

BAB II

TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Adapun penelitian terdahulu tentang sistem pengontrolan pintu otomatis yang dibuat oleh saudara ONY KISWANTORO yang juga sama-sama menggunakan IC 89C51, sebagai inputan pengontrol pintu otomatis pada gedung-gedung perkantoran, adapun persamaan dan perbedaan dari penelitian terdahulu adalah sebagai berikut :

2.1.1 Persamaan

Adapun persamaan dari penelitian terdahulu adalah :

1. Sama_sama menggunakan mikrokontroler 89C51.
2. Sama_sama menggunakan motor DC sebagai penggerak pintu.
3. Sama-sama menggunakan driver motor DC dan Relay.

2.1.2 Perbedaan

Adapun perbedaan dari penelitian terdahulu :

Bila penelitian terdahulu sensor yang digunakan adalah sensor jarak, untuk buka tutup pintu pada gedung-gedung perkantoran, maka dari itu kita mengacu pada penelitian terdahulu, maka kita coba untuk membuat yang lain yang lebih berguna dimasyarakat, yaitu membuat pintu air, pada pintu air ini kami menggunakan sensor apung yang digunakan sebagai batas ketinggian air pada buka tutup pintu air otomatis yang kami buat.

2.2 Landasan teori

2.2.1 Mikrokontroler AT89C51

Penggunaan mikrokontroler 89C51 memiliki beberapa keuntungan dan keunggulan antara lain : tingkat kehandalan yang cukup tinggi ,implementasi dengan jumlah komponen yang lebih sedikit sehingga memerlukan waktu perencanaan dan pembuatan yang relative singkat,kemudahan dalam hal pemrogramannya , dan penghematan dalam segi biaya, karena *Flash* PEROM yang dimasukan ke dalam *Chip* Mikrokontroler 89C51 ini sejenis dengan *Flash* PEROM yang digunakan dalam penyetingan BIOS sebuah PC saat ini yang harganya sangat murah.

Mikrokontroler 89C51 dikemas dalam bentuk *Single IC* 40 pin dengan memori program internal (*flash memory*) yang mudah untuk dihapus dan diprogram kembali secara berulang-ulang .Dengan beberapa kelebihan tersebut menjadikan mikrokontroler 89C51 banyak digunakan di berbagai industri dalam bidang control .

Spesifikasi teknik dari mikrokontroler 98C52 adalah sebagai berikut:

- 64 Kbytes 8-bit *Central Processing Unit*
- 4096 (4 K) bytes of *In-System Reprogrammable Flash Program Memory*
- 128 jalur masukan dan keluaran yang bersifat dua arah dikelompokkan menjadi 4 byte port
- Mode yang lebih bervariasi (*Multiplex Mode*), dan kemampuan pemrograman dengan kecepatan tinggi untuk setiap port serial
- *Full Duplex Serial Port*
- 2 buah 16 bit pewaktu atau counter

- Kemampuan pengalamat ke memori program dan memori data eksternal masing-masing mencapai 64 K byte memori

Konfigurasi dari AT89C51 dapat dilihat pada gambar 2.1 dibawah ini :

P1.0	1	40	VCC
P1.1	2	39	P0.0
P1.2	3	38	P0.1
P1.3	4	37	P0.2
P1.4	5	36	P0.3
P1.5	6	35	P0.4
P1.6	7	34	P0.5
P1.7	8	33	P0.6
RST	9	32	P0.7
RSTRXD (3.0)	10	31	EA/VPP
TXD (P3.1)	11	30	ALE
INT (P3.2)	12	29	PSEN
INT1 (P3.3)	13	28	P2.7
T0 (P3.4)	14	27	P2.7
T1 (P3.5)	15	26	P2.7
WR (P3.6)	16	25	P2.7
RD (P3.7)	17	24	P2.7
XTAL2	18	23	P2.7
XTAL1	19	22	P2.7
GND	20	21	P2.7
	???		

