



	<u>16</u>	<u>10</u>	<u>2</u>
	<u>Hex</u>	<u>Dec</u>	<u>Bin</u>
B0	00	0	0000
B1	r1	1	0001
B2	r2	2	0010
B3	r3	3	0011
B4	r4	4	0100
B5	r5	5	0101
B6	r6	6	0110
B7	r7	7	0111
B8	r0	8	1000
B9	r1	9	1001
B10	r2	10	1010
B11	r3	11	1011
B12	r4	12	1100
B13	r5	13	1101
B14	r6	14	1110
B15	r7	15	1111
B16	r0	16	1111
B17	r1	17	1
B18	r2	18	
B19	r3	19	
B20	r4	20	
B21	r5	21	
B22	r6	22	
B23	r7	23	
B24	r0	24	
B25	r1	25	
B26	r2	26	
B27	r3	27	
B28	r4	28	
B29	r5	29	
B30	r6	30	
B31	r7	31	
B32	32	0010 0000	
B33	33	0010 1100	
B34	34	0010 1101	
B35	35	0010 1110	
B36	36	0010 1111	
B37	37	0011 0000	
B38	38	0011 0001	
B39	39	0011 0010	
B40	40	0011 0011	
B41	41	0011 0100	
B42	42	0011 0101	
B43	43	0011 0110	
B44	44	0011 0111	
B45	45	0011 1000	
B46	46	0011 1001	
B47	47	0011 1010	
B48	48	0011 1011	
B49	49	0011 1100	
B50	50	0011 1101	
B51	51	0011 1110	
B52	52	0011 1111	
B53	53	0100 0000	
B54	54	0100 0001	
B55	55	0100 0010	
B56	56	0100 0011	
B57	57	0100 0100	
B58	58	0100 0101	
B59	59	0100 0110	
B60	60	0100 0111	
B61	61	0100 1000	
B62	62	0100 1001	
B63	63	0100 1010	
B64	64	0100 1011	
B65	65	0100 1100	
B66	66	0100 1101	
B67	67	0100 1110	
B68	68	0100 1111	
B69	69	0101 0000	
B70	70	0101 0001	
B71	71	0101 0010	
B72	72	0101 0011	
B73	73	0101 0100	
B74	74	0101 0101	
B75	75	0101 0110	
B76	76	0101 0111	
B77	77	0101 1000	
B78	78	0101 1001	
B79	79	0101 1010	
B80	80	0101 1011	
B81	81	0101 1100	
B82	82	0101 1101	
B83	83	0101 1110	
B84	84	0101 1111	
B85	85	0110 0000	
B86	86	0110 0001	
B87	87	0110 0010	
B88	88	0110 0011	
B89	89	0110 0100	
B90	90	0110 0101	
B91	91	0110 0110	
B92	92	0110 0111	
B93	93	0110 1000	
B94	94	0110 1001	
B95	95	0110 1010	
B96	96	0110 1011	
B97	97	0110 1100	
B98	98	0110 1101	
B99	99	0110 1110	
B100	100	0110 1111	

8 = bit  $\Rightarrow$  1 byte

16 bit = 2 byte = 1 word  
4 bit = nibble

## Memory

d: 0x00

rom address 11100111

c: 0x00

program address 11100111  
1k byte

0010 0000  
0010 1100  
0010 1101  
0010 1110  
0011 0000  
0011 0001  
0011 0010  
0011 0011  
0011 0100  
0011 0101  
0011 0110  
0011 0111  
0100 0000  
0100 0001  
0100 0010  
0100 0011  
0100 0100  
0100 0101  
0100 0110  
0100 0111  
0110 0000  
0110 0001  
0110 0010  
0110 0011  
0110 0100  
0110 0101  
0110 0110  
0110 0111  
0110 1000  
0110 1001  
0110 1010  
0110 1011  
0110 1100  
0110 1101  
0110 1110  
0110 1111

Bit-Byte

setb C

mov 21h, C

clr C

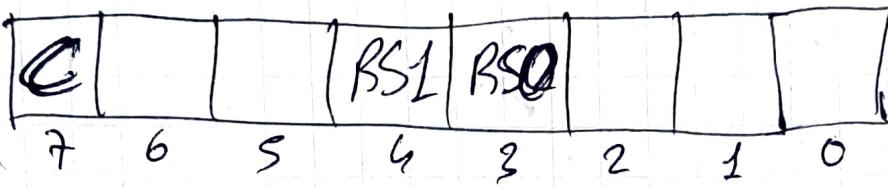
mov 21h, C

setb 21h

31	28	29	30	31
30	27	22	23	24
29	14	15	16	17
28	7	8	9	10
27	1	2	3	4



PSW



$$RS0 = PSW.3$$

$$RS1 = PSW.4$$

RS1	RS0	Aktif Sokakracı Deposu
0	0	Bank 0
0	1	Bank 1
1	0	Bank 2
1	1	Bank 3

TEMMUZ PAZAR JULY SUNDAY

13

clr → clear komutu bitset bizeyi;

mov c, P0.1  
mov P0.2, c ) Bitsel toplama carry bit'i ile

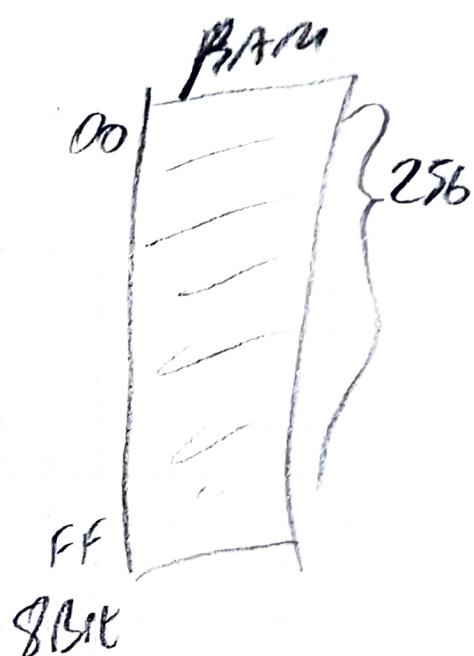
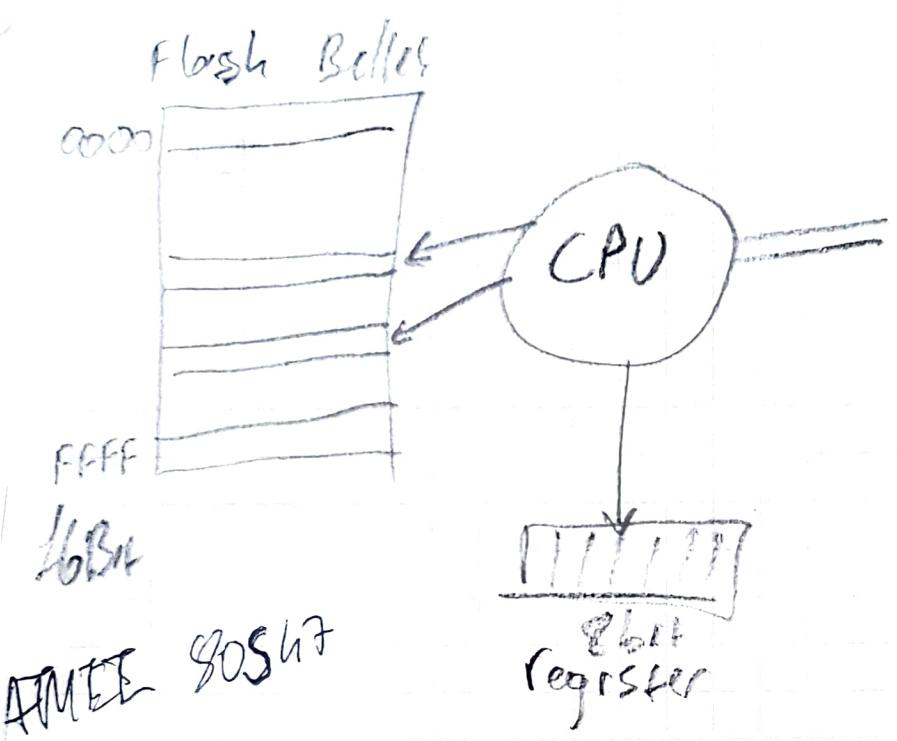
mov R0, R1 → Yanlış

