



ریزپردازنده

دانشکده کامپیوتر دانشگاه یزد

نیم سال دوم تحصیلی ۹۶-۹۷

ارائه دهنده : پریسا استواری



مجموعه دستورات 8051

انواع دستورات

- دستورات محاسباتی (Arithmetic)
- دستورات منطقی (Logic)
- دستورات انتقال اطلاعات (Data Transfer)
- دستورات بر روی بیت یا متغیرهای بولین (Boolean Variables)
- دستورات کنترل یا انشعاب برنامه (Program Branching)

دستورات روی بیت

- دستورات روی بیت‌ها یکی از ویژگی‌های میکروکنترلرهای خانواده 8051 است.
- این ویژگی در میکروکنترلرهای دیگر به ندرت یافت می‌شود.

- حافظه داخلی RAM دارای ۱۲۸ بیت آدرس‌پذیر (از آدرس 00H تا 7FH) است.

- ثبات‌های کاربرد خاص نیز دارای ۸۸ بیت آدرس‌پذیر دیگر هستند. (از آدرس 80H تا F7H)

- پورت‌ها به صورت بیتی دسترس پذیرند.
- یکی از ویژگی‌های 8051

7F	General-purpose RAM							
30								
2F	7F	7E	7D	7C	7B	7A	79	78
2E	77	76	75	74	73	72	71	70
2D	6F	6E	6D	6C	6B	6A	69	68
2C	67	66	65	64	63	62	61	60
2B	5F	5E	5D	5C	5B	5A	59	58
2A	57	56	55	54	53	52	51	50
29	4F	4E	4D	4C	4B	4A	49	48
28	47	46	45	44	43	42	41	40
27	3F	3E	3D	3C	3B	3A	39	38
26	37	36	35	34	33	32	31	30
25	2F	2E	2D	2C	2B	2A	29	28
24	27	26	25	24	23	22	21	20
23	1F	1E	1D	1C	1B	1A	19	18
22	17	16	15	14	13	12	11	10
21	0F	0E	0D	0C	0B	0A	09	08
20	07	06	05	04	03	02	01	00
1F	Bank3							
18								
17	Bank2							
10								
0F	Bank1							
08								
07	Default register Bank for R0-R7							
00								

<

مثال : بیت و بایت مرتبط با هر دستور را در RAM پیدا کنید.

- (a) SETB 42H, (b) CLR 67H, (c) CLR 0FH
(d) SETB 28H, (e) CLR 12, (f) SETB 05

Solution:

(a) D2 of RAM location 28H

(b) D7 of RAM location 2CH

(c) D7 of RAM location 21H

(d) D0 of RAM location 25H

(e) D4 of RAM location 21H

(f) D5 of RAM location 20H

	D7	D6	D5	D4	D3	D2	D1	D0
2F	7F	7E	7D	7C	7B	7A	79	78
2E	77	76	75	74	73	72	71	70
2D	6F	6E	6D	6C	6B	6A	69	68
2C	67	66	65	64	63	62	61	60
2B	5F	5E	5D	5C	5B	5A	59	58
2A	57	56	55	54	53	52	51	50
29	4F	4E	4D	4C	4B	4A	49	48
28	47	46	45	44	43	42	41	40
27	3F	3E	3D	3C	3B	3A	39	38
26	37	36	35	34	33	32	31	30
25	2F	2E	2D	2C	2B	2A	29	28
24	27	26	25	24	23	22	21	20
23	1F	1E	1D	1C	1B	1A	19	18
22	17	16	15	14	13	12	11	10
21	0F	0E	0D	0C	0B	0A	09	08
20	07	06	05	04	03	02	01	00

Bit-addressable RAM

- برای اینکه آدرس‌های 00H – 7FH برای آدرس‌دهی بایت‌ها و بیت‌ها اشتباه نشود:
 - ۱۲۸ بایت حافظه RAM از آدرس 00H تا 7FH
 - با دستورات بایتی به صورت آدرس‌دهی مستقیم یا غیرمستقیم استفاده می‌شوند.
- ۱۶ بایت حافظه RAM (از 20H تا 2FH) که آدرس بیتی 00H تا 7FH دارند
 - با دستورات بیتی و تنها با آدرس‌دهی مستقیم استفاده می‌شوند.

