

Temel Bilgisayar Yapısı ve Devreleri

43

Bu bölümde anlatılan Temel Bilgisayar, ticari bir bilgisayara kıyasla hem daha basit bir yapıya hem de anlaşılması daha kolay bir manitöra sahiptir. Diğer bölümde bu temel bilgisayarı komut kütüphanesi kullanılarak bazı basit programlar yazılaraktır.

Buyruklar, veriler ile beraber, bellekte bulunurlar. Buyruklar bir kağıt parçasından oluşurlar. Buyrugun en önemli parçası işlem kodu kısmıdır. Bu kısmındaki bitler toplama, çıkarma, veleme, veyalama, kaydırma ve türleme gibi bazı işlemleri tanımlar. Buyruk içindeki işlem bitlerinin sayısı bilgisayar içindeki mevcut işlemlerin sayısına bağlıdır. 64 farklı işlem ($2^6 = 64$) yapabilen bir bilgisayarın 6 bit işlem kodudur. Mesela toplama işleminin işlem kodu 011001 olabilir.

Her bir buyruk ikili bir kod olup bilgisayara belli bir işlem yapmasını söyler. Denetim birimi buyruğu bellekten alıp işlem kodunun bitlerini anlamlandırır. Bilgisayar yazacılardan bazı mikro işlemleri başlatmak üzere bir dizi denetim sinyali üretir. Her bir işlem kodu işin denetim birimi bir dizi mikro işlemin yapılmasını sağlar.

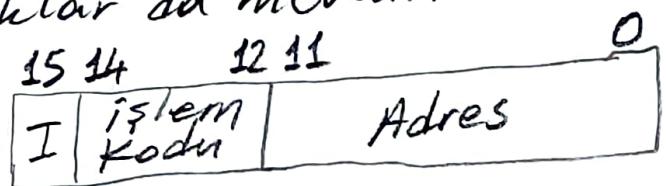
Buyruk kodu sadece işlemleri değil aynı zamanda verilerin bulunduğu yazış ve bellek adresleri ile sonucun yazılıcılara yazış ve bellek adreslerini de içermelidir. Her bilgisayarn kendine has buyruk kod düzenini vardır. Bu bİsteler bilgisayar mimarisini tasarılayanlar tarafından belirlenir.

Eğer bir buyruk kodu işin içindeki işlem, bellekten bir veriye ihtiyaç duymuyorsa, buyruk içindeki diğer bitler başka amaslar işin kullanılabilir. Örneğin işlemci yazacını temizle ($AC \leftarrow 0$), işlemci yazacını yükle ($AC \leftarrow \overline{AC}$), işlemci yazacını bir artır ($AC \leftarrow AC + 1$) gibi.

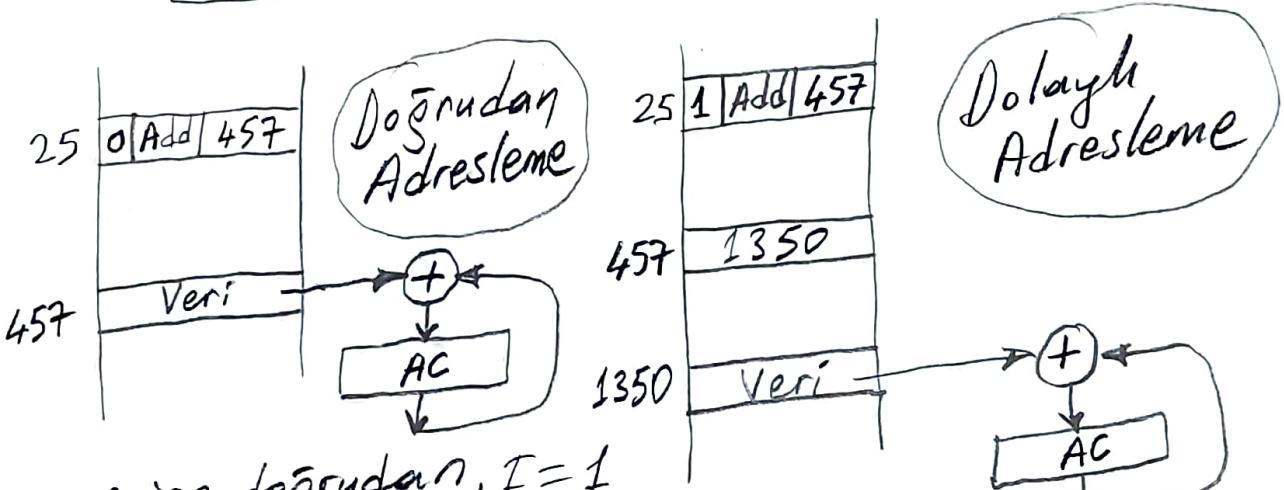
Tek bir işlemci yazacı bulunan bilgisayarlarda bu işlemci yazacına Birikç (AC - Accumulator) denir. İşlem bellekten alınan veri ile işlemci yazacının içeriği arasında perçehlesir. Sonuç işlemci yazacısında doldurulur. Bellekte işlemci yazacısı arasında dosyadaki işlem olmaz. Önce bellekteki bilgiin Veri Yazacına aktarılması lazımdır. Bu iş için bellekteki verinin adresini bulmak işin bir de Adres Yazacına ihtiyaç vardır.

44

Büyruklar belleğin bir kısmında, veriler ise başka bir kısmında bulunur. 4096 kelimelik bir bellek birimi için 12 bit adres uzunluğu gereklidir. Çünkü $2^{12} = 4096$ dir. Eğer bir buyruğu 16 bitlik bellek kelimesine yerleştirirse işlem kodu 4'ün 4 bitlik bir yer kahr. Denetim birimi 16 bitlik buyruğu, belleğin program kısmından okur. Buyruğun 12 bitlik adres kısmını kullanarak belleğin veri kısmından 16 bitlik veriyi okur. Sonra 4 bitlik işlem kodu tarafından belirlenen işlemi icra eder. Bu şekilde icra edilen buyruk bellek adreslemeli buyruktur. Yazış adreslemeli buyruklar ve giriş-çıkış adreslemeli buyruklar da mevcuttur.



Buyruk Bisimi



$I=0$ ise doğrudan, $I=1$ ise dolaylı adreslemedir.

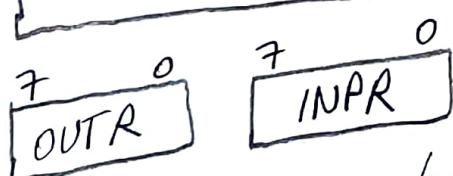
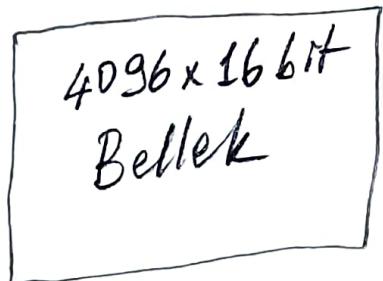
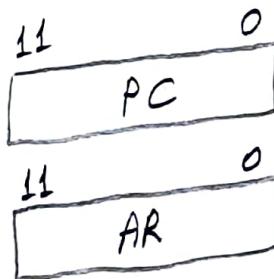
Ortak Veri Yolu

Temel bilgisayarda 8 yazış ve bir bellek bulunmaktadır. Bir de denetim birimi vardır. Yazışlar arasında ve yazışlar ile bellek arasında verilerin aktarımını sağlayan yollara ihtiyaç vardır. Yolun tek ve ortak olması karışıklığı azaltır. Ortak veri yoluna hangi yazıştan veya bellekten veri aktarılacağına denetim birimi karar verir. Herhangi bir yazışın yükle biti aktif edilirse veri yolundan bilgiyi alır. Belleğe bilgi almak için ise oku biti aktif edilir.

Temel Bilgisayarın İzin Yazar Listesi

(45)

| Yazar Sembolü | Bit Sayısı | Yazar İsmi | Yazar Fonksiyonu |
|---------------|------------|----------------|----------------------------|
| DR | 16 | Veri Yazarı | Bellek Verisini Tutar |
| AR | 12 | Adres Yazarı | Bellek İzin Adresini Tutar |
| AC | 16 | Birikes | İşlemci Yazarı |
| IR | 16 | Buyruk Yazarı | Buyruk Kodunu Tutar |
| PC | 12 | Program Sayacı | Buyruk Adresini Tutar |
| TR | 16 | Geçici Yazarı | Geçici Veriyi Tutar |
| INPR | 8 | Giriş Yazarı | Giriş Karakterini Tutar |
| OUTR | 8 | Gökçes Yazarı | Gökçes Karakterini Tutar |



| S ₂ S ₁ S ₀ | Veri Yolu |
|--|-----------|
| 0 0 0 | — |
| 0 0 1 | AR |
| 0 1 0 | PC |
| 0 1 1 | DR |
| 1 0 0 | AC |
| 1 0 1 | IR |
| 1 1 0 | TR |
| 1 1 1 | Bellek |

INPR yazarının ortak veri yoluna bağlantısı yoktur. OUTR yazarından ortak veri yoluna bilgi gitmez. Fakat bilgi alabilir. AC yazarından ortak veri yoluna bilgi gider. Fakat bilgiyi ortak veri yolundan değil de ALU hikayesinden alır. Bunun dışında tüm yazarların gökçesinden ortak veri yoluyla bağlantıları vardır. ve belleğin ortak veri yoluyla bağlantıları vardır. Hem veri alabilir hem de veri verebilirler.

