: begin delay(1000);
      SoundAction;
      PopScr(2,8,77,21,0); inc(K);
      RestoreFrame;
    end; end;
  W := H+7-14*K;
  str(H:3, DigitStr); Wrt(2,W,$fe and WC, DigitStr);
  str(C1m1[H]:6, AddressCode); Wrt(9,W,$fe and WC, AddressCode);
  C1m2[H] := Time; Wrt(19,W,$fe and WC, Time);
  str(C1m4[H]:4, DigitStr); Wrt(43,W,$fe and WC, '0000.000'); { last month }
  ControlSignal := #200; if LookOnlyFlag = true then ControlSignal := #240;
  Transmit(AddressCode, AddressCode, ControlSignal);
  StringRx := Receive(4);
  GetData;
  Wrt(31,W,$fb and WC, StringRxDot);
  val(StringRx, DigitResult, FaultCode); C1m3[Recrd] := DigitResult;
  C1m5[Recrd] := DigitResult; Wrt(54,W,$fb and WC, StringRxDot);
  if FaultCode <> 0 then begin
    ErrorInfo(0, 'Data Error');
    TimeKey(11,23); PopScr(11,8,68,17,2); end;
  C1m6[Recrd] := DigitResult;
  DigitResult := DigitResult* Waterprice;
  str(DigitResult:9, StringTx); Wrt(66,W,$fb and WC, StringTx);
  str(DigitResult, StringTx);
  case length(stringTx) of
    1: StringTx := #60+#60+#60+#60+#60+StringTx;
    2: StringTx := #60+#60+#60+#60+StringTx;
    3: StringTx := #60+#60+#60+StringTx;
    4: StringTx := #60+#60+StringTx;
    5: StringTx := #60+StringTx;
    6: StringTx := StringTx;
  end;
  if LookOnlyFlag = true then goto LookOnlyLbl;
  Transmit(AddressCode, StringTx, #255);
  LookOnlyLbl;
  if keypressed = true then goto FinalLbl;
end;
FinalLbl;
PushScr(0,1,79,23,1);
PushScr(0,1,79,23,0);
TimeKey(11,23);
KeyFlag := true;
end;

```

```

procedure Demo1M;
const S : string[26] = ' ';
label FinalLbl, GoBack1Lbl, GoBack2Lbl, LookOnlyLbl;
var DigitStr, StringTx : string[6]; Recrd : byte;
    FaultCode : integer; DigitResult : longint;
    ControlSignal : char;
begin
  SoundAction;
  PushScr(5,3,74,21,5);
  SadowWind(5,3,71,20,$7f,0);

```

