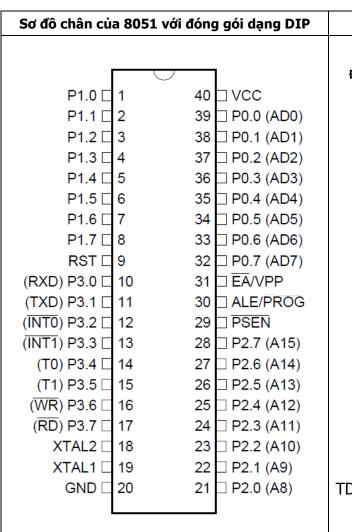
Tóm tắt phần cứng và tập lệnh 8051

Tổ chức của RAM nội (số dạng hex)



					•	_ •		
Địa chỉ								
7F								
30								
2F	7F	7E	7D	7C	7B	7A	79	78
	Vù	ng có	địa ch	ỉ bit từ	rđ/cb	yte 20	đến 2	F
	Địa	a chỉ b	it = (Đ	ịa chỉ	byte -	20) x 8	3 + Vị t	rí bit
		1	ı	1	ı		ı	
20	07	06	05	04	03	02	01	00
1F				R7				
		Ban	k 3 (R	RS1 =	1 và F	RS0 =	1)	
18				R0				
17				R7				
		Ban	k 2 (R	RS1 =	1 và F	RS0 =	0)	
10				R0				
0F				R7				
		Bank 1 (RS1 = 0 và RS0 = 1)						
80		R0						
07				R7				
		Ban	k 0 (R		0 và I	RS0 =	0)	
00				R0				

TD: Tìm địa chỉ bit thứ 5 của byte có địa chỉ 2F. Địa chỉ bit = $(2F - 20) \times 8 + 5 = 7D$

Địa chỉ		Ký hiệu
F0		В
E0		ACC
D0		PSW
В8		IP
В0		P3
A8		IE
Α0		P2
99	Không có địa chỉ bit	SUBF
98		SCON
90		P1
8D	Không có địa chỉ bit	TH1
8C	Không có địa chỉ bit	TH0
8B	Không có địa chỉ bit	TL1
8A	Không có địa chỉ bit	TL0
89	Không có địa chỉ bit	TMOD
88		TCON
87	Không có địa chỉ bit	PCON
83	Không có địa chỉ bit	DPH
82	Không có địa chỉ bit	DPL
81	Không có địa chỉ bit	SP
80		P0

Vùng các thanh ghi SFR (số dạng hex)

Các thanh ghi có địa chỉ tận cùng 0H hay 8H thì các bit có địa chĩ bit và công thức tính địa chĩ bit (=Ký hiệu.Vị trí bit): Địa chỉ bit = Địa chỉ byte + Vị trí bit

Reset và các cổng I/O

Sau khi Reset:

Hầu hết các thanh ghi đều bằng 0, trừ các thanh ghi sau:

SP = 07H

P0 = FFH

P1 = FFH

P2 = FFHP3 = FFH

Nghĩa là sau khi Reset thì CPU mặc nhiên chon bank 0 và stack có SP = 07H

Các cổng I/O:

- P0 không có điện trở kéo lên bên trong, do đó khi dùng như cổng I/O thì phải gắn điện trở kéo lên 10KΩ cho mỗi chân I/O.
- P1, P2, và P3 đều điện trở kéo lên bên trong.
- Đặt cấu hình I/O cho mỗi chân Pm.n: Pm.n = 0: Output, 1: Input TD: Cấu hình nhập cho 4 bit thấp của P1 MOV P1, #0FH

PSW (từ trạng thái chương trình)

7	6	5	4	3	2	1	0
CY	AC	F0	RS1	RS0	OV	ı	Р

Ý nghĩa các cờ:

 $CY = c\dot{\sigma} nh\dot{\sigma} (Carry)$

AC = cờ nhớ phụ (Auxiliary Carry)

F0 = cờ tạm dùng cho các phép toán Boole RS1 và RS0 dùng để chon bank thanh ghi

_		
RS1	RS0	Chọn bank
0	0	0
0	1	1
1	0	2
1	1	3

OV = cờ báo tràn

P = cờ Parity. Bit này sẽ là 0 hay 1 sao cho tổng số bit 1 trong thanh ghi A và P là số chẵn. Cập nhật tức thời khi A thay đổi!

