

Mikrokontroler Atmel 89C51

Pendahuluan

Features

- Compatible with MCS-51™ Products
- 4 Kbytes of In-System Reprogrammable Flash Memory
- Endurance: 1,000 Write/Erase Cycles
- Fully Static Operation: 0 Hz to 24 MHz
- Three-Level Program Memory Lock
- 128 x 8-Bit Internal RAM
- 32 Programmable I/O Lines
- Two 16-Bit Timer/Counters
- Six Interrupt Sources
- Programmable Serial Channel
- Low Power Idle and Power Down Modes

Perbedaan antara mikroprosesor dan mikrokontroler adalah sebagai berikut :

1. Mikroprosesor : adalah bagian dari CPU dari sebuah computer, tanpa memori tanpa I/O dan Peripheral. Contoh 8088 dan 80x86. Untuk dapat bekerja mikroprosesor membutuhkan perangkat pendukung berupa RAM, ROM, dan I/O
2. Mikrokontroler atau yang kita kenal dengan *Single Chip* mengkombinasikan CPU dengan memori dan I/O. Dengan demikian suatu mikrokontroler tidak membutuhkan tambahan RAM, ROM dan I/O

Perlengkapan Dasar Mikroprosesor/Mikrokontroler

1. CPU (Central Processing Unit)
2. Alamat/Address
3. Data
4. Pengendali
5. Memori
6. RAM
7. ROM
8. Input/Output

Mikrokontroler 8951

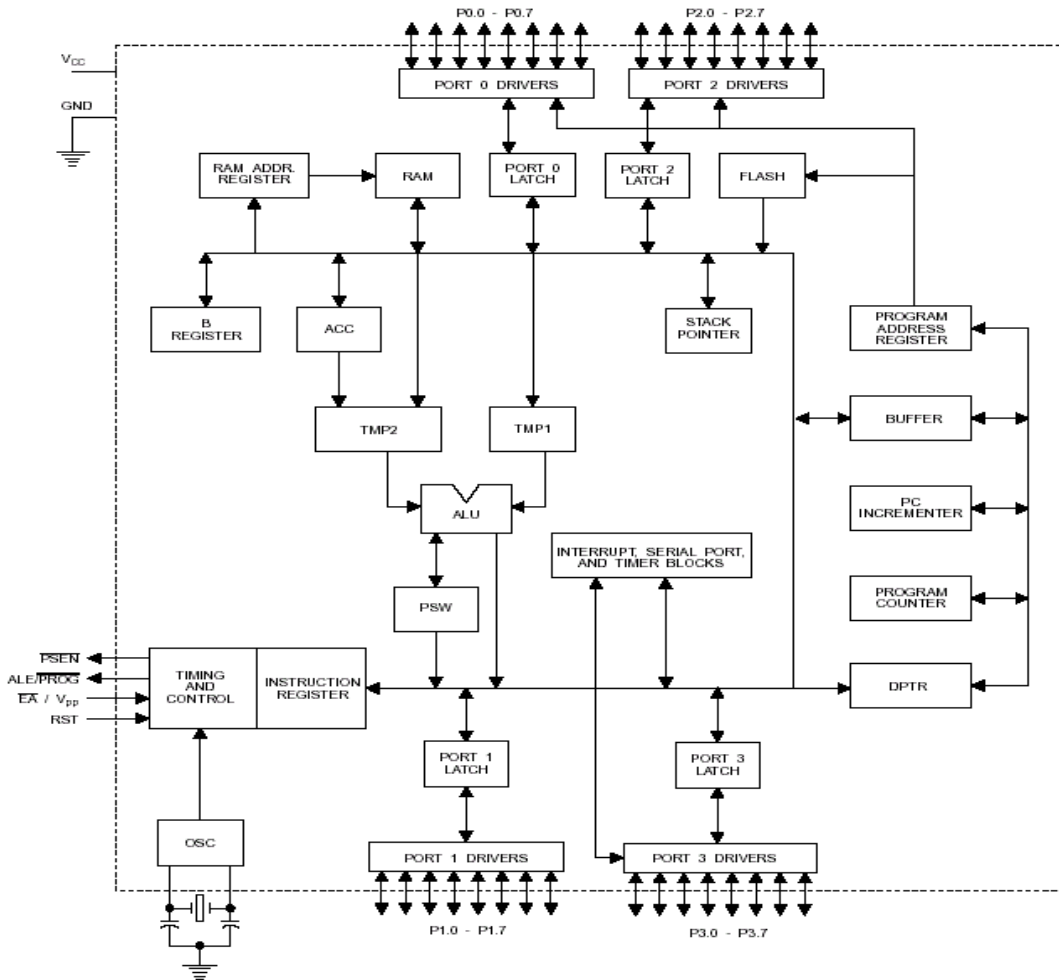
Mikrokontroler Memiliki ROM yang dapat ditulis dan dihapus dengan menggunakan listrik dan dikenal dengan EEPROM (Electrical Erasable Programming ROM)

Mikrokontroler 89c51 memiliki keistimewaan sebagai berikut :

- a) Sebuah CPU 8 bit
- b) Osilator internal dan pewaktu
- c) RAM internal 128 byte
- d) Empat buah programmable port I/O, masing masing terdiri atas 8 buah jalur I/O
- e) Dua Buah Timer/Counter 16 bit
- f) Lima buah jalur interupsi (2 buah interupsi eksternal dan 3 buah interupsi internal)
- g) Sebuah Port Serial dengan control serial Full Duplex UART
- h) Kemampuan melaksanakan operasi perkalian, pembagian dan Operasi Boolean

i) Kecepatan pelaksanaan intruksi dari 4 MHZ sampai 24 MHZ

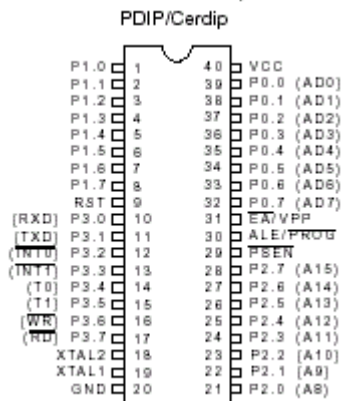
Block Diagram



Pena-Pena Mikrokontroler 89c51

Pena pena 89c51 diperlihatkan pada gambar 4. Penjelasan masing-masing pena adalah sebagai berikut :

- a) Pena 1 sampai 8 (Port 1) merupakan port paralel 8 bit dua arah (bidirectional) yang dapat digunakan untuk berbagai keperluan (general Purpose)
- b) Pena 9 (reset) merupakan rest aktif tinggi). Pulsa transisi dari rendah ke tinggi akan mereset 8951. Pena ini dihubungkan ke power on reset
- c) Pena 10 sampai 17 (port 3) adalah port paralel 8 bit dua arah yang memiliki fungsi pengganti. Fungsi pengganti meliputi TxD, RxD, Int0 (Interrupt 0), Int1 (Interrupt 1). T0, T1, WR(Write) dan RD(Read) Bila fungsi pengganti tidak dipakai, pena-pena ini dapat digunakan sebagai port paralel 8 bit serbaguna.
- d) Pena 18 (XTAL 1) adalah pena masukan ke rangkaian osilator internal. Sebuah osilator kristal atau sumber osilator luar dapat digunakan.