```

Wrt(35,3,$7f,' Edit ');
Wrt(08,05,$70,'Record                               Pencatatan bulan lalu (mc)');
Wrt(08,06,$3f,'                                     '); Wrt(43,06,$3f,'                                     ');
Wrt(08,09,$70,'Kode alamat / password             Pencatatan sekarang (mc)');
Wrt(08,10,$3f,'                                     '); Wrt(43,10,$3f,'                                     ');
Wrt(08,13,$70,'Selisih pencatatan (mc)            Jumlah Tagihan (rupiah)');
Wrt(08,14,$3f,'                                     '); Wrt(43,14,$3f,'                                     ');
ADK(15,17); ACancel(34,17); AHelp(55,17);
GoBack1Lbl;
  gotoxy(09,07); val(StrInput(2),Recrd,FaultCode); Clm0[Recrd] := Recrd;
  if KeyCode = 27 then goto FinalLbl;
  if FaultCode <> 0 then begin Wrt(08,06,$3f,S); goto GoBack1Lbl; end;
  str(Clm4[Recrd],DigitStr); Wrt(43,06,$3c,'0000.000');
GoBack2Lbl;
  gotoxy(09,11); AddressCode := StrInput(6);
  val(AddressCode,DigitResult,FaultCode); Clm1[Recrd] := DigitResult;
  if KeyCode = 27 then goto FinalLbl;
  if FaultCode <> 0 then begin Wrt(08,10,$3f,S); goto GoBack2Lbl; end;
ControlSignal := #200; if LookOnlyFlag = true then ControlSignal := #240;
Information(0,'Transmit & receive processing');
Transmit(AddressCode,AddressCode,ControlSignal); PopScr(11,8,68,17,2);
  StringRx := Receive(4);
  if StringRx = '???????' then goto FinalLbl;
  GetData;
  Wrt(43,10,$3c,StringRxDot);
  Wrt(31,19,$da,'Data diterima');
  val(StringRx,DigitResult,FaultCode); Clm3[Recrd] := DigitResult+DataRemain;
  Clm5[Recrd] := DigitResult+DataRemain; Wrt(8,14,$3c,StringRxDot);
  if FaultCode <> 0 then begin
    ErrorInfo(0,'Data Error');
    TimeKey(11,23); PopScr(11,8,68,17,2); end;
  DigitResult := DigitResult* Waterprice + Round(DataRemain*WaterPrice);
  Clm6[Recrd] := DigitResult;
  str(DigitResult,StringTx);
  Wrt(43,14,$3c,StringTx);
  case length(stringTx) of
    1: StringTx := #60+#60+#60+#60+StringTx;
    2: StringTx := #60+#60+#60+#60+StringTx;
    3: StringTx := #60+#60+#60+StringTx;
    4: StringTx := #60+#60+StringTx;
    5: StringTx := #60+StringTx;
    6: StringTx := StringTx;
  end;
if LookOnlyFlag = true then goto LookOnlyLbl;
Information(0,'Transmit jumlah tagihan');
Transmit(AddressCode,StringTx,#255); PopScr(11,8,68,17,2);
LookOnlyLbl:
  Clm2[Recrd] := Time;
  FinalLbl;
  SumOfData := Recrd;
  TimeKey(11,23); KeyFlag := true; PopScr(5,3,74,21,5);
end;

procedure Demo2M;
var StringRx : string[10]; StringTx : string;
    Nibble1,Nibble2 : byte;
begin
  SoundAction;

```

```

InitGraphics;
Init8051($5c);
port[ModemContReg] := $03;
delay(100);
AddressCode := '123777';
repeat;
  Transmit(AddressCode,AddressCode,#224);
  StringRx := Receive(5);
  ConvertToGraph(StringRx);
until keypressed;
GotoText;
CursorOFF(CurOff1,CurOFF2);
PopScr(0,0,79,24,0);
end;

procedure MenuRun;
label InitLbl,FinalLbl,StartLbl;
begin
StartLbl::
PopScr(0,0,79,23,0);
Wrt(39,0,$20,' Run '); Dot(40,0,$24,'R');
BlackBox(39,1,58,5);
Wrt(40,2,$70,'   OKE   '); Dot(45,2,$74,'0');
Wrt(40,3,$70,'   Demo 1  '); Dot(52,3,$74,'1');
Wrt(40,4,$70,'   Demo 2  '); Dot(52,4,$74,'2');
PushScr(39,1,58,5,1);
  Case Arr4 of
0: begin Wrt(40,2,$20,'   OKE   '); Dot(45,2,$24,'0'); end;
1: begin Wrt(40,3,$20,'   Demo 1  '); Dot(52,3,$24,'1'); end;
2: begin Wrt(40,4,$20,'   Demo 2  '); Dot(52,4,$24,'2'); end;
  end;
InitLbl::
if KeyFlag = false then TimeKey(11,23);
KeyFlag := false;
EnterFlag := false;
case KeyCode of
328: dec(Arr4);
336: inc(Arr4);
79,111: begin Arr4 := 0; EnterFlag := true; end; {0}
49: begin Arr4 := 1; EnterFlag := true; end; {1}
50: begin Arr4 := 2; EnterFlag := true; end; {2}
13: EnterFlag := true;
27,289,281,303,275,280,273,291,331,333,315,316,317,301,323,324: goto FinalLbl;
end;
  if Arr4 = $ff then Arr4 := 2;
  if Arr4 = 3 then Arr4 := 0;
case Arr4 of
0: begin
  PopScr(39,1,58,5,1);  {(0,0,79,12,1);}
  Wrt(40,2,$20,'   OKE   '); Dot(45,2,$24,'0');
  if EnterFlag = true then begin OKEM; goto StartLbl; end; end;
1: begin
  PopScr(39,1,58,5,1);  {(0,0,79,12,1);}
  Wrt(40,3,$20,'   Demo 1  '); Dot(52,3,$24,'1');
  if EnterFlag = true then Demo1M; end;
2: begin
  PopScr(39,1,58,5,1);  {(0,0,79,12,1);}

```