Gambar 2.1
Mikrokontroller AT89C51

Konfigurasi dari kaki-kaki mikrokontroller 89C51 diperlihatkan pada gambar

Fungsi-fungsi tiap kaki adalah sebagai berikut:

1. VSS

Dihubungkan dengan ground rangkaian

2. VCC

Dihubungkan dengan sumber tegangan

3. Port 0 (P0.0-P0.7)

Port 0 merupakan O/I 8 bit dua arah. Port ini digunakan sebagai multipleks bus alamat rendah dan bus data selama pengaksesan ke memori luar

4. Port 1 (P1.0-P1.7)

Port 1 dapat berfungsi sebagai input maupun output dan dapat bekerja baik untuk operasi bit maupun byte, tergantung dari pengaturan software

5. Port 2 (P.2.0-P2.7)

Port 2 dapat digunakan sebagai alamat bus byte tinggi selama adanya akses ke memori program luar atau memori data luar

6. Port 3 (P.3.0-P3.7)

Port ini selain berfungsi sebagai I/O juga mempunyai fungsi khusus sebagai berikut

- RD (P3.7) : sinyal membaca memori data luar
- RW (P3.6) : sinyal menuliskan memori data luar
- T1 (P3.5) : masukan dari pewaktu/pencacah 1
- T0 (P3.4) : masukan dari pewaktu/pencacah 2
- INT1 (P3.3) : masukan interupsi 1
- INT (P3.2) : masukan interupsi 2
- TXD (P3.1) : keluaran pengiriman data untuk serial port (asinkron) atau sebagai keluaran clock (sinkron)
- RXD (P3.0) : keluaran penerimaan data untuk serial port (asinkron) atau sebagai keluaran clock (sinkron)

7. RSD/VPD

Merupakan pin input yang aktif tinggi, jika pin ini aktif selama dua siklus mesin ketika osilator bekerja akan mereset peralatan.

8. ALE (Address Latch Enable) / prog (Pulse Program)

Pin ALE aktif tinggi mengeluarkan pulse output untuk me-latch satu byte alamat rendah selama mengakses ke memori luar . ALE dapat mengendalikan 8 beban TTL .Pin ini merupakan input program yang aktif rendah selama pemrograman EPROM

Pada operasi normal ,ALE dikeluarkan pada suatu kecepatan yang konstan yaitu 1/6 dari frekwensi osilator , dan dapat digunakan untuk timing eksternal atau untuk tujuan peng-clock-an.

9. PSEN (Program Strobe Enable)

Pin ini aktif rendah yang merupakan Strobe pembaca ke program memory eksternal

10. XTAL 1 DAN XTAL 2

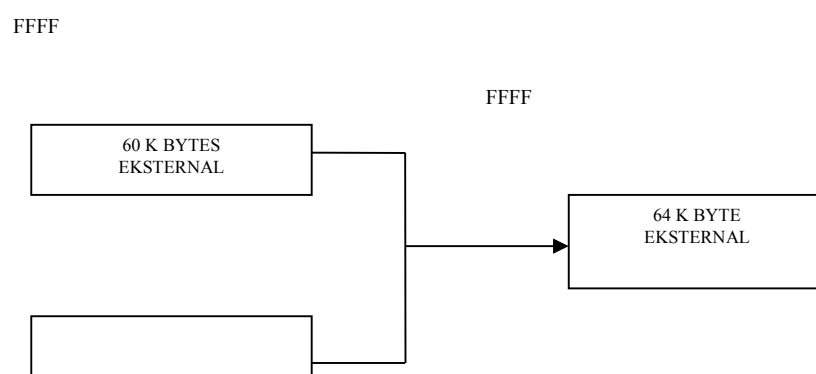
Pin XTAL 1 ini merupakan pin penguatan osilator pembalik dan pin XTAL 2 merupakan pin output dari penguatan osilator pembalik

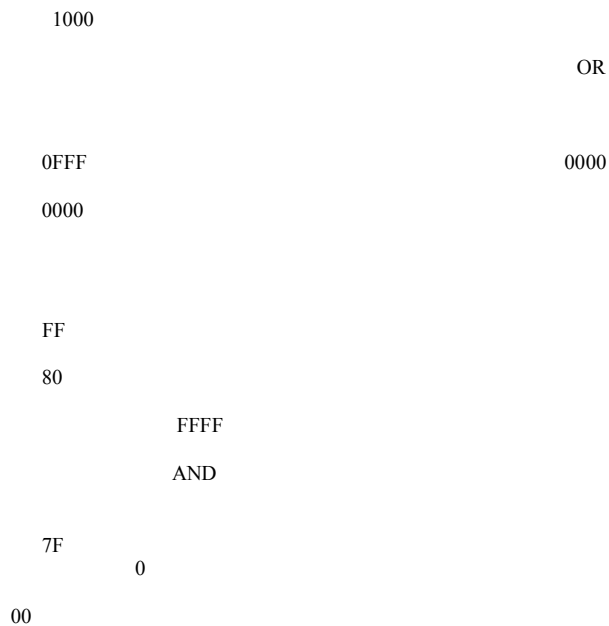
11. EA/VVP (Eksternal Acces/Programming supply Voltage)

Pin Ea harus di hold rendah secara eksternal atau digroundkan agar 8951 dapat mengakses kode mesin dari program memori eksternal dengan lokasi \$0000 sampai \$0FFF.

