



8. HAFTA

BLM320

BİLGİSAYAR MİMARİSİ

Yrd. Doç. Dr. Salih GÖRGÜNOĞLU

sgorgunoglu@karabuk.edu.tr

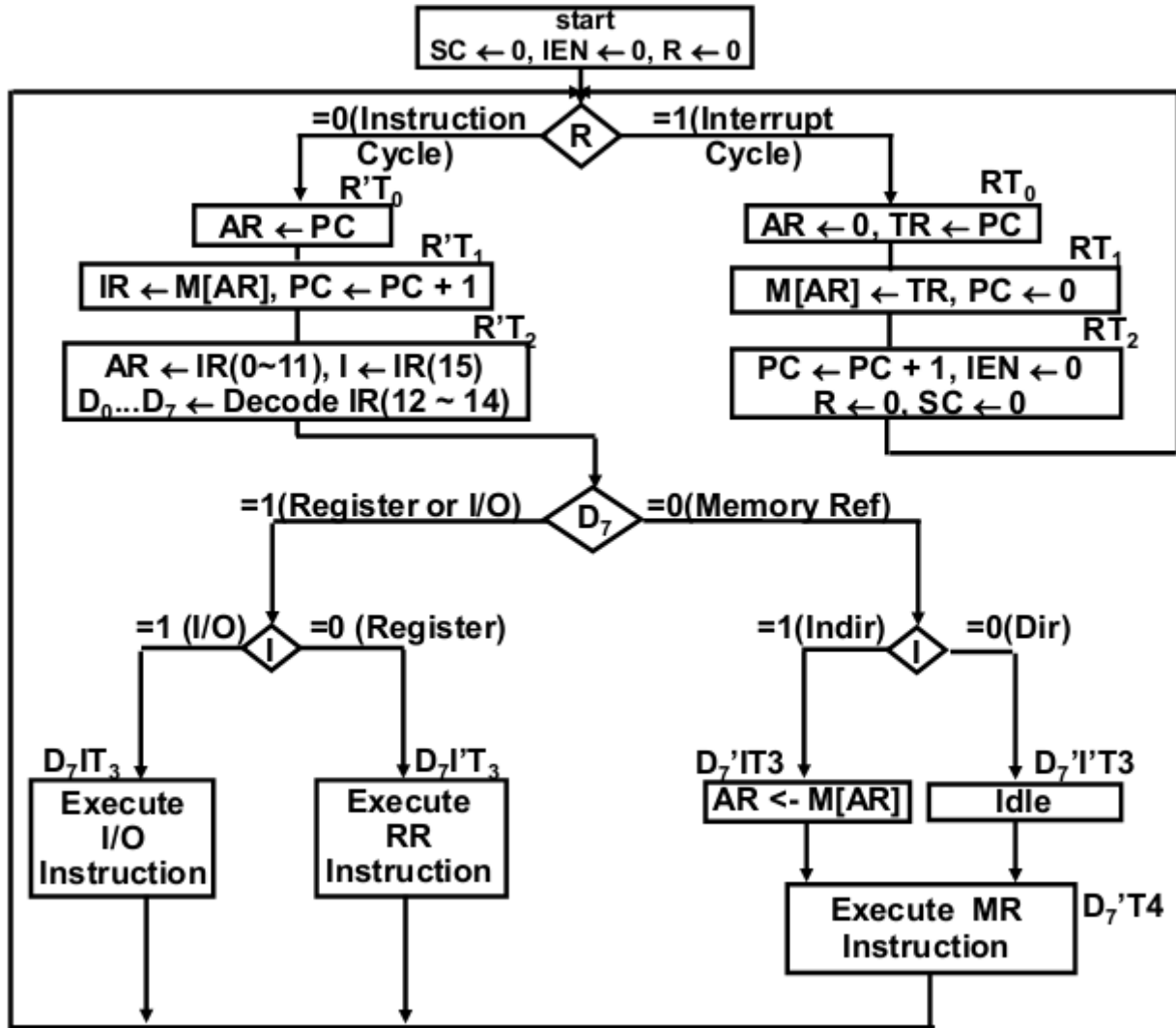
KBUZEM

Karabük Üniversitesi

Uzaktan Eğitim Uygulama ve Araştırma Merkezi

8. Temel Bilgisayarın Donanı tasarımı

Bilgisayar komutlarının tamamının çalışmasını tanımlayan akış çizelgesi



Tablo I: Bilgisayar komutlarının tamamının çalışmasını tanımlayan tüm mikroişlemler

