

## Z80'DE PARALEL GİRİŞ/ÇIKIŞ

Aykut DALGIÇ  
Elektrik Mühendisliği Bölümü  
Orta Doğu Teknik Üniversitesi

### GİRİŞ

Konunun daha iyi izlenebilmesi amacıyla önce Z80 hakkında (onun bacak gösterimi, kayıtlayın yapısı, bayrak yapısı, komut kümesi ve adresleme biçimleri hakkında) kısa bir özet yapılmıştır. Daha sonra Z80 komut kümesinin giriş ve çıkış grubu komutları incelenmiş ve bunlarla

ilgili kısa program örneklerine yer verilmiştir. Son olarak 11 Z80 Paralel Giriş /Çıkış Devresi (PIO) ele alınmış ve onun programlanması kısaca ve örneklerle gözden geçirilmiştir.

### A.1. Z80

Z80, Intel 8080'nin çok geliştirilmiş bir modelidir. Intel 8080 makina dilinde yazılmış herhangi bir program Z80'de de işlenebilir. Z80, 8080 gibi 8 bitlik bir mikroislemcidir. Adresleyebilme kapasitesi  $2^{16} = 64$  K Bayt'tır. Z80'de, 8080'in aritmetik işlemlerdeki eksikliklerini karşılamak üzere fazladan komutlar bulunmaktadır. İki bağımsız indis kayıtlayıcısı vardır. Bunlar Z80'nin adresleme yeteneğini arttırmaktadır.

Z80 donanımsal açıdan da 8080'den ileri düzeydedir. Örneğin 8080'de üç ayrı güç kaynağının kullanılmasına karşın Z80'de tek güç kaynağı (+5V) kullanılmaktadır.

Z80'de kesme olarak, maskelenebilen kesme girişi  $\overline{INT}$  ve maskelenemeyen kesme girişi  $\overline{NMI}$  bulunmaktadır.

Şekil 1'de tek bir tümleşik devreden oluşan Z80'nin bacak gösterimi görülmektedir.

Z80'de 18 tane 8 bitlik ve 4 tane 16 bitlik kayıtlayıcı vardır. Birikeç (akümülatör) ve bayrak (flag) kayıtlayıcıları ikişer tanedir. Z80'deki kayıtlayıcılar, Şekil 2'deki Z80'nin basit genel yapısında da gösterildiği gibi, aşağıdaki kayıtlayıcılardan oluşmaktadır:

Ana Kayıtlayıcılar Kümesi: A (Birikeç), F (Bayrak), B, C, D, E, H ve L.

Seçenek Kayıtlayıcıları Kümesi: A' (Birikeç), F' (Bayrak), B \ C \ D', E', H' ve L \

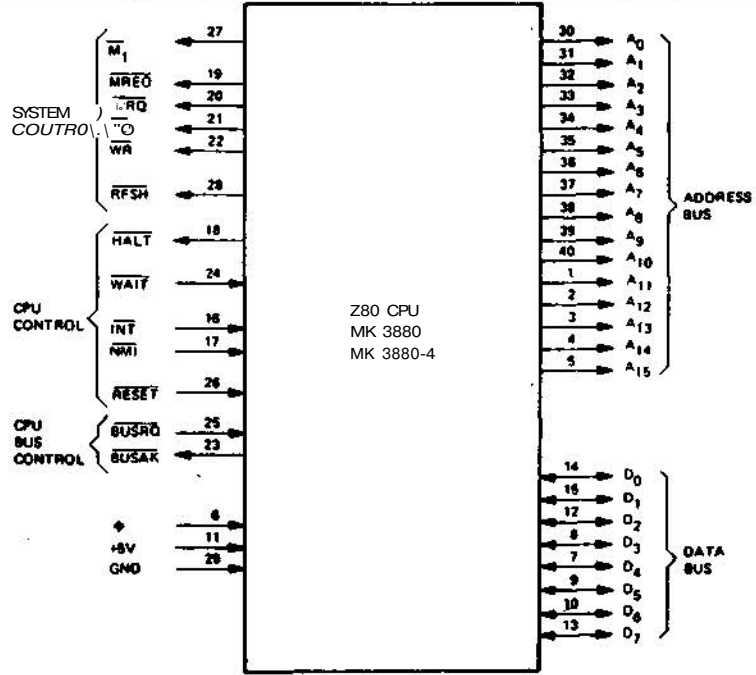
Genel Amaçlı Kayıtlayıcılar: Indis Kayıtlayıcıları IX ve IY, Yiğit Göstericisi (SP), Program Sayıcısı (PC), Kesme Vektörü (I), Bellek Yenileme (R).