2.2.1.1 Organisasi Memori Mikrokontroller 89C51

Pada mikrokontroler keluarga MCS 51, memiliki memori program dan memori data yang terpisah sesuai yang ditunjukkan pada gambar 1.1 pemilihan dilakukan secara logika,sehingga CPU dapat mengakses sampai 64 K byte memori program dari 64 K byte memori data, lebar memori data internal adalah 8 bit dan 16 bit (register PC dan register DPTR





2.2.1.2 Memori Program

Memori program menggunakan alamat sepanjang 64 K Byte dengan 4 K Byte (alamat \$0000 sampai \$0FFF) yang merupakan memori internal , sehingga 60 K Byte merupakan memori eksterna ,atau dapat juga menggunakan 64 K Byte memori eksternal sebagaimana ditunjukkan pada gambar 2.2 diatas

Memori program merupakan tempat penyimpanan data yang permanen .Memori program hanya dapat dibaca atau lebih dikenal dengan sebutan *Read Only Memory* (ROM). Data dalam ROM tidak akan terhapus meskipun catu daya dimatikan (bersifat *non volatile*) Oleh karena itu ROM hanya data digunakan untuk menyimpan program

Ada beberapa tipe *Read Only Memory* (ROM) , diantaranya ROM murni yaitu memori yang sudah deprogram oleh pabrik,ROM, EPROM, dan EEPROM. *Programmable Read Only Memory* (PROM) merupakan memori yang dapat diprogram

hanya sekali saja *Eraseable Programmable Read Only Memory (EPROM)* merupakan PROM yang dapat diprogram ulang. *Electrical Eraseable Programmable Read Only Memory (EEPROM)* pada prinsipnya sama dengan EPROM. Perbedaannya terletak pada pengisian dan penghapusan program. Untuk mengisi program baru, pada EPROM harus dihapus dulu dengan ultraviolet, sedangkan pada EEPROM pengisian dapat dilakukan secara langsung, yaitu program yang lama langsung di tinds dengan program yang baru maka secara otomatis program lama akan terhapus, sehingga program EEPROM lebih fleksibel di bandingkan dengan EPROM, selain itu harganya pun lebih murah

2.2.1.3 Memori Data

Memori data adalah tempat untuk menyimpan data yang sifatnya sementara, sehingga pada memori data bersifat *volatile* yaitu data akan hilang jika catu daya mati. Memori data lebih dikenal dengan sebutan nama *Random Access memory (RAM)* yaitu dapat melakukan membaca dan menulis data pada alamat yang tersedia. memori data pada 51 dapat memiliki pengalamat sampai 64 k byte yang diperoleh dari memory data internal sebagai mana yang ditunjukkan pada gambar no2.3

Selain pada eksternal, MCS51 Juga mempunyai 128 byte internal ditambah dengan sejumlah register fungsi khusus atau *Spesial Funtion Register (SFRs)*

Pada keluarga mikrokontroller8951, ruang data memory internal terbagi menjadi 3 blok yang disebut sebagai *lower 128*, *upper 128* dan ruang SRFs sebagaimana tertera pada gambit dibawah ini

Pada lower 128 lokasi memori terbagi menjadi 3 bagian yaitu

1. Register Bank 0-3 lokasi Register bank di mulai dari alamat 00h – 1f h yang terdiri dari 32 byte. Register bank ini terdiri dari 4 buah register 8 bit yang dapat dipilih melalui pengaturan program *Status Word Register*.