Fetch	$R'T_0:$	$AR \leftarrow PC$
	$R'T_1:$	$IR \leftarrow M[AR], PC \leftarrow PC + 1$
Decode	$R'T_2:$	$D_0, \dots, D_7 \leftarrow \text{Decode } IR(12 \sim 14),$ $AR \leftarrow IR(0 \sim 11), I \leftarrow IR(15)$
Indirect	$D_7'IT_3:$	$AR \leftarrow M[AR]$
Interrupt	$T_0'T_1'T_2'(IEN)(FGI + FGO):$	$R \leftarrow 1$
	$RT_0:$	$AR \leftarrow 0, TR \leftarrow PC$
	$RT_1:$	$M[AR] \leftarrow TR, PC \leftarrow 0$
	$RT_2:$	$PC \leftarrow PC + 1, IEN \leftarrow 0, R \leftarrow 0, SC \leftarrow 0$
Memory-Reference		
AND	$D_0T_4:$	$DR \leftarrow M[AR]$
	$D_0T_5:$	$AC \leftarrow AC \wedge DR, SC \leftarrow 0$
ADD	$D_1T_4:$	$DR \leftarrow M[AR]$
	$D_1T_5:$	$AC \leftarrow AC + DR, E \leftarrow C_{out}, SC \leftarrow 0$
LDA	$D_2T_4:$	$DR \leftarrow M[AR]$
	$D_2T_5:$	$AC \leftarrow DR, SC \leftarrow 0$
STA	$D_3T_4:$	$M[AR] \leftarrow AC, SC \leftarrow 0$
BUN	$D_4T_4:$	$PC \leftarrow AR, SC \leftarrow 0$
BSA	$D_5T_4:$	$M[AR] \leftarrow PC, AR \leftarrow AR + 1$
	$D_5T_5:$	$PC \leftarrow AR, SC \leftarrow 0$
ISZ	$D_6T_4:$	$DR \leftarrow M[AR]$
	$D_6T_5:$	$DR \leftarrow DR + 1$
	$D_6T_6:$	$M[AR] \leftarrow DR, \text{ if}(DR=0) \text{ then } (PC \leftarrow PC + 1),$ $SC \leftarrow 0$
Register-Reference		
	$D_7IT_3 = r$	(Common to all register-reference instr)
	$IR(i) = B_i$	($i = 0, 1, 2, \dots, 11$)
	$r:$	$SC \leftarrow 0$
CLA	$rB_{11}:$	$AC \leftarrow 0$
CLE	$rB_{10}:$	$E \leftarrow 0$
CMA	$rB_9:$	$AC \leftarrow AC'$
CME	$rB_8:$	$E \leftarrow E'$
CIR	$rB_7:$	$AC \leftarrow shr AC, AC(15) \leftarrow E, E \leftarrow AC(0)$
CIL	$rB_6:$	$AC \leftarrow shl AC, AC(0) \leftarrow E, E \leftarrow AC(15)$
INC	$rB_5:$	$AC \leftarrow AC + 1$
SPA	$rB_4:$	If($AC(15)=0$) then ($PC \leftarrow PC + 1$)
SNA	$rB_3:$	If($AC(15)=1$) then ($PC \leftarrow PC + 1$)
SZA	$rB_2:$	If($AC = 0$) then ($PC \leftarrow PC + 1$)
SZE	$rB_1:$	If($E=0$) then ($PC \leftarrow PC + 1$)
HLT	$rB_0:$	$S \leftarrow 0$
Input-Output		
	$D_7IT_3 = p$	(Common to all input-output instructions)
	$IR(i) = B_i$	($i = 6, 7, 8, 9, 10, 11$)
	$p:$	$SC \leftarrow 0$
INP	$pB_{11}:$	$AC(0-7) \leftarrow INPR, FGI \leftarrow 0$
OUT	$pB_{10}:$	$OUTR \leftarrow AC(0-7), FGO \leftarrow 0$
SKI	$pB_9:$	If($FGI=1$) then ($PC \leftarrow PC + 1$)
SKO	$pB_8:$	If($FGO=1$) then ($PC \leftarrow PC + 1$)
ION	$pB_7:$	$IEN \leftarrow 1$
IOF	$pB_6:$	$IEN \leftarrow 0$