```

Wrt(40,4,$20, ' Demo 2 '); Dot(52,4,$24,'2');
if EnterFlag = true then Demo2M; end;
end;
goto InitLbl;
FinalLbl;; KeyFlag := true; end;

```

```

Procedure DetectKey;
begin case Arrow of
0: MenuZero; 1: MenuFile; 2: MenuPrint; 3: MenuVerify; 4: MenuRun;
5: MenuOption;
6: MenuWindows; 7: MenuHelp;
end;
end;
end;

```

```

===== }
{                               }
{           * MAIN PROGRAM *   }
===== }
label ScanMenuLbl,GoToDosLbl,TakeActionLbl,GetOutLbl;
begin
Init8051($5c);
LookOnlyFlag := false; SoundFlag := false;
Arrow := 1;
delay(10);
AniRatio := 4;
EnterFlag := true;
Wind(0,1,79,23,WC);
PushScr(0,0,79,24,0);
PushScr(0,0,79,24,1);
AboutOpeningProgram;
TimeKey(11,23);
Att(0,1,79,23,WC,176);
Att(0,23,2,23,WC,16); Wrt(3,23,$1f and WC,' Time = ');
PushScr(0,0,79,24,0);
PushScr(0,0,79,24,1);
KeyFlag := true;
ScanMenuLbl;
EnterFlag := false;
if KeyFlag = false then TimeKey(11,23);
KeyFlag := false;
case KeyCode of
289: Arrow := 1;   { Alt-F ,File  }
281: Arrow := 2;   { Alt-P ,Print  }
303: Arrow := 3;   { Alt-V ,Verify }
275: Arrow := 4;   { Alt-R ,Run   }
280: Arrow := 5;   { Alt-O ,Option }
273: Arrow := 6;   { Alt-W ,Window }
291: Arrow := 7;   { Alt-H ,Help  }
301: goto GoToDosLbl; { Alt-X ,Exit }
315: begin EnterFlag := true; MenuHelp; TimeKey(11,23); end; { F1 Help }
316: SaveM; { F2 Save }
317: begin OpenM; TimeKey(11,23); end; { F3 Open }
323: OKEM; { F9 Run }
324: begin goto TakeActionLbl; end; { F10 Menu }
331: dec(Arrow);
333: inc(Arrow);
13: EnterFlag := true;

```

```
324: goto TakeActionLbl;
27: begin PopScr(0,0,79,23,0); goto ScanMenuLbl; end;
75,107: goto TakeActionLbl; { O K }
  else goto ScanMenuLbl; end;
if (Arrow = $ff) then Arrow := 7;
if (arrow = 8) then Arrow := 0;
TakeActionLbl;;
DetectKey;
goto ScanMenuLbl;
GoToDosLbl;;
PushScr(11,8,68,17,4); { at mode 3 }
Information(1,FileName + ' not Save. Save? (Y/N)');
TimeKey(11,23); if upcase(chr(KeyCode)) = 'Y' then begin SaveM;
  PopScr(11,8,68,17,4);
  if ErrorCode <> 0 then goto ScanMenuLbl;
  goto GetOutLbl; end;
PopScr(11,8,68,17,2);
GetOutLbl;;
SoundAction; CursorOn(6,7); textattr := $07; clrscr; writeln('all is done.');
```