- e) Pena 19 (Xtal 2) adalah pena keluaran ke rangkaian osilator internal. Pena ini dipakai bila menggunakan osilator kristal
- f) Pena 20 (Ground) dihubungkan ke Vss atau Ground
- g) Pena 21 sampai 28 (Port 2) adalah port paralel 2 (P2) selebar 8 bit dua arah. Port 2 ini digunakan sebagai pengalamat bila dilakukan pengaksesan memori eksternal
- h) Pena 29 adalah pena PSEN (Program Store Enable) yang merupakan sinyal pengontrol yang membolehkan program memory eksternal masuk ke dalam bus selama proses pemberian/pengambilan instruksi (fetching)
- i) Pena 30 adalah pena ALE (Address Latch Enable) yang digunakan untuk menahan alamat memori eksternal selama proses pelaksanaan instruksi
- j) Pena 31 (EA). Bila pena ini diberi logika tinggi (H) mikrokontroler akan melaksanakan instruksi dari ROM/EPROM. Ketika isi program counter kurang dari 4096. Bila diberi logika rendah (L), mikrokontroler akan melaksanakan seluruh instruksi dari program memori program luar.
- k) Pena 32 sampai 39 (Port 0) merupakan port paralel 8 bit open drain dua arah. Bila digunakan untuk mengakses memori luar, port ini akan memultipleks alamat memori dengan data.
- l) Pena 40 (Vcc) dihubungkan dengan ke Vcc + 5 volt

Perancangan Rangkaian Minimum ATMEL 89C51

Yang dimaksud rangkaian Minimum adalah rangkaian yang secara minimal harus ada agar mikroprosesor dapat bekerja.

Mikroprosesor dapat bekerja minimal komponen yang harus ada yaitu :

- CPU (Central Prosesor Unit)
- Mem ori Program umumnya menggunakan EEPROM (Read Only Memory)
- Memory Data menggunakan RAM (Random Access Memory)
- Port I/O
- Pewaktuan CPU(Crystall (4 – 24 Mhz))
- Reset (optional) Timer/Counter 0 Hugh Byte
- Power Supply 5 Volt
- EA VPP dihubungkan ke VCC

} *Intern*

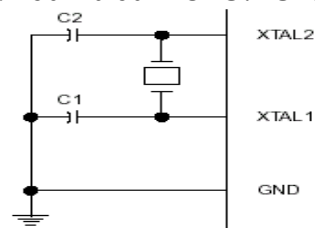
} *Extern*

Untuk Komponen Intern telah tersedia di dalam mikrokontroler, sedang untuk eksternal menggunakan rangkaian berikut :

Pewaktuan CPU (Crystal)

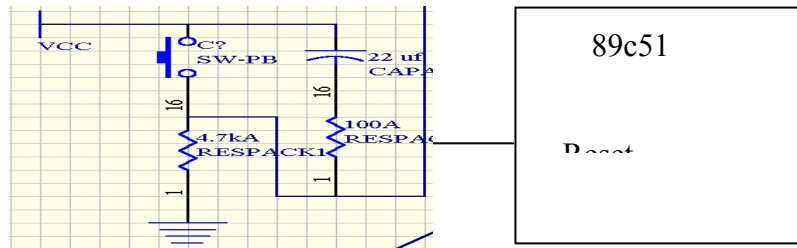
Mikrokontroler 8951 memiliki osilator internal bagi sumber clock CPU. Untuk menggunakan osilator internal diperlukan kristal antara XTAL1 dan XTAL 2 an sebuah kapasitor ground seperti terlihat pada gambar berikut

Untuk kristalnya dapat digunakan frekuensi dari **4 sampai 24 MHZ**. Sedang untuk kapasitor dapat bernilai **20 pF sampai 40 pF**. Bila menggunakan clock eksternal rangkaian dihubungkan seperti berikut :



Reset

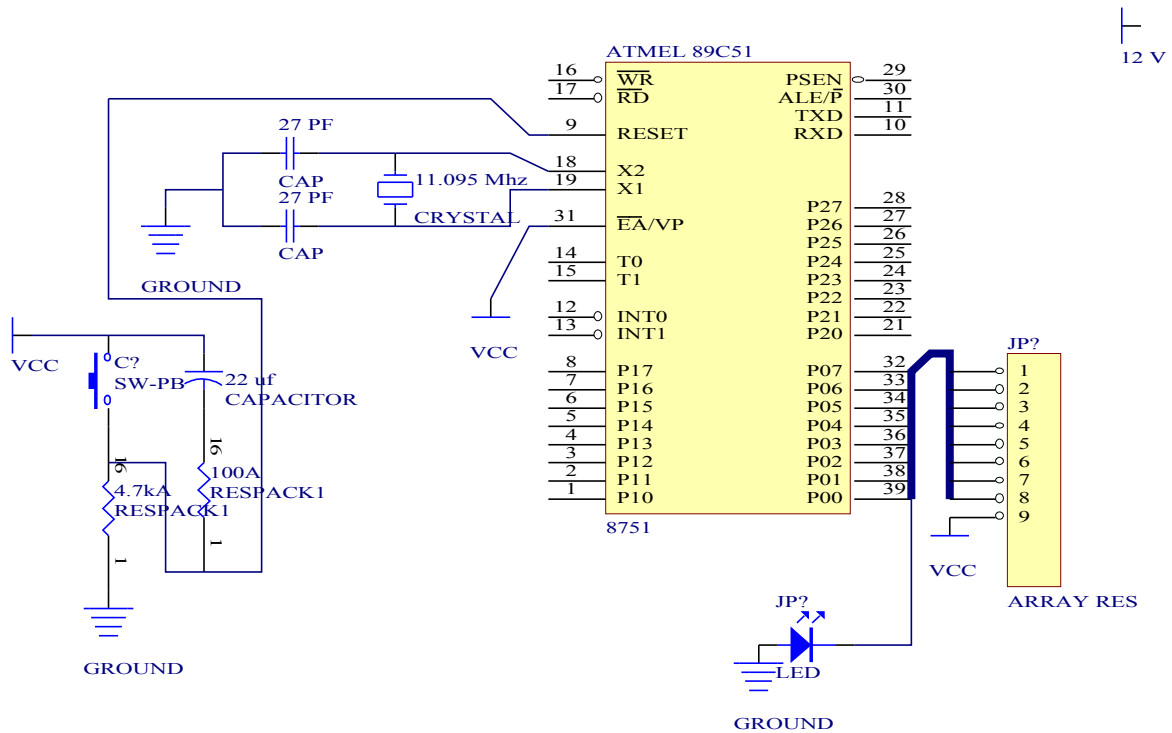
Rangkaian Reset mikrokontroler tampak pada gambar berikut
 Sehingga rangkaian minimum sebagai berikut :



Power Supply

Rangkaian Power Supply dapat menggunakan rangkaian berikut :

Sehingga rangkaian Minimum dapat dirancang sebagai berikut :



Register – register di mikrokontroler 89C51

Register di Mikrokontroler dapat diklasifikasikan berdasar penggunaan sebagai berikut