Các thanh ghi liên quan Timer

TMOD (Chế độ timer)

Với 4 bit cao định nghĩa chế độ Timer 1, và 4 bit thấp định nghĩa chế độ Timer 0. GATE: thường cho 0 nếu không dùng chung với /INT0 hay /INT1

C/T = 0 chọn Timer và 1 chọn Counter

M1 M0 Chế độ

0 0 (Timer 13 bit) 0 1 1 (Timer 16 bit) 2 (Timer 8 bit, tự động nạp 1 0 lại) 1 1 3 (2 timer 8 bit)

TCON (Điều khiển Timer)

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

Ý nghĩa các cờ dùng cho Timer:

TFn = cờ báo tràn cho Timer Tn

TRn = cờ điều khiển Timer Tn (0: dừng Timer, 1: cho Timer chay)

Các thanh ghi liên quan cổng nối tiếp

SCON (Điều khiển nối tiếp)

SM0	SM	1 SM2	REN	TB8	RB8	TI	RI			
Ý ngh	Ý nghĩa các bit:									
	SM		hế độ		-	Tốc đ				
0	0 1 baud									
0	0	0 (thanh ghi dịch) cố định								

0	1	•	baud
0	0	0 (thanh ghi dịch)	cố định
0	1	1 (UART 8 bit)	chỉnh được
1	0	2 (UART 9 bit)	cố định
1	1	3 (UART 9 bit)	chỉnh được

SM2 = 0 khi không dùng đa xử lý

REN = 1 cho phép thu nối tiếp, 0:cấm TB8 = bit 8 khi phát dùng UART 9 bit

RB8 = bit 8 khi thu dùng UART 9 bit

TI = 1 khi phát nối tiếp hoàn tất RI = 1 khi thu nối tiếp hoàn tất

PCON (điều khiển công suất) SMOD (MSB của PCON) với chế độ 2:

Tốc độ baud = $F_{XTAL} \times 2^{SMOD} / 64$

F_{XTAL} là tần số bộ dao động

Bảng giá trị nạp TH1 với UART 8 bit

Tốc độ baud	Tần số XTAL(MHz)	SMOD	Trị nạp	Sai số
9600	12.000	1	-7(F9H)	7%
2400	12.000	0	-13 (F3H)	0.16%
1200	12.000	0	-26 (E6H)	0.16%
19200	11.059	1	-3 (FDH)	0
9600	11.059	0	-3 (FDH)	0
2400	11.059	0	-12 (F4H)	0
1200	11.059	0	-24 (E8H)	0

TD: Lập trình 8051 nhận nối tiếp các byte và gửi chúng ra cổng P1. Đặt tốc độ baud 2400 với UART 8 bit. Biết FXTAL = 11.059 MHz Bài giải.

MOV TMOD,#20H; Timer 1, mode 2 MOV TH1,#-12; 2400 baud

MOV SCON, #50H; UART 8 bit và REN=1 SETB TR1; cho Timer 1 chạy

Loop: JNB RI, Loop ; đợi nhận hoàn tất MOV A, SBUF

MOV P1, A

CLR RI ; xóa cờ RI để nhận tiếp SJMP Loop

Các ngắt trong 8051

IE (Cho phép ngắt)

7	6	5	4	3	2	1	0
EA	ı	-	ES	ET1	EX1	ET0	EX0

EA = 1 cho phép ngắt toàn cục, 0: cấm. ES = 1 cho phép ngắt cổng nối tiếp

ETn = 1 cho phép ngắt Timer n (n = 0,1) EXn = 1 cho phép ngắt ngoài n (n = 0,1)

IP (Ưu tiên ngắt)

١,	,	1	⊥: ≙ ∽	مام	~~~~	~ <u>^′:</u> ⊥:	<u>^</u>	
	1	_	_	PS	PT1	PX1	PT0	PX0
	7	6	5	4	3	2	1	0

PS = 1 ưu tiên cho cống nối tiếp PTn = 1 ưu tiên cho Timer n (n = 0,1) PXn = 1 ưu tiên ngắt ngoài n (n = 0,1)

Ngắt thứ	Ngắt	Địa chỉ ISR	Độ ưu tiên
0	ngoài 0	0003H	cao nhất
1	timer 0	000BH	
2	ngoài 1	0013H	
3 timer 1		001BH	
4	nối tiếp	0023H	thấp nhất