Temel Bilgisayar Buyruk Bisimleri

(46)

| | | | | | |
|-------|-------|---|-----|-------|---|
| 15 14 | 12 11 | I | Kod | Adres | O |
|-------|-------|---|-----|-------|---|

Bellek Adreslemeli Buyruk

$I=0$ igin doğrudan, $I=1$ igin dolaylı adresleme.
Kod 000 ile 110 arası ($0-6$ veya $8-14$ arası)

| | | |
|---------|--------------|---|
| 15 | 12 11 | O |
| 0 1 1 1 | Yazış İşlemi | |

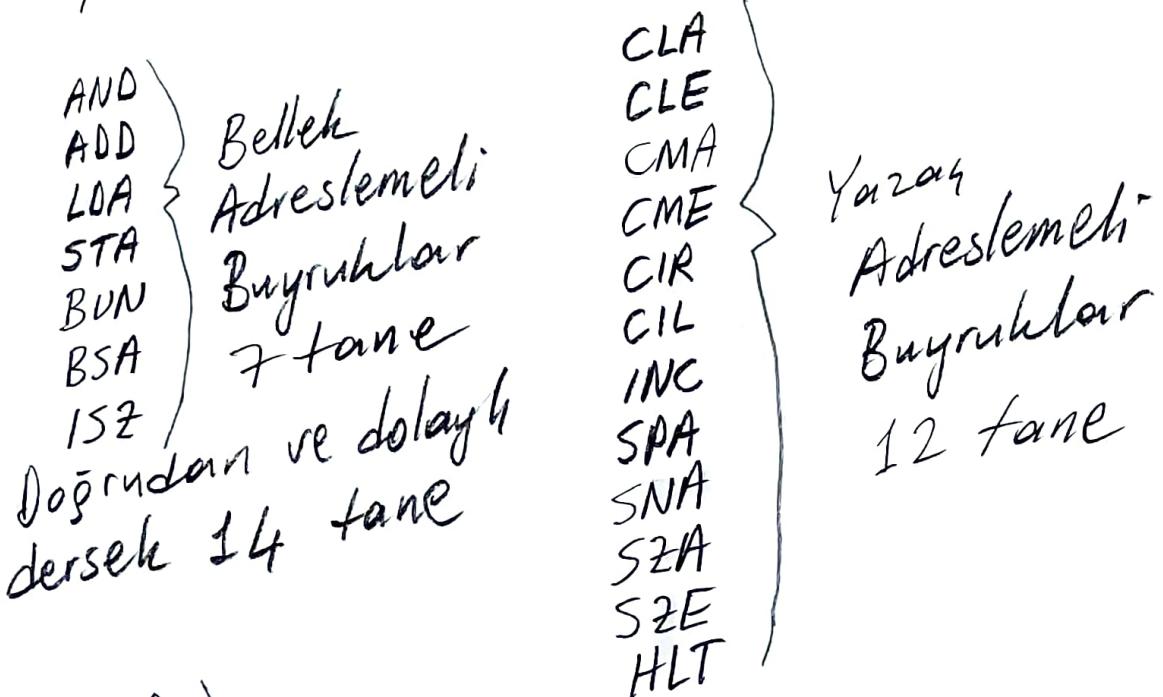
Yazış Adreslemeli Buyruk

Kod = 111, $I=0$ (İşlem kodu 7)

| | | |
|---------|------------------|---|
| 15 | 12 11 | O |
| 1 1 1 1 | Giriş-Gıç İşlemi | |

Giriş-Gıç Adreslemeli Buyruk

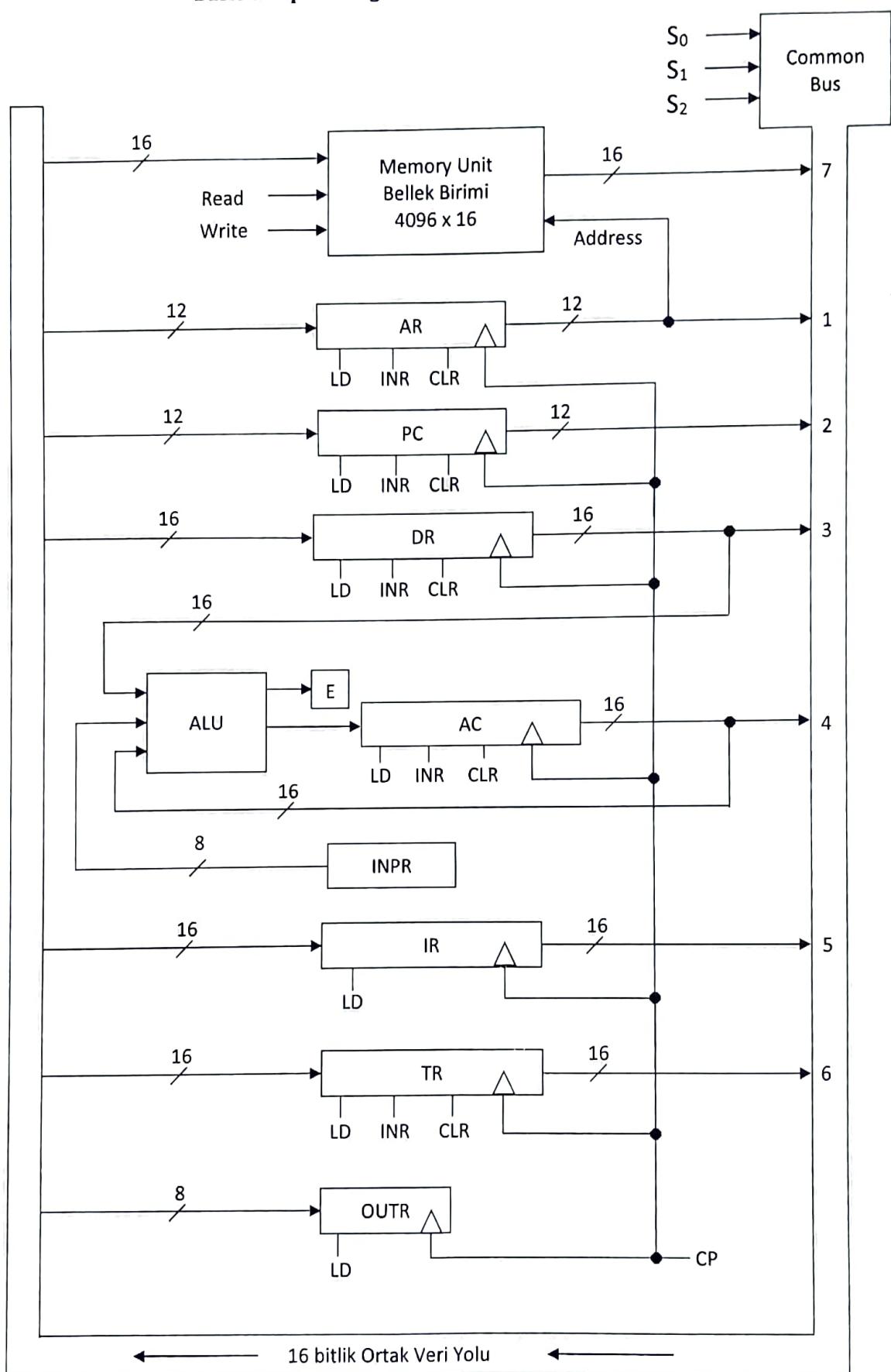
Kod = 111, $I=1$ (İşlem kodu 15)



INP
OUT
SKI
SKO
ION
IOF

Giriş-Gıç
Adreslemeli
Buyruklar
6 fane

Temel Bilgisayar Yazaçlarının bir Ortak Veri Yoluna Bağlanması
Basic computer registers connected to a common bus