Klasifikasi Register	Nama Register	Nama	Alamat	fungsi
Register Untuk Aritmatika dan Logik (8 Byte)	A	Accumulator	E0H	
	B		F0H	
Register I/O (8 Byte)	P0	Port 0	80H	
	P1	Port 1	90H	
	P2	Port 2	A0H	
	P3	Port 3	B0H	
	SBUF	Serial Buffer	98H	
Register Control (8 Byte)	IP	Interrupt Priority Control	B8H	
	IE	Interrupt Enable Control	A8H	
	TMOD	Timer/Counter Mode Control	89H	
	TCON	Timer/Counter Control	88H	
	TH0	Timer/Counter 0 High Byte	8CH	
	TL0	Timer/Counter 0 Low Byte	8AH	
	TH1	Timer/Counter 1 High Byte	8DH	
	TL1	Timer/Counter 0 Low Byte	8BH	
	PCON	Power Control	87H	
Register Bantu (8 Byte)	R0.. R7		00H..07H	
	R0'..R7'		08H..0FH	
	R0''..R7''		10H..17H	
	R0'''..R7'''		18H..1FH	
Register 16 Bit	DPTR		82H..83H	
Register Pointer	SP		81H	

Membuat Program dengan bahasa Assembly MCS-51

Perhatikan Contoh soal berikut :

Rancang lampu Hias sederhana 8 bit dengan menggunakan LED dengan Nyala berkedip kedip.

Contoh 0000 0000 berikut 1111 1111 berikut 0000 0000

Rancang Hardware dan Program

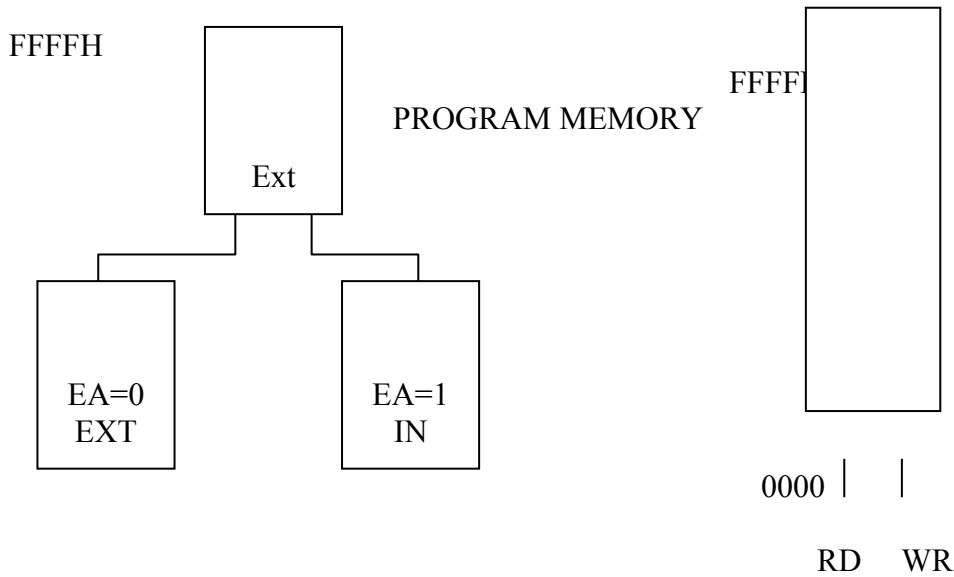
Jawab

Definisikan Hidup ---> 0 ; Mati ---> 1

Organisasi memory

Mikrokontroler MCS 51 memiliki pembagian ruang alamat untuk program dan data. Memori program hanya dapat dibaca tidak dapat ditulisi. Sedang memory data dapat ditulisi. Program yang berukuran lebih dari kapasitas EEPROM (4 Kbite untuk 8951, dan 8 Kbit untuk 8952) disimpan di EEPROM eksternal.

Sinyal yang membolehkan pembacaan dari memoy program eksternal adalah dari pena PSEN (Program Store Enable)



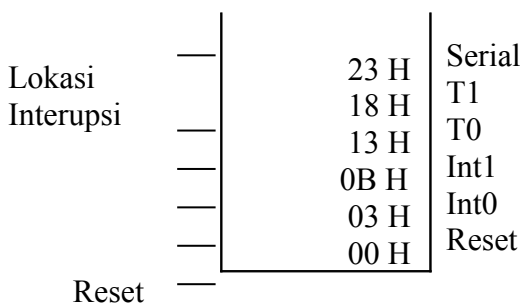
Struktur memori Mikrokontroler 8951

Mikrokontroler Intel 8951 memiliki 5 buah ruang alamat,yaitu

1. Ruang alamat kode (code addresss space) sebanyak 4k, yang seluruhnya merupakan ruang alamat kode eksternal (off chip)
2. Ruang alamat data internal yang dapat dialamat secara langsung, yang terdiri atas:
 - ♣ RAM sebanyak 128 byte
 - ♣ Hardware register sebanyak 128 byte
3. Ruang alamat data internal yang dialamat secara tidak langung sebanyak 128 byte, seluruhnay diakses dengan pengalamatan tidak langsung.
4. Ruang alamat data eksternal 64K byte yang dapat ditambahkan oleh pemakai.
5. Ruang alamat bit. Dapat diakses dengan pengalamatan langsung.

Memori Program

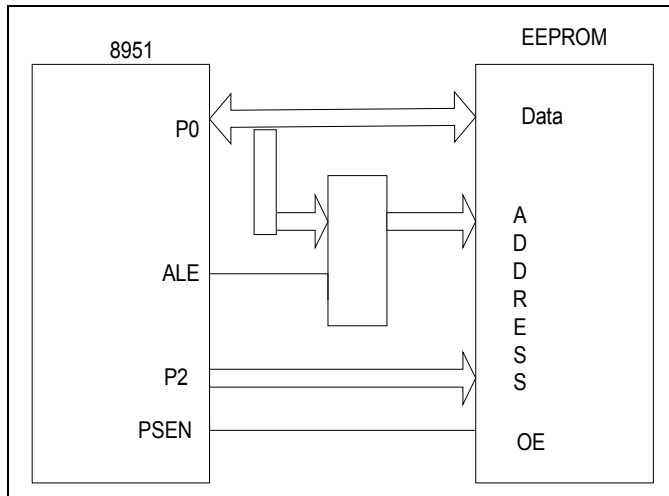
Gambar dibawah memperlihatkan bagian bawah dari memori program. Setelah reset CPU memulai eksekusi dari lokasi 0000H



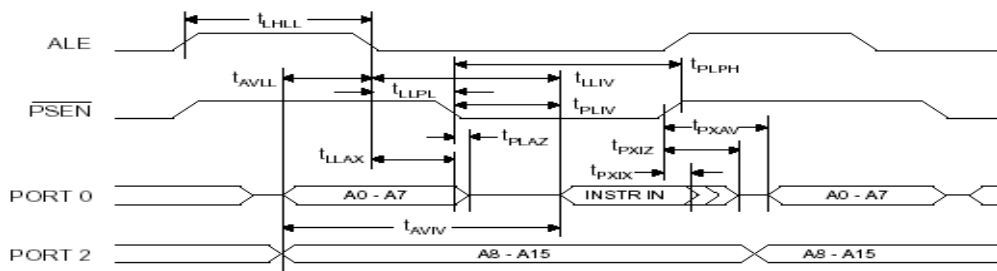
Memori Program

Setiap interupsi mempunyai lokasi tetap dalam memori program. Interupsi menyebabkan CPU melompat lokasi yang ada diatas yang merupakan alamat-alamat lokasi interupsi. Terdapat sub rutin yang harus dilaksanakan.