Herhangi bir anda, Ana Kayıtlayıcılar ve Seçenek Kayıtlayıcıları kümelerinin sadece birindeki genel amaçlı kayıtlayıcılar kullanılmaktadır. Bir değiştirme komutuyla Ana ya da Seçenek Kayıtlayıcı Kümesinin hangisinin kullanılacağı belirlenebilir.

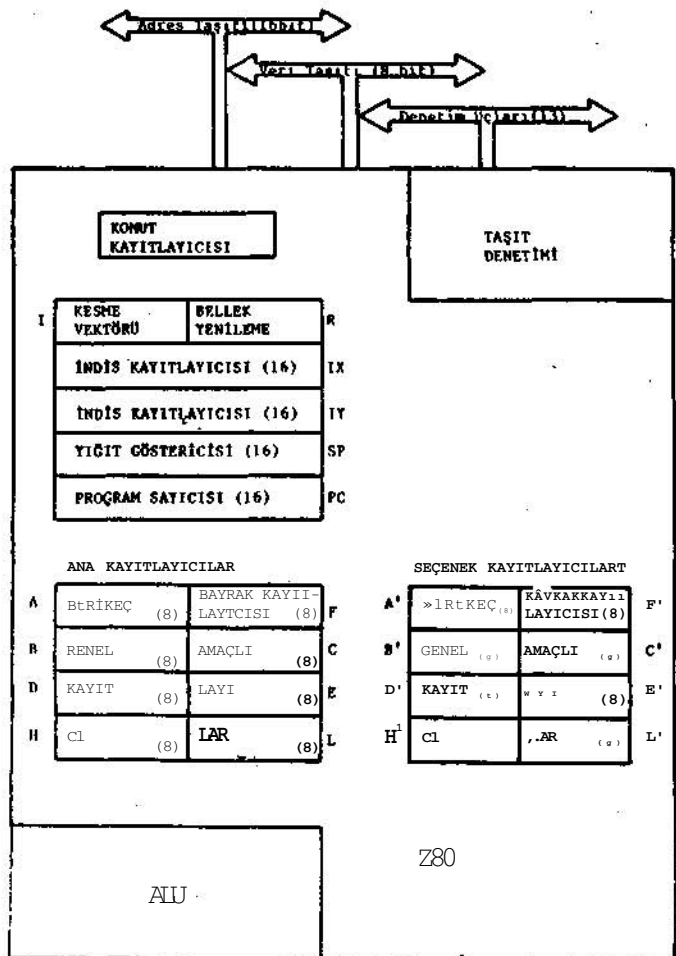
Şekil 3'te Z80'nin işlevsel blok şeması gösterilmektedir.

Z80'nin Aritmetik - Mantık Biriminde (ALU'da) CPU'nun 8 bitlik aritmetik ve mantıksal komutları işlenmektedir. ALU tarafından gerçekleştirilebilen işlemler şunlardır :