2. Bit Addressable Area

Terdiri dari 16 byte yang mulai dari 20h -21h. Masing-masing dari 128 bit lokasi ini dapat dialamati secara langsung ,dari \$00h sampai \$7fh

3. scratch Pat Area

lokasinya dari alamat 30h sampai 7Fh atau sebanyak 80 byte yang dapat digunakan sebagai data bagi RAM

Pada 128 byte atas (*upper*) ditempati suatu fungsi register yang mempunyai fungsi khusus yang disebut *Special Function Register* (SFRs),ruang dari register fungsi khusus ini adalah 80H sampai FFh. Berikut ini adalah contoh dari *vector* alamat pada *Special Function Register*

1. Accumulator (Acc) atau register A dan B

Kedua register ini digunakan untuk operasi perkalian dan pembagian

2. Program Status Word

Register ini meliputi bit-bit:CY (carry),AC (auxiliary carry) ,FO (sebagai Flag),RSO dan RSI (untuk pemilihan register bank), OV (Over Flow),Parity (Parity Flag)

3. Stack Pointer

SP merupakan register yang digunakan untuk petunjuk alamat. Register ini berguna apabila suatu routine pada program utama.

4. Data Pointer High (DPH) dan Data Pointer Low (DPL)

DPTR adalah suatu register yang digunakan untuk pengalamatan tidak langsung. Register ini selain digunakan untuk mengakses memori program baik eksternal maupun internal, juga digunakan untuk alamat external data. DPTR ini dikontrol oleh dua buah register 8 bit yaitu DPL dan DPH.

5. Port 0, Port 1, Port 2, Port 3

Pada keluarga 8051 masing-masing port dapat dialamatkan baik secara byte maupun bit. Masing-masing port merupakan port bi-directional yang artinya dapat berjalan dua arah (input dan output). Port 0 dan port 2 digunakan untuk pengalamatan dari luar, port 1 digunakan untuk I/O, sedang port 3 berisi sinyal-sinyal control seperti interrupt serial, WR dan RD.

6. Register Prioritas Interupt

Merupakan register yang berisi bit-bit untuk mengaktifkan prioritas dari suatu interupsi yang ada pada mikrokontroler pada tahap yang diinginkan.

7. Interrupt Enable Register

EI merupakan register yang berisi bit-bit untuk menghidupkan atau mematikan sumber interrupt

8. Timer./ Counter Control Register

TCON merupakan suatu register yang berisi bit-bit untuk memulai / menghentikasn pencacah/ pewaktu

9. Serial Control Buffer(SBUF)

Register ini digunakan untuk menampung data dari masukan (SBUF IN) atau keluaran (SBUF OUT) dari serial port.

2.2.1.4 Sistem Interrupt

Mikrokontroler 89C51 mempunyai 5 buah sumber interrupt yang dapat membangkitkan interrupt request yaitu:

INT0 :permintaan interrupt luar dari kaki P3,2

INT1 :permintaan interrupt luar dari kaki p3,3

Timer/Counter 0 :bila terjadi over low

Timer/Counter 1 :bila terjadi over low

Port serial : bila pengirirman / penerimaan satu frame telah lengkap

Saat terjadi *interrupt mikrokontroler* secara otomatis akan menuju ke *subroutine* pada alamat tersebut ,setelah *interrupt service* selesai dikerjakan, *mikrokontroler* akan mengerjakan program semula

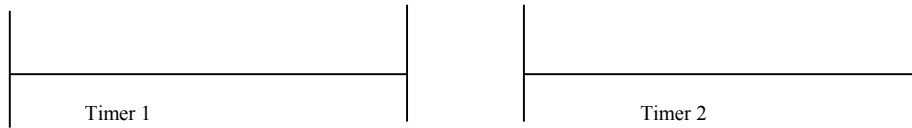
Dua sumber *internal* adalah INT0 dan INT1, kedua *interrupt* ini akan *aktif level* atau aktif transisi tergantung ini IT0 dan IT1 pada *register* TCON, *Interrupt Timer 0* dan *timer 1* aktif pada saat *timer* yang sesuai mengalami *sollover* ,*Interrupsi* serial dibangkitkan dengan menggunakan operasi OR pada R1 dan T1. Tiap tiap sumber *interrupsi* dapat *enable* atau *disable* secara *software* .

Tingkat prioritas semua sumber *interrupsi* dapat diprogram sendiri-sendiri dengan set atau *clear bit* pada SFRS IP (*Interrupt priority*) *interrupsi* tingkat rendah dapat di *interrupsi* oleh *interrupt* yang memiliki tingkatan lebih tinggi, tetapi tidak sebaliknya ,Walaupun demikkian *interrupsi* yang mempunyai tingkat lebih tinggi tidak bias menginterrupsi semua intrupsi lain.