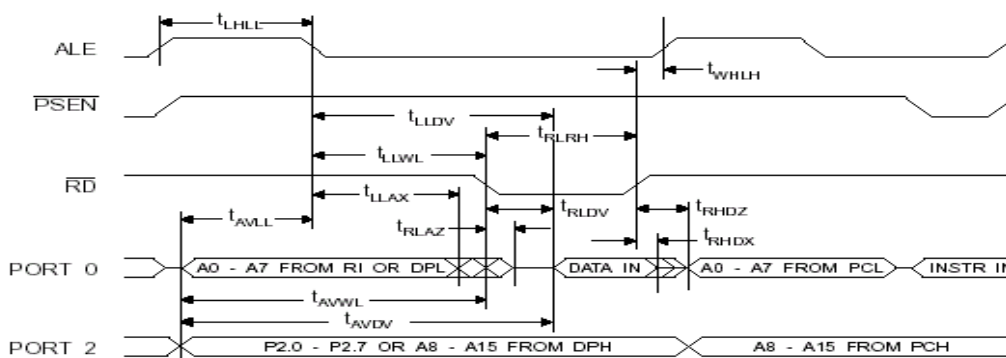
kita dapat menambahkan EEPROM eksternal bila dirasa ukuran program lebih dari 4 Kbyte. Tampak pada blok diagram dibawah ini :



External Program Memory Read Cycle



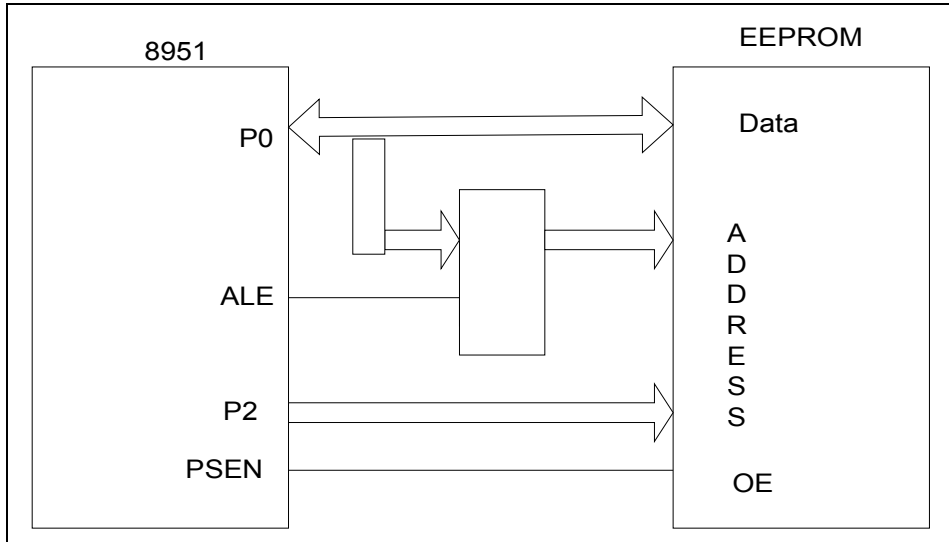
External Data Memory Read Cycle



Memori Data

Gambar dibawah memperlihatkan hubungan mikrokontroler untuk mengakses RAM Eksternal. Untuk melakukan pembacaan dan penulisan, mikrokontroler akan mengirimkan RD atau WR.

men
an

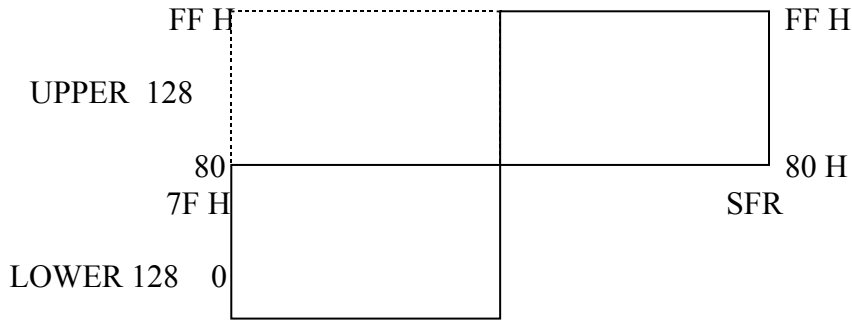


ghubungk

mikrokontroler dengan RAM Eksternal

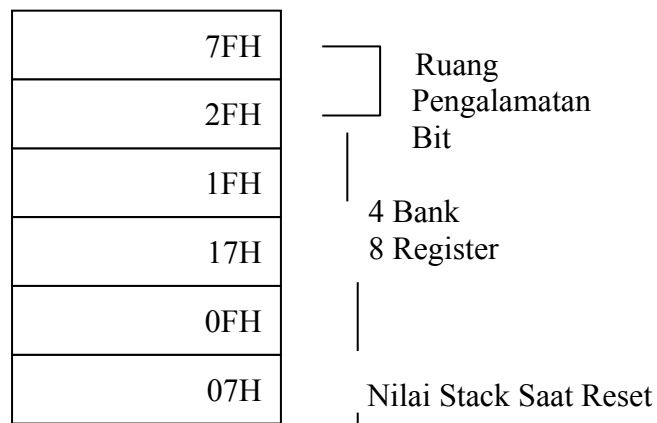
Memori data internal dipetakan seperti pada gambar 10. Ruang memorinya dibagi menjadi tiga blok, yaitu sebagai Lower 128, upper 128 dan ruang SFR (Special

Function Register).

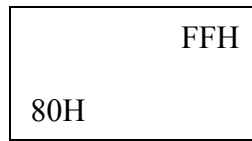


Memori data Intenal

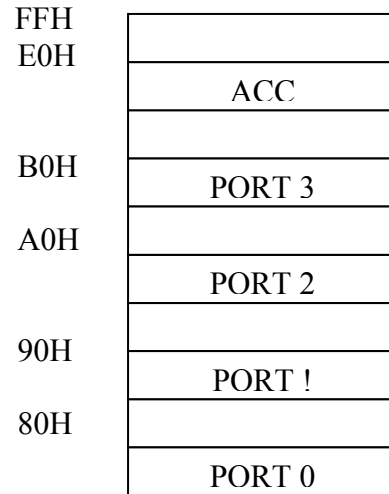
Bagian Bawah dari 128 Byte RAM dipetakan seperti terlihat pada gambar 11. Tiga puluh byte paling bawah dikelompokkan dalam 4 bank (8 Register) yaitu R0 sampai R7. Dua bit dalam psw memilih register bank yang digunakan.