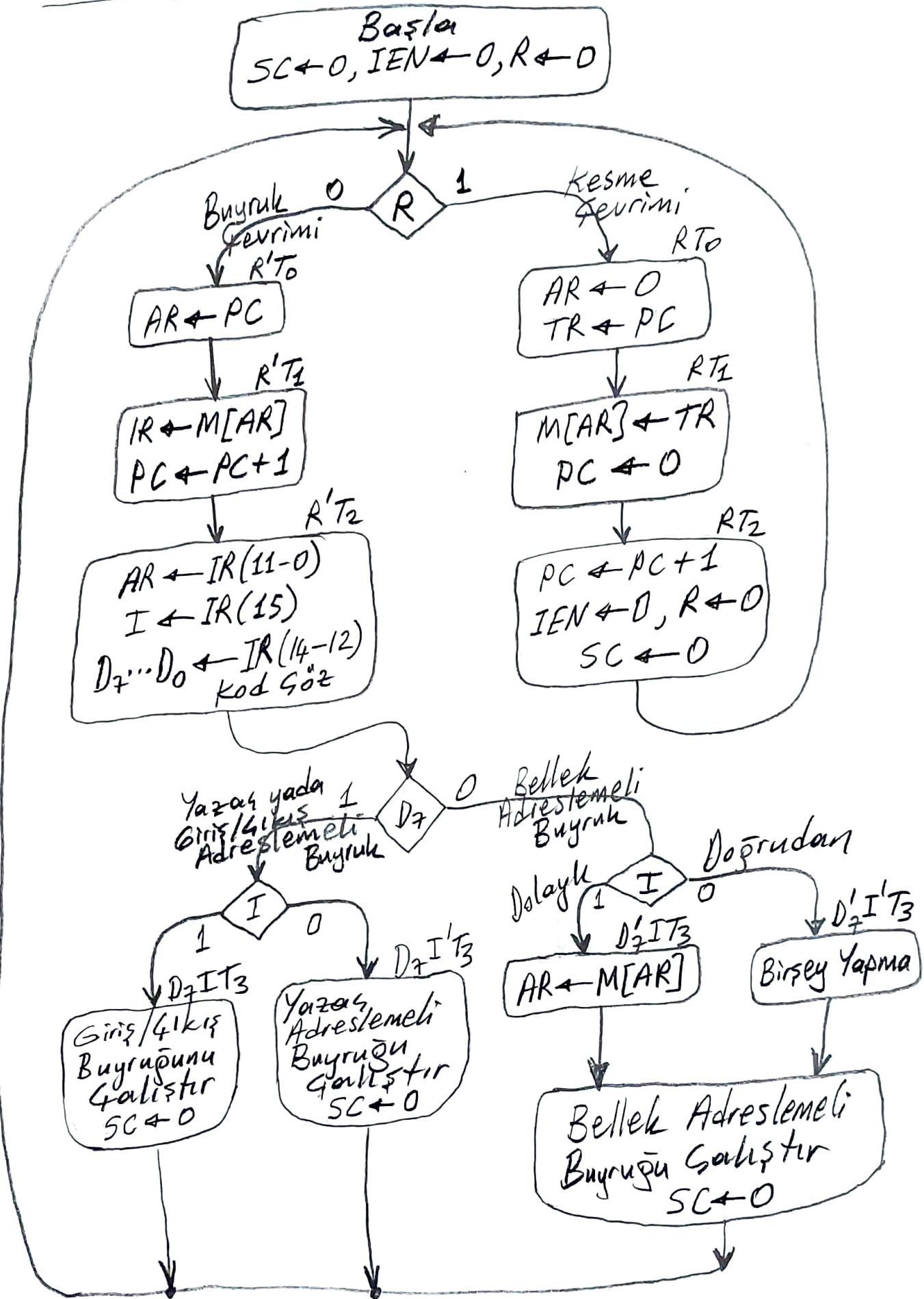


Temel Bilgisayar İçin Denetim Fonksiyonları ve Mikroişlemler Control functions and microoperations for the basic computer

| | |
|---|---|
| Fetch – Al Getir | $R' T_0 : AR \leftarrow PC$ $R' T_1 : IR \leftarrow M[AR], PC \leftarrow PC + 1$ |
| Decode – Kod Çöz | $R' T_2 : I \leftarrow IR(15), D_7 \dots D_0 \leftarrow \text{Decode } IR(14-12), AR \leftarrow IR(11-0)$ |
| Indirect – Dolaylı | $D'_1 T_3 : AR \leftarrow M[AR]$ |
| Interrupt – Kesme | $(T_0 + T_1 + T_2)' \cdot IEN \cdot (FGI + FGO) : R \rightarrow 1$ $RT_0 : AR \leftarrow 0, TR \leftarrow PC$ $RT_1 : M[AR] \leftarrow TR, PC \leftarrow 0$ $RT_2 : PC \leftarrow PC + 1, IEN \leftarrow 0, R \leftarrow 0, SC \leftarrow 0$ |
| Memory Reference Instructions – Bellek Adreslemeli Buyruklar | |
| AND | $D_0 T_4 : DR \leftarrow M[AR]$ $D_0 T_5 : AC \leftarrow AC \wedge DR, SC \leftarrow 0$ |
| ADD | $D_1 T_4 : DR \leftarrow M[AR]$ $D_1 T_5 : AC \leftarrow AC + DR, E \leftarrow C_{out}, SC \leftarrow 0$ |
| LDA | $D_2 T_4 : DR \leftarrow M[AR]$ $D_2 T_5 : AC \leftarrow DR, SC \leftarrow 0$ |
| STA | $D_3 T_4 : M[AR] \leftarrow AC, SC \leftarrow 0$ |
| BUN | $D_4 T_4 : PC \leftarrow AR, SC \leftarrow 0$ |
| BSA | $D_5 T_4 : M[AR] \leftarrow PC, AR \leftarrow AR + 1$ $D_5 T_5 : PC \leftarrow AR, SC \leftarrow 0$ |
| ISZ | $D_6 T_4 : DR \leftarrow M[AR]$ $D_6 T_5 : DR \leftarrow DR + 1$ |
| | $D_6 T_6 : M[AR] \leftarrow DR, \text{If } DR = 0 \text{ then } PC \leftarrow PC + 1, SC \leftarrow 0$ |
| Register Reference Instructions – Yazaç Adreslemeli Buyruklar | |
| $D_7 I' T_3 = r, \quad IR(i) = B_i (i=0, 1, 2, \dots, 11)$ | $r : SC \leftarrow 0$ |
| CLA | $r B_{11} : AC \leftarrow 0$ |
| CLE | $r B_{10} : E \leftarrow 0$ |
| CMA | $r B_9 : AC \leftarrow AC'$ |
| CME | $r B_8 : E \leftarrow E'$ |
| CIR | $r B_7 : AC \leftarrow shr AC, AC(15) \leftarrow E, E \leftarrow AC(0)$ |
| CIL | $r B_6 : AC \leftarrow shl AC, AC(0) \leftarrow E, E \leftarrow AC(15)$ |
| INC | $r B_5 : AC \leftarrow AC + 1$ |
| SPA | $r B_4 : \text{If } AC(15) = 0 \text{ then } PC \leftarrow PC + 1$ |
| SNA | $r B_3 : \text{If } AC(15) = 1 \text{ then } PC \leftarrow PC + 1$ |
| SZA | $r B_2 : \text{If } AC = 0 \text{ then } PC \leftarrow PC + 1$ |
| SZE | $r B_1 : \text{If } E = 0 \text{ then } PC \leftarrow PC + 1$ |
| HLT | $r B_0 : S \leftarrow 0$ |
| Input Output Instructions – Giriş Çıkış Buyrukları | |
| $D_7 I' T_3 = p, \quad IR(i) = B_i (i=6, 7, 8, 9, 10, 11)$ | $p : SC \leftarrow 0$ |
| INP | $p B_{11} : AC(7-0) \leftarrow INPR, FGI \leftarrow 0$ |
| OUT | $p B_{10} : OUTR \leftarrow AC(7-0), FGO \leftarrow 0$ |
| SKI | $p B_9 : \text{If } FGI = 1 \text{ then } PC \leftarrow PC + 1$ |
| SKO | $p B_8 : \text{If } FGO = 1 \text{ then } PC \leftarrow PC + 1$ |
| ION | $p B_7 : IEN \leftarrow 1$ |
| IOF | $p B_6 : IEN \leftarrow 0$ |

