

ตัวอย่างโปรแกรม J31E21.ASM

Assembly

Machine - Code

```

1  ; FILENAME      J31E21.ASM
2  ; DESCRIPTION  SEGMENT SHIFT
3  ; HARDWARE    JAZZ31 V2.0
4  ; ASSEMBLER   SXA51
5  ; START-DATE  17/11/37
6  ; COMPANY     SILA RESEARCH CO.,LTD.
7
8  ORG      8100H
9
10 CONA     EQU    0F800H      ;SEGMENT (0-7)
11 CONB     EQU    0F801H      ;DIGIT (0-3)
12 SOUND    EQU    00A2H      ;SOUND SYSTEM CALL
13
14 ; ***** MAIN *****
15
16 EXAM:     MOV     R7,#8      ;8 DIGIT LOOP
17           MOV     R6,#0      ;START DIGIT
18
19 EXAM1:     MOV     DPTR,#CONB ;SET DIGIT
20           MOVX    A,@DPTR    ;READ,MODIFY,WRITE
21           ANL     A,#11111000B ;AND BIT 0-2
22           ORL     A,R6       ;DIGIT BY R6
23           MOVX    @DPTR,A
24
25           MOV     R5,#8      ;8 SEGMENT LOOP
26           MOV     R4,#1      ;START SEGMENT
27           MOV     DPTR,#CONA ;WRITE SEGMENT
28           MOVX    A,R4
29           MOVX    @DPTR,A
30           RL      A           ;SHIFT LEFT
31           MOV     R4,A
32           CALL    EXAMD       ;BACK LOOP
33           DJNZ    R5,EXAM2    ;SOUND (FREQUENCY BY R6)
34
35           MOV     A,R6
36           ORL     A,#20H
37           MOV     R2,A
38           MOV     R3,#08H
39           CALL    SOUND
40           INC     R6
41           DJNZ    R7,EXAM1    ;NEXT DIGIT
42           SJMP    EXAM        ;BACK DIGIT LOOP
43                               ;LOOP FOREVER
44
45 ; ***** EXAMD SUB *****
46 ; DELAY SUBROUTINE
47
48 EXAMD:     MOV     R3,#80H
49 EXAMD1:    MOV     R2,#0
50           DJNZ    R2,$
51           DJNZ    R3,EXAMD1
52           RET
53
54 END

```

ตัวอย่างโปรแกรม J31E22.ASM

```

1  ; FILENAME      J31E22.ASM
2  ; DESCRIPTION  10 SECOND SOFT TIMER
3  ; HARDWARE    JAZZ31 V2.0
4  ; ASSEMBLER   SXA51
5  ; START-DATE  12/04/38
6  ; COMPANY     SILA RESEARCH CO.,LTD.
7
8  ORG      8100H
9
10 CONA     EQU    0F800H      ;SEGMENT
11 CONB     EQU    0F801H      ;DIGIT & LED
12 DTSEC    EQU    0042H      ;DELAY SYSTEM CALL
13 UBEEP     EQU    00A5H      ;USER BEEP SYSTEM CALL
14 ENDVEC    EQU    0036H      ;END VECTOR
15
16 ; ***** MAIN *****
17
18 EXAM:     MOV     R7,#10      ;10 SECOND LOOP
19           MOV     DPTR,#EXAMTB
20
21 EXAM1:     CLR     A           ;LOAD FROM TABLE
22           MOVX    A,@A+DPTR   ;MEMORY TO R2
23           MOV     R2,A
24
25           PUSH    DPH
26           PUSH    DPL
27           MOV     DPTR,#CONB ;SET DIGIT=0
28           MOVX    A,@DPTR
29           ANL     A,#11111000B ;B0-B2=0
30           MOVX    @DPTR,A
31           MOV     DPTR,#CONA ;SET SEGMENT
32           MOV     A,R2
33           MOVX    @DPTR,A
34           MOV     DPTR,#CONB ;FLASH LED
35           MOVX    A,@DPTR
36           SETB    ACC.4       ;LED ON (1)
37           MOVX    @DPTR,A
38           MOV     R2,#5       ;DELAY 0.5 SECOND
39           CALL    DTSEC
40           CLR     ACC.4       ;LED OFF (0)
41           MOVX    @DPTR,A
42           MOV     R2,#5       ;DELAY 0.5 SECOND
43           CALL    DTSEC
44           POP     DPL
45           POP     DPH
46           ;--/
47
48           INC     DPTR        ;NEXT TABLE
49           DJNZ    R7,EXAM1    ;NEXT LOOP
50           CALL    UBEEP
51           JMP     ENDVEC      ;TO MONITOR
52
53 EXAMTB:    DB      6FH,7FH,07H,7DH ;9876
54           DB      6DH,66H,4FH,5BH ;5432
55           DB      06H,3FH          ;10
56
57 END