mov (R0, R1) → Dogrular

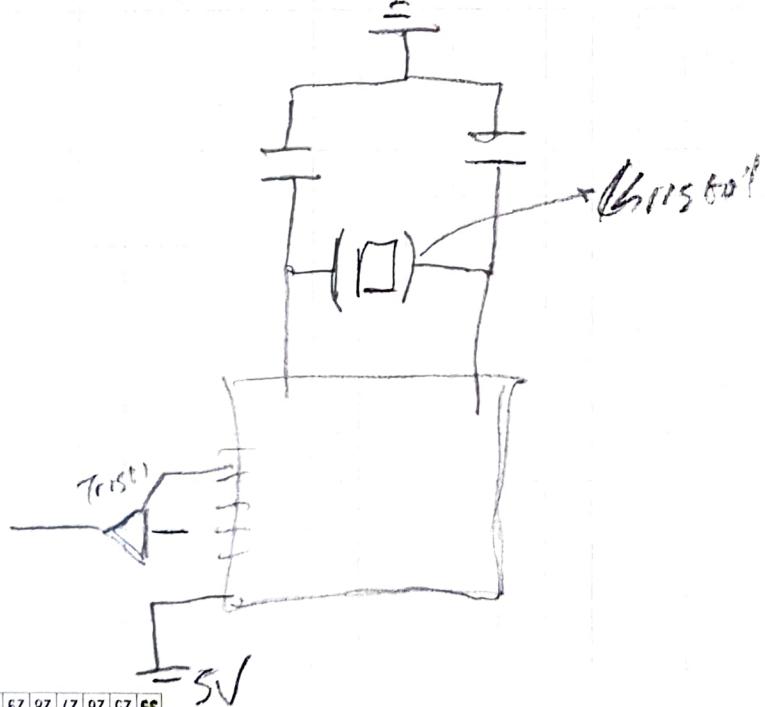
H	P	S	G	P	C	C1	P2
31	28	29	30	31			
30	21	22	23	24	25	26	27
29	14	15	16	17	18	19	20
28	7	8	9	10	11	12	13
27	1	2	3	4	5	6	

TEMMUZ CUMARTESİ JULY SATURDAY

12



255	1	1	1	1	1	1	1	1
127	0	1	1	1	1	1	1	1
128	1	0	0	0	0	0	0	0



35	25	26	27	28	29	30	31
34	18	19	20	21	22	23	24
33	11	12	13	14	15	16	17
32	4	5	6	7	8	9	10
31							
30							
29							
28							
27							
26							
25							



PLC → bit bit okular you 128x8 true bit okular  
yolcu her bit 1,0 you ON, OFF onbor  
bit.

2. barko ges  
Bark okular  
note NO, #5d

mov 10h #5d

Dalekh Ashes leone  
15/12

PSW 

C	I		R1	R0		
---	---	--	----	----	--	--

0 0 → BANK 0

0 1

1 0  
1 1

(Microglomer en boite)  
BANK 0

BANK 0

mov r7, #0f4

Setb rs0 ⇔ setb PSW.3

clr R51 ⇔ clr PSW.4

mov r7, #12h

(mov PSW, #00001000b) 1sten neyer yonken

AUGUSTOS | PAZARTESİ AUGUST MONDAY

18

35	25	26	27	28	29	30	31
34	18	19	20	21	22	23	24
33	11	12	13	14	15	16	17
32	4	5	6	7	8	9	10
31				1	2	3	

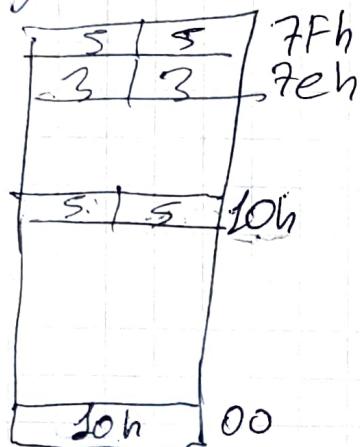
AUGUSTOS 2014

HAZİRAN 2014

H	P	S	C	P	C	Ct	Pz
23	2	3	4	5	6	7	10
24	9	10	11	12	13	14	15
25	16	17	18	19	20	21	22
26	23	24	25	26	27	28	29
27	30						

(Cine)

→ Dörtlü Adreslene



(r0 ve r2 : b)

27

HAZİRAN JUNIE

ÇETEVEN

FRIDAY

VAR

MOV 10h, #55h  
MOV 7Fh, #55h

MOV 7Fh, @r0

r0'in degerini adres yap  
o adreslerde verilecektir.

MOV 7eh, #33h

MOV @r0, 7eh  
#33

→ TSM RAM'ı sıfırlamak;

MOV r0, #7fh

X:

MOV @r0, #00h

dfnz r0, x

r0 dekrement → decrement  
azalt jump not zero  
azalt after 0 dekrement looks

bruto → okunusuz



# 28

HAZIRAN | CUMARTESİ  
JUNE | SATURDAY

DPTR 'ye ontanak ran S.53 'deki kod

HAZIRAN 2014									
H	P	S	Ç	P	C	Ct	Pz		
23	2	3	4	5	6	7	8		
24	9	10	11	12	13	14	15		
25	16	17	18	19	20	21	22		
26	23	24	25	26	27	28	29		
27	30								

#include <adc841.h>

org 0000h

bşlsr!

```

mov    dptr, #sayi
mov    a, #10
movc   a,@d+dptr
sjmp   bşlsr
org    020h
sayi: db 5,8,11,14,15
end

```

# 29

HAZIRAN | PAZAR  
JUNE | SUNDAY

Tel Foto



# 05

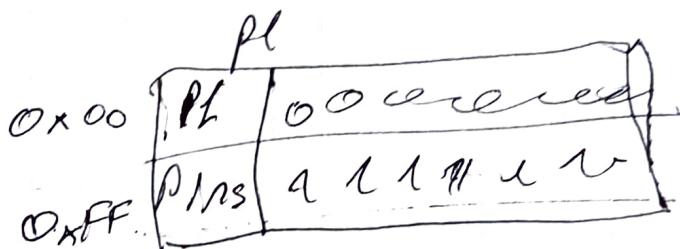
AGUSTOS | SALI  
AUGUST | TUESDAY

ADUC  
AD<sub>uc</sub>  
Analog Digital      Mikro Controller



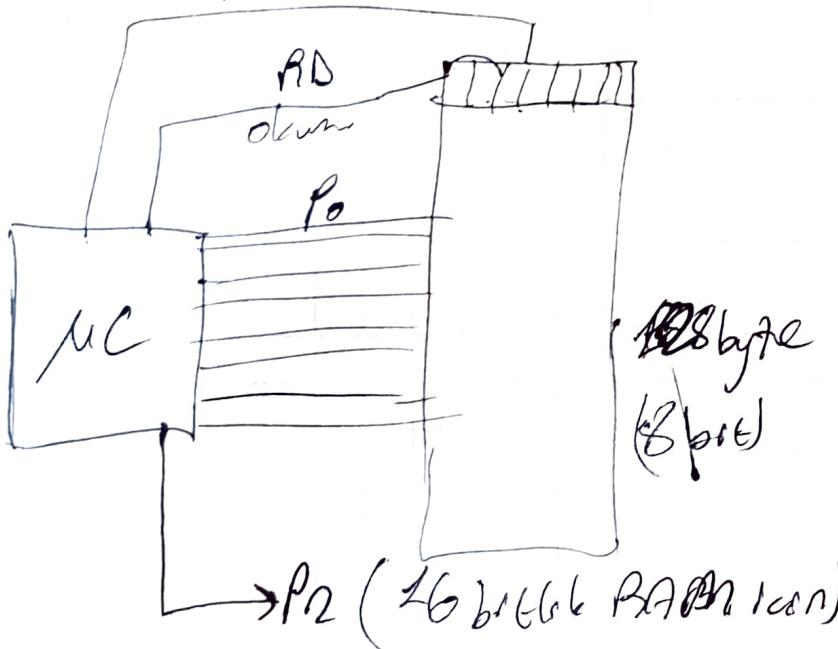
1101 0011

Mov P1, #00H



Sunday 8/05/14  
guru

RR



ADC S/I → P1 besinnlike wirks obrot kullenbotz

Mov P0, #00h  
K: jnb P1.0, X  
sjmp K  
X: setb P0.0

Mov P0, #00h  
K: jnb P1.0, X  
sjmp K  
X: jnb P1.0, X  
setb P0.0  
sjmb K



UNIVERSIDAD NACIONAL DE LA PLATA

AGÜSTOS 2014

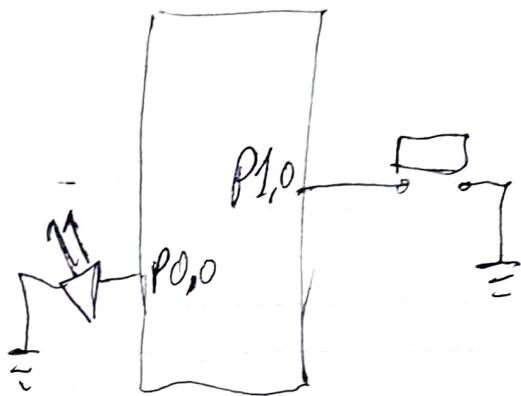
H	P	S	C	P	Ct	Pz
31				1	2	3
32	4	5	6	7	8	9 10
33	11	12	13	14	15	16 17
34	18	19	20	21	22	23 24
35	25	26	27	28	29	30 31