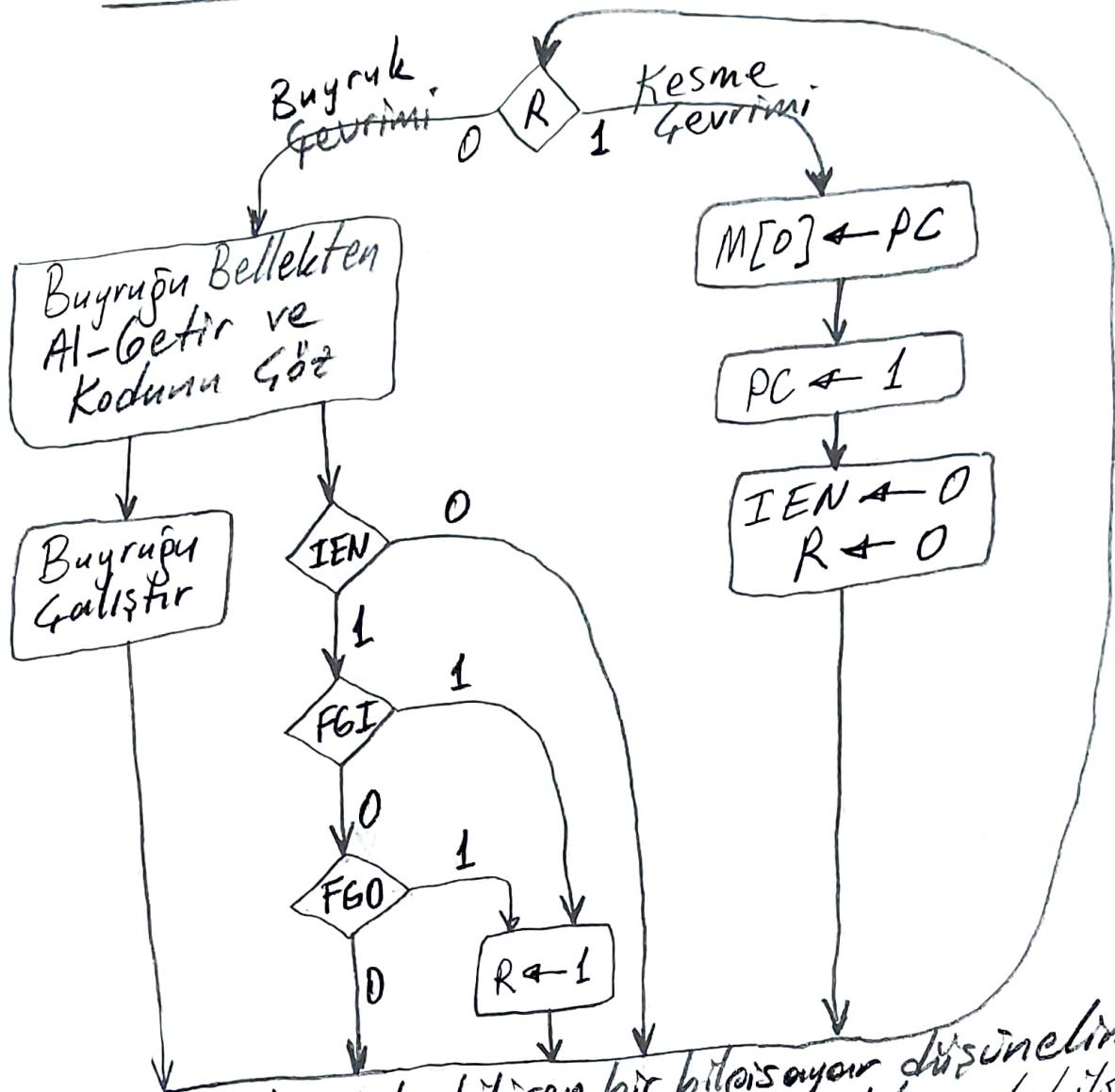
Temel Bilgisayar İşlemleri İçin Akış Şeması

49

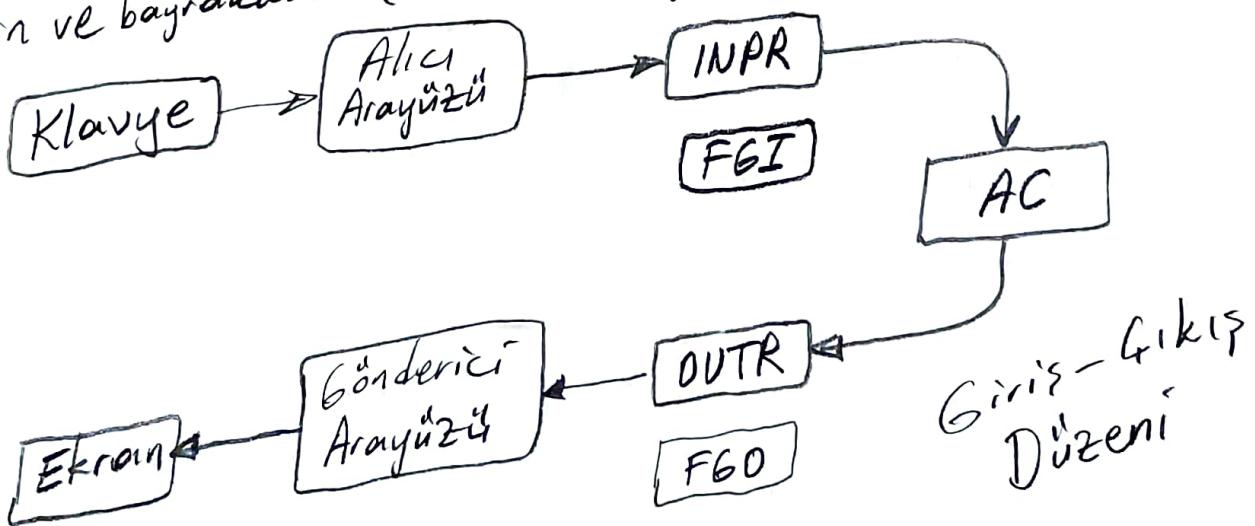


Kesme Süreci işin Akış Şeması

(50)



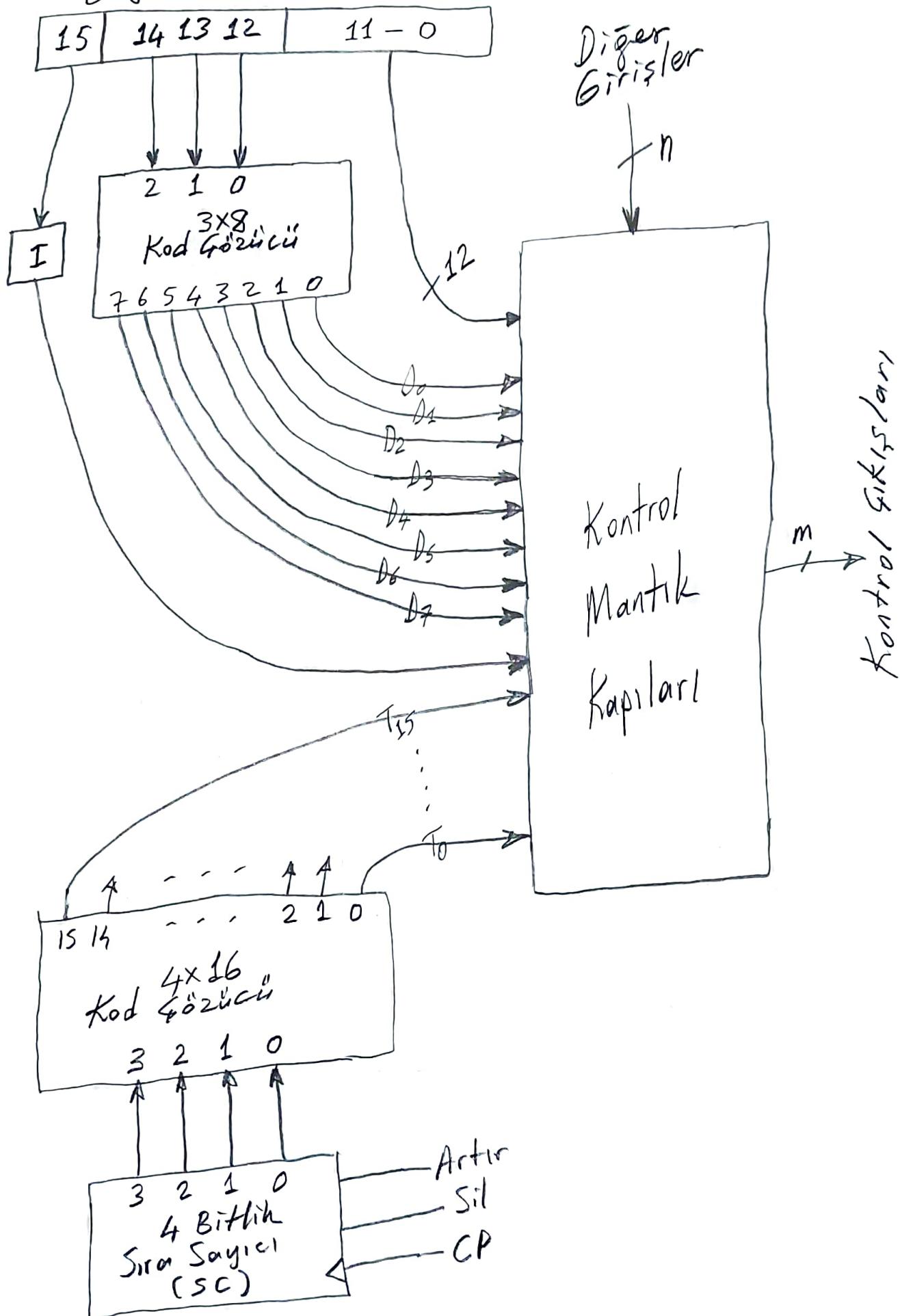
Buyruk sürecini 1 ms'de bitiren bir bilgisayor düşünelim.
 Giriş/Gökis aleti saniyede maksimum 10 karakter alabilir.
 Bu karakter başına 100.000 buyruk süreci ontamına gelir.
 Bu durumda devamlı Giriş/Gökis aletini kontrol etmek zorunlu
 tayıbdir. Bunun yerine IEN biti 1 olduğundan bayrakları kontrol
 etsin ve bayraklardan biri 1 ise R yaz-bozunu 1 yapmak daha uygundu
 lutsin



