



ORGANISASI MEMORI MIKROKONTROLER MCS-51

Yoyo Somantri dan Erik Haritman
Dosen Jurusan Pendidikan Teknik Elektro
FPTK Universitas Pendidikan Indonesia

Pendahuluan

Dalam bab ini akan dibahas tujuan perkuliahan, organisasi memori program, memori data, Flash memory, dan mode pengalamatan.

Tujuan Perkuliahan

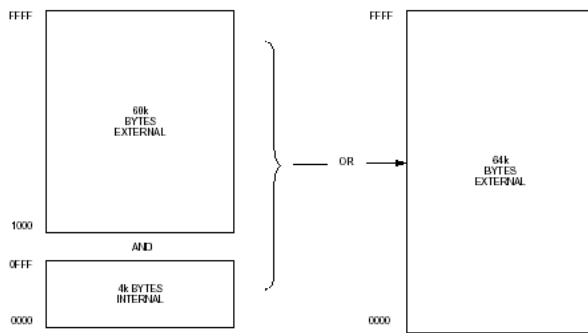
Setelah mempelajari bab ini, diharapkan mahasiswa mampu untuk :

1. Memahami organisasi memori mikrokontroler
2. Memahami memori program
3. Flash memori
4. Memahami mode pengalamatan.

1. Memori Program

Mikrokontroler MCS-51 memiliki ruang alamat memori data dan program yang terpisah. Memori program adalah tempat program (*binary program*) yang akan dijalankan. Ruang maksimum untuk memori program sebesar 64 KB, akan tetapi untuk memaksimalkan memori tersebut perlu menggunakan memori luar (eksternal memori) disamping memori dalam (internal memori). Tidak semua mikrokontroler dari keluarga MCS-51 mempunyai memori dalam seperti 8051, dan untuk yang mempunyai memori dalam kapasitasnya juga berbeda-beda.

Gambar di bawah ini memperlihatkan peta memori program dari 8051,

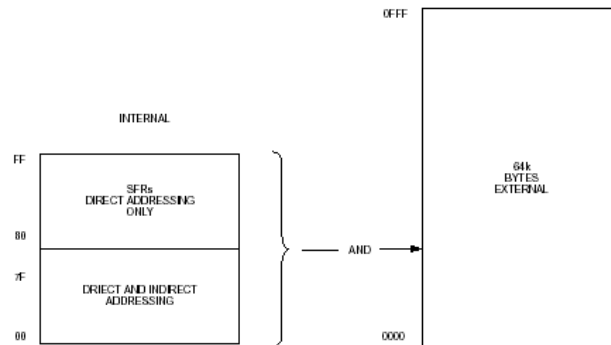


Gambar 1. Peta memori program

2. Memori Data

Memori data merupakan tempat menyimpan data-data yang dibutuhkan untuk proses program. Semua keluarga MCS-51 selalu mempunyai Memori Data Dalam (*Internal Data Memory*) akan tetapi kapasitasnya berbeda yaitu ada yang 256 byte (8051) dan ada yang 384 byte (8052). Sedangkan Memori Data Luar (*External Data Memory*) mempunyai kapasitas maksimum 64 kb.

Gambar di bawah ini memperlihatkan peta memori data dari 8051



Gambar 2. Peta memori data

3. Flash PEROM AT 89C51

Flash PEROM, merupakan teknologi terbaru ROM yang diperkenalkan oleh berbagai perusahaan semikonduktor IC MCS-51 dapat dianggap sebagai IC Flash PEROM. Sebagai Flash PEROM ada 5 hal yang dapat dilakukan, yakni :

- a. Menghapus isi *Flash* PEROM semuanya sekaligus.
- b. Mengisi *Flash* PEROM byte per byte.
- c. Mengambil isi *Flash* PEROM byte per byte.
- d. Selain itu untuk mencegah pembajakan, *Flash* PEROM dapat dikunci, sehingga program yang disimpan di dalamnya tidak bisa diambil keluar dari chip. Disediakan tiga macam pengaman yang berbeda tingkat.
- e. Mikrokontroler produksi Atmel masing-masing mempunyai kode produksi, ini memudahkan program di komputer mengenali *chip* mikrokontroler jenis apa yang dipasangkan pada *flash* PEROM Programmer.