02  
13 → X

04

AGÜSTOS | PAZARTESİ  
AUGUST | MONDAY

Mov P0, \$00h  
L1 jnb P1.0, b  
M: jnb P1.0, m  
setb P0.0  
sjmp b



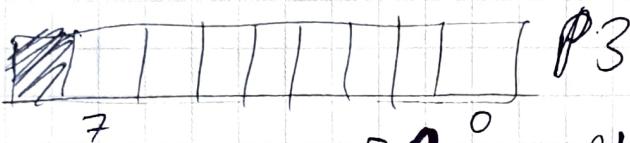
HAZİRAN						2012		
H	P	S	Ç	P	C	Ct	Pz	
22						1	2	3
23	4	5	6	7	8	9	10	
24	11	12	13	14	15	16	17	
25	18	19	20	21	22	23	24	
26	25	26	27	28	29	30		

$$\begin{array}{r}
 100 \\
 10 \\
 \hline
 1
 \end{array}
 \quad
 \begin{array}{r}
 7)3 \\
 4)2 \\
 2)1 \\
 1)1
 \end{array}$$

**HAZİRAN**  
*JUNE*

**PERŞEMBE**  
*THURSDAY*

14



~~JB~~<sup>0</sup> P<sub>10</sub>, X<sub>1</sub>  
X<sub>1</sub>: MOV RO, #10000000B

X:

Mov p3, r0

DENZ RO, X

Kl:

1	000	0000	—	28	)7
0	100	0000	—	22	)6
0	010	0000	—	16	)5
0	001	0000	—	11	)4
0	000	1000	—	7	)3
0	000	0100	—	6	)2
0	000	0010	—	2	)1
0	000	0001	—	1	)1

80 H	128 D
60 H	64 D
20 H	32 D

ANL 10, #00001111B



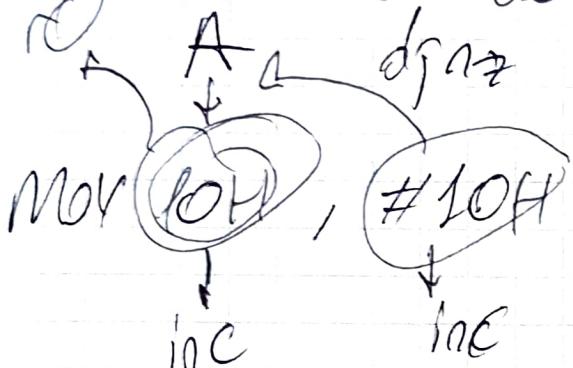


MOV R0, #16d

X2: mov R1, #1

X1: dec A

DJNZ R0, X1



R0 10H

dec R0

A #10H → @R0

dec A



CLR C  
MOV R0, #10H

#2

X1:

MOV 11H, R0  
MOV @R0, 11H  
ADDC A, @R0  
MOV 12H, A  
DJNZ R0, X1

ANL 0000 1111

MOV R0, #5D

#3

X: MOV B, R0  
MUL AB  
MOV 10H, A  
DJNZ R0, X

10

PER SEMBE  
THURSDAY  
MAYIS  
MAY

H	P	S	C	P	C1	C2	P2
18	1	2	3	4	5	6	
19	7	8	9	10	11	12	13
20	14	15	16	17	18	19	20
21	21	22	23	24	25	26	27
22	28	29	30	31			

MAYIS 2012

#5



OAG COOH

## BASLA:

MOV R0, #12H  
MOV R1, #34H  
  
MOV 10H, R0  
ANL 10H, #00001111B ; R0 LOW NIBBLE  
MOV 11H, R0  
ANL 11H, #11110000B ; R0 HIGH NIBBLE  
MOV 13H, R1  
ANL 13H, #00001111B ; R1 LOW NIBBLE  
MOV 14H, R1  
ANL 14H, #11110000B ; R1 HIGH NIBBLE  
  
MOV A, 14H  
ADD A, 10H  
Mov P0, A  
  
MOV A, 13H  
ADD A, 11H  
Mov PL, A  
  
LJMP BASLA  
END

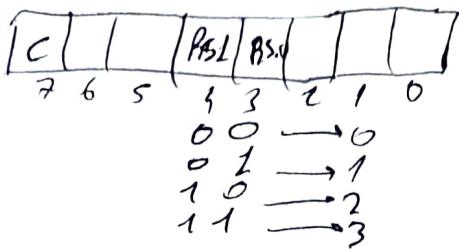
FRIDAY  
CUIMA  
MAY  
MAYIS

II

# 18

AĞUSTOS | PAZARTESİ  
AUGUST | MONDAY

AĞUSTOS 2014							
H	P	S	Ç	P	C	Ct	Pz
31				1	2	3	
32	4	5	6	7	8	9	10
33	11	12	13	14	15	16	17
34	18	19	20	21	22	23	24
35	25	26	27	28	29	30	31



AND  $\Leftrightarrow$  ANL

$$\begin{array}{ccc} X & \wedge & Y = Z \\ 0 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 0 & 0 \\ 1 & 1 & 1 \end{array}$$

OR  $\Leftrightarrow$  ORL

$$\begin{array}{ccc} X & \vee & Y = Z \\ 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{array}$$

XOR  $\Leftrightarrow$  XRL

$$\begin{array}{ccc} X \oplus Y = Z \\ 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1 & 0 & 1 \\ 1 & 1 & 0 \end{array}$$

CPL A  $\rightarrow$  A'ının her bitinin tersi

A'ının yerine OXEO dir.

ADD & ADDC  $\rightarrow$  Toplama

SUBB  $\rightarrow$  Çıkarma

INC A  $\rightarrow$  A+1

DEC A  $\rightarrow$  A-1

MUL AB  $\rightarrow$  A = BxA

DA  $\rightarrow$  Ondalik sayı

SWAP  $\rightarrow$  nibble deşifre

ANL

ORL

XRL

CPL  $\rightarrow$  Bit veya Byte tersleme



H	P	S	C	P	C	Cl	Pz
23	2	3	4	5	6	7	1
24	9	10	11	12	13	14	15
25	16	17	18	19	20	21	22
26	23	24	25	26	27	28	29
27	30						

X = 1236h  $\rightarrow$  Gilbert  
Y = 12EFh

# 30

HAZIRAN | PAZARTESİ  
JUNE | MONDAY

Mov 00h, #12h  
Mov 01h, #36h  
Mov 02h, #12h  
Mov 03h, #0eth

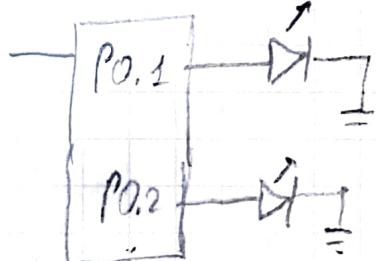
18d  
52d  
18d  
239d

; toplama işlem  
clr c

Mov a, 01h  
Add a, 03h  
Mov 10h, a

$\rightarrow$  a'yı toplama işleminden sonra  
sayfa

Mov a, 00h  
Add a, 02h  
Mov 11h, a



carry taken after  
jc x-ekleme  
setb p0.1  
x-ekleme; setb p0.2

baska:

Mov a, #16d  
Subb a, #5d

Jc x  
Clr p0.1  
Sjmp y

x:  
Clr p0.2

y:  
Sjmp basla  
end

if/else/ for/while

Jc x

Jnc x  $\rightarrow$  choice'e

Jz x

Jnz x

choice k, l, m

$\hookrightarrow$  k, l, m ye esit degilse  
m'ye otur

Djnz



# 01

TEMMUZ | SALI  
JULY | TUESDAY

X Data Pointer

TEMMUZ 2014											
H	P	S	C	P	C	Ct	P				
27		1	2	3	4	5	6				
28	7	8	9	10	11	12	13				
29	14	15	16	17	18	19	20				
30	21	22	23	24	25	26	27				
31	28	29	30	31							

mov DPTB, # X

mov r0, # 65

X: db 1, 2, 4, 8, 16, 32, 64  
           |   |   |   |   |   |  
           0   1   2   3   4   5   6

Mov a, # 04h

move a, @ a + DPTB

a = a & r0      - & +  
       V   /    } 1bit   V 1bit

Bank3 → a ⇒ 00011000  
       orl a, PSW

Bank0 → a ⇒ 11100111  
       and a, PSW

clr c  
   mov a, # 250d       $250 + 7 + 8 = ?$   
   add a, # 7d  
   addc a, # 8d

Databy mov r0, # 02h  
 datby mov 02h, # 12h  
 datby mov a, # 23h  
 datby add a, @ r0





		P1 - S	?	P2.1	P2.2	gak sa	gak sa
Sonu		negatif				yahsa	
"		positif				yahsa	

b0s0r:	mov	P2, #00h
	mov	a, P1
	clr	c
	subb	a, #5d
	sjmp	Y
X:	setb	P2.1
	sjmp	b0s0r
Y:	jc	X
	setb	P2.2
	sjmp	b0s0r

clr	c
subb	a, #5d
jc	x
setb	p2.2
sjmp	bosb
x: setb	p2.1
sjmp	bosbr

JNC son P2. lern yerleri degisildi

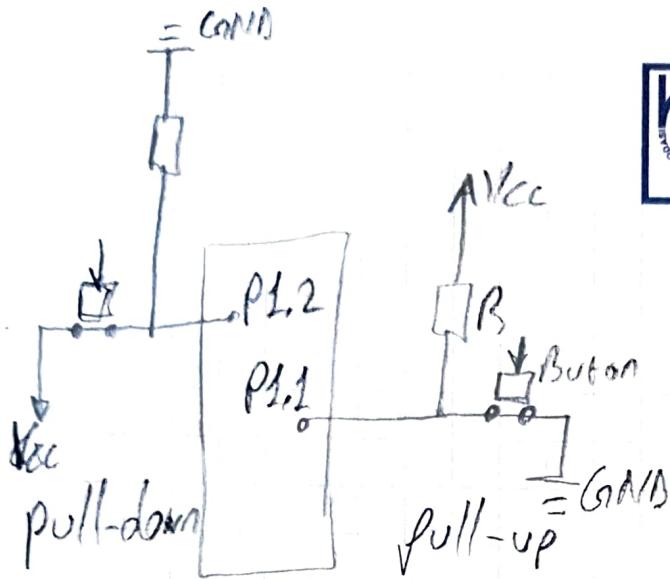
```

        mov P3, #00h
        mov P0, a
        CJNE a, #15d, X
        mov P3, #0FFh
Y:    sjmp Y
X:    mov P3, #00H
        sjmp bas6

```

31	28	29	30	31
30	21	22	23	24
29	14	15	16	17
28	7	8	9	10
27	1	2	3	4

TEMUZU 2014



64K → 1Byte

MOV C, psw.3

MOV 21H, C

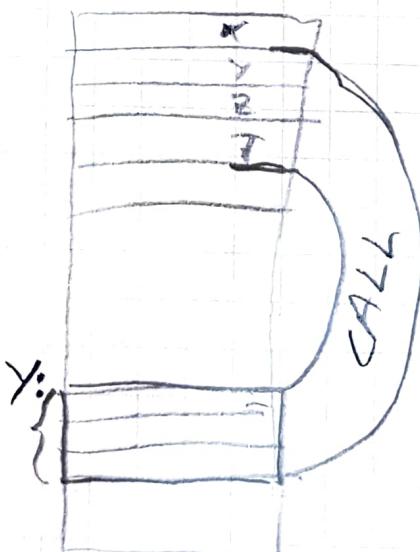
bit addressee  
bit in byte  
byte defit

MOV 20H, 00H

MOV C, 00H

byte to bit adrs

Nop → no operation (gesture softbar)  $\frac{1}{12\text{MHz}}$



JULY MONDAY  
TEMUZ PAZARTESİ

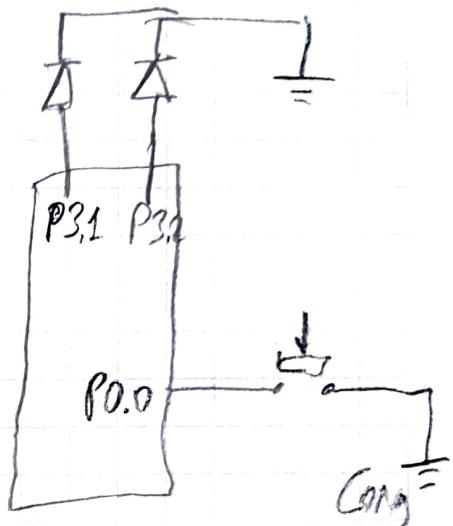
28

H	P	S	G	P	C	C1	P2
31	28	29	30	31			
30	21	22	23	24	25	26	27
29	14	15	16	17	18	19	20
28	7	8	9	10	11	12	13
27	1	2	3	4	5	6	



PSW.3 = 1 ise X'e döller  
Aksa Taktende devam et

$P0.0 = 0$  ise  $Ram = 0$   
 0       $P3.1 = 1$   
 1       $P3.2 = 0$   
 Aksa Takt.       $P3.2 = 1$   
 1       $P3.1 = 0$



Ye portu düşgündə təkrarla

ORG 0000H

```

basa: mov P0, #0FFh
      jb P0.0, x
      mov P0, #7fh
k: mov @r0, #00h
   djnz r0, k
   setb P3.1
   clr P3.2
   sjmp basa
x: setb P3.2
   clr P3.1
   sjmp basa

```

H	P	S	G	P	C	G	P
27	1	2	3	4	5	6	
28	7	8	9	10	11	12	13
29	14	15	16	17	18	19	20
30	21	22	23	24	25	26	27
31	28	29	30	31			



$$\frac{1}{12 \text{ MHz}} = 8,3 \cdot 10^{-8} = 0,83 \text{ nS}$$

15 kes ortlama

Mov d, #00h

MOV r0, #16d

X: nov PO, a

fnc d

DJNZ RO, X

; PPTB 1000 olmasini isteyiriz.

mov B2, #2d

mov DPTR, #00h

Mov R1, #4d

Yi

MOV R0

18

TEMUZ GARSAMBAY WEDNESDAY JULY

03

31	28	29	30	31
30	21	22	23	24
29	14	15	16	17
28	7	8	9	10
27	1	2	3	4
H	P	S	G	P
	C	C	C	Pz

TEMMUZ 2014

AGUSTOS 2012

H	P	S	C	P	C	Ct	Pz
31			1	2	3	4	5
32	6	7	8	9	10	11	12
33	13	14	15	16	17	18	19
34	20	21	22	23	24	25	26
35	27	28	29	30	31		