```


Table 11. Instruction Opcodes in Hexadecimal Order

Hex Code	Number of Bytes	Mnemonic	Operands	Hex Code	Number of Bytes	Mnemonic	Operands
00	1	NOP		33	1	RLC	A, #data
01	2	AJMP	code addr	34	2	ADDC	A, #data
02	3	LJMP	code addr	35	2	ADDC	A, data addr
03	1	RR	A	36	1	ADDC	A, @R0
04	1	INC	A	37	1	ADDC	A, @R1
05	2	INC	data addr	38	1	ADDC	A, R0
06	1	INC	@R0	39	1	ADDC	A, R1
07	1	INC	@R1	3A	1	ADDC	A, R2
08	1	INC	R0	3B	1	ADDC	A, R3
09	1	INC	R1	3C	1	ADDC	A, R4
0A	1	INC	R2	3D	1	ADDC	A, R5
0B	1	INC	R3	3E	1	ADDC	A, R6
0C	1	INC	R4	3F	1	ADDC	A, R7
0D	1	INC	R5	40	2	JC	code addr
0E	1	INC	R6	41	2	AJMP	code addr
0F	1	INC	R7	42	2	ORL	data addr, A
10	3	JBC	bit addr, code addr	43	3	ORL	data addr, #data
11	2	ACALL	code addr	44	2	ORL	A, #data
12	3	LCALL	code addr	45	2	ORL	A, data addr
13	1	RRC	A	46	1	ORL	A, @R0
14	1	DEC	A	47	1	ORL	A, @R1
15	2	DEC	data addr	48	1	ORL	A, R0
16	1	DEC	@R0	49	1	ORL	A, R1
17	1	DEC	@R1	4A	1	ORL	A, R2
18	1	DEC	R0	4B	1	ORL	A, R3
19	1	DEC	R1	4C	1	ORL	A, R4
1A	1	DEC	R2	4D	1	ORL	A, R5
1B	1	DEC	R3	4E	1	ORL	A, R6
1C	1	DEC	R4	4F	1	ORL	A, R7
1D	1	DEC	R5	50	2	JNC	code addr
1E	1	DEC	R6	51	2	ACALL	code addr
1F	1	DEC	R7	52	2	ANL	data addr, A
20	3	JB	bit addr, code addr	53	3	ANL	data addr, #data
21	2	AJMP	code addr	54	2	ANL	A, #data
22	1	RET		55	2	ANL	A, data addr
23	1	RL	A	56	1	ANL	A, @R0
24	2	ADD	A, #data	57	1	ANL	A, @R1
25	2	ADD	A, data addr	58	1	ANL	A, R0
26	1	ADD	A, @R0	59	1	ANL	A, R1
27	1	ADD	A, @R1	5A	1	ANL	A, R2
28	1	ADD	A, R0	5B	1	ANL	A, R3
29	1	ADD	A, R1	5C	1	ANL	A, R4
2A	1	ADD	A, R2	5D	1	ANL	A, R5
2B	1	ADD	A, R3	5E	1	ANL	A, R6
2C	1	ADD	A, R4	5F	1	ANL	A, R7
2D	1	ADD	A, R5	60	2	JZ	code addr
2E	1	ADD	A, R6	61	2	AJMP	code addr
2F	1	ADD	A, R7	62	2	XRL	data addr, A
30	3	JNB	bit addr, code addr	63	3	XRL	data addr, #data
31	2	ACALL	code addr	64	2	XRL	A, #data
32	1	RETI		65	2	XRL	A, data addr