```

; *****
; *
; * 8031 Assembler on Subscriber Unit *
; * Tugas Akhir, Maret 1994 *
; * 288 220 1115 *
; * Copyright (c) 1993, 1994 M. Abdul Rozaq *
; *
; *****

```

```

AlarmFlag bit 20h.0      ; bit area
MsgOrDgt bit 20h.1
DatInFlag bit 20h.2
TxFlag bit 20h.3
RxDirFlag bit 20h.4
HoldOnFlg bit 20h.5
Addres0 data 21h        ; 21h-26h, for address / password code
RxHold data 27h         ; variable location
AccSend data 28h
Locate data 29h
DgtMinus data 2ah       ; 2bh-2fh unused
SevSeg0 data 30h        ; 30h-38h, buffer SevSeg scanning, 38's reserved
Digit0 data 39h         ; 39h-3fh, buffer BCD data
Messeg0 data 40h        ; 40h-47h, buffer Messege data
Memory data 48h         ; 48h-4dh, buffer BCD Memory

```

```

org 0000h
ljmp Init
org 03h
ljmp Int03h
org 0bh
lcall Int0bh
reti
org 13h
ljmp Int13h
org 23h
ljmp Int23h

```

```

org 100h
Init: mov sp,#50h
      mov 20h,#00h      ; init boolean bit
      mov ie , #10010111b ; init interrupt routine
      mov ip , #00000010b ; init interrupt priority
      mov scon,#01010000b ; init serial routine
      mov 87h,#10000000b ; init pcon
      mov th1,#204      ; 1200 baud on oscilator 12 MHz
      mov tmod,#00100110b ; timer 1 mode 2, counter 0 mode 2
      mov tcon,#01010001b ; to run timer 1 and counter 0
      mov th0,#0ffh    ; autoreload for counter 0
      mov t10,#0ffh
      mov pl,#03fh     ; init I/O port, enable Rx Relay
      mov Addres0,#01h ; 21h-26h ---> init password area
      mov 22h,#02h
      mov 23h,#03h
      mov 24h,#07h
      mov 25h,#07h

```

```

mov 26h,#07h
mov Locate,#00h      ; init variable
mov AccSend,#00h
mov DgtMinus,#00h
setb p1.0
jb p1.0,Start       ; verify all incoming data or not
setb RxDirFlag
Start: acall MakeZero
setb p1.1           ; read from RAM or init counter
jb p1.1,MainProg
mov r0,#Digit0     ; init counter digit
mov r2,#07h
mov dptr,#00h
RamToDgt: movx a,@dptr      ; copy data from RAM to Volume (DGT)
mov @r0,a
inc r0
inc dptr
djnz r2,RamToDgt
mov r0,#Messeg0
mov r2,#08h
mov dptr,#10h
RamToMsg: movx a,@dptr     ; copy data from RAM to Messege
mov @r0,a
inc r0
inc dptr
djnz r2,RamToMsg
mov r0,#Memory
mov r2,#06h
mov dptr,#20h
RamToMem: movx a,@dptr    ; copy data from RAM to Volume Memory
mov @r0,a
inc r0
inc dptr
djnz r2,RamToMem

MainProg: acall ChoseAdr   ; main program handle
acall Transmit2          ; transmit if HoldOnFlg is set
acall Copy2Ram
mov c,MsgOrDgt
orl c,/p1.5
jnc MaProg1             ; menu digit displ. or messege displ.
acall Mes2Sev
sjmp MaProg2
MaProg1: acall Dig2Sev
MaProg2: jb p1.5,MaProg3  ; erase MsgOrDgt status
clr MsgOrDgt
MaProg3: acall Scan       ; seven segment scanning
acall Transmit          ; transmit if TxFlag is set
EndSend: jb p1.4,ExMaProg ; check external counter simulator
acall IncCnt
ExMaProg: ajmp MainProg  ; repeat main program