Hal-hal tersebut diatas diatur lewat kombinasi sinyal yang diberikan pada P3₆ dan P3₇ (kaki no 16 dan 17) serta P2₆ dan P2₇ (kaki no 27 dan 28). Selain sinyal-sinyal itu perlu pula diatur tegangan yang diumpankan ke Vpp (kaki no 31), untuk keperluan pengisian informasi ke dalam *Flash* PEROM diperlukan tegangan 12 Volt yang disertai dengan pulsa pada \overline{PROG} (kaki no 30). Sedangkan untuk pembacaan cukup memakai tegangan Vpp 5 Volt. Mode kerja *Flash* PEROM AT 89C51 diperlihatkan pada tabel dibawah ini .

Tabel 1. Mode Kerja Flash PEROM AT89C51.

Mode Kerja	P3 ₇	P3 ₆	P2 ₇	P2 ₆	Vpp	PROG
Menghapus Isi PEROM	L	L	L	H	12 V	L
Mengisi PEROM	H	H	H	L	12 V	L
Membaca PEROM	H	H	L	L	5 V	H
Proteksi bit 1	H	H	H	H	12 V	L
Proteksi bit 2	L	L	H	H	12 V	L
Proteksi bit 3	L	H	L	H	12 V	L
Membaca tanda ATMEL	L	L	L	L	5 V	H

Daerah indirect address

Yaitu lokasi memori data yang diakses dengan cara penunjukkan alamat yang tidak langsung. Sebagai gambaran diperlihatkan mengisi data pada 8052 dengan masing-masing pada alamat 80H.

Contoh 1 : MOV 80H, #11H

Contoh2 : MOV R0, #80H

 MOV @R0, #22H

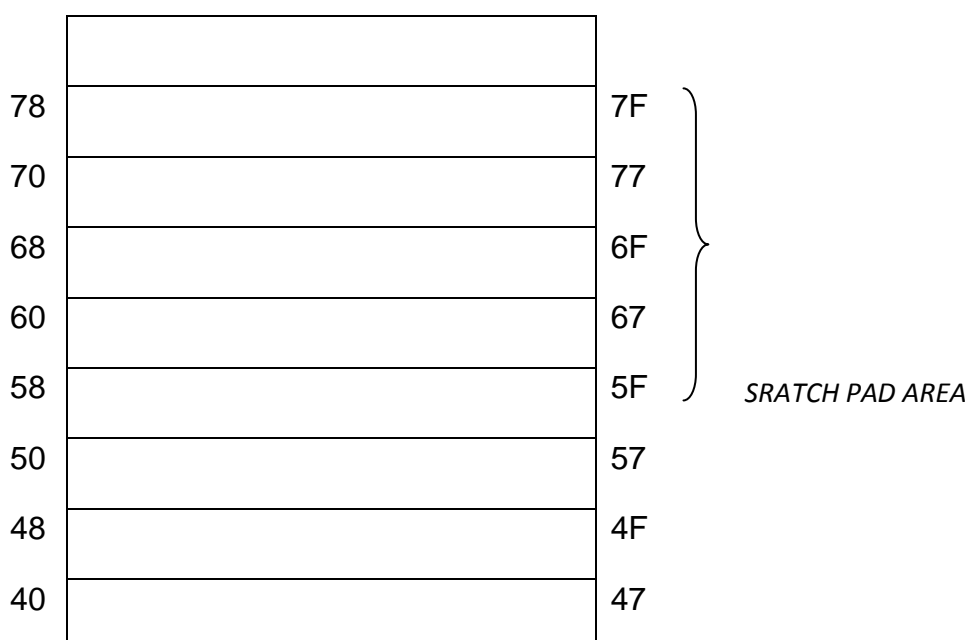
Pada contoh 1 transfer data direct address berlaku pengisian data ke blok memori SFRs (P0) yang beralamat 80H, sedangkan pada contoh 2 transfer data indirect address berlaku pengisian data ke memori data yang biasanya beralamat 80H (hanya pada 8052). Untuk daerah memori data dari alamat 00H sampai 7FH tidak ada perbedaan lokasi data antara 8951 dan 8952.