Bagian Bawah 128 Byte RAM Internal



bagian atas 128 byte RAM Internal



Ruang special function register

Interupsi

Ada 2 jenis interupsi dalam mikrokontroler 8951 yaitu :

1. Interupsi yang tak dapat dihalangi oleh perangkat lunak (Non maskable Interrupt) misalnya reset
2. Interupsi yang dapat dihalangi perangkat lunak (maskable interrupt) contoh interupsi jenis ini adalah INT 0 dan INT 1 serta timer/Counter 0, Timer/Counter 1 dan interupsi serial (internal)

Alamat layanan rutin interupsi dari setiap sumber interupsi diperlihatkan pada table dibawah ini

Nama	Lokasi	Alat Interupsi
Reset	00 H	Power on reset
Int 0	03H	INT 0
Timer 0	0BH	Timer 0
Int 1	13 H	INT 1
Timer 1	1 BH	Timer 1

Sint	23 H	Port I/O serial
------	------	-----------------

Mikrokontroler atmel 89c51 menyediakan 5 sumber interupsi: 2 interupsi eksternal, 2 interupsi timer, dan satu interupsi port serial.. Register yang mengontrol interupsi yaitu IE(Interrupt enable) dan IP (Interrupt priority)

Interrupt enable

MSB							LSB
EA	-	-	ES	ET1	EX1	ET0	EX0

Simbol	Posisi	Fungsi
EA	IE.7	Melumpuhkan semua interupsi. Jika EA = 0 tidak ada interupsi yang akan dilayani. Jika EA = 1 setiap sumber interupsi dapat dijalankan atau dilumpuhkan secara individual
-	IE.6	Kosong
-	IE.5	Kosong
ES	IE.4	Bit pembuat enable port serial
ET1	IE.3	Bit pembuat enable timer 1
EX1	IE.2	Bit pembuat enable $\overline{INT1}$
ET0	IE.1	Bit Pembuat enable timer 0
EX0	IE.0	Bit pembuat enable $\overline{INT0}$

Jika akan mengaktifkan interupsi 0 ($\overline{INT0}$) misalnya nilai yang harus diberikan ke IE adalah 81H(memberikan logika 1 ke EA dan EX0)

Prioritas interupsi

Setiap sumber interupsi dapat deprogram secara individual menajdi satu atau dua tingkat prioritas dengan mengatur bit SFR yang bernama IP (Interrupt priority). NInterupsi dengan prioritas rendah dapat diinterupsi oleh interupsi yang memiliki prioritas lebih tinggi.

Bit bit pada IP adalah :

MSB							LSB
-	-	-	PS	PT1	PX1	PT0	PX0

Priority bit = 1 menandakan prioritas tinggi
 Priority bit = 0 menandakan prioritas rendah

Timer/Counter

Bila timer/counter diaktifkan pada frekuensi kerja 12 MHZ , timer/counter akan melakukan perhitungan setiap 1 mikrodetik secara independent. Apabila peroda waktu telah dilampaui, timer/counter segera menginterupsi

Perioda waktu timer/counter dirumuskan sebagai berikut :

a. Sebagai timer/counter 8 bit

$$T = (255 - TLx) * 1 \mu s$$

Dimana TLx isi register TL0 atau TL1

b. Sebagai timer counter 16 bit

$$T = (65535 - THxTLx) * 1 \mu s$$

THx = isi register TH0 atau TH1

TLx = isi register TL0 atau TL1

Pengontrol kerja timer/counter adalah register timer control (TCON) bit pada timer counter adalah sebagai berikut :

MSB				LSB			
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

Simbol	Posisi	Fungsi
TF1	TCON.7	Timer 1 overflow flag. Di-set oleh perangkat keras saat timer/counter menghasilkan limpahan (overflow)
TR1	TCON.6	Bit untuk menjalankan Timer 1. Diset/clear oleh software untuk membuat timer on atau off
TF0	TCON.5	Timer 0 overflow flag. Di-set oleh perangkat keras.
TR0	TCON.4	Bit untuk menjalankan Timer 0. Di-set/clear oleh software untuk membuat timer on atau off.
IE1	TCON.3	External interrupt 1 edge flag
IT1	TCON.2	Interrupt 1 type control byte. Set/clear oleh software untuk menspesifikasikan sis turun/level rendah trigger dari interupsi eksternal
IE0	TCON.1	External interrupt 0 edge flag
IT0	TCON.0	Interrupt 0 type control bit

Pengontrol pemilihan mode operasi timer/counter adalah register timer mode (TMOD) yang mana definisi bit- bitnya adalah sebagai berikut :

MSB				LSB			
Timer/counter 1				Timer/counter 0			
GATE	C/T	M1	M0	GATE	C/T	M1	M0

Gate : saat TRx dalam TCON diset 1 dan gate = 1, timer/counter x akan berjalan ketika TRx = 1 (Timer dikontrol software)

C/T : Pemilih fungsi timer atau counter. Clear (0) untuk operasi timer dengan masukkan dari system clock internal. Set (1) untuk operasi counter dengan masukkan dari pena T0 atau T1.

M1 dan M0 Bit pemilih mode

Kombinasi Mo dan M1 adalah sebagai berikut :

M1	M0	Mode	Operasi
0	0	0	Timer 13 Bit
0	1	1	Timer/counter 16 Bit
1	0	2	Timer auto reload 8 bit (pengisian otomatis)
1	1	3	TL.0 adalah timer/counter 8 bit yang dikontrol oleh control bit standar Timer 0 TH0 adalah timer 8 Bit dan dikontrol oleh Kontrol bit Timer 1

Perangkat lunak mikrokontroler atmel 8951

Sebuah mikrokontroler tidak akan bekerja bial tidak diberikan program kepadanya. Program tersebut memberitahu mikrokontroler apa yang harus dilakukan.

Sistem Bilangan

Bilangan Biner

Bilangan biner merupakan bilangan dengan dasar 2. Bilangan ini hanya mempunyai lambing bilangan 0 dan 1.

Sebagai contoh bilangan biner 1011 memiliki decimal :

$$\begin{matrix}
 1 & 0 & 1 & 1 \\
 (1*8) + (0*4) + (1*2) + (1*1) = 11 \text{ desimal}
 \end{matrix}$$

Bilangan heksadesimal

Bilangan Heksadesimal merupakan bilangan berbasis 16, dengan symbol bilangan

0 1 2 3 4 5 6 7 8 9 A B C D E F → Keenam huruf mewakili bilangan decimal 10 sampai 15

Contoh A9H = 1010 1001

Simbol Assembler Khusus

Assembler telah menyediakan beberapa symbol tampak pada table di bawah ini

Simbol Khusus	Arti
A	Accumulator
R0..R7	Register serbaguna
DPTR	Data Pointer register 16 bit
PC	Program Counter. Register 16 bit yang berisi alamat instruksi berikutnya yang akan dijalankan
C	Carry Flag
AB	Akumulator/register B, pasangan register untuk perkalian dan pembagian

Pengalamatan tak langsung

ADD a,@R0 ; Tambahkan isi RAM yang lokasinya ditunjukkan oleh register R0 ke akumulator

DEC @R1 ; Kurang satu isi RAM yang alamatnya ditunjukan register R1

MOVX @DPTR,A ; Pindahkan isi akumulator ke memori luar yang lokasinya ditunjukkan oleh data pointer (DPTR)

Pengalamatan Langsung

MOV A, #01H ;isi akumulator dengan 01

MOV DPTR,#19AB ;isi DPTR dengan 19AB

Pengalamatan BIT

SETB TR1 ; Set TR1

SETB 88H.6 ;Set bit 6 pada lokasi 88H

Instruksi 8951 sebagai berikut

Microcontroller Instruction Set

For interrupt response time information, refer to the hardware description chapter.