Table 11. Instruction Opcodes in Hexadecimal Order (Continued)

Hex Code	Number of Bytes	Mnemonic	Operands	Hex Code	Number of Bytes	Mnemonic	Operands
66	1	XRL	A, @R0	99	1	SUBB	A, R1
67	1	XRL	A, @R1	9A	1	SUBB	A, R2
68	1	XRL	A, R0	9B	1	SUBB	A, R3
69	1	XRL	A, R1	9C	1	SUBB	A, R4
6A	1	XRL	A, R2	9D	1	SUBB	A, R5
6B	1	XRL	A, R3	9E	1	SUBB	A, R6
6C	1	XRL	A, R4	9F	1	SUBB	A, R7
6D	1	XRL	A, R5	A0	2	ORL	C, bit addr
6E	1	XRL	A, R6	A1	2	AJMP	code addr
6F	1	XRL	A, R7	A2	2	MOV	C, bit addr
70	2	JNZ	code addr	A3	1	INC	DPTR
71	2	ACALL	code addr	A4	1	MUL	AB
72	2	ORL	C, bit addr	A5		reserved	
73	1	JMP	@A + DPTR	A6	2	MOV	@R0, data addr
74	2	MOV	A, #data	A7	2	MOV	@R1, data addr
75	3	MOV	data addr, #data	A8	2	MOV	R0, data addr
76	2	MOV	@R0, #data	A9	2	MOV	R1, data addr
77	2	MOV	@R1, #data	AA	2	MOV	R2, data addr
78	2	MOV	R0, #data	AB	2	MOV	R3, data addr
79	2	MOV	R1, #data	AC	2	MOV	R4, data addr
7A	2	MOV	R2, #data	AD	2	MOV	R5, data addr
7B	2	MOV	R3, #data	AE	2	MOV	R6, data addr
7C	2	MOV	R4, #data	AF	2	MOV	R7, data addr
7D	2	MOV	R5, #data	B0	2	ANL	C, bit addr
7E	2	MOV	R6, #data	B1	2	ACALL	code addr
7F	2	MOV	R7, #data	B2	2	CPL	bit addr
80	2	SJMP	code addr	B3	1	CPL	C
81	2	AJMP	code addr	B4	3	CJNE	A, #data, code addr
82	2	ANL	C, bit addr	B5	3	CJNE	A, data addr, code addr
83	1	MOVC	A, @A + PC	B6	3	CJNE	@R0, #data, code addr
84	1	DIV	AB	B7	3	CJNE	@R1, #data, code addr
85	3	MOV	data addr, data addr	B8	3	CJNE	R0, #data, code addr
86	2	MOV	data addr, @R0	B9	3	CJNE	R1, #data, code addr
87	2	MOV	data addr, @R1	BA	3	CJNE	R2, #data, code addr
88	2	MOV	data addr, R0	BB	3	CJNE	R3, #data, code addr
89	2	MOV	data addr, R1	BC	3	CJNE	R4, #data, code addr
8A	2	MOV	data addr, R2	BD	3	CJNE	R5, #data, code addr
8B	2	MOV	data addr, R3	BE	3	CJNE	R6, #data, code addr
8C	2	MOV	data addr, R4	BF	3	CJNE	R7, #data, code addr
8D	2	MOV	data addr, R5	C0	2	PUSH	data addr
8E	2	MOV	data addr, R6	C1	2	AJMP	code addr
8F	2	MOV	data addr, R7	C2	2	CLR	bit addr
90	3	MOV	DPTR, #data	C3	1	CLR	C
91	2	ACALL	code addr	C4	1	SWAP	A
92	2	MOV	bit addr, C	C5	2	XCH	A, data addr
93	1	MOVC	A, @A + DPTR	C6	1	XCH	A, @R0
94	2	SUBB	A, #data	C7	1	XCH	A, @R1
95	2	SUBB	A, data addr	C8	1	XCH	A, R0
96	1	SUBB	A, @R0	C9	1	XCH	A, R1
97	1	SUBB	A, @R1	CA	1	XCH	A, R2
98	1	SUBB	A, R0	CB	1	XCH	A, R3