Temel Bilgisayarın Denetim Birimi

Bağrak Yazarı (IR)

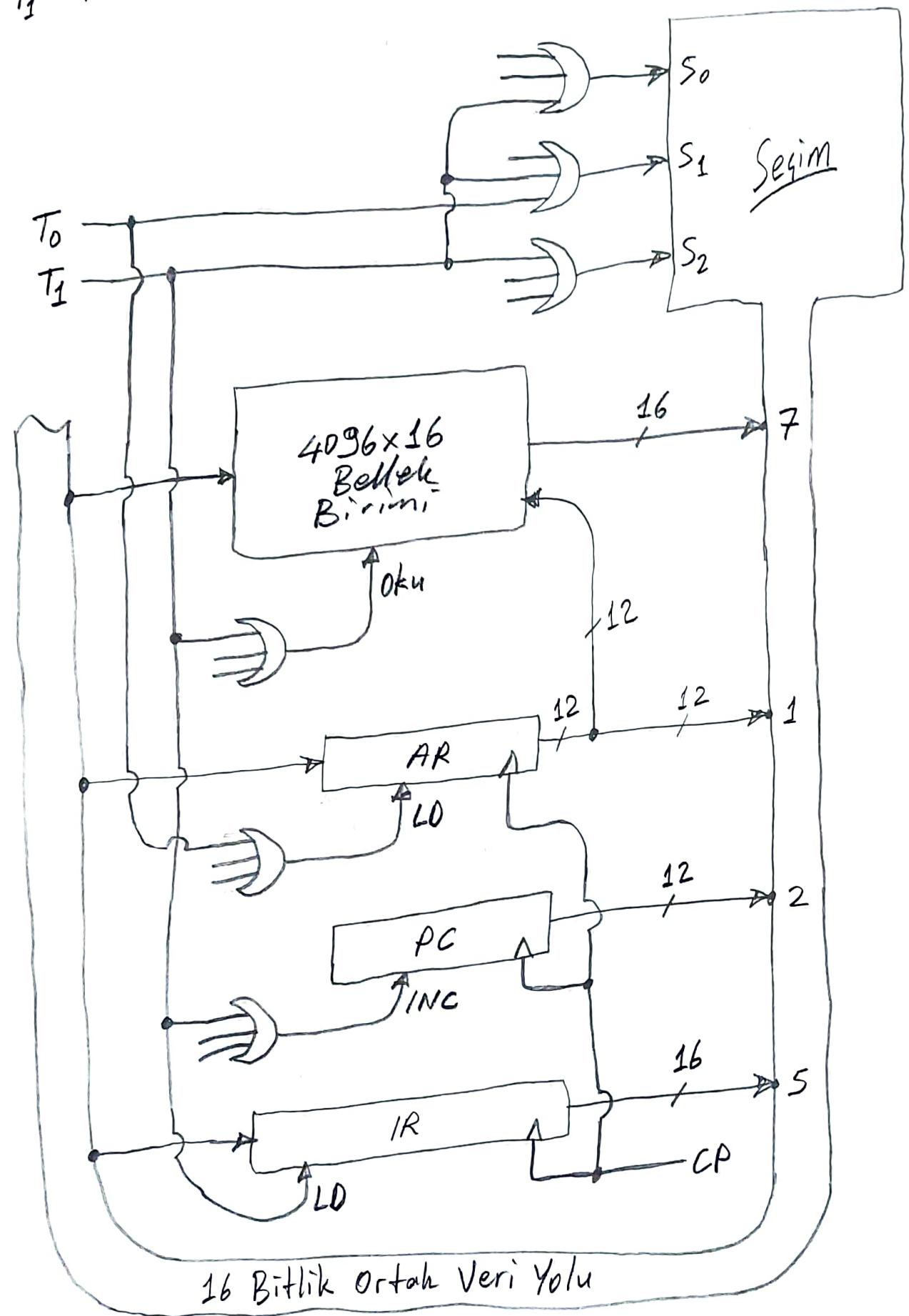
(51)



Al-Getir Evresi isin Yaras Aktarimlari

$T_0 : AR \leftarrow PC$

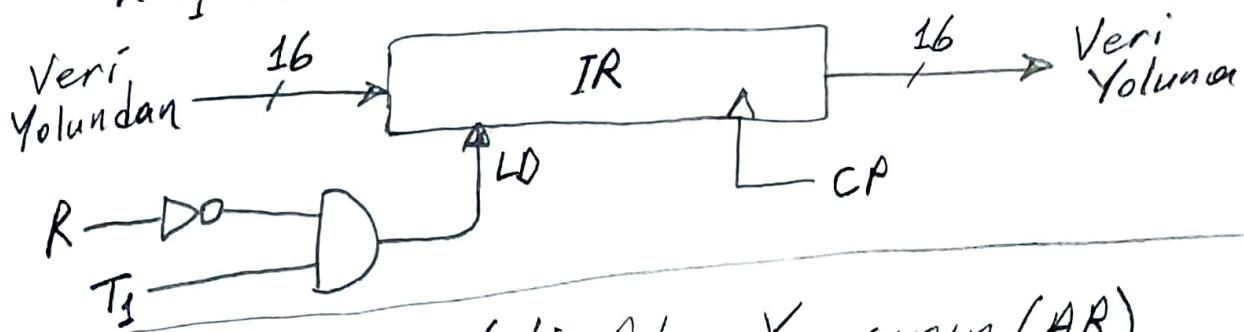
$T_1 : IR \leftarrow M[AR], PC \leftarrow PC + 1$



16 Bitlik Ortak Veri Yolu

Temel Bilgisayardaki Buyruk Yazarinin (IR)
LD denetimi işin kapı tasarımını siziniz.

$$R'T_1 : IR \leftarrow M[AR]$$



Temel Bilgisayardaki Adres Yazarinin (AR)
LD, INC, CLR denetimi işin kapı tasarımını siziniz.