ITn = 1 chọn kích cạnh xuống cho ngắt ngoài n

Nhóm lệnh chuyển dữ liệu									
Mnemo	nic	Opcode	#bytes	#MCs					
MOV	A,Rn	E8H+n	1	1					
MOV	A,direct	E5H	2	1					
MOV	A,@Ri	E6H+i	1	1					
MOV	A,#data	74H	2	1					
MOV	Rn,A	F8H+n	1	1					
MOV	Rn,direct	A8H+n	2	2					
MOV	Rn,#data	78H+n	2	1					
MOV	direct,A	F5H	2	1					
MOV	direct,Rn	88H+n	2	2					
MOV	direct, direct	85H	3	2					
MOV	direct,@Ri	86H+i	2	2					
MOV	direct,#data	75H	3	2					
MOV	@Ri,A	F6H+i	1	1					
MOV	@Ri,direct	A7H	2	2					
MOV	@Ri,#data	76H+i	2	1					
MOV	DPTR,#d16	90H	3	2					
MOVC	A,@A+DPTR	93H	1	2					
MOVC	A,@A+PC	83H	1	2					
MOVX	A,@Ri	E3H	1	2					
MOVX	A,@DPTR	E0H	1	2					
MOVX	@Ri,A	F3H	1	2					
MOVX	@DPTR,A	F0H	1	2					
PUSH	direct	C0H	2	2					
POP	direct	D0H	2	2					
XCH	A,Rn	C8H+n	1	1					
XCH	A,direct	C5H	2	1					
XCH	A,@Ri	C6H+i	1	1					
XCHD	A,@Ri	D6H+i	1	1					

	Nhón	ı lệnh số	học	
Mnemo	onic	Opcode	#bytes	#MCs
ADD	A,Rn	28H+n	1	1
ADD	A,direct	25H	2	1
ADD	A,@Ri	26H+i	1	1
ADD	A,#data	24H	2	1
ADDC	A,Rn	38H+n	1	1
ADDC	A,direct	35H	2	1
ADDC	A,@Ri	36H+i	1	1
ADDC	A,#data	34H	2	1
SUBB	A,Rn	98H+n	1	1
SUBB	A,direct	95H	2	1
SUBB	A,@Ri	96H+i	1	1
SUBB	A,#data	94H	2	1
INC	Α	04H	1	1
INC	Rn	08H+n	1	1
INC	direct	05H	2	1
INC	@Ri	06H+i	1	1
INC	DPTR	АЗН	1	2
DEC	A	14H	1	1
DEC	Rn	18H+n	1	1
DEC	direct	15H	2	1
DEC	@Ri	16H+i	1	1
MUL	AB	A4H	1	4
DIV	AB	84H	1	4
DA	Α	D4H	1	1

Chú ý:

- MUL: AxB cho byte cao ở B, byte thấp ở A
- DIV: A/B cho thương số ở A và dư số ở B

Nhóm lệnh luận lý				
Mnemo	Mnemonic		#bytes	#MCs
ANL	A,Rn	58H+n	1	1
ANL	A,direct	55H	2	1
ANL	A,@Ri	56H+i	1	1
ANL	A,#data	54H	2	1
ANL	direct,A	52H	2 2 3	1
ANL	direct,#data	53H	3	2
ORL	A,Rn	48H+n	1	1
ORL	A,direct	45H	2	1
ORL	A,@Ri	46H+i	1	1
ORL	A,#data	44H	2	1
ORL	direct,A	42H	2	1
ORL	direct,#data	43H	3	2
XRL	A,Rn	68H+n	1	1
XRL	A,direct	65H	2	1
XRL	A,@Ri	66H+i	1	1
XRL	A,#data	64H	2	1
XRL	direct,A	62H	2	1
XRL	direct,#data	63H	3	2
CLR	Α	E4H	1	1
CPL	Α	F4H	1	1
RL	Α	23H	1	1
RLC	Α	33H	1	1
RR	Α	03H	1	1
RRC	Α	13H	1	1
SWAP	Α	C4H	1	1
Chú ý: Không có lệnh dịch bịt, muốn dịch bịt tạ				

Chú ý: Không có lệnh dịch bit, muốn dịch bit ta phải xóa MSB/LSB để tạo dịch bit. TD: dịch trái Ta dùng 2 lệnh RL A và CLR ACC.0

Nhóm lệnh rẽ nhánh				
Mnemonic		Opcode	#bytes	#MCs
ACALL	addr11	xem CT	2	2
LCALL	addr16	12H	3	2
RET		22H	1	2
RETI		32H	1	2
AJMP	addr11	xem CT	2	2
LJMP	addr16	02H	3	2
SJMP	rel	80H	2	2
JMP	@A+DPTR	73H	1	2
JZ	rel	60H	2	2
JNZ	rel	70H	2	2
CJNE	A,direct,rel	B5H	3	2
CJNE	A,#data,rel	B4H	3	2
CJNE	Rn,#data,rel	B8H+n	3	2
CJNE	@Ri,#data,rel	B6H+i	3	2
DJNZ	Rn,rel	D8H+n	2	2
DJNZ	direct,rel	D5H	3	2
NOP		00H	1	1