Temel Bilgisayarın Donanımsal tasarımı

Temel bilgisayarı oluşturan donanımsal elemanlar

Bellek: 4096x16

Kaydediciler: AR, PC, DR, AC, IR, TR, OUTF, INPR, ve SC

Flip Floplar: I, S, E, R, IEN, FGI ve FGO

Dekoderler: 3x8 opcode decoder (D0 ... d7)

4x16 zamanlama dekode (T0 ... T15)

Ortak yol: 16 bit

Kontrol Sinyallerini üreten Kontrol birimi

- Kaydedicilere yükleme yapan LD, INC, CLR girişlerin kontrolü
- Bellekten okuma ve yazma yaptıran Read, Write sinyalleri
- Filip floplar Set, Clear, complement yapan girişler
- Ortak yolu kullanmak için kaydedicileri seçen S2, S1, S0 sinyalleri
- AC, ALU kontrolü yapan sinyallerin üretimi

Kontrol biriminin tasarımı

Kontrol birimi içinde her bir kaydedici ve birimlerin kontrolü için üretilmesi gereken kontrol sinyallerini üreten devrenin tasarımı ayrı ayrı verilecektir.

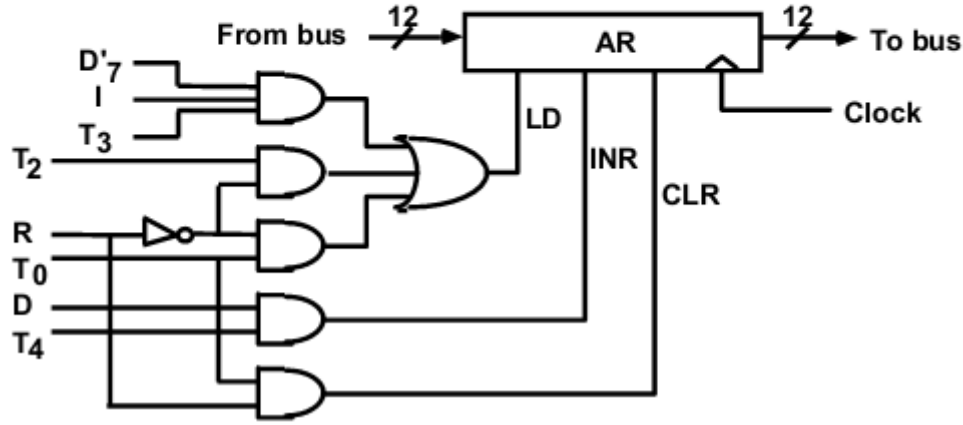
Adres Kaydedicisi AR için gerekli kontrol sinyallerinin üretilmesi.

AR ile ilgili tüm işlemler Tablo I incelenerek bulunur.

$R'T_0:$	$AR \leftarrow PC$	$LD(AR)$
$R'T_2:$	$AR \leftarrow IR(0-11)$	$LD(AR)$
$D_7IT_3:$	$AR \leftarrow M[AR]$	$LD(AR)$
$RT_0:$	$AR \leftarrow 0$	$CLR(AR)$
$D_5T_4:$	$AR \leftarrow AR + 1$	$INR(AR)$

Daha sonra AR kaydedicisini yükleyen (LD), içeriğini temizleyen (CLR) ve içeriğini bir artıran (INR) mikroişlemler olarak bunlar kendi içinde aşağıda görüldüğü gibi ilgili giriş için gruplanır. Ve şeması çizilir.

$$\begin{aligned} \text{LD(AR)} &= R'T_0 + R'T_2 + D'_7IT_3 \\ \text{CLR(AR)} &= RT_0 \\ \text{INR(AR)} &= D_5T_4 \end{aligned}$$