Sumber Interupsi	Alamat Awal
Interupt luar	003H

TMOD: Timer Counter Mode Control Register

GATE	C/T	M1	M0	GATE	C/T	M1	M0
------	-----	----	----	------	-----	----	----



Gate : Bila gate = 1, Timer Counter x enable hanya saat pin INTx tinggi dan TRx1, saat gate = 0, Timer/Counter enable jika TRx 1

CT : Jika C/T = 0, maka Timer /Counter x akan berfungsi sebagai timer jika CT=1, maka Timer/Counter x berfungsi sebagai Counter

M1	M0	OPERATING MODE
0	0	Timer 13 bit
0	1	Timer/ Counter 16 bit
1	0	8 bit Auto Reload Timer/Counter
1	1	TL0 dari Timer adalah 8 bit Timer/Counter dikendalikan oleh control bit timer 0. TH0 adalah Timer 8 bit yang dikendalikan oleh Timer 1 kontrol bit

M1 dan M0 : Penentu Mode

Pengaturan time dan counter dilakukan pertama kali dengan mengatur TMOD. Disini kita akan menentukan fungsi sebagai *Timer* atau *counter* lengkap dengan spesifikasinya, demikian juga dengan pengaktifan yang berhubungan dengan penggunaan mode ini.

2.2.2 Bahasa Pemrograman Mikrokontroller

Sebuah *mikrokontroler* tidak akan bekerja bila tidak diberi program kepalanya. Program tersebut memberitahu *mikrokontroller* apa yang akan dia lakukan. Sebuah *mikrokontroler* yang telah bekerja baik dengan suatu program, tidak akan bekerja lagi bila programnya diganti

Instruksi perangkat lunak berbeda untuk masing-masing jenis *mikrokontroler*. Instruksi-instruksi ini hanya bias dipahami oleh jenis *mikrokontroler* yang bersangkutan, sebuah *mikrokontroler* tidak dapat memahami instruksi-instruksi yang berlaku pada *mikrokontroler* lain. *seagaimanconton* *mikrokontroler* buatan *intel* dengan *mikrokontroler* buatan *Motorola* memiliki perangkat industri yang berbeda. Instruksi yang berbeda. Instruksi-instruksi ini disebut sebagai bahasa pemrograman *mikrokontroler*

2.2.2.1 Pengalamatan Tak Langsung

Operasi pengalamat yang tidak langsung menunjukkan ke sebuah register yang berisi lokasi alamat memori yang akan digunakan dalam operasi. Lokasi yang nyata tergantung pada isi register saat instruksi dijalankan. Untuk melakukan pengalamatan tak langsung digunakan simbol @, misalnya:

ADD A,@R0 ;Tambahkan isi RAM pada register 0 ke akumulator

DEC @R1 :Kurang 1 isi RAM pada alamat R1

2.2.2.2 Pengalamatan langsung

Pengalamatan dilakukan dengan memberi suatu nilai ke register secara langsung .Untuk melaksanakan hal tersebut digunakan tanda #,missal;

MOV A,# 01h ;Isi akumulator dengan bilangan 01H

MOV DPTR, # 19ABH ;Isi DPTR dengan bilangan 19ABH

Pengalamatan data langsung dari 0 sampai 127 akan mengakses RAM internal.sedangkan pengalamatan dari 128 sampai255 akan mengakses register perangkat keras .

MOV P3,A ;Pindahkan isi akumulator ke alamat data BOH (alamat port 3)

DEC @RI ;Naikkan lokasi 50 (decimal) ke dalam memori

2.2.2.3 Pengalamatan Bit

Pengalamatn bit adalah penunjukan alamat lokasi bit baik dalam *RAM internal* (*byte* 32 sampai 47) atau bit perangkat keras.Untuk melakukan pengalamat bit digunakan simbul titik (.),misalnya *FLAGS.3.40.5.21H.5*. dan *ACC.7*.