AGUSTOS

AUGUST

PERŞEMBE

THURSDAY

09

00000000

böslə:

cpl	p1.0
dcall	birsonje
sjmp	böslə

1\$1 → 11,058,200

kədə dələq

birsonje:

mov	R1, #15d
X2: MOV	R0, #255d
X1: DJNZ	R0, X1
DJNZ	R1, X2

MOV R3, #K3

X3: MOV R1, #K2

X2: MOV R0, #K1

X1: DJNZ R0, X1 (3K1+2) Kə Vəz

3K1+2 K3 Kəz

DJNZ R1, X2

DJNZ R3, X3

$$(3K_1 + 2)K_2 + 3K_2 + 2$$

$$3K_1K_2K_3 + 5K_2K_3 + 5K_3 + 2 = 11,058,200$$

250 60

Hərəkət tənzəsi eməkzə K3 de

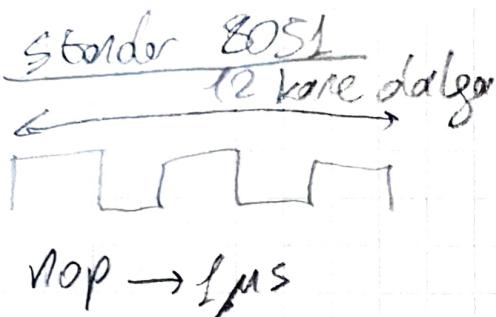
$$\begin{aligned} K_1 &= 2^{43} \\ K_2 &= 60 \\ K_3 &= 60 \\ K_4 &= 60 \end{aligned}$$



# 10

AĞUSTOS  
AUGUST  
CUMA  
FRIDAY

AGUSTOS 2012						
H	P	S	C	P	C	Pz
31			1	2	3	4 5
32	6	7	8	9 10	11 12	
33	13	14	15 16	17	18 19	
34	20	21	22 23	24	25 26	
35	27	28	29	30	31	



$$\left( 12 \cdot \frac{1}{12 \text{ mikrosaniye}} \right) = 1 \mu\text{s}$$

1nop çalışma süreci moderni gevrimi

ADUC 861



enfisiyel yapılandırır

11,0592 MHz

$$\frac{1}{11,0592 \times 10^{-6}} = 90,4 \text{nS}$$

X: Jb P1.1, X

Y: Snb P1.1, Y

setb P0.1

D Zamanlayıcı 1sn (Donge)  
clr P0.1

MOV R1, #K2

X2: mov R0, #K1

X1:

DJNZ R0, X1

DJNZ R1, X2

, 3K2

$$\frac{1\text{sn}}{90,4\text{nS}} = 11062 \mu\text{s}$$

Aşağıda 1nop konusunda

$$(3K_1 + 2)K_2 + 3K_2 + 2$$

$$(3K_1 + K_1)K_2 + 3K_2 + 2$$



# 08

AĞUSTOS

AUGUST

ÇARŞAMBA

WEDNESDAY

MOD 1

AĞUSTOS		2012						
H	P	S	C	P	C	Ct	Pz	
31				1	2	3	4	5
32	6	7	8	9	10	11	12	
33	13	14	15	16	17	18	19	
34	20	21	22	23	24	25	26	
35	27	28	29	30	31			

Ayar:

```

    mov    TMOD, #0000 0001
    mov    TL0, #8BH
    mov    TH0, #OFFH
    setb   TR0
  
```

X: jnb TF0, X

clr TF0

setb led2

mov TL0, #9BH

mov TH0, #OFFH

sjmp X

Otomatik  
yöntemde  
dezenfekte

MOD 2

1. yarılışta sayer 1'den  
T1 → P3.5 1 → 0  
T0 → P3.6 1 → 0

(255'e kavur)  
(155'den 100)  
Sayer

Ayar:

```

    mov    TMOD, #0000 0020
  
```

mov TL0, #155d

mov TH0, #155d

setb TR0

X: jnb TF0, X

clr TF0

sjmp X



AGUSTOS 2012							
H	P	S	Ç	P	C	Cl	Pz
31			1	2	3	4	5
32	6	7	8	9	10	11	12
33	13	14	15	16	17	18	19
34	20	21	22	23	24	25	26
35	27	28	29	30	31		

PRESSCALUAR → 9 AM

25.08.2012

Yazılım 35

AGUSTOS

AUGUST

SALI

TUESDAY

07

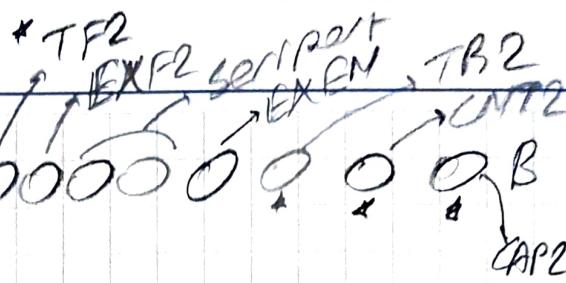
Ayar:

```

    mov    tmod, #00000001b ; mod1, timer0
; mov    TL0, #0FS
; mov    TH0, #0FFH
    mov    dptr, #65525d
    mov    TL0, DPL
    mov    TH0, DPH
    setb   TR0
X: jnb    TF0, X
    clr
    mov    TL0, DPL
    mov    TH0, DPH
    SJMP   X
    end
  
```



# Timer 2



MOV T2CON, #00000000

11058200 → 1s

110582 → 1ms

110582 → 10ms

$$2 \text{ koz sigma } \frac{110582}{2} = 55826$$

$$65535 - 55826 = 10233 \quad \text{Duty} \rightarrow P1.0 CPL$$

MOV T2CON, #00000000

MOV DPTR, #10233

MOV TL2, DPL

MOV TH2, DPH

MOV RCAP2L, DPL

MOV RCAP2H, DPH

MOV R0, #2d

Setb TB2

TF2 → 1 olsys

Oberhalb gelben

Oberhalb RCAP

gen yellow.

X: jnb TF2, (X) → TF2

clr TF2

DJNZ R0, X

CPL P1.0

Sjmp X

H	P	S	G	P	C	C1	P2
22				1	2	3	
23	4	5	6	7	8	9	10
24	11	12	13	14	15	16	17
25	18	19	20	21	22	23	24
26	25	26	27	28	29	30	

HAZIRAN JUNE CUMLA FRIDAY

29

# Tinned Kesmeli



ORG 002BH  
SJMP T2 kesme adt  
MOV T2COM, #00h  
MOV DPTR, #10233f  
MOV TL2, TH2, BCAP2L, BCAP2H = CPL, DPH  
MOV R0, #2d  
SEBB TB2

X:

SJMP X

T2 kesme adt:

JNZ ZD, Y  
CPL  
MOV P1.0  
Y: RETI R0, #2d

# 28

HAZIRAN JUNE  
PERSEMBE THURSDAY

H	P	S	G	P	C	C1	P2
22	4	5	6	7	8	9	10
23	11	12	13	14	15	16	17
24	18	19	20	21	22	23	24
25	25	26	27	28	29	30	

## Quick Sort

enable  
 $\text{ex} \rightarrow \text{external}$

①



Horia lesme Ø

Setb  $\cap \emptyset$  (Düzel kesişir kesmez)

set b exp (Horror kecne aktif)

Set b ed (400 beats) 1

Set-b ed (floor break) [enable all]

## TFO Këshmesi

sets et $\phi$  (enable timer)

set b ed

Sat [skeler artor gelir] ustebelam Ø olmasi

May tmod, # 0000 1101 mod#

\* Timer 0 100 adet aranı sayımı yapacak (kesme)  
offer: ve set b p2,0 yapacak  
Nun local #22222222 1.00

May 6 mod, #00009 101b → mod

Mov Dptr, #65435d note → ClF

nov TLφ, DPL

MOV TH $\varnothing$ , DPH

Setb et al.

See Ed

seb b TRP

X:

Samp X

**CUMA  
FRIDAY**  
**JULY**  
**TEMMUZ**

90



alt-gined;

clr	TFØ
setb	P2.0
not	TLØ, DPL
mov	THØ, DPH

reti

013 0000

Simp ayar

org 003h

Simp alt-grain → 2 Byte → scroll fein

00Bh

Simp alt - grer 2

0405:

X:

10

Simp X

alt grivent:

9

reti

05

**PERSEMBER JULY**

29	28	27	26	25	24	23	30	31	32
22	21	20	19	18	17	16	15	14	13
15	14	13	12	11	10	9	8	7	6
13	12	11	10	9	8	7	6	5	4
12	11	10	9	8	7	6	5	4	3
Pz	C1	P	C	S	G	H	M	U	Z

②

Gate  $\rightarrow$  1  $\rightarrow$  setb TH $\phi$  sadace struktur

P3.2  $\rightarrow$   0'01 s'e rekesker

P3.2  $\rightarrow$  Kosme  sadace objektsensor  
Jbilir (alle degi)

Gate = 1 timer disorden bas, Gribod.