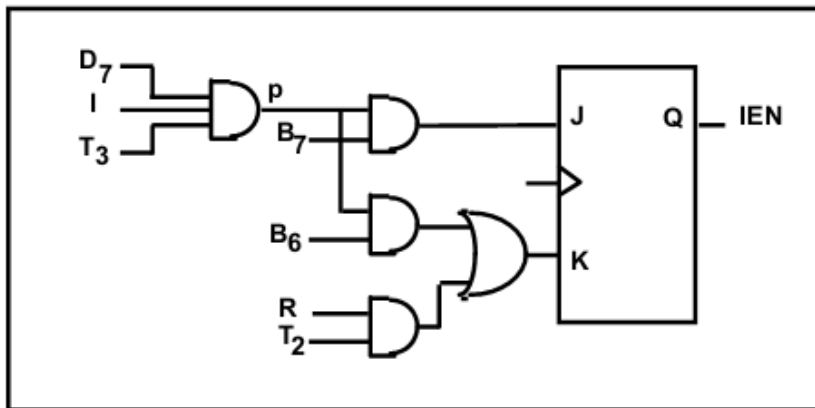
IEN (interrupt enable flag) flip flopunun kontrolü

pB_7 : $IEN \leftarrow 1$ (I/O Instruction)

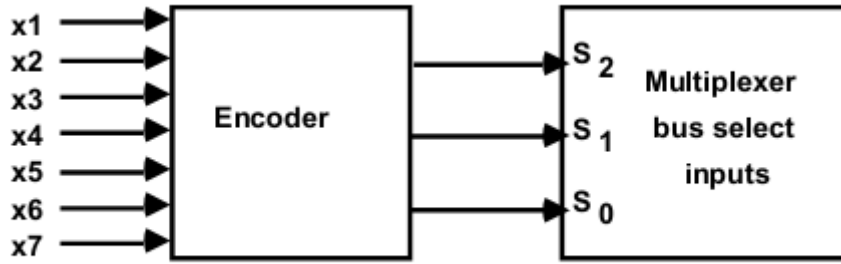
pB_6 : $IEN \leftarrow 0$ (I/O Instruction)

RT_2 : $IEN \leftarrow 0$ (Interrupt)

$p = D_7IT_3$ (Input/Output Instruction)

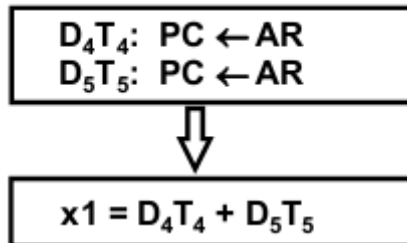


Ortak yolun kontrolü



x1	x2	x3	x4	x5	x6	x7	S2	S1	S0	selected register
0	0	0	0	0	0	0	0	0	0	none
1	0	0	0	0	0	0	0	0	1	AR
0	1	0	0	0	0	0	0	1	0	PC
0	0	1	0	0	0	0	0	1	1	DR
0	0	0	1	0	0	0	1	0	0	AC
0	0	0	0	1	0	0	1	0	1	IR
0	0	0	0	0	1	0	1	1	0	TR
0	0	0	0	0	0	1	1	1	1	Memory

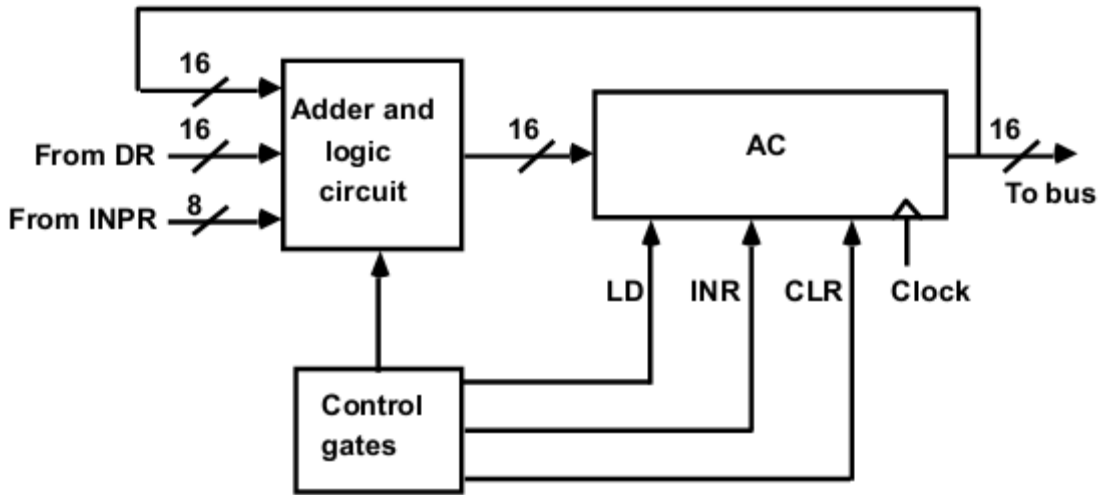
Örnek olarak AR kaydedicisinin ortak yolu kullanması için encoderin x1 girişine verilmesi gereken giriş aşağıdaki gibi olmalıdır. Diğer kaydediciler için de Tablo I taranarak ilgili girişler bulunabilir.