ChoseAdr: mov 24h,#07h    ; address/password option
mov 25h,#07h            ; '123777' or '123000'
mov 26h,#07h
setb p1.2

```

```

        jb pl.2,OkayAdr
        mov 24h,#00h
        mov 25h,#00h
        mov 26h,#00h
OkayAdr: ret

Mes2Sev: mov 30h,#20h           ; display address or bill messege
        mov 37h,#20h
        mov r0,#SevSeg0+6       ; copy data from Messege to SevSeg (for $7000)
        mov a,Messeg0+3         ; getting control signal
        cjne a,#0cfh,AddrDis
        mov r0,#SevSeg0+5       ; copy data from Messege to SevSeg (for addr.)
        mov 37h,#4bh
        mov 36h,#7bh

AddrDis: mov r1,#Messeg0
        mov r2,#03h

Mes22:  mov a,@r1
        anl a,#0f0h
        swap a
        mov dptr,#SevTabl2      ; dec. to Sev. Seg. conversion
        movc a,@a+dptr
        mov @r0,a
        dec r0
        mov a,@r1
        anl a,#0fh
        mov dptr,#SevTabl2      ; dec. to Sev. Seg. conversion
        movc a,@a+dptr
        mov @r0,a
        dec r0
        inc r1
        djnz r2,Mes22
        mov r5,#5h

Wait:   setb AlarmFlag          ; make warning voice
        mov r3,#1fh

Sound1: acall Scan
        djnz r3,Sound1
        clr AlarmFlag
        mov r3,#39h

Sound2: acall Scan
        djnz r3,Sound2
        djnz r5,Wait
        clr MsgOrDgt
        ret

Dig2Sev: mov r0,#SevSeg0        ; copy data from Digit to SevSeg
        mov r2,#04h
        mov 33h,#20h
        mov r1,#Digit0+2
        jb pl.3,DigSev1
        mov r1,#Digit0+3
        ajmp DigSev3             ; start at high byte

DigSev1: cjne r0,#33h,DigSev2
        inc r0

DigSev2: mov a,@r1
        anl a,#0fh
        mov dptr,#SevTabl      ; dec. to Sev. Seg. conversion

```

```

        movc a,@a+dptr
        mov @r0,a
        cjne r0,#37h,Go600n
        mov @r0,#00h
Go600n: inc r0
        cjne r0,#33h,DigSev3
        inc r0
DigSev3: mov a,@r1
        anl a,#0f0h
        swap a
        mov dptr,#SevTabl ; dec. to Sev. Seg. conversion
        movc a,@a+dptr
        mov @r0,a
        inc r0
        inc r1
        djnz r2,DigSev1 ; be careful, ended at 38h
        ret

Transmit: jnb TxFlag,EndTran ; transmit string BCD digit to Tx
        setb pl.6
        mov r6,#099h
Trans1:  acall Scan
        djnz r6,Trans1
        clr es
        acall HandShak
        mov r6,#4
        mov r0,#3fh
NextTx1: mov sbuf,@r0
        jnb ti,$
        clr ti
        dec r0
        acall Delay
        djnz r6,NextTx1
        setb es
        mov r6,#20h
Trans2:  acall Scan
        djnz r6,Trans2
        clr pl.6
        clr TxFlag
EndTran: ret

Transmit2: jnb HoldOnFlg,EndTran2 ; transmit if HoldingTransmit is set
        setb pl.6
        clr es
        mov r6,#99h
TranScan: acall Scan
        djnz r6,TranScan
        acall HandShak
        mov r6,#5
        mov r0,#3fh
NextTx:  mov sbuf,@r0
        jnb ti,$
        clr ti
        dec r0
        acall Delay
        djnz r6,NextTx

```

```

        setb es
        clr pl.6
        clr HoldOnFlg
EndTran2: ret
HandShak: mov sbuf,#0ffh      ; make hand shaking 8031 with sentral
          jnb ti,$            ; software handshaking use passwords
          clr ti
          acall Delay
          mov sbuf,#0eeh
          jnb ti,$
          clr ti
          acall Delay
          mov sbuf,#0ddh
          jnb ti,$
          clr ti
          acall Delay
          ret