دستور SETB و CLR

SETB bit

- مقدار یک بیت را یک می کند. (SET Bit)

CLR bit

- مقدار یک بیت را صفر می کند. (Clear)
- **bit آدرس مستقیم یک بیت حافظه است.**
- مثال :

SETB C	مقدار پرچم نقلی (carry) را یک می کند.
CLR P1.3	مقدار بیت ۳ پورت یک را صفر می کند. دقت شود در هنگام اسمبل شدن آدرس 93H به جای P1.3 قرار می گیرد.
SETB ACC.4	مقدار بیت ۴ اکومولاتور را یک می کند.
CLR B.5	مقدار بیت ۵ ثبات B را صفر می کند.
SETB 0A4H	مقدار بیت در خانه A4H حافظه را یک می کند. در واقع مقدار بیت ۴ پورت ۲ را یک می کند.
CLR 87H	مقدار بیت در خانه 87H حافظه را صفر می کند. در واقع مقدار بیت ۷ پورت صفر را صفر می کند.

دستور CPL

CPL bit

- مقدار یک بیت را مکمل می کند. (Complement)
- bit آدرس مستقیم خانه حافظه است.

• مثال :

CPL C	مقدار پرچم نقلی (carry) را مکمل می کند.
CPL P1.3	مقدار بیت ۳ پورت یک را مکمل می کند. دقت شود در هنگام اسمبل شدن آدرس 93H به جای P1.3 قرار می گیرد.
CPL ACC.4	مقدار بیت ۴ اکومولاتور را مکمل می کند.
CPL B.5	مقدار بیت ۵ ثبات B را مکمل می کند.
CPL 0A4H	مقدار بیت در خانه A4H حافظه را مکمل می کند. در واقع مقدار بیت ۴ پورت ۲ را مکمل می کند.
CPL 87H	مقدار بیت در خانه 87H حافظه را مکمل می کند. در واقع مقدار بیت ۷ پورت صفر را مکمل می کند.

دستور SETB و CLR و CPL

تعداد سیکل ماشین	تعداد بایت	ترجمه دستور	دستور
1	1	11010011	SETB C
1	2	11010010 bbbbbbbb	SETB bit
1	1	11000011	CLR C
1	2	11000010 bbbbbbbb	CLR bit
1	1	10110011	CPL C
1	2	10110010 bbbbbbbb	CPL bit

• یادآوردی : دستور CLR و CPL برای A هم بود.

- CLR A
- CLR C
- CPL A
- CPL C

دستور MOV

MOV C, bit

MOV bit, C

- مقدار یک بیت در حافظه‌ی آدرس‌پذیر بی‌تی RAM یا فضای ثبات‌های کاربرد خاص (SRF) را می‌توان بین بیت پرچم نقلی جابجا کرد.
- دستور MOV همیشه با C انجام می‌شود.
- bit آدرس مستقیم خانه حافظه است.

- بیت نقلی (Carry) (C یا CY) به عنوان یک اکومولاتور یک بی‌تی است.

- مثال : دستورات زیر مقدار بیت خانه 18H حافظه را به پین صفر پورت P1 منتقل می‌کند.

MOV C, 18H

MOV P1.0, C

- مثال : دستور MOV C, 1 مقدار C را یک نمی‌کند بلکه محتوای خانه ی RAM 01H را در C میریزد.

دستور MOV

دستور	ترجمه دستور	تعداد بایت	تعداد سیکل ماشین
MOV C, bit	10100010 bbbbbbbb	2	1
MOV bit, C	10010010 bbbbbbbb	2	2

دستور ANL و ORL

- در این دستورات C به عنوان اکومولاتور تک بیتی است و یک عملوند در C است و نتایج در C ریخته می شود.

ANL C, bit

• $C = C \text{ AND } (\text{bit})$

ANL C, /bit

• $C = C \text{ AND NOT}(\text{bit})$

ORL C, bit

• $C = C \text{ OR } (\text{bit})$

ORL C, /bit

• $C = C \text{ OR NOT}(\text{bit})$

- bit آدرس مستقیم خانه حافظه است.