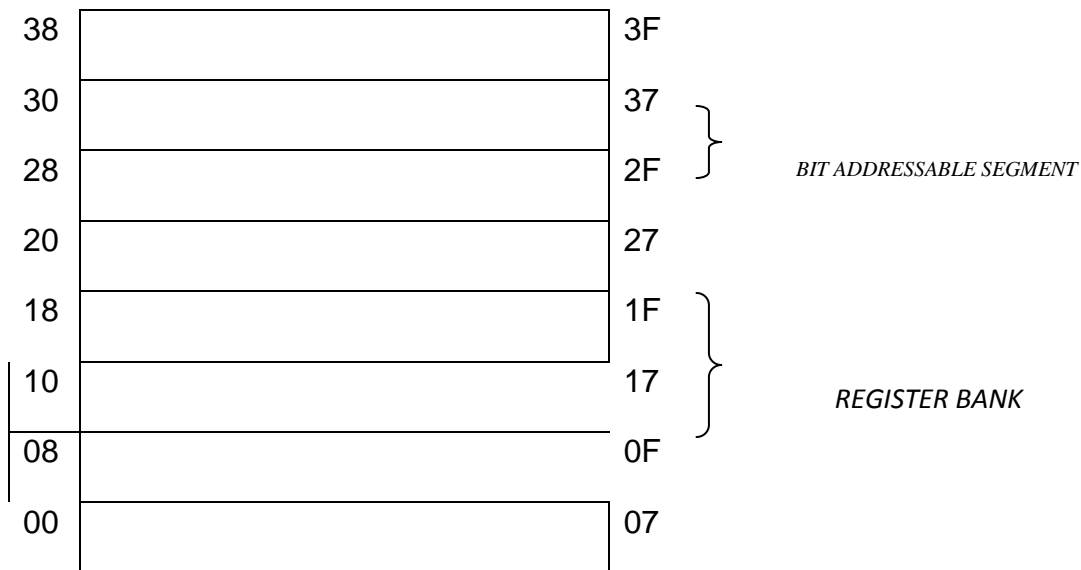
Daerah *direct address*

Yaitu lokasi memori data yang diakses dengan cara langsung menuliskan alamat lokasi memorinya.

Contoh : MOV 30H, #33H (Lokasi memori data yang beralamat 30H diisi data 33H)

Daerah memori data dari alamat 00h s/d 7fh





Daerah memori data dari alamat 80h s/d ffh

Terdapat dua kategori memori data pada alamat 80H s/d FFH untuk keluarga MCS-51, yaitu :

- Kontroler yang mempunyai 1 blok data memori
- Kontroler yang mempunyai 2 blok data memori yang beralamat sama.

Untuk kategori yang mempunyai 2 blok data memori masing-masing blok disebut blok SFRs (*Special Function Register*) dan blok memori data biasa.

Tabel 2. Komposisi SFRs

Simbol	Nama	Alamat
* ACC	<i>Accumulator</i>	0E0H
* B	<i>B Register</i>	0F0H
* PSW	<i>Program Status Word</i>	0D0H
SP	<i>Stack Pointer</i>	81H
DPTR	<i>Data Pointer 2 Bytes</i>	
DPL	<i>Low Byte</i>	82H
DPH	<i>High Byte</i>	83H
*P0	<i>Port 0</i>	80H

*P1	<i>Port 1</i>	90H
*P2	<i>Port 2</i>	0A0H
*P3	<i>Port 3</i>	0B0H
*IP	<i>Interrupt Priority Control</i>	0B8H
*IE	<i>Interrupt Enable Control</i>	0A8H
TMOD	<i>Timer/ Counter Mode Control</i>	89H
*TCON	<i>Timer/ Counter Control</i>	
*+T2CON	<i>Timer/ Counter 2 Control</i>	
TH0	<i>Timer/ Counter Control 0 high byte</i>	
TL0	<i>Timer/ Counter Control 0 low byte</i>	
TH1	<i>Timer/ Counter Control 1 high byte</i>	
TL1	<i>Timer/ Counter Control 1 low byte</i>	
+TH2	<i>Timer/ Counter Control 2 high byte</i>	
+TL2	<i>Timer/ Counter Control 2 low byte</i>	
+RCAP2H	<i>T/C Capture Reg. High Byte</i>	
+RCAP2L	<i>T/C Capture Reg. Low Byte</i>	
*SCON	<i>Serial Control</i>	
SBUF	<i>Serial Data Buffer</i>	
PCON	<i>Power Control</i>	

Tabel 3. Isi SFRs setelah *power on* atau *reset*

<i>Register</i>	Data (Biner)
* ACC	00000000
* B	00000000
* PSW	00000000
SP	00000111
DPTR	
DPL	00000000
DPH	00000000
*P0	11111111
*P1	11111111
*P2	11111111
*P3	11111111
*IP	8051 xxx00000 8052 xx000000
*IE	8051 0xx00000 8052 0x000000
TMOD	00000000
*TCON	00000000
*+T2CON	00000000
TH0	00000000
TL0	00000000
TH1	00000000
TL1	00000000
+TH2	00000000
+TL2	00000000