Int03h:  push acc              ; display from Volume memory to sev. seg#.
          push psw
          setb rs0              ; at bank 1
          clr rsl
          mov r0,#SevSeg0      ; copy data from Digit to SevSeg
          mov r2,#04h
          mov 33h,#20h
          mov r1,#Memory
          jb pl.3,MemSev1
          mov r1,#Memory+1
          ajmp MemSev3         ; start at high byte
MemSev1:  cjne r0,#33h,MemSev2
          inc r0
MemSev2:  mov a,@r1
          anl a,#0fh
          mov dptr,#SevTabl    ; dec. to Sev. Seg. conversion
          movc a,@a+dptr
          mov @r0,a
          cjne r0,#37h,GoGoMem
          mov @r0,#00h
GoGoMem:  inc r0
          cjne r0,#33h,MemSev3
          inc r0
MemSev3:  mov a,@r1
          anl a,#0f0h
          swap a
          mov dptr,#SevTabl    ; dec. to Sev. Seg. conversion
          movc a,@a+dptr
          mov @r0,a
          inc r0
          inc r1
          djnz r2,MemSev1      ; be careful, ended at 38h
          mov r5,#0ffh
ScanMem:  lcall Scan
          djnz r5,ScanMem
          pop psw
          pop acc
          reti

```

```

Int0bh:  push acc                ; timer/counter 0 interrupt routine
         push psw
         clr rs0                ; set bank 2
         setb rsl
         clr c
         mov a,#89h            ; adding with 8.888889
         add a,DgtMinus
         da a
         mov a,#88h
         addc a,Digit0
         da a
         mov Digit0,a          ; 10e-3 and 10e-4
         mov a,#88h
         addc a,3ah
         da a
         mov 3ah,a             ; 10e-1 and 10e-2
         mov a,#08h           ; in m1
         addc a,3bh
         da a
         mov 3bh,a
         jnc ExC0
         mov a,#01h           ; in m1 / 1
         add a,3ch
         da a
         mov 3ch,a
         jnc ExC0
         mov a,#01h           ; in 1
         add a,3dh
         da a
         mov 3dh,a
         jnc ExC0
         mov a,#01h           ; mC
         add a,3eh
         da a
         mov 3eh,a
         jnc ExC0
         mov a,#01h           ; mC
         add a,3fh
         da a
         mov 3fh,a
ExC0:    acall Copy2Mem
         pop psw
         pop acc
         ret

Int13h:  setb pl.6            ; test data transmit
         clr es
         mov sbuf,#01h
         jnb ti,$
         clr ti
         mov sbuf,#23h
         jnb ti,$
         clr ti
         mov sbuf,#45h
         jnb ti,$
         clr ti
         mov sbuf,#67h

```

```

        jnb ti,$
        clr ti
        mov sbuf,#89h
        jnb ti,$
        clr ti
        lcall Scan
        setb es
        reti

Int23h:  push acc           ; serial interupt routine
        push b
        push psw
        setb rs0           ; with bank 1
        setb rs1
        jnb ri,$
        mov RxHold,sbuf
        clr ri
        jb RxDirFlag,Busy  ; received and display all incoming data ?
        mov a,RxHold
        cjne a,#0aah,Skip  ; password referens
        mov AccSend,#00h  ; special security, stage 2
        mov Locate,#00h
        clr DatInFlag
Skip:    jnb DatInFlag,Leap
Busy:    mov a,Locate
        mov r1,#Messeg0
        add a,r1
        mov r1,a
        mov @r1,RxHold
        inc Locate
        mov a,Locate
        cjne a,#4,ExSeri
        jb RxDirFlag,GoThere2
        setb TxFlag
        mov a,Messeg0+3
        cjne a,#0b0h,GoThere1 ; the same as #224 on main program (Demo2)
        setb HoldOnFlg
        clr TxFlag
GoThere1: cjne a,#0cfh,GoThere2 ; receive billing (Demo1 & OK) > #255
        setb MsgDrDgt
        acall Copy2Mem
        acall MakeZero
        clr TxFlag
GoThere2: mov Locate,#00h
        clr DatInFlag
        ajmp ExSeri
Leap:    mov r0,#Addres0   ; special security, stage 1
        mov a,AccSend
        add a,r0
        mov r0,a
        mov a,RxHold
        mov b,@r0
        cjne a,b,Refuse
        inc AccSend
        mov a,AccSend
        cjne a,#06h,ExSeri
        setb DatInFlag

```