Instructions that Affect Flag Settings⁽¹⁾

Instruction	Flag			Instruction	Flag		
	C	OV	AC		C	OV	AC
ADD	X	X	X	CLR C	0		
ADDC	X	X	X	CPLC	X		
SUBB	X	X	X	ANL C,bit	X		
MUL	0	X		ANL C,/bit	X		
DIV	0	X		ORL C,bit	X		
DA	X			ORL C,/bit	X		
RRC	X			MOV C,bit	X		
RLC	X			CJNE	X		
SETB C	1						

Note: 1. Operations on SFR byte address 208 or bit addresses 209-215 (that is, the PSW or bits in the PSW) also affect flag settings.

The Instruction Set and Addressing Modes

R_n	Register R7-R0 of the currently selected Register Bank.
direct	8-bit internal data location's address. This could be an Internal Data RAM location (0-127) or a SFR [i.e., I/O port, control register, status register, etc. (128-255)].
@ R_i	8-bit internal data RAM location (0-255) addressed indirectly through register R1 or R0.
#data	8-bit constant included in instruction.
#data 16	16-bit constant included in instruction.
addr 16	16-bit destination address. Used by LCALL and LJMP. A branch can be anywhere within the 64K byte Program Memory address space.
addr 11	11-bit destination address. Used by ACALL and AJMP. The branch will be within the same 2K byte page of program memory as the first byte of the following instruction.
rel	Signed (two's complement) 8-bit offset byte. Used by SJMP and all conditional jumps. Range is -128 to +127 bytes relative to first byte of the following instruction.
bit	Direct Addressed bit in Internal Data RAM or Special Function Register.

Langkah Langkah dalam perancangan rangkaian Menggunakan Komponen tambahan

1. Pemetaan Memory

Pemetaan ini bertujuan untuk memudahkan kita mengalokasikan memory dan data tidak saling bertabrakan yang kita gunakan terutama bila kita banyak menggunakan device luar

Langkah pemetaan memory

Contoh :

- bila mikrokontroler terhubung ke sejumlah peralatan PPI, RAM, RTC buat table seperti berikut :

Kita buat table

Ram	RTC	PPI	
0	0	1	Ram Aktif
0	1	0	RTC aktif
1	0	0	PPI Aktif

- Dari table yang kita buat kita dapat alamat untuk mengaktifkan peralatan dengan menggunakan MUX yang mempunyai rumus $2^{\text{input}} = \text{Output}$

Buat blok dari table tersebut

	0000H
RAM	0999H 1000H
RTC	1FFFH 2000H
PPI	2FFFH 4000H 4FFFH

Petakan pin yang terhubung :

Mengaktifkan RAM

P2.7 P2.6 P2.5 P2.4 P2.3 P2.2 P2.1 P2.0 P0.7 P0.6 P0.5 P0.4 P0.3 P0.2 P0.1 P0.0
0 0 0 1 0 0 0 0 0 0 0 0 0 0 0 0

P2.7 P2.6 P2.5 P2.4 P2.3 P2.2 P2.1 P2.0 P0.7 P0.6 P0.5 P0.4 P0.3 P0.2 P0.1 P0.0
0 0 0 1 1 1 1 1 1 1 1 1 1 1 1

Mengaktifkan RTC

P2.7 P2.6 P2.5 P2.4 P2.3 P2.2 P2.1 P2.0 P0.7 P0.6 P0.5 P0.4 P0.3 P0.2 P0.1 P0.0
0 0 1 0 0 0 0 0 0 0 0 0 0 0 0

P2.7 P2.6 P2.5 P2.4 P2.3 P2.2 P2.1 P2.0 P0.7 P0.6 P0.5 P0.4 P0.3 P0.2 P0.1 P0.0

0 0 1 0 1 1 1 1 1 1 1 1 1 1 1 1

Mengaktifkan PPI

P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0

P2.7	P2.6	P2.5	P2.4	P2.3	P2.2	P2.1	P2.0	P0.7	P0.6	P0.5	P0.4	P0.3	P0.2	P0.1	P0.0
0	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1

Program Bantu

Ada dua program Bantu yang dapat digunakan untuk mempelajari ATMEL 8951 yaitu AVSIM51 dan ALDS. AVSIM51 merupakan suatu program yang dapat mensimulasikan proses yang terjadi dalam prosesor 8951 seperti bagaimana kondisi output port, kondisi accumulator, menampilkan isi carry flag, dan banyak lagi.. Sedang untuk mengcompile program yang kita buat kita dapat menggunakan program ALDS yang mempunyai tampilan editor teks mirip dengan turbo Pascal.

Untuk memakai program AVSIM51 ketikkan AVSIM51 dan tekan enter akan muncul pesan sebagai berikut :

AVSIM 8051 Simulator/Debugger

Serial # 01036 Licensed by Avocet Systems, Inc.

Copyright (C) 1984,1985,1986 by Ken Anderson Software
All Rights Reserved

Intel 8051 Family Microcomputers

HMOS ROM HMOS ROMless CMOS ROM

A: 8051/8751	C: 8031	E: 80C51
B: 8052/8752	D: 8032	F: 80C31

Choose a CPU for simulation:

. Setelah itu kita pilih jenis mikrokontroler untuk 8951 pilih (A).

ALDS V1.26 File Assembly Processor MCS51 Debug
Line 1 Col 1 Insert Indent Tab C:NONAME.ASM

F2-Save F3-Load F9-Assembly Alt:F3-Pick F10-Menu STTS 10141991

Membuka program

File → Load → <nama file>

Simpan Program/Save As

File → Write to → <tulis nama file .asm>
Atau tekan F2

Kompile program

Tekan F9 sebelumnya cek dulu di menu assembly pengarah kompilernya :

- Buat Hex bila dibuka di program AVSIM51
- Buat BIN/HEX bila akan dimasukkan ke Prosesor

Jangan lupa pastikan prosesor bertuliskan Processor MCS51

Contoh Program

Mengaktifkan Port Pararel

Untuk mengaktifkan port pararel dapat digunakan instruksi MOV. Sebagai contoh :

MOV A,P2 : Masukkan data dari Port 2 ke akumulator
MOV P2,A : Keluarkan data dari akumulator ke port 2

Sebuah port memiliki 8 jalur saluran karena itu untuk membedakannya kedelapan jalur dinamai sesuai urutan nama port. Sebagai contoh berikut ini

Port	P1.7	P1.6	P1.5	P1.4	P1.3	P1.2	P1.1	P1.0
Data	1/0	1/0	1/0	1/0	1/0	1/0	1/0	1/0

Data diatas bernilai 1 atau 0 tergantung data yang akan kita kirim ke port.