دستور ANL و ORL

• مثال :

ANL C, ACC.7	مقدار بیت ۷ اکومولاتور با مقدار C and می شود و حاصل در C قرار می گیرد. دقت شود اسمبلر آدرس ACC.7 یعنی E7H را به جای ACC.7 قرار می دهد.
ANL C, /OV	مقدار مکمل بیت پرچم سرریز (OV) با مقدار پرچم سرریز (C) and شده و حاصل در C قرار می گیرد.
ANL C, 0B6H	محتوای خانه حافظه B6H (پین ۶ پورت ۳) با C and شده و حاصل در C قرار می گیرد.
ORL C, /P1.4	مکمل محتوای پین ۴ پورت ۱ با C or شده و حاصل در C قرار می گیرد.
ORL C, /18H	مکمل محتوای خانه 18H حافظه با C or شده و حاصل در C قرار می گیرد.
ORL C, 7FH	محتوای خانه 7FH حافظه با C or شده و حاصل در C قرار می گیرد.

دستور ANL و ORL

- مثال : برنامه‌ای بنویسید که مقدار موجود در پین 0 و 1 پورت یک را AND کرده و حاصل را در پین 2 پورت یک بریزد.

```
SETB P1.0
SETB P1.1
LOOP : MOV C, P1.0
ANL C, P1.1
MOV P1.2, C
SJMP LOOP
```

- مثال : برنامه‌ای بنویسید که مقدار موجود در پین 0 پورت یک را به پین 0 پورت دو انتقال دهد.

```
MOV C, P1.0
MOV P2.0, C
```

دستور ANL و ORL

تعداد سیکل ماشین	تعداد بایت	ترجمه دستور	دستور
2	2	10000010 bbbbbbbb	ANL C, bit
2	2	10110000 bbbbbbbb	ANL C, /bit
2	2	01110010 bbbbbbbb	ORL C, bit
2	2	10100000 bbbbbbbb	ORL C, /bit

دستور JC

JC rel

- در صورتی که بیت پرچم نقلی (C) برابر با یک باشد به آدرس مقصد پرش می‌کند.
- Jump Carry

- آدرس مقصد به صورت یک عدد ۸ بیتی علامت‌دار است. یعنی پرش به ۱۲۷ بایت پایین‌تر یا به ۱۲۸ بایت بالاتر صورت می‌پذیرد. (یادآوری اسلاید ۳)
- rel می‌تواند آدرس مقصد یا یک برجسب باشد.
- دستور دو بایتی است.

- عملکرد :

$$PC \leftarrow PC + 2$$

اگر $C=1$ باشد پس

$$PC \leftarrow PC + \text{بایت دوم دستور}$$

دستور JNC

JNC rel

- در صورتی که بیت پرچم نقلی (C) برابر با صفر باشد به آدرس مقصد می‌پرد.
- Jump Not Carry

- آدرس مقصد به صورت یک عدد ۸ بیتی علامت‌دار است. یعنی پرش به ۱۲۷ بایت پایین‌تر یا به ۱۲۸ بایت بالاتر صورت می‌پذیرد.
- rel می‌تواند آدرس مقصد یا یک برجسب باشد.
- دستور دو بایتی است.

- عملکرد :

$$PC \leftarrow PC + 2$$

اگر $C=0$ باشد پس

$$PC \leftarrow PC + \text{دستور}$$

مثال

• دو سری دستورات زیر معادلند.

JC label1

CPL C

JC label2

JC label1

JNC label2

label1:

label2:

دستور JB

JB bit, rel

- در صورتی که محتوای بیت آدرس‌پذیر bit برابر با یک باشد، به آدرس مربوطه پرش می‌کند.
- Jump Bit
- bit آدرس مستقیم خانه حافظه است.

- آدرس مقصد همانند دستور JC به صورت نسبی محاسبه می‌گردد.
- دستور سه بایتی است.