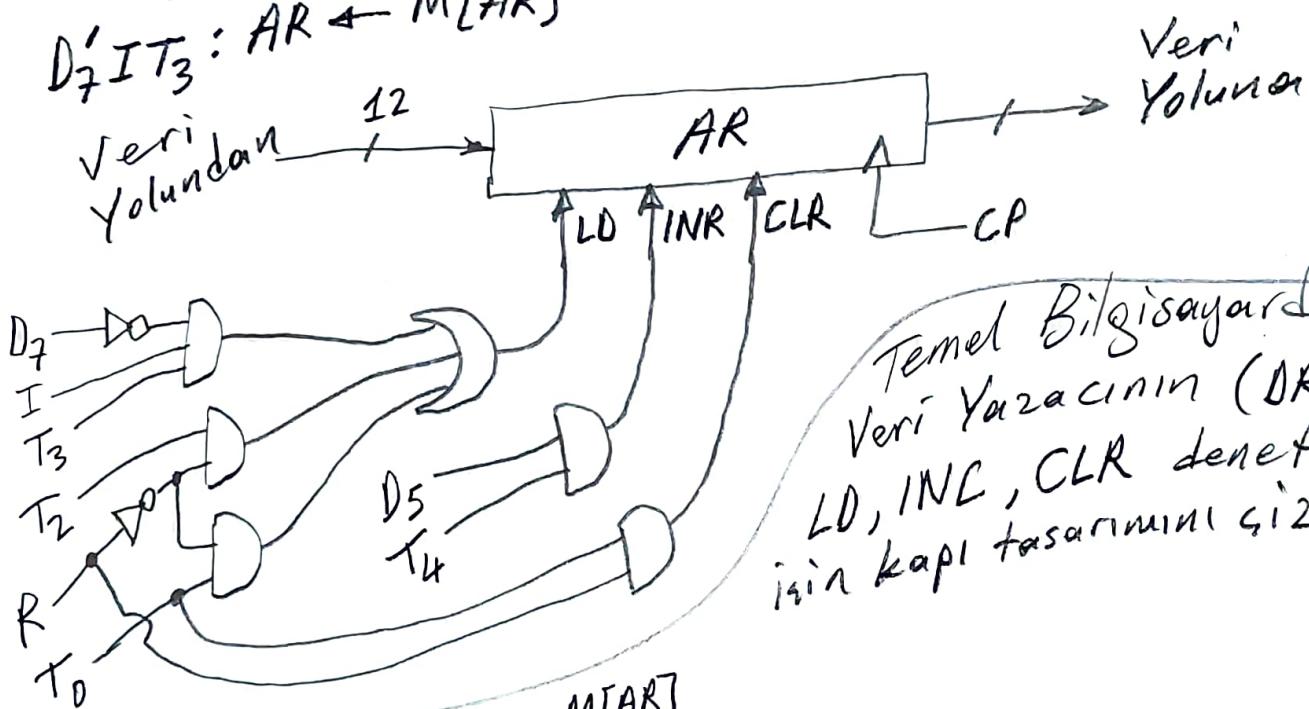
$$R'T_0 : AR \leftarrow PC$$

$$D_5 T_4 : AR \leftarrow AR + 1$$

$$R'T_2 : AR \leftarrow IR(11-0)$$

$$RT_0 : AR \leftarrow 0$$

$$D_7 IT_3 : AR \leftarrow M[AR]$$

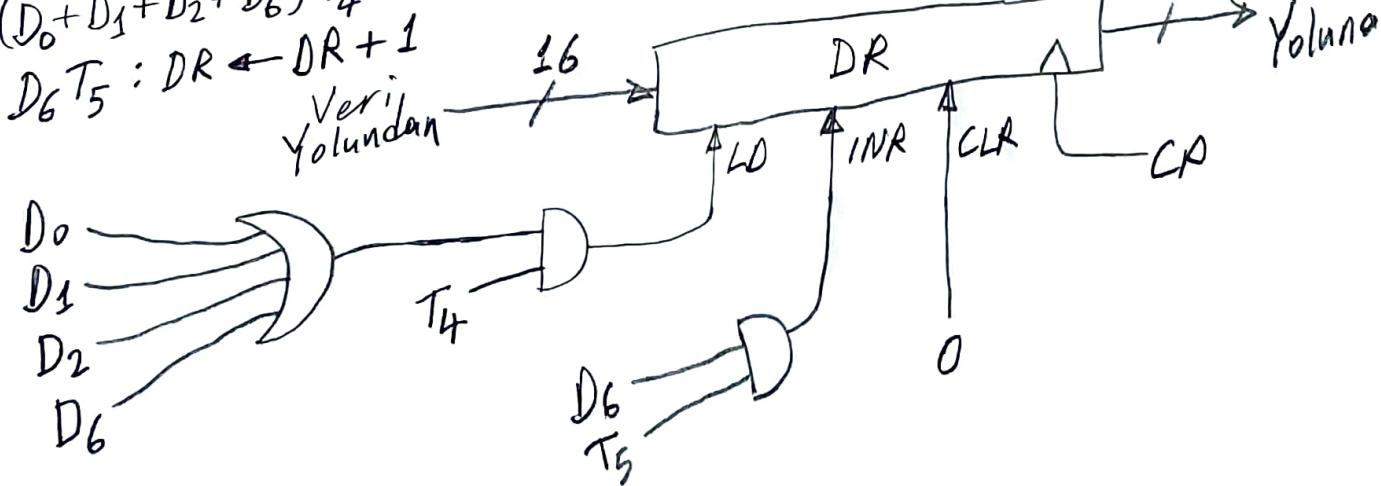


Temel Bilgisayardaki
Veri Yazarinin (DR)
LD, INC, CLR denetimi
işin kapı tasarımını siziniz.

$$(D_0 + D_1 + D_2 + D_6) T_4 : DR \leftarrow M[AR]$$

$$D_6 T_5 : DR \leftarrow DR + 1$$

$$Veri Yolundan$$



Temel Bilgisayardaki belleğin ve yaracağıları ortaak
veri yolunda bağlayan kodlayıcının girişlerini.
 $(x_1, x_2, x_3, x_4, x_5, x_6, x_7)$ ve çıkışlarına (S_0, S_1, S_2)
bulunuz.

$$\begin{array}{l} D_4 T_4 : PC \leftarrow AR \\ D_5 T_5 : PC \leftarrow AR \end{array} \quad \left. \begin{array}{l} \\ \end{array} \right\} x_1 = D_4 T_4 + D_5 T_5$$

$$\begin{array}{l} R'T_0 : AC \leftarrow PC \\ R T_0 : TR \leftarrow PC \end{array} \quad \left. \begin{array}{l} \\ \end{array} \right\} x_2 = T_0 + D_5 T_4$$

$$D_5 T_4 : M[AR] \leftarrow PC$$

$$\begin{array}{l} D_2 T_5 : AC \leftarrow DR \\ D_6 T_6 : M[AR] \leftarrow DR \end{array} \quad \left. \begin{array}{l} \\ \end{array} \right\} x_3 = D_2 T_5 + D_6 T_6$$

$$\begin{array}{l} D_3 T_4 : M[AR] \leftarrow AC \\ pB_{10} : DUTR \leftarrow AC(7-0) \end{array} \quad \left. \begin{array}{l} \\ \end{array} \right\} x_4 = D_3 T_4 + pB_{10}$$

$$p = D_7 IT_3, B_{10} = IR(10)$$

$$R'T_2 : I \leftarrow IR(15), AR \leftarrow IR(11-0) \quad \left. \begin{array}{l} \\ \end{array} \right\} x_5 = R'T_2$$

$$RT_1 : M[AR] \leftarrow TR \quad \left. \begin{array}{l} \\ \end{array} \right\} x_6 = RT_1$$

$$R'T_1 : IR \leftarrow M[AR]$$

$$D_7' IT_3 : AR \leftarrow M[AR]$$

$$(D_0 + D_1 + D_2 + D_6) T_4 : DR \leftarrow M[AR]$$

$$\left. \begin{array}{l} x_7 = R'T_1 + D_7' IT_3 \\ + (D_0 + D_1 + D_2 + D_6) T_4 \end{array} \right\}$$

| X | S_2 | S_1 | S_0 |
|-------|-------|-------|-------|
| x_1 | 0 | 0 | 1 |
| x_2 | 0 | 1 | 0 |
| x_3 | 0 | 1 | 1 |
| x_4 | 1 | 0 | 0 |
| x_5 | 1 | 0 | 1 |
| x_6 | 1 | 1 | 0 |
| x_7 | 1 | 1 | 1 |