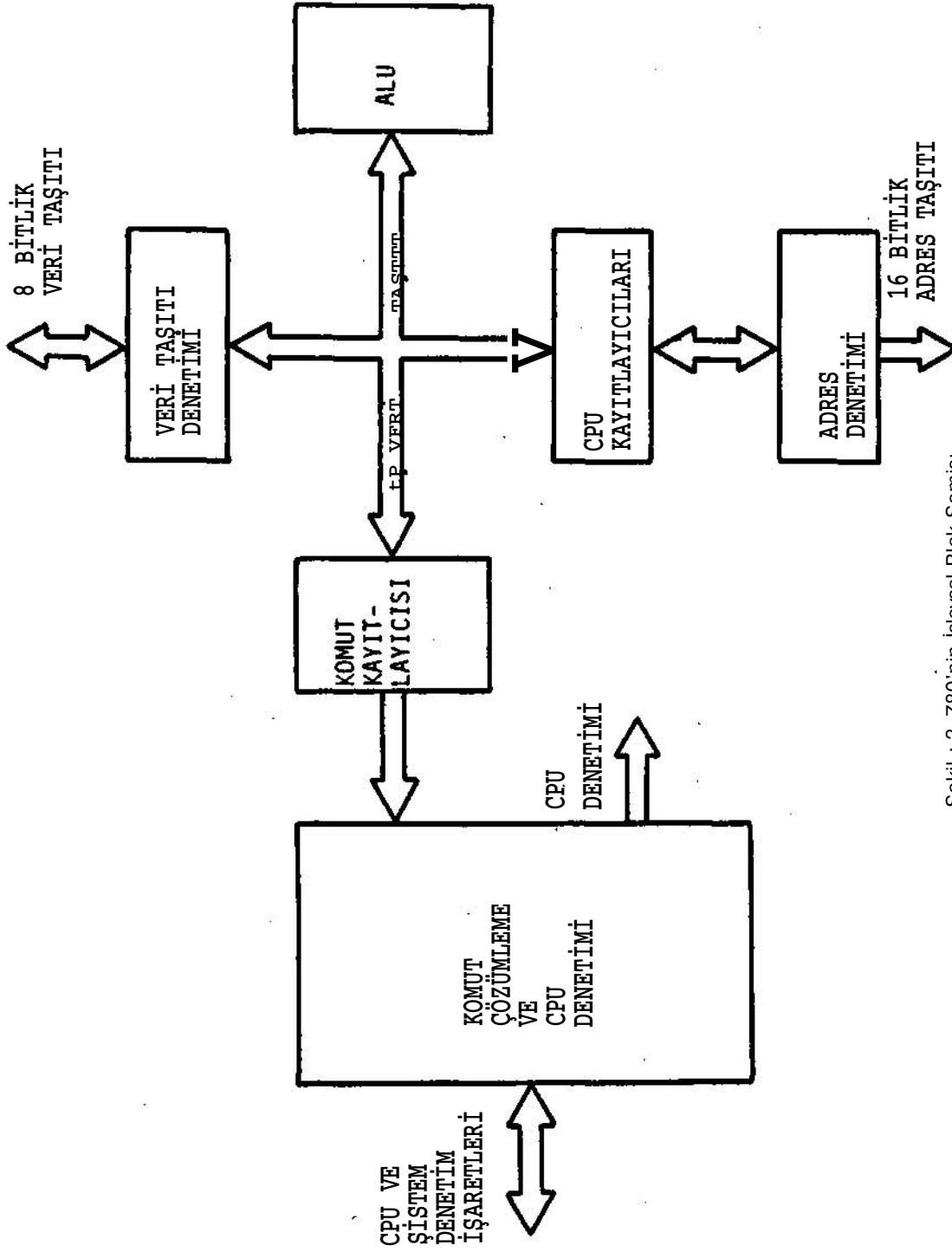
- Toplama
- Çıkarma
- Mantıksal VE (AND)
- Mantıksal VEYA (OR)



Şekil : 1 Z80'ün Üçlü Gösterimi



Şekil : 2 Z80'nin Basit Geni Yapısı



Şekil : 3 Z80'nin İşlevsel Blok Şemisi

## Mantıksal Dışlayan Veya (EXCLUSIVE-OR) Karşılaştırma

Sağa veya sola kaydırma ve döndürme (aritmetik ve mantıksal)