Misal port 1.7 dan port 1.5 yang aktif (misal untuk aktif bernilai 1) maka data yang kita kirimkan ke port yaitu : #10100000b atau #0A0H maka perintah yang kita berikan MOV P1,#10100000b atau MOV P1,#0A0H (ditambahkan angka 0 bila data yang kita kirim berupa kode huruf).

Tanda “#” menunjukkan data yang dikirim adalah nilai /value bukan alamat.

Selain menggunakan perintah MOV untuk men-set jalur dapat digunakan perintah SETB. Misal untuk menyalakan P1.0 dapat digunakan perintah

```
SETB P1.0
```

Program LOOP

```
Start:  MOV P1,#01H : isi port 1 dengan nilai 01h
Ulang:  NOP          : tanpa operasi
        SJMP ulang  : Menuju label ulang
```

Yang perlu diperhatikan pada program diatas penulisan label langsung diikuti “:” tanpa spasi. Jarak antara titik dua dan instruksi minimal dipisahkan oleh 1 space

Program Tunda

Subrutin tunda digunakan bila kita ingin respon alat yang dikontrol tidak terlalu cepat. Dibawah ini contoh program untuk membuat lampu berkedip

```

Start:      SETB P1.0
            ACALL DELAY : Panggil subrutin delay
            CLR P1.0
            ACALL DELAY

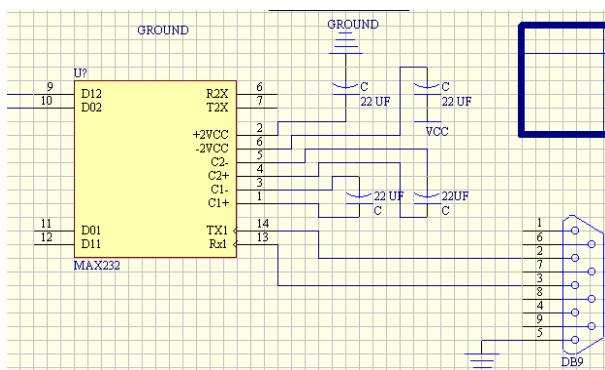
DELAY:      MOV R6,#0FFH
DEL1:       MOV R5,#0FFH
DEL2:       MOV R4,#0AH
            DJNZ R4,$      : Kurangkan nilai R4 sampai nilai 0 bila telah nol ke baris
                               berikutnya
            DJNZ R5, DEL2  : kurangkan nilai R5 bila belum nol lompat ke del2
            DJNZ R6,DEL1   : kurangkan nilai R6 bila belum nol lompat ke del1
            RET

END.
    
```

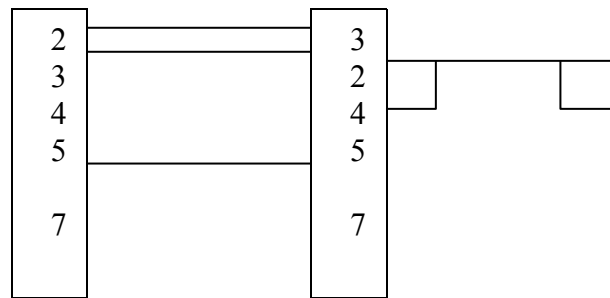
Port Serial.

Komunikasi serial adalah komunikasi dengan data yang datang secara berurutan tiap bitnya. Dalam prakteknya hanya membutuhkan 3 kabel saja yaitu kabel untuk transmit TX untuk receive RX.

Untuk dapat berhubungan dengan komputer ditambahkan MAX IC 232 dengan rangkaian sebagai berikut :



Nomor pin yang dihubungkan adalah



Mikro

COM 1

Port Serial dalam mikrokontroler memiliki sifat full duplex yang berarti dapat mengirim dan menerima secara bersamaan. Register pengirim dan penerima diakses pada register SBUF. Register pengontrol kerjanya yaitu SCON. Bit SCON didefinisikan sebagai berikut :

MSB				LSB			
SM0	SM1	SM2	REN	TB8	RB8	T1	R1

Simbol	Posisi	Fungsi
SM0	SCON.7	Pemilih Mode Port Serial
SM1	SCON.6	Pemilih Mode Port Serial
SM2	SCON.5	Membuat enable komunikasi multiprosesor dalam mode 2

		dan 3
REN	SCON.4	Set/Clear oleh Perangkat Lunak untuk menjalankan/melumpuhkan penerimaan
TB8	SCON.3	Bit ke 9 yang akan dikirim dalam mode 2 dan 3 secara software
RB8	SCON.2	Dalam Mode 2 dan 3 adalah bit ke 9 yang diterima. Dalam Mode 1 jika SM2 =0, RB8. Merupakan Bit stop yang diterima. Dalam mode 0 RB8 tidak diterima
T1	SCON.1	Transmisi interrupt flag. Diset oleh hardware pada akhir bit ke 8 dalam mode 0. atau pada permulaan dari bit stop dalam mode lainnya secara software
R1	SCON.0	Receive interupt flag. Diset oleh hardware pada akhir bit ke 8 dalam mode 0

Tabel Pemilihan mode port serial

SM0	SM1	Mode	Keterangan	Baud rate
0	0	0	Shift Register	Frek Osc/2
0	1	1	8 bit UART	Variabel
1	0	2	9 Bit UART	Frek osc/64 atau osc/32

Baud Rate

Pada Mode 0 baud rate adalah 1/12 frekuensi Oscilator dan hanya dibuthkan register SCON. Di Mode 1 satu baud rate dapat diubah. Baud rate dapat dihasilkan timer 1. Untuk ini dilakukan dalam mode 2 (auto reload)

$$\text{Baud Rate} = \frac{(K * \text{Frekuensi Oscilator})}{(32 * 12 * [256 - TH1])}$$

Nilai K ditentukan oleh SMOD dalam power Control register (PCON) Bila SMOD = 0 maka k = 1, bila SMOD = 1 maka k = 2Bila diketahui baud rate, nilai TH1 melalui persamaan berikut :

$$TH1 = 256 - \frac{(K * \text{FREKUENSI OSCILATOR})}{(384 * \text{BAUD RATE})}$$

Nilai TH1 dalam bentuk Integer. Untuk PCON dialamati dengan ORL PCON, #80H Untuk Mode 2 Baud Rate bila SMOD = 1 Baud Rate 1/32 SMOD = 0 Baud Rate 1/64, pada mode 3, baud rate diatur seperti mode 1.

Percobaan

1. Percobaan Penggunaan Instruksi MOV, EQU, RR, RL ,SJMP,ACALL

Dalam Percobaan ini digunakan register A Register Komunikasi P0, P1, P2 dan Register Bantu R0 – R7

Data EQU 040h
 Start: MOV Data,#80h

```

Putar:    MOV P1,Data
            MOV A,Data
            RR A
            MOV Data,A
            ACALL DELAY : Panggil subrutin delay
            SJMP Putar
DELAY:  MOV R6,#0FFH
            DJNZ R6,$   : Kurangkan nilai R6 sampai nilai 0 bila telah nol ke baris
                       berikutnya
            RET
END.
  