ตารางสรุปของ 8255 (MODE-0)

Table 11. Instruction Opcodes in Hexadecimal Order (Continued)

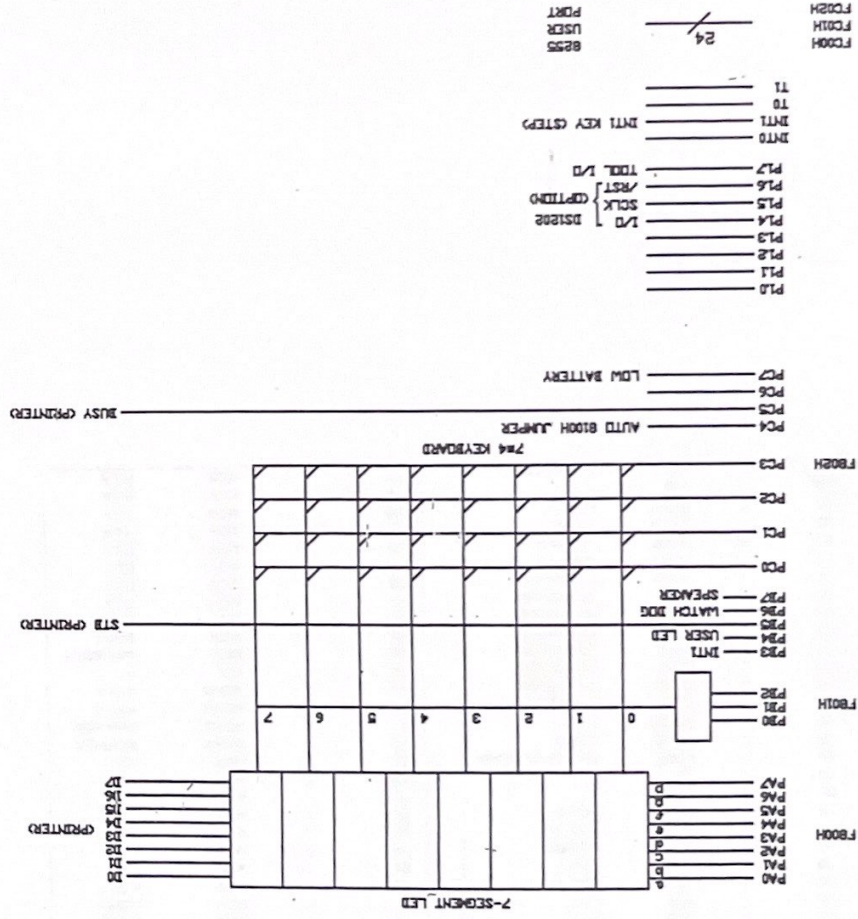
Hex Code	Number of Bytes	Mnemonic	Operands
CC	1	XCH	A,R4
CD	1	XCH	A,R5
CE	1	XCH	A,R6
CF	1	XCH	A,R7
D0	2	POP	data addr
D1	2	ACALL	code addr
D2	2	SETB	bit addr
D3	1	SETB	C
D4	1	DA	A
D5	3	DJNZ	data addr,code addr
D6	1	XCHD	A,@R0
D7	1	XCHD	A,@R1
D8	2	DJNZ	R0,code addr
D9	2	DJNZ	R1,code addr
DA	2	DJNZ	R2,code addr
DB	2	DJNZ	R3,code addr
DC	2	DJNZ	R4,code addr
DD	2	DJNZ	R5,code addr
DE	2	DJNZ	R6,code addr
DF	2	DJNZ	R7,code addr
E0	1	MOVX	A,@DPTR
E1	2	AJMP	code addr
E2	1	MOVX	A,@R0
E3	1	MOVX	A,@R1
E4	1	CLR	A
E5	2	MOV	A,data addr