```

Refuse:  mov AccSend,#00h
ExSeri:  pop psw
         pop b
         pop acc
         reti

IncCnt:  push psw                ; counter simulator
         push acc
         mov a,#01h            ; in 1
         add a,3dh
         da a
         mov 3dh,a
         jnc ExCnt
         mov a,#01h            ; mc
         add a,3eh
         da a
         mov 3eh,a
         jnc ExCnt
         mov a,#01h            ; mc
         add a,3fh
         da a
         mov 3fh,a
ExCnt:   acall Copy2Mem
         pop acc
         pop psw
         ret

Copy2Mem: mov r0,#Digit0+2      ; copy Volume data to memory
         mov r1,#Memory
         mov r2,#06h
CopyMem:  mov a,@r0
         mov @r1,a
         inc r0
         inc r1
         djnz r2,CopyMem
         mov r0,#Memory
         mov r2,#06h
         mov dptr,#20h
MemToRam: mov a,@r0            ; copy memory to RAM
         movx @dptr,a
         inc r0
         inc dptr
         djnz r2,MemToRam
Copy2Ram: mov r2,#07h          ; copy Volume and Messege data to RAM
         mov r0,#Digit0
         mov dptr,#00h
DgtToRam: mov a,@r0            ; copy Volume data to RAM
         movx @dptr,a
         inc r0
         inc dptr
         djnz r2,DgtToRam
         mov r0,#Messeg0
         mov r2,#08h
         mov dptr,#10h
MsgToRam: mov a,@r0            ; copy messege to RAM
         movx @dptr,a
         inc r0

```

```
inc dptr
djnz r2,MsgToRam
ret
```

```
MakeZero: mov r0,#Digit0      ; init counter with data 0
           mov r2,#07h
InitDgt:  mov @r0,#00h
           inc r0
           djnz r2,InitDgt
           ret
```

```
Scan:     mov r7,#8           ; rutine scanning seven segment
           mov r0,#SevSeg0
ScanSev:  mov a,@r0
           mov dptr,#1000h
           movx @dptr,a
           mov a,r7
           mov dptr,#SevLoca  ; get positon from table
           movc a,@a+dptr
           mov dptr,#800h
           movx @dptr,a
           acall Delay
           inc r0
           djnz r7,ScanSev
           ret
```

```
Delay:    jb AlarmFlag,Del1   ; give delay and setting alara
           mov r2,#0ffh
           djnz r2,$
           sjmp ExDel
Del1:     setb pl.7
           mov r2,#1ch
           djnz r2,$
           clr pl.7
           mov r2,#9fh
           djnz r2,$
ExDel:    ret
```