+RCAP2H	00000000
+RCAP2L	00000000
*SCON	00000000
SBUF	Indeterminate
PCON	0xxxxxxx

FUNGSI-FUNGSI REGISTER PADA SFRs

- PSW (*Program Status Word*)

PSW. 7	PSW.6	PSW.5	PSW.4	PSW.3	PSW.2	PSW.1	PSW. 0
CY	AC	F0	RS1	RS0	OV	-	P

- CY = *Carry Flag*
 AC = *Auxiliary Carry*
 F0 = Flag 0 disediakan bebas untuk pengguna
 RS1 = *Register Bank Selector 1*
 RS0 = *Register Bank Selector 0*
 OV = *Overflow Flag*
 P = *Parity Flag* (dari AC)

RS	RS0	Register Bank	Address
1			
0	0	0	00H – 07H
0	1	1	08H – 0FH
1	0	2	10H – 17H
1	1	3	18H – 1FH

Masing-masing Register Bank mempunyai 8 register dengan penamaan R0 s/d R7

- **IE (INTERRUPT ENABLE REGISTER)**

IE.7	IE.6	IE.5	IE.4	IE.3	IE.2	IE.1	IE.0
EA	-	ET2	ES	ET1	EX1	ET0	AX0

EA : *Enable All Interrupt*
ET2 : *Enable Timer 2 Overflow or Capture Interrupt*
ES : *Enable Serial Port Interrupt*
ET1 : *Enable Timer 1 Overflow Interrupt*
EX1 : *Enable External 1 Interrupt*
ET0 : *Enable Timer 0 Overflow Interrupt*
EX0 : *Enable External 0 Interrupt*

• **IP (INTERRUPT PRIORITY REGISTER)**

IP.7	IP.6	IP.5	IP.4	IP.3	IP.2	IP.1	IP.0
-	-	PT2	PS	PT1	PX1	PT0	PX0

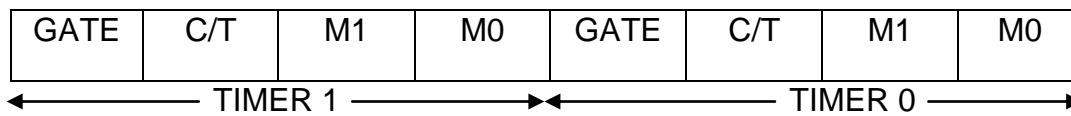
PT2 : *Interrupt Priority Timer 2*
PS : *Interrupt Priority Serial Port*
PT1 : *Interrupt Priority Timer 1*
PX1 : *Interrupt Priority External 1*
PT0 : *Interrupt Priority Timer 0*
PX0 : *Interrupt Priority External 0*

• **TCON (TIMER/ COUNTER CONTROL REGISTER)**

TCON. 7	TCON. 6	TCON. 5	TCON. 4	TCON. 3	TCON. 2	TCON. 1	TCON. 0
TF1	TR1	TF0	TR0	IE1	IT1	IE0	IT0

TF1 : *Timer 1 Overflow Flag*
TR1 : *Timer 1 Run Control Bit*
TF0 : *Timer 0 Run Control Bit*
IE1 : *External Interrupt 1 Edge Flag*
IT1 : *Interrupt 1 Type Control Bit*
IE0 : *External Interrupt 0 Edge Flag*
IT0 : *Interrupt 0 Type Control Bit*

• **TMOD (TIMER/ COUNTER MODE REGISTER)**



GATE : Menentukan pengontrolan *Timer/ Counter* oleh *Hardware* (INTx) atau *software* (TRx)

C/T : *Timer or Counter Selector*

M 1 : *Mode Slector Bit*

M 0 : *Mode Selector Bit*

M 1	M 0	Mode	Operasi
0	0	0	13 Bit <i>Timer</i>
0	1	1	16 Bit <i>Timer/ Counter</i>
1	0	2	8 Bit <i>Auto Reload Timer/ Counter</i>
1	1	3	<i>Timer 0 8 Bit Timer (TL0 & TH0), Timer 1 Stop</i>