- عملکرد :

$$PC \leftarrow PC + 3$$

- اگر محتوای خانه‌ی bit برابر با یک بود :

$$PC \leftarrow PC + \text{دستور}$$

دستور JNB

JNB bit, rel

- در صورتی که محتوای بیت آدرس پذیر bit برابر با صفر باشد، به آدرس مربوطه پرش می کند.
- Jump Not Bit
- bit آدرس مستقیم خانه حافظه است.

- آدرس مقصد همانند دستور JC به صورت نسبی محاسبه می گردد.
- دستور سه بایتی است.

- عملکرد :

$$PC \leftarrow PC + 3$$

- اگر محتوای خانه ی bit برابر با صفر بود :
- بایت سوم دستور $PC \leftarrow PC +$

دستور JBC

JBC bit, rel

- در صورتی که محتوای بیت آدرس‌پذیر bit برابر با یک باشد، به آدرس مربوطه پرش می‌کند و محتوای bit را صفر می‌کند.

• Jump Bit Clear

- bit آدرس مستقیم خانه حافظه است.

- آدرس مقصد همانند دستور JC به صورت نسبی محاسبه می‌گردد.
- دستور سه بایتی است.

• عملکرد :

• $PC \leftarrow PC + 3$

- اگر محتوای خانه‌ی bit برابر با یک بود :

• بایت سوم دستور $PC \leftarrow PC +$ و $bit \leftarrow 0$

دستورات تست بیت

دستور	ترجمه دستور	تعداد بایت	تعداد سیکل ماشین
JC rel	01000000 eeeeeeee	2	2
JNC rel	01010000 eeeeeeee	2	2
JB bit, rel	00100000 bbbbbbbb eeeeeeee	3	2
JNB bit, rel	00100000 bbbbbbbb eeeeeeee	3	2
JBC bit, rel	00010000 bbbbbbbb eeeeeeee	3	2

مثال

- فرض کنید تمامی ثبات‌ها و خانه‌های حافظه حاوی مقدار صفر باشند. کد زیر را تفسیر کنید.

SETB 7FH

SETB P1.2

label2 : JB P1.2, label1

JNB ACC.0, 5

label1 : CLR P1.2

JB 7FH, label2

JNC 0F1H

کد ترجمه دستورات :

D2	7F	D2	92	20	92	03	30	E0	05	C2	92	20	7F	F5	50	F1
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

مثال

- فرض کنید تمامی ثبات‌ها و خانه‌های حافظه حاوی مقدار صفر باشند. کد زیر را تفسیر کنید.

SETB 7FH ←

SETB P1.2

label2 : JB P1.2, label1

JNB ACC.0, 5

label1 : CLR P1.2

JB 7FH, label2

JNC 0F1H

این دستور بیت 7FH حافظه RAM را برابر با یک می‌کند.

از آنجایی که آدرس‌دهی مستقیم است آدرس بیت 7FH در بایت دوم دستور قرار گرفته است.

D2	7F	D2	92	20	92	03	30	E0	05	C2	92	20	7F	F5	50	F1
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

General-purpose RAM

Bank 3
Bank 2
Bank 1
Bank 0

Default register Bank for R0-R7

- # SETB 7FH

SETB P1.2

label2 : JB P1.2, label1

JNB ACC.0, 5

label1 : CLR P1.2

JB 7FH, label2

JNC 0F1H

این دستور پین ۲ پورت یک را برابر با یک می‌کند. آدرس این پین برابر با 92H است.

از آنجایی که آدرس دهی مستقیم است آدرس 92H در بایت دوم دستور قرار گرفته است.

D2	7F	D2	92	20	92	03	30	E0	05	C2	92	20	7F	F5	50	F1							
														83	not bit address			DPH					
														82	not bit address			DPL					
														81	not bit address			SP					
														80	87	86	85	84	83	82	81	80	P0

- فرض کنید تمامی ثبات‌ها و خانه های حافظه حاوی مقدار صفر باشند. کد زیر را

SETB 7FH

SETB P1.2

اگر مقدار P1.2 برابر با یک باشد
 به label1 می پرد. مقدار P1.2 ← JB P1.2, label1
 برابر با یک است.