$$S_0 = x_1 + x_3 + x_5 + x_7$$

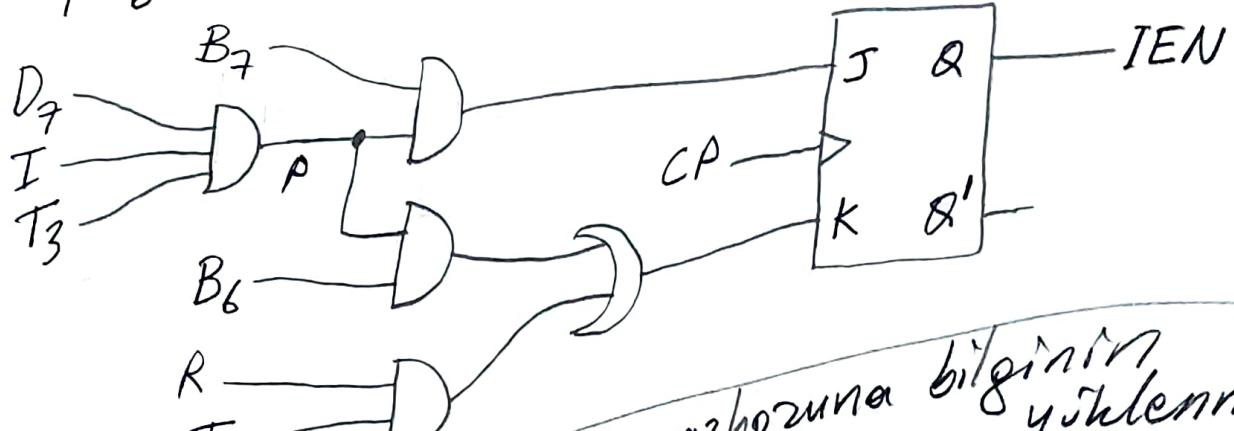
$$S_1 = x_2 + x_3 + x_6 + x_7$$

$$S_2 = x_4 + x_5 + x_6 + x_7$$

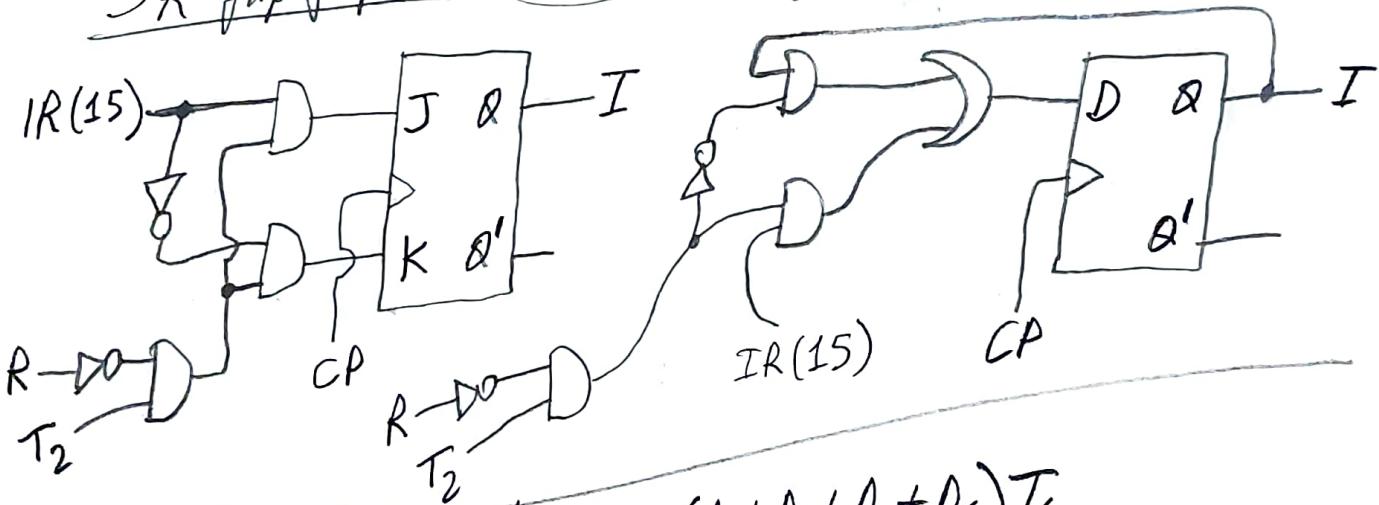
| Sembol | İşlem Kodu | Sembolik Gösterimi |
|--|------------|---|
| OR | 000 | $AC \leftarrow AC \vee M[AR]$ |
| SUB | 001 | $AC \leftarrow AC - M[AR]$ |
| XCH | 010 | $AC \leftarrow M[AR], M[AR] \leftarrow AC$ |
| ADM | 011 | $M[AR] \leftarrow M[AR] + AC$ |
| <p>Temel Bilgisayardaki bazı buyruklara yukarıdaki buyruklar ile değiştirelim. ALU devresinde herhangi bir değişiklik yapılmıyor. Her bir komut işin T_4 antından başlayarak gerekli mikro işlemleri sırasıyla yazınız.</p> | | |
| <u>OR</u> | $D_0 T_4$ | $DR \leftarrow M[AR], AC \leftarrow \overline{AC}$ |
| | $D_0 T_5$ | $DR \leftarrow AC, AC \leftarrow DR$ |
| | $D_0 T_6$ | $AC \leftarrow \overline{AC}$ |
| | $D_0 T_7$ | $AC \leftarrow AC \wedge DR$ |
| | $D_0 T_8$ | $AC \leftarrow \overline{AC}, SC \leftarrow 0$ |
| <u>SUB</u> | $D_1 T_4$ | $DR \leftarrow M[AR]$ |
| | $D_1 T_5$ | $DR \leftarrow AC, AC \leftarrow DR$ |
| | $D_1 T_6$ | $AC \leftarrow \overline{AC}$ |
| | $D_1 T_7$ | $AC \leftarrow AC + 1$ |
| | $D_1 T_8$ | $AC \leftarrow AC + DR, E \leftarrow C_{out}, SC \leftarrow 0$ |
| <u>XCH</u> | $D_2 T_4$ | $DR \leftarrow M[AR]$ |
| | $D_2 T_5$ | $M[AR] \leftarrow AC, AC \leftarrow DR, SC \leftarrow 0$ |
| <u>ADM</u> | $D_3 T_4$ | $DR \leftarrow M[AR]$ |
| | $D_3 T_5$ | $DR \leftarrow AC, AC \leftarrow AC + DR, E \leftarrow C_{out}$ |
| | $D_3 T_6$ | $M[AR] \leftarrow AC, AC \leftarrow DR, SC \leftarrow 0$ |

$$AVB = (A' \wedge B')$$

$$A \neg B = A + \overline{B} + 1$$

IEN'in Denetim Girişleri $PB_7 : IEN \leftarrow 1$ $RT_2 : IEN \leftarrow 0$ $PB_6 : IEN \leftarrow 0$ 

I yarbozuna bilginin yüklenmesi
JK flip-flop ile $R'T_2 : I \leftarrow IR(15)$ D flip-flop ile



$$\text{Bellek (Oku)} = R'T_1 + D_7 IT_3 + (D_0 + D_1 + D_2 + D_6) T_4$$

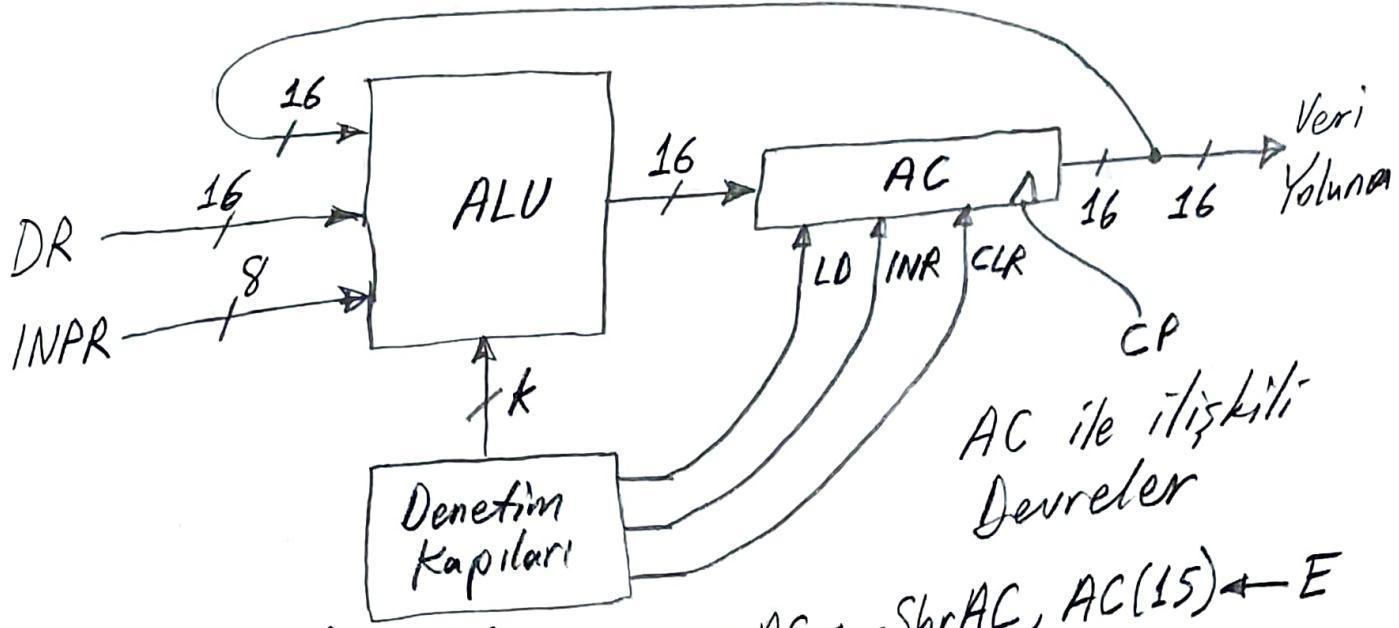
$$\text{Bellek (Yaz)} = RT_1 + (D_3 + D_5) T_4 + D_6 T_6$$

$$\begin{aligned} SC(CLR) = & RT_2 + (D_0 + D_1 + D_2 + D_5) T_5 + (D_3 + D_4) T_4 \\ & + D_6 T_6 + p + r \end{aligned}$$

$$p + r = D_7 IT_3 + D_7 I'T_3 = D_7 T_3 (I + I') = D_7 T_3$$

$$\begin{aligned} PC(INR) = & R'T_1 + RT_2 + D_6 T_6 Z_{DR} + r B_4 AC(15)' + r B_3 AC(15) \\ & + r B_2 Z_{AC} + r B_1 E' + p B_9 FGI + p B_8 F6O \end{aligned}$$