• **T2CON (TIMER/ COUNTER 2 CONTROL REGISTER)**

T2CO N.7	T2CO N.6	T2CO N.5	T2CO N.4	T2CO N.3	T2CO N.2	T2CO N.1	T2CO N.0
TF 2	EXF 2	RCLK	TCLK	EXEN2	TR2	C/ T2	CP/RL 2

TF2 : *Timer 2 Overflow Flag*

EXF2 : *Timer 2 External Flag*

RCCLK : *Receive Clock Flag*

TLCK : *Transmit Clock Flag*

EXEN2 : *Timer 2 External Enable Flag*

TR2 : *Software START/ STOP Control Timer 2*

C/ T2 : *Timer or Counter Select*

CP/ RL2 : *Capture/ Reload Flag*

• **SCON (SERIAL PORT CONTROL REGISTER)**

SCON .7	SCON. 6	SCON. 5	SCON. 4	SCON. 3	SCON. 2	SCON. 1	SCON .0
SM0	SM1	SM2	REN	TB8	RB8	TI	RI

SM0 : *Serial Port Mode*

SM1 : *Serial Port Mode*

SM2 : *Enable Komunikasi Multiprosessor pada mode 2 & 3*

REN : *Enable Serial Reception*

TB8 : *Bit ke 9 akan dikirim*

RB8 : *Pada mode 2 & 3, bit ke 9 akan diterima, pada mode 1 (SM2=0) RB8 = Stop bit.*

TI : *Transmit Interrupt Flag*

RI : *Receive Interrupt Flag*

SM 0	SM1	MODE	DESC	BAUD RATE
0	0	0	Shift Register	F. Osc/ 12
0	1	1	8 bit UART	Variable
1	0	2	9 bit UART	F.Osc/ 64 atau F.Osc/ 32
1	1	3	9 bit UART	Variable

3. Mode pengalamatan mcs-51

Terdapat 5 (lima) mode pengalamatan, yaitu :

1. *Direct Addressing* (pengalamatan langsung), *Operand* sumber berisi alamat data.
Contoh : MOV A, 7FH (7FH merupakan alamat memori data)
2. *Indirect Addressing* (pengalamatan tidak langsung), *Operand* sumber berisi alamat memori data yang datanya menunjukkan alamat data yang dimaksud.

Contoh : ADD, @R0 (R0 berisi alamat memori data)

3. *Register Addressing*, *Operand* adalah *register* yang berisi data.

Contoh : MOV A, R1

4. *Immediate Addressing*, *Operand* berisi data langsung yang akan ditransfer.

Contoh : MOV A, #12H

5. *Index Addressing*, Metode pengalamatan ini hanya digunakan pada memori program yaitu untuk mengambil data dengan menggunakan DPTR atau *Program Counter* sebagai alamat dasarnya (*Base of Table*) dan menggunakan Akumulator sebagai alamat relatifnya (*Table Entry Number*)

Contoh : MOVC A, @ A + DPTR atau MOVC A, @ A + PC

Referensi :

1. Sencer, (1997). *Programming Interfacing 8051 Microcontroller*. Mc Graw Hill.
2. Intel, (1994). *MCS'51 Microcontroller Family User Manual*.
3. Myke Predko, (1995). *Programming and Customizing The 8051 Microcontroller*. Mc Graw Hill.
4. Allen I Wyatt, (1995). *Using Assembly Language*. Que
5. Atmel, (2005). *Data Book Microcontroller*.
6. Ramakart Gayakwad, Leonard Sokolof. (1988). *Analog and Digital Control Systems*. Canada : Prentice- HallInternational, Inc.
7. Greenfield, Joseph D.(1992). *The 68HC11 Microcontroller*. Orlando, FL:
8. Peter Spasov, (2002). *Microcontroller Technology: The 68HC11*, Prentice-Hall. ISBN: 0-13-019579.
9. Toto Budiono (2005). *Pemograman Bahasa C dgn SDCC*. Gaya Media.
10. 89C51 Development Tools DT51 Version 3. User'S Guide. Manual Book.
11. Agus Bejo, (2008). *C & AVR Rahasia Kemudahan Bahasa C dalam mikrokontroler ATmega 8535*. Graha Ilmu. Yogyakarta.
12. Totok Budioko, (2005). *Belajar dengan mudah dan cepat Pemograman Bahasa C dengan SDCC (Small Device C Compiler)*.Penerbit Gaya Media. Yogyakarta.
13. Agfianto Eko Putra, (2004). *Belajar Mikrokontroler AT 89C51/52/55*. Penerbit Gaya Media. Yogyakarta.