JNB ACC.0, 5

label1 : CLR P1.2

JB 7FH, label2

JNC 0F1H

از آنجایی که آدرس دهی مستقیم است آدرس 92H در بایت دوم دستور قرار گرفته است.

آدرس نسبی پرش در بایت سوم است. یعنی مقدار PC به اضافه 3 می‌گردد.



FF	تفسیر کنید								
F0	F7	F6	F5	F4	F3	F2	F1	F0	B
E0	E7	E6	E5	E4	E3	E2	E1	E0	ACC
D0	D7	D6	D5	D4	D3	D2	-	D0	PSW
B8	-	-	-	BC	BB	BA	B9	B8	IP
B0	B7	B6	B5	B4	B3	B2	B1	B0	P3
A8	AF	AE	-	-	AB	AA	A9	A8	IE
A0	A7	A6	A5	A4	A3	A2	A1	A0	P2
99	not bit address								SBUS
98	9F	9E	9D	9C	9B	9A	99	98	SCO
90	97	96	95	94	93	92	91	90	P1
8D	not bit address								TH1
8C	not bit address								TH0
8B	not bit address								TL1
8A	not bit address								TL0
89	not bit address								TMO
88	8F	8E	8D	8C	8B	8A	89	88	TCON
87	not bit address								PCON
83	F5	50						F1	DPH
82	not bit address								DPL
81	not bit address								SP
80	87	86	85	84	83	82	81	80	P0

- فرض کنید تمامی ثبات‌ها و خانه‌های حافظه حاوی مقدار صفر باشند. کد زیر را

SETB 7FH

SETB P1.2

label2 : JB P1.2, label1

JNB ACC.0, 5

label1 : CLR P1.2 ↵

JB 7FH, label2

JNC 0F1H

این دستور پین ۲ پورت یک را برابر با صفر می‌کند. آدرس این پین برابر با 92H است.

از آنجایی که آدرس دهی مستقیم است آدرس 92H در بایت دوم دستور قرار گرفته است.

D2	7F	D2	92	20	92	03	30	E0	05	C2	92	20	7F
----	----	----	----	----	----	----	----	----	----	----	----	----	----

FF	تفسیر کنید.								
F0	F7	F6	F5	F4	F3	F2	F1	F0	B
E0	E7	E6	E5	E4	E3	E2	E1	E0	ACC
D0	D7	D6	D5	D4	D3	D2	-	D0	PSW
B8	-	-	-	BC	BB	BA	B9	B8	IP
B0	B7	B6	B5	B4	B3	B2	B1	B0	P3
A8	AF	AE	-	-	AB	AA	A9	A8	IE
A0	A7	A6	A5	A4	A3	A2	A1	A0	P2
99	not bit address								SBUS
98	9F	9E	9D	9C	9B	9A	99	98	SCO
90	97	96	95	94	93	92	91	90	P1
8D	not bit address								TH1
8C	not bit address								TH0
8B	not bit address								TL1
8A	not bit address								TL0
89	not bit address								TMO
88	8F	8E	8D	8C	8B	8A	89	88	TCO
87	not bit address								PCO
83	F5	50		F1		not bit address			DPH
82	not bit address								DPL
81	not bit address								SP
80	87	86	85	84	83	82	81	80	P0

مثال

- فرض کنید تمامی ثبات‌ها و خانه‌های حافظه حاوی مقدار صفر باشند. کد زیر را تفسیر کنید.

SETB 7FH

SETB P1.2

label2 : JB P1.2, label1

JNB ACC.0, 5

label1 : CLR P1.2

JB 7FH, label2

JNC 0F1H

اگر مقدار بیت 7FH برابر با یک باشد به label2 می‌پرد. محتوای بیت 7FH برابر با یک است.

از آنجایی که آدرس‌دهی مستقیم است آدرس 7FH در بایت دوم دستور قرار گرفته است.

آدرس نسبی پرش در بایت سوم است. یعنی مقدار PC به اضافه عدد علامت‌دار F5H می‌گردد. یعنی منهای 11 می‌گردد.