Kosme

ORa 0000

Sfmp Ayer

ORa 0000

Sfmp ext1

ayor:

setb It $\phi$

setb ex $\phi$

setb eo

X:

Sfmp X

ext1:

setb P2.0

!

reti

(Bas uch 1 kere deki kodu yox yox)

ext1:

Y: jnb P3.2, Y

setb P2.0

Ack oleyber. ram burovu

200ms yoxda buan bestekh  
kodu kodu bilmir.

reti

Horicu besmerrn payraf. yox  
arka zeren P3.2 bur beyn  
grevi, geyid.

X: jnb P3.2, X = X: jnb IE $\phi$ , X



H	P	S	G	P	C	C	P2
26	2	3	4	5	6	7	8
27	1						
28	9	10	11	12	13	14	15
29	16	17	18	19	20	21	22
30	23	24	25	26	27	28	29
31	30	31					

TEMMUZ	JULY
2012	PAZARTESİ
	MONDAY

02

③



2ms diabolo

Yani 1ms de bir P2.0 toggle yapılıca

11058200 → 1sn

11058,2 → 1 milisn

11058 → 1 Verbindungszeit

MOV TMOD, #00000001 noise

MOV DPTA, #54476d → 65536 - 11058 = 54476

MOV TL0, TH0 = DPL, DPH

setb TR0

X: jnb TF0, X

clr TF0

cpl P2.0

MOV TL0, TH0 = DPL, DPH

Sjmp X

OR3 0000

Sjmp ayor

ORG 00BH

Sjmp TF0 - kesme

Ayor:

setb EA0

Setb EA

Setb TR0

TF0 - kesmesi:

CPL P2.0

clr TF0

MOV TL0, TH0 = DPL, DPH

Retr

X:

Sjmp X

03

 TUESDAY  
 SALI  
 JULY  
 TEMMUZ

H	P	S	G	P	C	C1	P2				
26								1			
27	2	3	4	5	6	7	8				
28	9	10	11	12	13	14	15				
29	16	17	18	19	20	21	22				
30	23	24	25	26	27	28	29				
31	30	31									

TEMMUZ 2012

$T\phi \rightarrow P3.6$  2012/09/01

(SAYFA 123)

```

MOV TMOD, #0000 0101b ; mod3/Canal
MOV DPTR, #64535d
MOV TL $\phi$ , DPL
MOV TH $\phi$ , DPH
SETB TR $\phi$ 
X: J $\phi$ b TF $\phi$ , X
    T $\phi$ 
    CPL P2,2
    MOV TL $\phi$ , DPL
    MOV TH $\phi$ , DPH

```

ALL



ayar:

```

MOV TMOD, #0000 0101b
MOV DPTR, #64535d
MOV TH $\phi$ , DPL
MOV TL $\phi$ , DPH
SETB ET $\phi$ ; timer $\phi$  start
SETB ED
SETB TR $\phi$ 

```

ORG 0000
SJMP ayar
ORG 00BH
SJMP esme0f

X:
 \* SJMP X
 Kosne dt:
 Set P2.0

retI

H	P	S	G	P	C	C1	P2
26	2	3	4	5	6	7	8
27	9	10	11	12	13	14	15
28	16	17	18	19	20	21	22
29	23	24	25	26	27	28	29
30	31						
31							

04
 TEMMUZ
 JULY
 GARÇAMBA
 WEDNESDAY

# 03

EKİM | CUMA  
OCTOBER | FRIDAY

	R	P	S	C	P	C	C1	C2
40				1	2	3	4	5
41	8	1		9	10	11	12	
42	13	14	15	16	17	18	19	
43	20	21	22	23	24	25	26	27
	28	29	30	31				

## C:// Keil // Object Virtual Terminal

#include "at89x51.h"

ORG 0000

Mahn:

CLR RI,2  
MOV TMOD, #00100000B ;20H ;T1  
MOV TH1, #0FDH  
MOV TL1, #0F0H  
SETB SCON, #0L010000B ;50H

Y: MOV SBUF, #6FH ;'A'  
JNB TI,\$  
CLR TI  
SJMP Y

END

Hep A gonderye

Y: JNB RI,X

CLR RI

MOV A, SBUF

CJNE A, # 'A', X

MOV SBUF, # 'B'

JNB TI,\$

CLR TI

X: SJMP Y

A gonderliginde

B fazda





080 001

MOV ADCON1, #10001100B ; energ + datal  
MOV ADCON2, #00000001B ; 1. band

setb SCONV ; Gerinn backload

JNB ADCI, \$  
CLR ADCI

MOV R0, ADCDATAH

MOV R1, ADCDATAL

end

SEPTEMBER | SUNDAY  
EYLÜL | PAZAR

# 28

setb ECONV

ADC1, \$

ADC1

MOV A, ADCDATAH

ANL D, #0FH

) yolsel nibble safland

MOV R0, A

MOV R1, ADCDATAL

SJMP X

end

Setb RT0

Setb EXP

Setb EA

Setb ET0

Setb EA

TFO

KOSMO

Setb BADE

Setb EA

ADC

kesme

SEPTEMBER | SATURDAY  
EYLÜL | CUMARTESİ

# 27

101101	101100	101101	101100	101101	101100	101101	101100
09	10	09	10	09	10	09	10
28	27	28	27	28	27	28	27
26	25	26	25	26	25	26	25
24	23	24	23	24	23	24	23
22	21	22	21	22	21	22	21
19	20	19	20	19	20	19	20
18	17	18	17	18	17	18	17
16	15	16	15	16	15	16	15
11	10	11	10	11	10	11	10
12	13	12	13	12	13	12	13
13	14	13	14	13	14	13	14
15	16	15	16	15	16	15	16
17	18	17	18	17	18	17	18
18	19	18	19	18	19	18	19
20	21	20	21	20	21	20	21
21	22	21	22	21	22	21	22
22	23	22	23	22	23	22	23
24	25	24	25	24	25	24	25
25	26	25	26	25	26	25	26
26	27	26	27	26	27	26	27
27	28	27	28	27	28	27	28
28	29	28	29	28	29	28	29
29	30	29	30	29	30	29	30
30	31	30	31	30	31	30	31
31	32	31	32	31	32	31	32
32	33	32	33	32	33	32	33
33	34	33	34	33	34	33	34
34	35	34	35	34	35	34	35
35	36	35	36	35	36	35	36
36	37	36	37	36	37	36	37
37	38	37	38	37	38	37	38
38	39	38	39	38	39	38	39
39	40	39	40	39	40	39	40
40	41	40	41	40	41	40	41
41	42	41	42	41	42	41	42
42	43	42	43	42	43	42	43
43	44	43	44	43	44	43	44
44	45	44	45	44	45	44	45
45	46	45	46	45	46	45	46
46	47	46	47	46	47	46	47
47	48	47	48	47	48	47	48
48	49	48	49	48	49	48	49
49	50	49	50	49	50	49	50
50	51	50	51	50	51	50	51
51	52	51	52	51	52	51	52
52	53	52	53	52	53	52	53
53	54	53	54	53	54	53	54
54	55	54	55	54	55	54	55
55	56	55	56	55	56	55	56
56	57	56	57	56	57	56	57
57	58	57	58	57	58	57	58
58	59	58	59	58	59	58	59
59	60	59	60	59	60	59	60
60	61	60	61	60	61	60	61
61	62	61	62	61	62	61	62
62	63	62	63	62	63	62	63
63	64	63	64	63	64	63	64
64	65	64	65	64	65	64	65
65	66	65	66	65	66	65	66
66	67	66	67	66	67	66	67
67	68	67	68	67	68	67	68
68	69	68	69	68	69	68	69
69	70	69	70	69	70	69	70
70	71	70	71	70	71	70	71
71	72	71	72	71	72	71	72
72	73	72	73	72	73	72	73
73	74	73	74	73	74	73	74
74	75	74	75	74	75	74	75
75	76	75	76	75	76	75	76
76	77	76	77	76	77	76	77
77	78	77	78	77	78	77	78
78	79	78	79	78	79	78	79
79	80	79	80	79	80	79	80
80	81	80	81	80	81	80	81
81	82	81	82	81	82	81	82
82	83	82	83	82	83	82	83
83	84	83	84	83	84	83	84
84	85	84	85	84	85	84	85
85	86	85	86	85	86	85	86
86	87	86	87	86	87	86	87
87	88	87	88	87	88	87	88
88	89	88	89	88	89	88	89
89	90	89	90	89	90	89	90
90	91	90	91	90	91	90	91
91	92	91	92	91	92	91	92
92	93	92	93	92	93	92	93
93	94	93	94	93	94	93	94
94	95	94	95	94	95	94	95
95	96	95	96	95	96	95	96
96	97	96	97	96	97	96	97
97	98	97	98	97	98	97	98
98	99	98	99	98	99	98	99
99	100	99	100	99	100	99	100
100	101	100	101	100	101	100	101