```
SevTab1  db 0dfh,91h,3fh,0bdh,0f1h,0edh,0efh,99h,0ffh,0fdh
           db 20h,20h,20h,20h,20h,20h
SevLoca  db 10h,10h,08h,20h,40h,80h,01h,02h,04h ; first digit is neglected
SevTab12 db 0dfh,91h,3fh,0bdh,0f1h,0edh,0efh,99h,0ffh,0fdh
           db 4bh,7bh,00h,00h,00h,00h
```

USULAN TUGAS AKHIR

1. Judul Tugas Akhir : Pemantauan Pemakaian Air pada Meter Air Pelanggan PDAM Yang Dikomunikasikan ke IBM PC Sebagai Sentral Pemantau Melalui Jalur Band VHF

2. Ruang Lingkup :
 - Elektronika Komunikasi
 - Transmisi Data
 - Sistem Komunikasi
 - Mikroelektronika
 - Rangkaian Linier Aktip

3. Latar Belakang : Selama ini untuk mengetahui data penunjukkan pemakaian air minum pelanggan PDAM diperlukan petugas yang mencatat data pemakaian air yang dipakai oleh pelanggan setiap bulannya. Dengan memanfaatkan kemajuan di bidang elektronika dan telekomunikasi tugas pencatatan ini dapat digantikan dengan menggunakan perangkat elektronik. Dalam pencatatan secara elektronik tidak

lagi diperlukan petugas-petugas yang dikirimkan ke rumah-rumah pelanggan. Data pemakaian air cukup dipantau melalui unit radio pada pelanggan dan kantor pusat PDAM. Dengan menggunakan sistem pemantauan data secara elektronik ini maka semua proses pengolahan data serta pencetakan besar rekening yang harus dibayar oleh pelanggan sepenuhnya dapat ditangani oleh komputer, manusia dalam hal ini akan lebih banyak berperan sebagai pengawas.

Komunikasi data digital pada jarak jauh bisa berlangsung bila sinyal digital tersebut dimodulasi dengan sinyal pembawa (carrier signal). Tanpa adanya modulasi, sinyal digital hanya dapat menempuh jarak 1 s/d 1,5 km. Oleh sebab itu di sini dipakai sinyal carrier pada band VHF agar jarak komunikasi dapat diperjauh.

4. Penelaahan Studi : - Mempelajari mikrokontroller 8031 yang dipakai sebagai jantung

pemroses data pada Meter Air PDAM

- Mempelajari prinsip kerja komunikasi data secara serial
- Mempelajari unit radio pemancar dan penerima yang bekerja pada band VHF

5. Tujuan : Merancang dan membuat meter air digital yang dapat berkomunikasi dengan suatu komputer sentral (sentral pemantau).

6. Langkah - langkah :
1. Studi literatur
 2. Perencanaan dan pembuatan alat
 3. Pengujian alat
 4. Penulisan naskah

7. Jadwal Kegiatan :

K E G I A T A N	BULAN KE-					
	1	2	3	4	5	6
1. Studi literatur	██	██	██			
2. Perencanaan dan pembuatan alat		██	██	██		
3. Pengujian alat					██	
4. Penulisan Naskah					██	██

8. Relevansi

: Diharapkan bahwa meter air pelanggan PDAM yang selama ini menggunakan sistem mekanis digantikan dengan meter air digital yang terhubung dengan komputer IBM PC yang difungsikan sebagai komputer pusat pengolah data.

RIWAYAT HIDUP



MOKHAMAD ABDUL ROZAQ, dilahirkan di Madiun pada tanggal 2 April 1969. Penulis adalah putra kedua dari 4 bersaudara dari bapak Syaiin Hasyim dan ibu Siti Alifah.

Terdaftar sebagai mahasiswa Institut Teknologi Sepuluh Nopember Surabaya Fakultas Teknologi Industri Jurusan Teknik Elektro pada tahun 1988 dengan nomer pokok 2882201115. Selama menjadi mahasiswa pada tahap sarjana pernah menjadi asisten praktikum Rangkaian Listrik, praktikum Elektronika, dan praktikum Elektronika Lanjutan II.

Pendidikan yang telah ditempuh sampai saat ini :

- SD Wahid Hasyim Malang, lulus tahun 1982
- SMP Negeri 1 Malang, lulus tahun 1985
- SMA Negeri 3 Malang, lulus tahun 1988
- Jurusan Teknik Elektro, FTI-ITS Surabaya, tahun 1988.

Dan diharapkan pada ujian sarjana FTI - Jurusan Teknik Elektro periode Maret 1994 ini dapat menyelesaikan studinya.