```

Cara menghitung lama waktu delay diatas yaitu sebagai berikut :

Atmel untuk mengeksekusi program dibutuhkan total 12 cycle. Bila Kristal yang kita gunakan 12 MHZ maka:

```

1 cycle → 12(cycle)/12(Mhz) : 1 u second.
Contoh untuk kasus delay diatas
Acall 2 cycle
MOV 1 cycle
DJNZ 3 cycle
RET 1 cycle
  
```

Maka untuk menghitung delay :

Acall Delay (2) + mov r6,#0FFH (1) + DJNZ (3*0ffh) + RET (1)

2. Penggunaan Timer dan Counter

Register Yang Digunakan

TMOD → Untuk Mengatur Inisialiasi awal

TCON → Untuk ON OFF Timer atau Counter

THx,TLx → Untuk mengatur Waktu Timer

Berikut Program Untuk membuat timer selama 1 Detik

Memakai Timer

Atmel 8951 memiliki dua buah timer/counter 16 bit. Timer atau counter dapat diaktifkan melalui perangkat keras maupun perangkat lunak. Untuk menjalankan timer/counter melalui perangkat, bit TRx harus diset. Bila hitungan timer/counter overflow bit TFx akan diset.

Timer/counter dapat dijalankan melalui perangkat keras dengan memberikan transisi negative pada pena INT dengan syarat Bit TRx sudah diset terlebih dahulu. Program berikut digunakan untuk mengedipkan Port 1.0 dengan tundaan melalui timer/counter 0

```

Start:     SETB P1.0
            ACALL DELAY
            CLR P1.0
            ACALL DELAY
            SJMP START
  
```

DELAY:

```

MOV R0,#00H
LAGI:  MOV TM0D,#01H
      MOV TH0,#0D8H
      MOV TL0,#0EFH
      SETB TR0
ULANG: NOP
      JBC TF0, HITUNG
      SJMP ULANG
HITUNG: INC R0
      CJNE R0,#64H,LAGI
      RET

```

Tundaan dibuat menggunakan timer/counter 0 pada mode 1, yaitu timer 16 bit. Dengan mengisi nilai TH dan TL sesuai program diatas timer akan selesai menghitung selama kira-kira 0,01 detik (frekuensi clock 12 MHZ) register R0 digunakan sebagai pembagi 100 sehingga lampu akan berkedip selama kira-kira 1 detik.

Bila program akan dijalankan dengan timer 1, instruksi yang harus diganti

```

MOV TM0D,#01H diganti MOV TM0D,#10H
SETB TR0          diganti SETB TR1

```

Memakai Counter

ATMEL 89c51 memiliki 2 buah counter yaitu counter 0 dan counter 1. Untuk mengaktifkan counter 0

Set Bit C/T dengan 1

mode operasi bila 16 bit kode untuk M0 dan M1 yaitu 01

Aktifkan bit TR0/TR1. Untuk mengambil hasil counter di register TL0.

Contoh program counter dibawah ini :

```

MOV TM0D,#050H ; SET COUNTER 0. MODE 1 (16 BIT)
SETB TR0       ; AKTIFKAN TR1
MOV A,TL0      ; BACA ISI COUNTER
MOV P1,A       ; KIRIM KE PORT 1

```

3. Menggunakan Fasilitas Interupsi

Program berikut akan menyalakan port 1.0 Namun bila interupsi eksternal 0 diaktifkan dengan membuat rendah pena $\overline{INT\ 0}$ port 1.7 yang akan dinyalakan sementara port 1.0 dimatikan.

```

Start:  ORG 00h
        AJMP LANJUT
        ORG 03H      : ALAMAT INT 0
        MOV P1,#80H
        RETI
LANJUT: MOV IE,#81H  : IZINKAN MASUK INT 0
        MOV IP,#01H  : PRIORITASKAN
        MOV P1,#01H  : NYALAKAN PORT
ULANG:  NOP
        SJMP ULANG

```

Bila digunakan interupsi 1 maka

ORG 03 diganti ORG 13H

Isi IE diganti #84H

Isi IP diganti #04H

4 Percobaan Serial

Untuk Serial ada dua Program yang dirancang yaitu program di Mikrokontroler dan Program Di PC

Register Yang digunakan :

Register TH0 → Untuk Mengatur Baud Rate

Register SCON → Untuk Inisialisasi

Register TCON/TR1 → Untuk ON OFF Timer

Register SBUF Untuk Penampung data

Register TI Untuk Flag Transmit

Register RI Untuk Flag Receive

Program Di Mikrokontroler sebagai Berikut :

ORG 00H

MOV SCON,#50H

MOV TMOD,#20h

MOV TH1,#250

SETB TRI

TERIMA: JNB RI,TERIMA

MOV A,SBUF

CLR RI

MOV P0,A

MOV P1,A

MOV P2,A

CLR TI

MOV SBUF,A

KIRIM: JNB TI,KIRIM

SJMP TERIMA

Program Di Komputer

Private Sub Command1_Click()

MSComm1.PortOpen = True

MSComm1.Settings = "4800,n,8,1"

MSComm1.Output = Chr(Text1.Text)

Text2.Text = MSComm1.Input

MSComm1.PortOpen = False

End Sub

Soal .soal.

1. Buat program yang menyalakan lampu di port 1 dari kiri ke kanan bila pin interupsi 0(p3.2) disentuh nyala lampu berubah dari kanan ke kiri
2. Microcontroler pertama dihidupkan seluruh lampu di port 1 akan menyala berkedip-kedip bila pin interupsi 0(p3.2) disentuh lampu akan menyala dari kanan ke kiri
3. Microcontroler pertama dihidupkan lampu di port 1.0 akan berkedip-kedip bila pin interupsi 0 (p2.3) disentuh seluruh lampu di port 1 akan menyala kemudian bergeser dari kiri ke kanan
4. Buat program bila telah menerima counter 5 kali maka lampu yang terhubung di port 1 akan berkedip sebanyak 2 kali kemudian lampu di port 2 akan berubah dari 0Fh menjadi F0h kemudian kembali lagi menjadi 0Fh dan proses berulang kembali ke awal
5. Buat Program Penggeser data Komunikasi Serial antara PC dan Mikrokontroler yaitu bila Command >> bergeser ke kanan bila Command << bergeser ke kiri

Daftar Pustaka

- Moh. Ibnu Malik & Anistradi, Bereksprimen dengan Mikrokontroler 8031, PT ELEX MEDIA KOMPUTINDO, 1999
- Chacha, Ir P. Insap Santosa, MSC, Bagaimana mendayagunakan sepasang komputer, Andi Offset, 1995
- Myke Predko, Programing and Customizing the 8051 microcontroler, McGraw-Hill, 1999
- Sencer Yeralan, Ashutosh Ahluwalia, Programing and Interfacing the 8051 Microntroller, addison-Wesley Publishing Company, 1995
- Data Sheet. [http\\:www.atmel.com](http://www.atmel.com)
- Paulus Andi Nalwan, teknik antarmuka dan pemrograman mikrokontroler AT 89C51, Elex Media Komputindo