Kesne



AN1 → 2.5 2016

ORG H0000

SIMP bosta

ORG 0033H

SIMP ade lares

bosta

Mov ade cont, # 11001100B

Mov ade cont2, # 00000001B

SETH ade

SETH ea

SETH econt

SIMP \$

ade lares:

Mov a, adedatab

ORL a, & 0fh

Mov t\$, a

Mov t\$, adedatal

RET

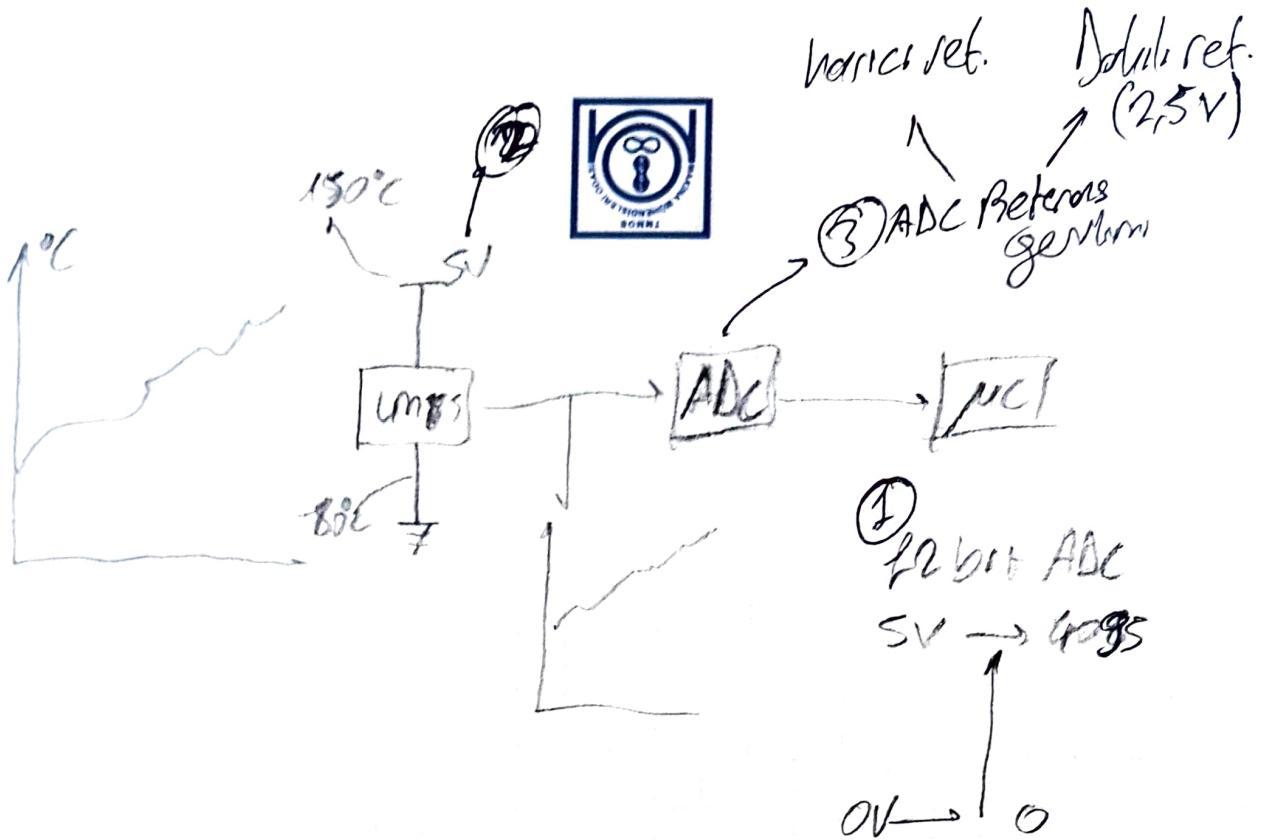
end

SEPTEMBER | FRIDAY  
EYLUL | CUMA

26

36	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
37	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
38	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
39	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
40	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31

EYLUL 2014



### Adilo Referenz

Fr. SV down max 4095 (12 bit rot) voraus.  
 6V also 616 4095 voraus.

$$5V \rightarrow 4095$$

$$4,5V = 6094 \text{ step}$$

$$4,0V = 6093 \text{ step}$$

$$\vdots$$

$$0,0024 = f^2$$

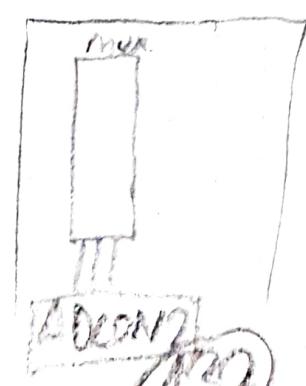
$$0,0012 = f^3$$

$$\text{step} = \frac{5V}{4095} = 0,0012$$

Häufigkeit

0,0024 → obere Toleranz  
 0,0012 → untere Toleranz

ADC ager oder kann seineslet  
 abweichen bei der obige



	40	29	28			
40	22	23	24	25	26	27
39	15	16	17	18	19	20
38	8	9	10	11	12	13
37	1	2	3	4	5	6
36	H	P	S	C	P	C
35	P	S	C	P	C	C
34	P	C	C	C	C	C



ADCON 1, 2 → Mod 1  
ADCON 4, 6 → Motor/Drehz.

1.5 }  
1.4 }  
1.3 }  
1.2 }  
1.1 → Motor  
1.0 → Motorico bo ms. 2

1. Tek conv (Single Conv) sets SCONV
2. Devr kon (Continuous Conv) sets CCONV
3. Torer 2 basincı
4. P3.5

X: Jnb ADC, X  
Jnb TFØ, \$

SEPTEMBER | MONDAY  
EYLÜL PAZARTESİ

29

H	P	S	G	P	C	C	P2
36	1	2	3	4	5	6	7
37	8	9	10	11	12	13	14
38	15	16	17	18	19	20	21
39	22	23	24	25	26	27	28
40	29	30					



DAC to the once H serial

12bit 10bit

ADC → DAC1 → DAC

W35 → DAC → Vref

2048 → DAC → Vref/2

066 → DAC → ADC → OV

DAC0

DAC1

8bit  
12bit

8bit  
16bit

ADC DATAH  
ADC DATAL

DACIH DACOH  
DACL DACOL

MOV DACPH, #081

MOV DACDL, #00H      0800H x 2048

DACCON →

mod  
16bit

Vref = 2.5 [mV]

2.5 → 2.5 [mV]

SEPTEMBER | THURSDAY

25

EYLUL 2014	P	S	C	P	C	C1	P2
40	29	30					
39	22	23	24	25	26	27	28
38	15	16	17	18	19	20	21
37	8	9	10	11	12	13	14
36	1	2	3	4	5	6	7





OKC 004

stop border

136

110AC008

MOV A\$CON1, #80H ; 10000000

mov DACCON, #00001101B

Nov DPTB, #2048d

Nov 1984 DACTH, dph

Nov DACOL, dPL

sfmp \$

end

Aleksi gewen dikt filter → DAC wünsch.  
Zero Order Hold

SEPTEMBER | WEDNESDAY  
EYLUL GARŞAMBA

24

# 06

EKİM | PAZARTESİ  
OCTOBER | MONDAY

(PWM)

H	P	S	Ç	P	C	C1	Pz
40		1	2	3	4	5	
41	6	7	8	9	10	11	12
42	13	14	15	16	17	18	19
43	20	21	22	23	24	25	26
44	27	28	29	30	31		

%20 doluluk  
Ort → 2V



1-Periyot  
2-On surez

Oscilatörde yarım周期da fSN uygulanabilir

$$1000\text{Hz} \times 1000 = f_{SN}$$

TMS

KK

$$200\text{Hz} \text{ periyodu } \frac{1}{200} = 5\text{ms} \rightarrow T_{PWM}$$