کد ترجمه دستورات :

D2	7F	D2	92	20	92	03	30	E0	05	C2	92	20	7F	F5	50	F1
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

PC جدید

+F5 = -11

PC قدیم

- # SETB 7FH

SETB P1.2

مقدار P1.2 صفر است پس پرش انجام نمی شود.

JNB ACC.0, 5

label1 : CLR P1.2

JB 7FH, label2

JNC 0F1H

کد ترجمه دستورات :

D2	7F	D2	92	20	92	03	30	E0	05	C2	92	20	7F	F5	50	F1
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----

↑
PC

مثال

• فرض کنید تمامی ثبات‌ها و خانه‌های حافظه حاوی مقدار صفر باشند. کد زیر را تفسیر

SETB 7FH

SETB P1.2

label2 : JB P1.2, label1

JNB ACC.0, 5

label1 : CLR P1.2

JB 7FH, label2

JNC 0F1H

اگر مقدار پین ACC.0 که دارای
آدرس E0 است برابر با صفر باشد
به آدرس جدید می‌پرد. آدرس
جدید برابر است با مقدار فعلی
PC + 5

کنید.

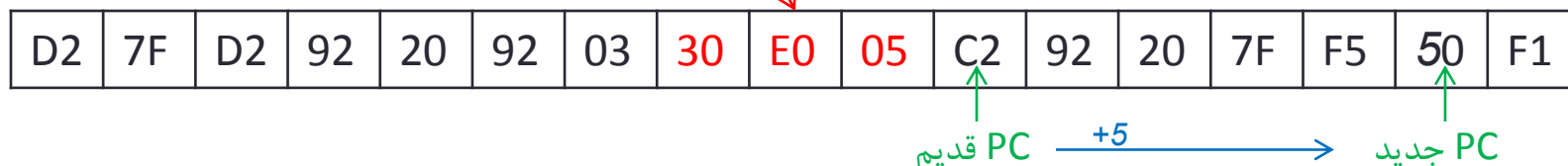
FF									
F0	F7	F6	F5	F4	F3	F2	F1	F0	B
E0	E7	E6	E5	E4	E3	E2	E1	E0	ACC
D0	D7	D6	D5	D4	D3	D2	--	D0	PSW
B8	--	--	--	BC	BB	BA	B9	B8	IP
R0	B7	B6	B5	B4	B3	B2	B1	B0	D3

به جای برچسب از عدد استفاده شده است.

آدرس پین صفر ACC که E0H است در بایت
دوم دستور مشخص شده است.

مقدار PC + 5 می‌شود تا آدرس مقصد پرش
مشخص شود.

کد ترجمه دستورات :



مثال

- فرض کنید تمامی ثبات‌ها و خانه‌های حافظه حاوی مقدار صفر باشند. کد زیر را تفسیر کنید.

SETB 7FH

SETB P1.2

label2 : JB P1.2, label1

JNB ACC.0, 5

label1 : CLR P1.2

JB 7FH, label2

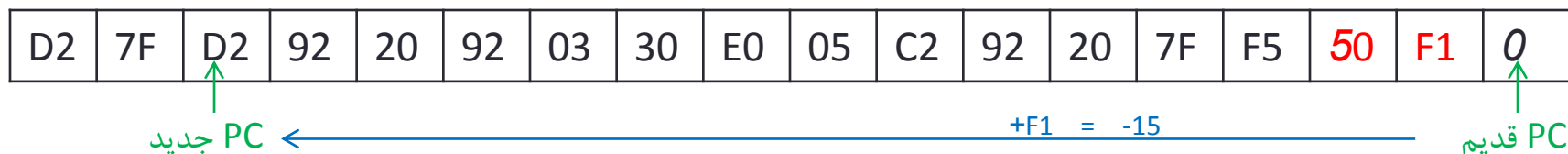
JNC 0F1H

اگر مقدار C برابر با صفر باشد
پرش می‌کند. آدرس مقصد برابر
است با $PC + F1$ که F1 یک
عدد علامتدار و منفی است.

مقدار $PC + F1$ می‌شود تا آدرس مقصد پرش
مشخص شود. F1 عدد علامتدار -15 است.

به جای برچسب از عدد استفاده شده است.