Hex Code	Number of Bytes	Mnemonic	Operands
E6	1	MOV	A,@R0
E7	1	MOV	A,@R1
E8	1	MOV	A,R0
E9	1	MOV	A,R1
EA	1	MOV	A,R2
EB	1	MOV	A,R3
EC	1	MOV	A,R4
ED	1	MOV	A,R5
EE	1	MOV	A,R6
EF	1	MOV	A,R7
F0	1	MOVX	@DPTR,A
F1	2	ACALL	code addr
F2	1	MOVX	@R0,A
F3	1	MOVX	@R1,A
F4	1	CPL	A
F5	2	MOV	data addr,A
F6	1	MOV	@R0,A
F7	1	MOV	@R1,A
F8	1	MOV	R0,A
F9	1	MOV	R1,A
FA	1	MOV	R2,A
FB	1	MOV	R3,A
FC	1	MOV	R4,A
FD	1	MOV	R5,A
FE	1	MOV	R6,A
FF	1	MOV	R7,A

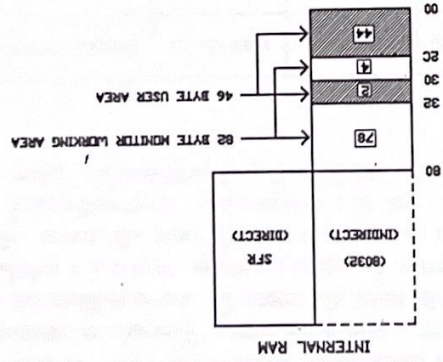
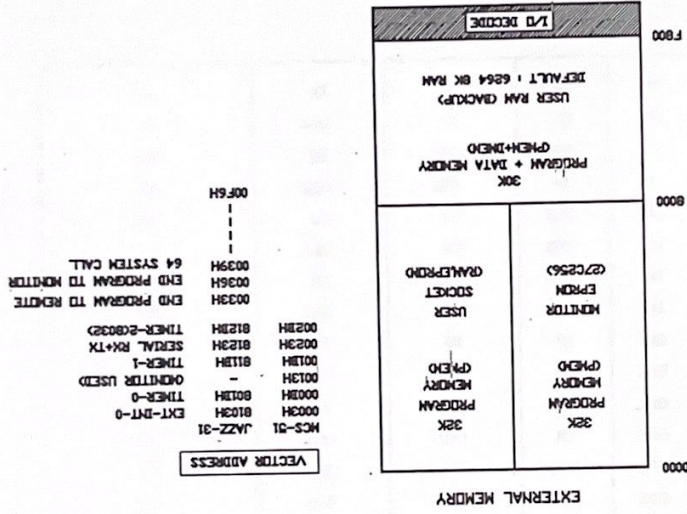
8000 74 40 MOV A,#40H ; นำค่า 40H ใส่ในรีจิสเตอร์ A
 9002 F8 MOV R0,A ; นำค่าในรีจิสเตอร์ A เก็บใน R0 (ดังนั้น R0 = A = 40H)
 2003 F9 MOV R1,A ; นำค่าใน A เก็บใน R1 (ดังนั้น R1 = A = R0 = 40H)
 9004 FA MOV R2,A ; นำค่าใน A เก็บใน R2 (ดังนั้น A = R0 = R1 = R2 = 40H)
 9005 7B 45 MOV R3,#45H ; นำค่า 45 เก็บใน R3 (ดังนั้น R3 = 45H)
 9007 EB MOV A,R3 ; นำค่าใน R3 เก็บใน A (ดังนั้น A = R3 = 45H)