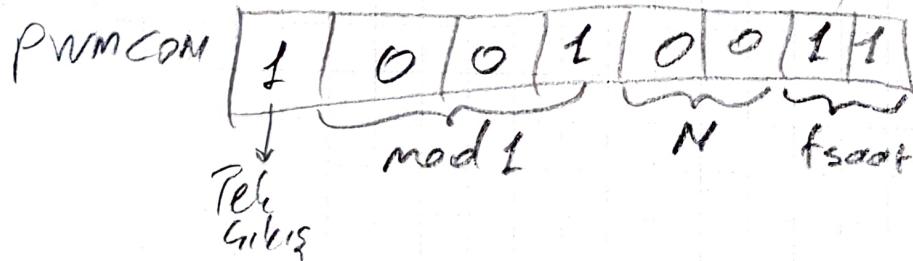
%25 on  
pwm adadı = 1

f saat = fosc

pwm çıkış, P2.7 N=1

Fomerde  $f_{SN} \rightarrow \frac{11053200}{200} = 55236$

$$65535 - 55236 = 10299$$



MİLLİ  
DÜNYA



# 07

EKİM SALI  
OCTOBER TUESDAY

$$625 \rightarrow \frac{55236}{4} = \text{Ans} \\ 65535 - \text{Ans} = 51726$$

EKİM 2014						
W	S	C	P	C	Ct	Pz
40		1	2	3	4	5
41	6	7	8	9	10	11
42	13	14	15	16	17	18
43	20	21	22	23	24	25
44	27	28	29	30	31	

MOV PWM1H, #10300D  
MOV PWM1L, #10300D

MOV PWM0H, #51726D  
MOV PWM0L, #51726D

MOV PWMCON, #10010011B ; PWM Alesif  
%25 duty  
SMS

1Hz periode vs 1sn periyod (%25 on)

fosc (15xBF) seviyesi

$$\frac{11059200}{15 \times 64} = 11520 \text{ Hz}$$

BF → CFG841 den  
N bilgisi

PWMCON [ ] 0 0 1 0 0 0 0  
Single N Fosc/15xBF

$$65535 - 11520 = 54015$$

MOV PWM1H, 1L #54015D  
MOV PWM0H, 0L #62655D  
MOV PWMCON, #10010000B

$$-\frac{11520}{4} + 65535 = 62655$$

(CFG841 den 69 bul) S.131



```

1 // mikroislemci mithakkak bir sonuz dongu icinde calismasi gereklidir
2
3 //N:
4 // kedlar bu araya yazılacak
5 //sjmp N
6
7 #include "ADUC841.H"
8
9     org    0000h
10 basla: //*****IVEDI ADRESLEME OPNEKLERİ*****
11
12
13     mov     a,#02h
14     mov     20h,#20h
15     mov     r0,#0AAh
16     mov     DPTR,#1FE7h
17     mov     p0,#0CCh
18
19 //asagida ayni adreslere farkli degerler yazilmistir
20 //RAM SFR bolgesinden degisikler kontrol edilebilir
21     mov     a,#0A2h
22     mov     20h,#0A0h
23     mov     r0,#0FAh
24     mov     DPTR,#2FE9h
25     mov     p0,#0FFh
26
27
28
29     mov     00h,#00001111b
30     mov     01h,#54d
31     mov     02h,#98h
32     mov     03h,#0AEh
33     mov     03h,#0Ch
34
35 // 03 adresinin icerigi degisir
36
37
38 // su anda mikrosilemci ilk basladiginda
39 // bank0 dayiz
40
41     r0,#54h      // buarasi 00h adresi ile ayni ram adresi
42     r1,#10100011b // binary ile de sayı girisi yapilabilir
43     r0h,#23d      // decimal ile de sayı girisi yapilabilir
44
45 //bank1'e gecelim, PSW.3 ve PSW.4 nolu bitlerini degistirelim
46     setb   PSW.3      // bank1 gecis yapildi
47
48 // demek oluyorki PSW'nin bitlerine bit erisim kabul ediliyor
49 // bir ustteki kodda PSW.3 biti rs0 olarak adlandirilmisitir
50 // bir ustteki kodun yerine rs0'i birleyerek te yapabiliriz, yani
51
52     clr   PSW.3      // tekrar bank0a gecis yapildi
53
54     setb   rs0          // = setb PSW.3, tekrar bank1e gecis yapildi
55
56     mov     r0,#23h      // 008 adresi
57 // burada yine direkt ram adresleri ile istenilen bolyegeye erisilebilir
58 // ancak r0 gibi isimlendirmelerde hangi bankta isek oradaki ram bolgesi degistiriliyor
59
60     mov     12h,#66h      // Baska bankin adresine direkt erisebildik ancak
61     r0,#34h           // ile hangi bankta isek oranin icerigi degistirildi
62
63
64     mov     p0,#0FOh      // Portlara bayt adresleme yapabiliyoruz
65
66
67     mov     DPTR,#01FEh    // DPTR'ye direkt 16 bit aktarabiliriz
68
69 //*****POGPULAN ADRESLEME OPNEKLERİ*****
70     r0h,a            // 02h adresine a'nin icerigini yaz
71     a,01h // 01 adresinin icerigini 01h adresine yazdik ancak
72 // asagidakı kod adres icerigini yazmaz. Direkt veridir
73     a,#01h
74

```



```

75      mov     30h,01h // adres içeriği diğer adresinin içeriğine aktarıldı
76
77      mov     r5,01h // hangi bankta ise o bankın r5 ine 01h adresinin içeriği aktarıldı
78
79      mov     p1,a
80      mov     p0,01h
81
82
83
84      // bit taşıma işlemleri carry biti üzerinden yapılıyor carry biti psw'nin 7. bitidir.
85      mov     c,acc.6
86      mov     c,p0.1
87      mov     p0.2,c
88
89      // ancak böyle bir kullanım yoktur
90      //mov     p0.1,psw.7
91
92      mov     01h,c // bit belgesi 01h adresine bit bit aktarıldı
93      setb    07h
94      mov     c,07h // bit belgesi içeriği c'ye aktarıldı
95      mov     p0.1,c
96
97      //*****SAKLAYICI DOGRUDAN ADRESLEME ORNEKLERİ*****
98      mov     a,r5           // hangi bankta ise oradaki rl içeriği akumulatore aktarılır
99
100     setb   rs0
101     mov     r0,a           // banka dikkat et.
102
103     ;DOLAYLI ADRESLEME örnekleri
104     // dolaylı adresleme için sadece r0 ve rl kullanılıyor
105
106
107     // mov psw,#00h
108
109     mov     r0,#02h
110     mov     02h,#33d
111
112     mov     a,@r0
113     mov     03h,@r0
114
115     setb   rs0
116     setb   rs1 ; bank3e geçtiğ
117
118     mov     a,@r0
119     mov     03h,@r0
120
121     ; DOLAYLI ADRESLEME RAM SIFIRLAMA
122     // djnz komutu kullanılacak
123     mov psw,#00h           // bank0 da olmamız gereklili Yani
124
125     mov     r0,#7fh          // 00h adresinde 128 yazıyoruz
126     mov     a,#00h
127     dng:
128     mov     @r0,a
129     djnz   r0,dng          ; eğer r0 içeriği 0 değilse dng'e dallan
130     // en son djnz komutunda r0=00h olarak kalır yani 00h adresi 0 verisine ulaşılır
131     // ve bir alttaki kod işler
132
133     ; -----
134     //*****SAKLAYICI İNDISLI ADRESLEME ORNEKLERİ*****
135     // PROGRAM HAFIZA içeriğine erişmek için kullanılır
136     mov     a,#04h
137     mov     DPTR,#1000h
138     movc   a,@a+DPTR
139     mov     p0,a
140     // sjmp basla
141     // istediğiniz bir program hafıza belgesine istediğiniz verileri yazabilirsınız
142     // örneğin alt tarafta program hafızanın 1000,1001,1002,1003 ve 1004 adreslerine
143     // rastgele sayılar atanmıştır
144
145     org    1000h
146     dng:   db 0,1,2,3,4
147
148     ljmp basla

```