کد ترجمه دستورات :



مثال

- فرض کنید تمامی ثبات‌ها و خانه‌های حافظه حاوی مقدار صفر باشند. کد زیر را تفسیر کنید.

SETB 7FH

SETB P1.2

دستور بعدی که اجرا می‌شود این دستور است.

label2 : JB P1.2, label1

JNB ACC.0, 5

label1 : CLR P1.2

JB 7FH, label2

JNC 0F1H

کد ترجمه دستورات :

D2	7F	D2	92	20	92	03	30	E0	05	C2	92	20	7F	F5	50	F1	0
----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	----	---

مثال

- برنامه‌ای بنویسید که چک کند مقدار بایت 37H حافظه RAM عددی زوج است یا نه. اگر زوج است آن را به پورت ۲ بفرستد، اگر فرد است آن را زوج کرده و به پورت ۲ بفرستد.

```

MOV      A,37H      ;load RAM 37H into ACC
JNB      ACC.0,YES   ;if D0 of ACC 0? If so jump
INC      A           ;it's odd, make it even
YES:     MOV      P2,A      ;send it to P2

```

- مقدار بیت‌های P1.2 و P1.3 باید قبل از تغییر کردنشان ذخیره شود. برنامه‌ای بنویسید که مقدار این دو بیت را در خانه‌های 06 و 07 حافظه ذخیره کند.

```

CLR      06          ;clear bit addr. 06
CLR      07          ;clear bit addr. 07
JNB      P1.2,OVER   ;check P1.2, if 0 then jump
SETB     06          ;if P1.2=1,set bit 06 to 1
OVER:    JNB      P1.3,NEXT ;check P1.3, if 0 then jump
SETB     07          ;if P1.3=1,set bit 07 to 1
NEXT:    ...

```

مثال

- یک سویچ به پین P1.7 متصل است. برنامه‌ای بنویسید که حالت این سویچ را چک کرده و تصمیمات زیر را بگیرد.

- (a) If SW = 0, send "0" to P2
(b) If SW = 1, send "1" to P2

Solution:

```

SW      EQU  P1.7
MYDATA  EQU  P2
HERE:    MOV    C, SW
          JC     OVER
          MOV    MYDATA, #'0'
          SJMP   HERE
OVER:    MOV    MYDATA, #'1'
          SJMP   HERE
          END

```

SW	EQU	97H
MYDATA	EQU	0A0H

مثال

- برنامه‌ای بنویسید که مقدار بیت های 40H و 50H را XOR کرده و حاصل را در بیت 60H بریزد.

40H	50H	60H
0	0	0
0	1	1
1	0	1
1	1	0

JB 40H, YEK

SEFR: MOV C, 50H
MOV 60H, C
SJMP EXIT

YEK: MOV C, 50H
CPL C
MOV 60H,C

EXIT:

مرجع سریع دستورات انتقال اطلاعات

CLR C
CLR bit
SETB C
SETB bit
CPL C
CPL bit
ANL C, bit
ANL C, /bit

ORL C, bit
ORL C, /bit
MOV C, bit
MOV bit, C
JC rel
JNC rel
JB bit, rel
JNB bit, rel
JBC bit, rel

راهنما

Rn	آدرس‌دهی ثبات R0 تا R7
direct	آدرس ۸ بیتی حافظه داده (RAM) داخلی (00H-FFH)
@Ri	آدرس‌دهی غیرمستقیم با استفاده از ثبات‌های R0 یا R1
source	بایت منبع که هر یک از ثبات‌های Rn، آدرس مستقیم (direct)، یا آدرس غیرمستقیم @Ri می‌تواند باشد.
dest	بایت مقصد که هر یک از ثبات‌های Rn، آدرس مستقیم (direct)، یا آدرس غیرمستقیم @Ri می‌تواند باشد.
#data	عدد ۸ بیتی در دستور
#data16	عدد ۱۶ بیتی در دستور
bit	آدرس ۸ بیتی یک بیت اطلاعات
rel	آدرس نسبی یا آفست ۸ بیتی علامت‌دار
addr11	آدرس ۱۱ بیتی برای صفحه 2k بایتی حافظه
addr16	