ชิพเบอร์ 8255 เป็นชิพประเภท INPUT/OUTPUT ที่นิยมใช้กันเป็นอย่างมาก ตัว 8255 จะมีจำนวนพอร์ทให้ใช้งานอยู่ 3 พอร์ท คือ PORT-A PORT-B และ PORT-C ซึ่งหมายถึง 24 INPUT/OUTPUT BIT นั่นเอง การกำหนดให้แต่ละพอร์ทเป็น INPUT หรือ OUTPUT นั้น สามารถทำได้ด้วยการส่ง CONTROL CODE ไปก่อน โดยส่งไปที่พอร์ท CONTROL ซึ่งจะอยู่ที่หมายเลขท้ายสุด จากจำนวน 4 พอร์ทที่มองเห็นได้จากตัวไมโครคอนโทรลเลอร์ พอร์ท A และ B จะต้องกำหนดให้เป็น INPUT หรือ OUTPUT เหมือนกันทั้ง 8 BIT ส่วนพอร์ท C สามารถแยกเป็น 4 BIT ได้ 2 ส่วนอิสระจากกัน การกำหนดที่กล่าวมานี้ เป็นการทำงานใน MODE-0 ของชิพ 8255 เท่านั้น แต่ก็เป็นที่นิยมนำมาใช้ในการใช้งานมากที่สุด โดยตารางของ CONTROL CODE จะมีดังต่อไปนี้

PORT A	PORT B	PORT C0-C3	PORT C4-C7	CONTROL CODE (HEX)
OUT	OUT	OUT	OUT	80
OUT	OUT	OUT	IN	88
OUT	OUT	IN	OUT	81
OUT	OUT	IN	IN	89
OUT	IN	OUT	OUT	82
OUT	IN	OUT	IN	8A
OUT	IN	IN	OUT	83
OUT	IN	IN	IN	8B
IN	OUT	OUT	OUT	90
IN	OUT	OUT	IN	98
IN	OUT	IN	OUT	91
IN	OUT	IN	IN	99
IN	IN	OUT	OUT	92
IN	IN	OUT	IN	9A
IN	IN	IN	OUT	93
IN	IN	IN	IN	9B



ATTENTION I/O PORT



ATTENTION MEMORY MAP

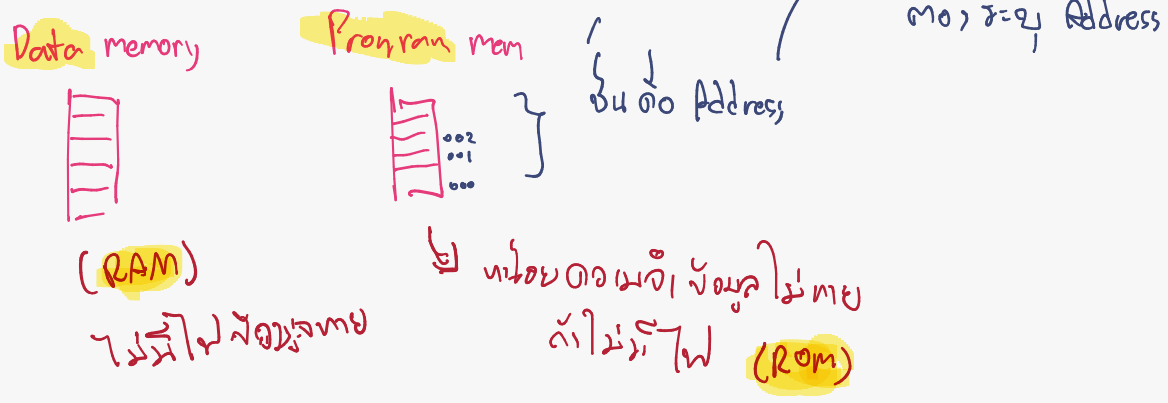
MOV R7 #8 7F

MOV R6, #0 7E

MOV DPTR

✖ 6030 MCS-51

- ↳ memory 2 ส่วนคือ
- ↳ 1. memory สำหรับข้อมูล
- ↳ 2. memory สำหรับโปรแกรม



Pmem			
10h - 20h	FF	10	
20h - 30h	FF	11	
30h - 40h	FF	12	
40h - 50h	20	13	
50h - 60h	6C	14	
60h - 70h	03	15	
70h - 80h	02	16	
80h - 90h	05	17	
90h - 100h	6C	18	
100h - 110h	22	19	
110h - 120h	FF	20	