$$r = D_7 I'T_3, p = D_7 IT_3, B_i = IR(i)$$



$D_0 T_5 : AC \leftarrow AC \wedge DR$

$D_1 T_5 : AC \leftarrow AC + DR$

$D_2 T_5 : AC \leftarrow DR$

$P B_{11} : AC(7-0) \leftarrow INPR$

$r B_g : AC \leftarrow \overline{AC}$

ALU'den Gelir

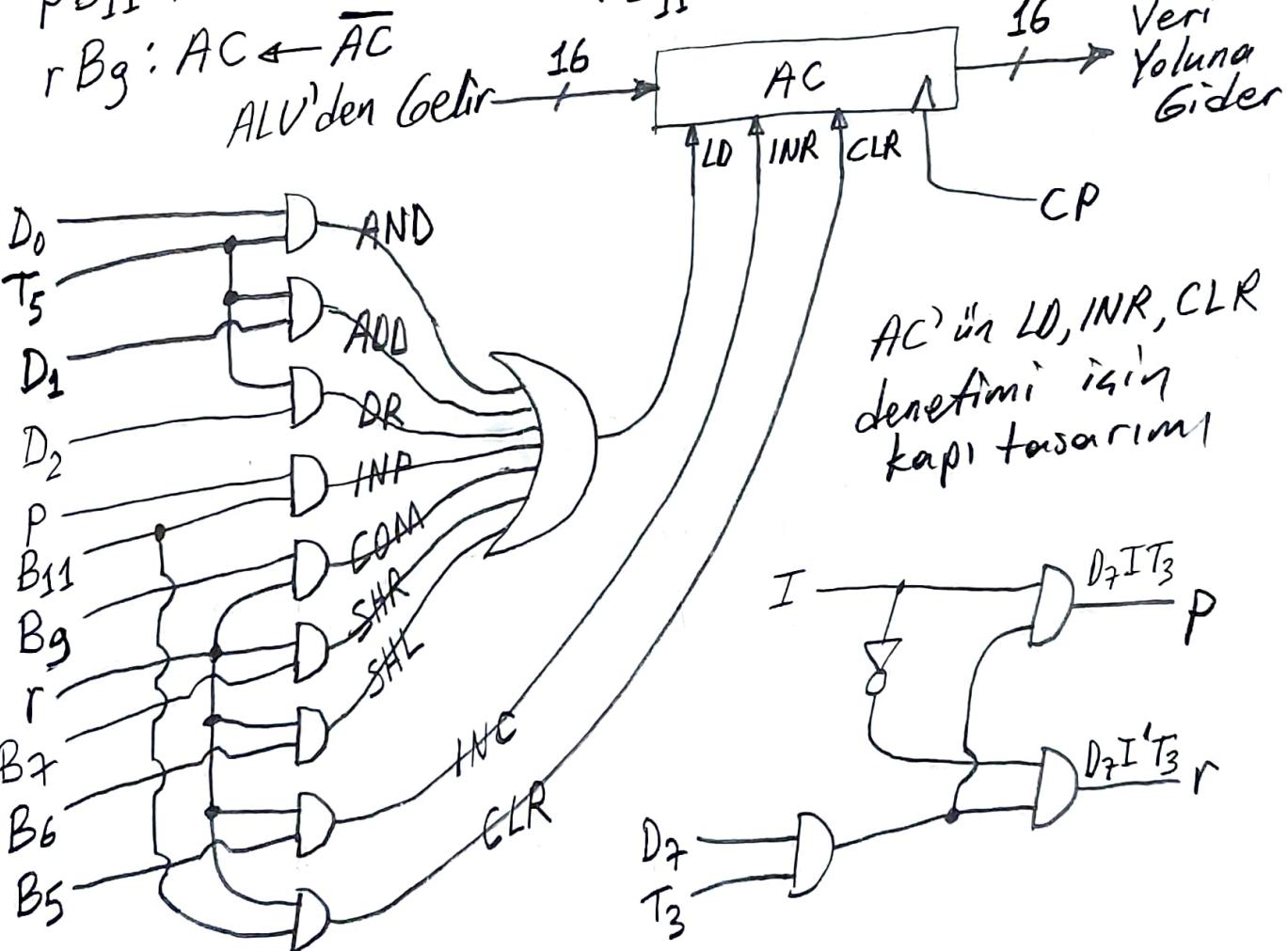
$r B_7 : AC \leftarrow ShrAC, AC(15) \leftarrow E$

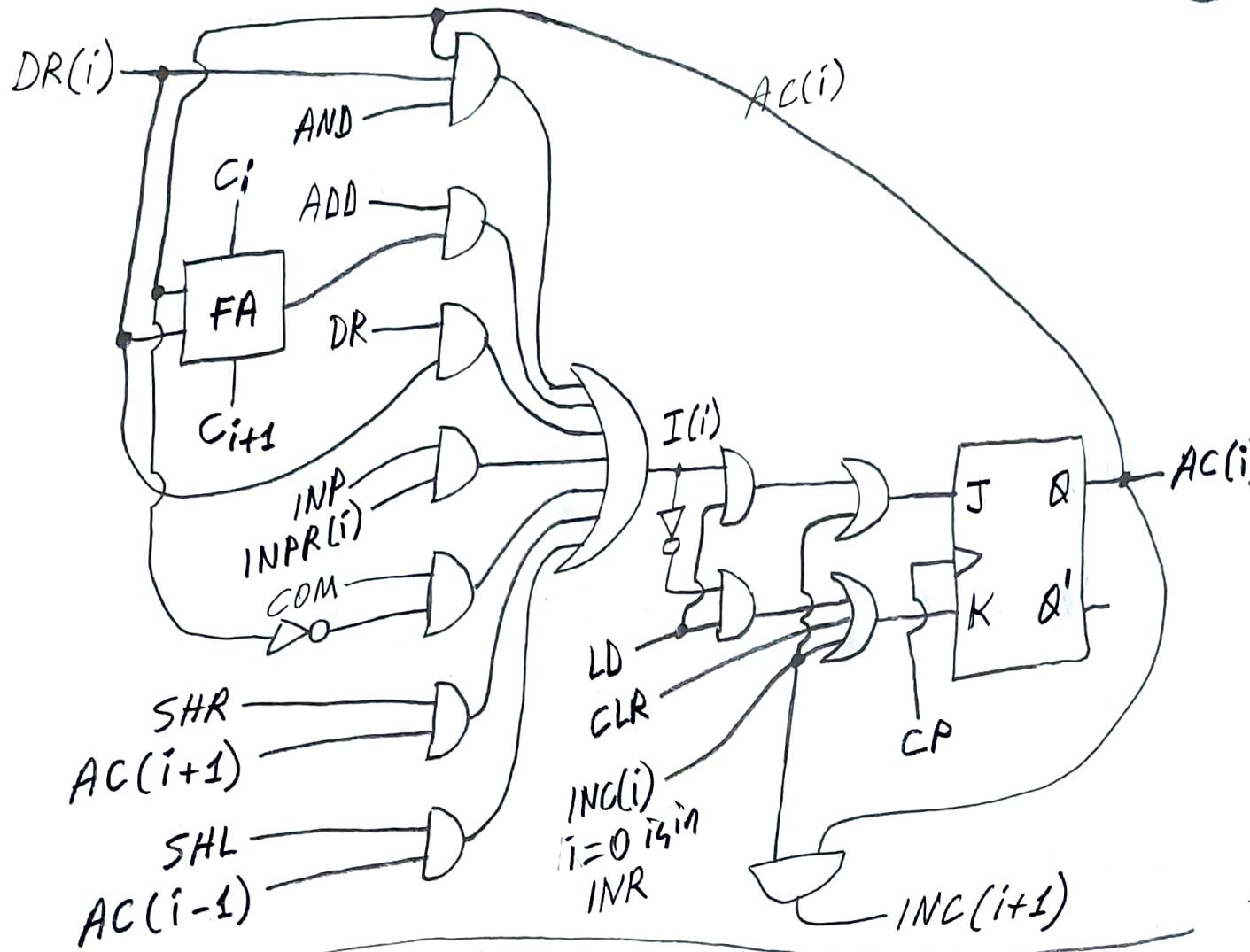
$r B_6 : AC \leftarrow ShlAC, AC(0) \leftarrow E$

$r B_5 : AC \leftarrow AC + 1$

$r B_{11} : AC \leftarrow 0$

AC ile ilişkili
Devreler





$$r = D_7 I' T_3$$

$$r B_6 : E \leftarrow AC(15)$$

$$r B_7 : E \leftarrow AC(0)$$

$$r B_8 : E \leftarrow E'$$

$$r B_{10} : E \leftarrow 0$$

$$\begin{array}{|c|c|} \hline J & K \\ \hline 0 & 0 \\ \hline \end{array} \quad \begin{array}{c} \delta_{t+1} \\ \delta_t \\ \hline 0 & 0 \\ 0 & 1 \\ 1 & 0 \\ 1 & 1 \\ \hline \end{array}$$

$$\begin{array}{|c|c|} \hline 0 & 0 \\ \hline \end{array} \quad \begin{array}{c} \delta_t \\ \delta'_t \\ \hline 0 \\ 1 \\ 1 \\ \hline \end{array}$$

$$\begin{array}{|c|c|} \hline 0 & 1 \\ \hline \end{array} \quad \begin{array}{c} \delta_t \\ \delta'_t \\ \hline 0 \\ 1 \\ 1 \\ \hline \end{array}$$

$$\begin{array}{|c|c|} \hline 1 & 0 \\ \hline \end{array} \quad \begin{array}{c} \delta_t \\ \delta'_t \\ \hline 1 \\ 0 \\ 1 \\ \hline \end{array}$$

$$\begin{array}{|c|c|} \hline 1 & 1 \\ \hline \end{array} \quad \begin{array}{c} \delta_t \\ \delta'_t \\ \hline 1 \\ 1 \\ 1 \\ \hline \end{array}$$

Temel Bilgisayardaki
E yarbozunu tasarılmış