x2,x3,x4,x5,x6,x7=?

AC kaydedicisinin kontrolü

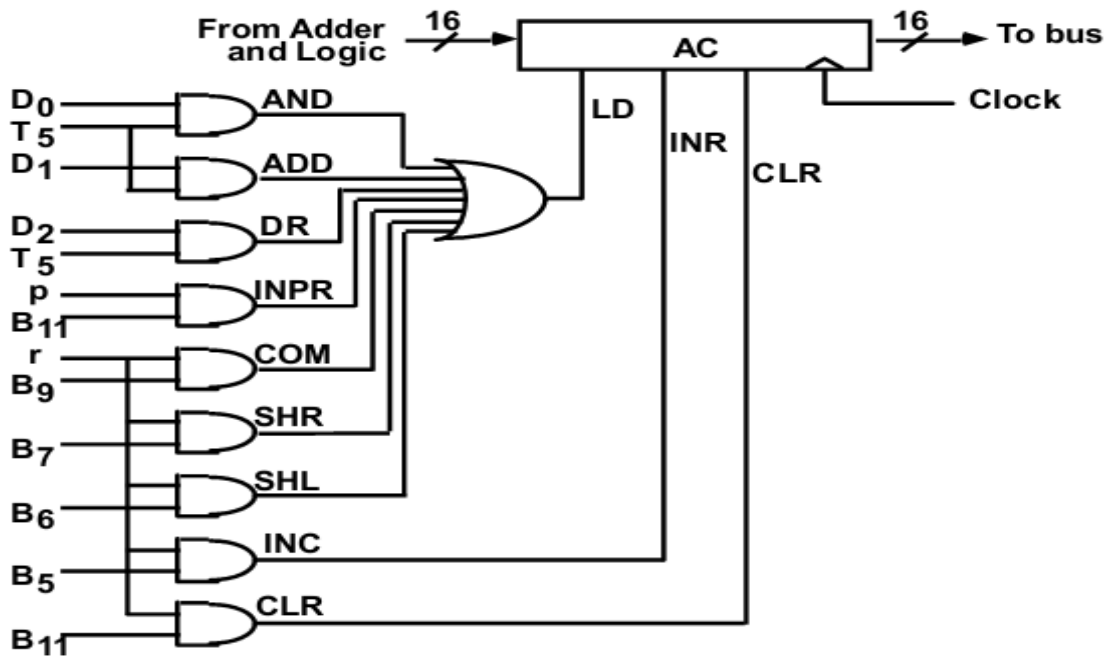
AC kaydedicisi diğer kaydedicilerden biraz farklıdır. Çünkü ALU ile birlikte çalışmaktadır. Ve ALU dan üretilen sonuçlar AC de saklanmaktadır.



AC nin içeriğini değiştiren bütün miroişlemler şu şekildedir.

D_0T_5 :	$AC \leftarrow AC \wedge DR$	AND with DR
D_1T_5 :	$AC \leftarrow AC + DR$	Add with DR
D_2T_5 :	$AC \leftarrow DR$	Transfer from DR
pB_{11} :	$AC(0-7) \leftarrow INPR$	Transfer from INPR
rB_9 :	$AC \leftarrow AC'$	Complement
rB_7 :	$AC \leftarrow shr AC, AC(15) \leftarrow E$	Shift right
rB_6 :	$AC \leftarrow shl AC, AC(0) \leftarrow E$	Shift left
rB_{11} :	$AC \leftarrow 0$	Clear
rB_5 :	$AC \leftarrow AC + 1$	Increment

AC kaydedicisini kontrol eden devre aşağıdaki gibidir.



ALU devresinin bir biti