Nhóm lệnh xử lý biến Boole				
Mnemonic		Opcode	#bytes	#MCs
CLR	С	СЗН	1	1
CLR	bit	C2H	2	1
SETB	С	D3H	1	1
SETB	bit	D2H	2	1
CPL	С	взн	1	1
CPL	bit	B2H	2	1
ANL	C,bit	82H	2	2
ANL	C,/bit	ВОН	2	2
ORL	C,bit	72H	2	2
ORL	C,/bit	A0H	2	2
MOV	C,bit	A2H	2	1
MOV	bit,C	92H	2	2
JC	rel	40H	2	2
JNC	rel	50H	2	2
JB	bit,rel	20H		2
JNB	bit,rel	30H	3	2
JBC	bit,rel	10H	3	2

	Các lệnh ảnh hưởng cờ				
Lênh	Сờ		Lônh	Сờ	
LĖIII	С	OV	AC	Lệnh -	С
ADD	Χ	Χ	Χ	SETB C	1
ADDC	Χ	Χ	Χ	CLR C	0
SUBB	Χ	Χ	Χ	CPL C	Χ
MUL	0	Χ		ANL C,bit	Χ
DIV	0	Χ		ANL C,/bit	Χ
DA	Χ			ORL C,bit	Χ
RLC	Χ			ORL C,/bit	Χ
RRC	Χ			MOV C,bit	Χ
CJNE	Χ				

Chú thích (CT):

- 1. Qui ước trong bảng tóm tắt tập lệnh:
 - Opcode = mã lệnh

#bytes = số byte

#MCs = số chu kỳ máy (Machine Cycle)

 $1 MC = 12/F_{XTAL}$

data = hằng số dữ liệu 8 bit d16 = hằng số dữ liệu 16 bit

Rn = thanh ghi 8 bit (n=0,1,...,7)

Ri = thanh ghi 8 bit (i=0 hay 1)

direct = địa chỉ trực tiếp byte (00H-FFH)

bit = địa chỉ trực tiếp của bit

rel = độ dời (-128 đến +127)

addr11 = dia chi A10...A0

addr16 = dia chi A15... A0

2. Mã máy của định địa chỉ tuyệt đối:

AJMP addr11 có mã máy

A7 A6 A5 A4 A3 A2 A1 A0

A10 A9 A8 1 0 0 0 1

A7 A6 A5 A4 A3 A2 A1 A0

- 3. Các lệnh 3 byte có **addr16** gồm có byte 1 là mã lệnh, byte 2 là byte cao của addr16 và byte 3 là byte thấp của addr16.
- Các lệnh 3 byte CJNE có byte 1 là mã lệnh, byte 2 là *direct* (hoặc *data*), byte 3 là độ dời *rel*.

- 5. Các lệnh 3 byte MOV
 - MOV DPTR,#d16 có byte 1 là mã lệnh, byte 2 là byte cao của d16 và byte 3 là byte thấp của d16
 - MOV direct,#data có byte 1 là mã lệnh, byte 2 là direct và byte 3 là data
 - MOV direct, direct có byte 1 là mã lệnh, byte 2 là direct toán hạng nguồn và byte 3 là direct toán hạnh đích
- Các lệnh 3 byte ANL, ORL và XRL với toán hạng direct,#data có byte 1 là mã lệnh, byte 2 là *direct* và byte 3 là *data*
- 7. Lệnh DJNZ direct,rel có byte 1 là mã lệnh, byte 2 là *direct* và byte 3 là đô dời *rel*
- Các lênh JB, JNB và JBC với toán hạng bit,rel có byte 1 là mã lệnh, byte 2 là địa chỉ bit và byte 3 là độ dời rel

Một số chỉ thị của trình biên dịch hợp ngữ

Chỉ thị	Mô tả
ORG	Cho biết lệnh/data tiếp theo bắt
	đầu từ địa chỉ theo sau ORG
EQU	Định nghĩa giá trị ký hiệu
BIT	Định nghĩa địa chỉ trong vùng bit
DB	Định nghĩa 1 hay nhiều giá trị byte
DW	Định nghĩa ≥ 1 giá trị word
DS	Dành 1 số byte với số theo sau DS
END	Kết thúc chương trình
HIGH	Lấy byte cao của biểu thức
LOW	Lấy byte thấp của biểu thức

TD:

COUNT EQU 100

LED EQU P1.0 ; hoặc dùng LED BIT P1.0

LOAD_TO EQU 50000

ORG 0

SETB LED

MOV A,#COUNT

MOV TL0,#LOW(-LOAD_T0) MOV TH0,#HIGH(-LOAD_T0)

MOV DPTR,#TABLE

MOV DPTR,#STR

ORG 100H

TABLE: DB 12, 0F5H, 'B', 10110101B, 'A'-'a'

STR: DB "Hello the world!"