2.2.2.6 Perangkat Instruksi

Mikrokontroler Intel 89C51 memiliki 256 perangkat instruksi .seluruh instruksi dapat dikelompokkan menjadi 4 bagian yang meliputi instruksi 1 *byte* sampai 4 *byte*

Apabila *frekuwensi mikrokontroler* yang digunakan adalah 12 *MHZ*.kecepatan pelaksanaan instruksi akan bervariasi dari 1 hingga 4 *mikro* detik .Perangkat instruksi *mikrokontroler intel 89C51* dapat dibagi menjadi 5 kelompok sebagai berikut :

❖ Instruksi *Transfer Data*

Instruksi ini memindahkan data antara *register-register*, *memori-memori*, *register-memori*, *antar muka-register*, dan *antar muka-memori*

❖ Instruksi *Aritmatika*

Instruksi ini melakukan operasi *aritmatika* yang meliputi penjumlahan, pengurangan, penambahan satu (*increment*), pengurangan satu (*decrement*), perkalian dan pembagian

❖ Instruksi Logika dan Manipulasi Bit

Melaksanakan operasi logika *AND*, *OR*, *XOR*, pertandingan, penggeseran dan komplemen data

❖ Instruksi Percabangan

Instruksi ini mengubah urutan normal pelaksanaan suatu program. Dengan instruksi ini program yang sedang dilaksanakan akan mencabang ke suatu alamat tertentu, instruksi percabangan dibedakan atas percabangan bersyarat dan percabangan tanpa syarat

❖ Instruksi *Stack*, *I/O*, dan Kontrol

Instruksi ini mengatur penggunaan *stack*, membaca / menulis *port I/O*, serta pengontrolan-pengontrolan

❖ Perangkat Instruksi 89C51

Perangkat instruksi mikrokontroler intel 89C51 secara lengkap adalah sebagai berikut :

ACALL

Absolute Call

Mnemonic : ACALL
Operand : Alamat kode
Format : ACALL alamat kode

Keterangan instruksi ini mempunyai isi yang telah dinaikan dari pencacah program pada stack. Byte bawah dari pencacah program selalu ditempatkan pada stack pertama

ADD

Add Immediate Data

Mnemonic : ADD
Operand : A Akumulator
: data $-256 \leq \text{data} \leq +256$
Format : ADD A, # data

Keterangan instruksi ini menambah 8 bit data langsung ke dalam isi akumulator dan menyimpan hasilnya pada akumulato

ADD

Add Indirect Data

Mnemonic : ADD
Operand : A Akumulator
: Rr Register $0 \leq r \leq 1$
Format : ADD A, @Rr

Keterangan instruksi ini menambahkan isi data memori yang lokasinya di tunjukan oleh nilai register r ke isi akumulator dan menyimpan hasilnya dalam akumulator

Contoh : ADD A, @R1

ADD

Add Register

Mnemonic : ADD

Operand : A Akumulator

: Rr Register 0 sampai 7

Format : ADD A, Rr

Keterangan instruksi ini menambahkan isi register r ke isi akumulator dan menyimpan hasilnya dalam akumulator

Contoh : ADD A, R6

ADD

Add memori

Mnemonic : ADD

Operand : A Akumulator

: Alamat data 0 <= alamat data,=256

Format : ADD A, alamat data

Keterangan instruksi ini menambahkan isi alamat data ke isi akumulator dan menyimpan hasilnya dalam akumulator

Contoh : ADD A, 32H

ADDC

Add Carry Plus Immediate Data to Accumulator

Mnemonic : ADDC

Operand : A Akumulator
: data -256<= data <= +256

Format : ADD A, # data

Keterangan instruksi ini menambahkan isi carry flag (0 atau 1) kedalam isi akumulator. Dan langsung 8 bit ditambahkan ke akumulator

Contoh :ADD A, #0AFH

ADDC

Add Carry Plus Indirect Address to Accumulator

Mnemonic : ADDC

Operand : A Akumulator
: Rr Register 0 <=r<=1

Format : ADD A, @Rr

Keterangan instruksi ini menambahkan isi carry flag (0 atau 1) dengan isi akumulator. Isi data memori pada lokasi yang di tunjukkan oleh register Rr ditambahkan dan hasilnya disimpan di akumulator

Contoh :ADD A, @R1

ADDC

Add Carry Plus Register to Accumulator

Mnemonic : ADDC

Operand : A Akumulator
: Rr Register 0<= r <=7

Format : ADDC A,Rr

Keterangan instruksi ini menambahkan isi carry flag dengan isi akumulator, isi register r ditambahkan dan hasilnya disimpan di akumulator

Contoh :ADD C A, R7

ADDC

Add Carry Plus Indirect Address to Accumulator

Mnemonic : ADDC

Operand : A Akumulator

: Alamat data 0 <= alamat data,=255

Format : ADD A, Alamat data

Keterangan instruksi ini menambahkan isi *carry flag* dengan isi akumulator, dari alamat data tertentu ditambahkan pula, dan hasilnya disimpan di akumulator

Contoh :ADD C A, 25H

AJMP

Absolute Jump Within 2K Byte Page

Mnemonic : AJMP

Operand : Alamat kode

Format : AJMP Alamat kode

Keterangan instruksi ini meletakkan bagian bawah 11 bit dari pencacah program dengan 11 bit alamat yang dikodekan

ANL

Logical AND Indirect Address to Accumulator

Mnemonic : ANL

Operand : A Akumulator
: R Register 0 $\leq r \leq 1$
Format : ANL A, @Rr

Keterangan instruksi ini meng-AND-kan isi memori yang lokasinya ditunjukkan oleh isi register r dengan isi akumulator

Contoh : ANL A, @Rr

ANL

Logical AND Immediate Data to Accumulator

Mnemonic : ANL
Operand : A Akumulator
: data $-256 \leq \text{data} \leq +256$
Format : ANL A, #data

Keterangan instruksi ini meng-AND-kan data 8 bit secara langsung dengan isi akumulator

Contoh : ANL A, #00001000B

ANL

Logical AND Register to Accumulator

Mnemonic : ANL
Operand : A Akumulator
: R Register 0 $\leq Rr \leq 7$
Format : ANL A, Rr

Keterangan instruksi ini meng-AND-kan isi register r dengan isi akumulator

ANL

Logical AND Memory to Accumulator

Mnemonic : ANL

Operand : A Akumulator

: Alamat data 0 <= alamat data <=255

Format : ANL A, Alamat data

Keterangan instruksi ini meng-AND-kan isi alamat tertentu dengan isi akumulator

Contoh : ANL A, 37H

ANL

Logical AND Bit to Carry Flag

Mnemonic : ANL

Operand : C Carry Flag

: Alamat bit 0 <= alamat bit <=255

Format : ANL A, Alamat bit

Keterangan instruksi ini meng-AND-kan isi alamat tertentu dengan isi *carry flag*. Jika keduanya 1 maka hasilnya 1, selain itu hasilnya 0. Hasilnya ditempatkan pada *carry flag*.

Contoh : ANL A, @R0

2.2.2 Sensor level

Sensor level air adalah sensor yang mengalami bentuk dari tekanan air mendekati titik tertentu dan mengakibatkan perubahan tenaga listrik menjadi mekanik. Sensor level dibagi menjadi dua bagian misalnya : sensor sebagai masukan dan sebagai pembuka dan juga disebut sebagai *switch*. Cara kerja dari sensor tersebut adalah pertama sensor yang dialiri oleh listrik agar dapat terhubung dengan tekanan air maka akan menjalankan motor DC selanjutnya mengerakan *relay* maka terbukalah daun pintu

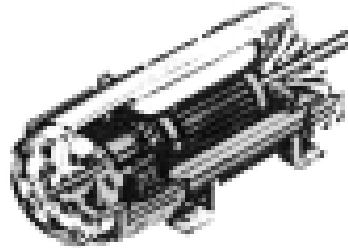
Pada tugas akhir ini digunakan sensor level air sebagai sensor untuk mengontrol ada tidaknya keluaran atau output (sebagai masukan dari tekana yang ada) bila tekanan air masuk / naik maka akan terbuka dan bila tekanan air menurun maka akan tertutup kembali

2.2.3 Motor DC

Untuk dapat membuka dan menutup pintu air dlm tugas akhir ini menggunakan motor DC. Motor DC adalah mesin yang berfungsi utk mengubah tenaga listrik searah menjadi tenaga gerak atau mekanik berupa putaran motor. Motor DC terdiri dari bagian yang diam disebut stator dan bagian yang berputar disebut rotor. Mesin DC banyak digunakan dalam industri modern.

Alasan mengapa industri modern lebih tertarik pada penggunaan motor DC adalah :

Karena kecepatan kerja motor-motor DC mudah diatur selain itu arah putaran dari motor DC ini juga dapat diatur atau disesuaikan dengan kebutuhan.



Gambar 2.2.3
Motor DC