Arttırma

A/altma

Biti birleme

Biti sıfırlama

- Biti test etme

Z80'in Bayrak Yapısı :

Z80'in her iki Bayrak Kayıtlayıcısı, çeşitli CPU işlemleriyle değiştirilebilen (sıfırlanan ya da birlenen) 6 bitlik bir bilgi taşırlar. Bunların 4 tanesi test edilebilir, yani bunlar atlama (Jump), çağırma (Call) ve geri dönme (Return) komutları için birer koşul olarak kullanılabilirler. Bu 4 bit şunlardır :

1) Elde Bayrağı (C): Bu, birikecin (akümülatörün) en önemli bitinden olan eldeyi gösterir. Örneğin bir toplama işlemi sırasında birikecin en önemli bitinden bir elde yaratılırsa o zaman Elde Bayrağı 1 olur.

2) Sıfır Bayrağı (Z): Yapılan işlemin sonucu sıfırsa bu bayrak 1 olur. Aksi takdirde 0'dir.

3) İşaret Bayrağı (S): Bu bayrağın işaretlenmiş sayılarla kullanılması amaçlanmış olup işlemin sonucu eksi ise işaret bayrağı 1, artı ise 0 olmaktadır.

4) Eşlik/Taşma Bayrağı (P/V): Bu iki amaçlı bayrak, birikeçte mantıksal işlemler yapıldığı zaman sonucun eşliğini belirtir. Eğer işaretlenmiş sayılarla 2'nin tümleyenine göre aritmetik işlemler yapılıyorsa, o zaman bu bayrak taşmayı belirtir.

Örneğin ikili (binary) sayı 11101110'nin eşliği = 1, 11101100'nin ise 0'dir. Eğer ikili sayıdaki 1'lerin toplamı çift ise Eşlik 1, tek ise Eşlik 0 olmaktadır.

Taşma ile ilgili olarak ta aşağıdaki örnek ele alınabilir.

$$+ 120 = 0111 1000$$

$$+ 104 = 0110 1000$$

elde(C) = 0 1110 0000 = - 32 (+224 bu yedi hane için olan maksimumu yani + 127'yi geçtiği için taşma var.)

Yukarıdaki örnekten de görüldüğü gibi, işaretlenmiş sayılarla 2'nin tümleyenine göre yapılan aritmetik işlemlerde, taşma elde (C) yaratılmadan da meydana gelebilmektedir.

Bu saydığımız 4 test edilebilir bayrak dışında test edilemeyen 2 bayrak daha vardır. Onların ikisi de BCD aritmetiğinde kullanılırlar. Bunlar : 1) Yarım Elde Bayrağı (H) ve 2) Topla/Çıkar Bayrağı (N)'dir.

8 bitlik olan Bayrak Kayıtlayıcısında, bu saydığımız bayrakların dağılımı şu şekildedir.

7 6 5 4 3 2 1 0

[S | Z | ~X | J | H | X | P/V | "N" | C ]

(X belirsiz anlamında kullanılmıştır.)

## Z80'in Komut Kümesi

Z80, 8080A'nın 78 değişik çeşitteki komutu dahil olmak üzere 158 değişik çeşitteki komutu işleyebilmektedir. Z80'nin işleyebildiği bu 158 değişik çeşitteki komutlar aşağıdaki gibi gruplandırılabilir :

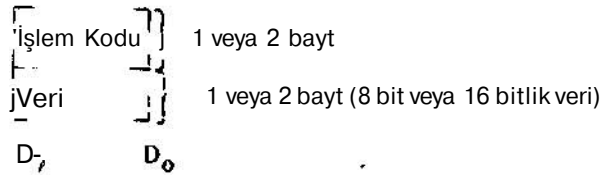
- Yükleme ve Değiştirme Grubu
- Blok Aktarma ve Arama Grubu
- Aritmetik ve Mantıksal İşlemler Grubu
- Döndürme ve Kaydırma Grubu
- Bit Düzenlemesi Grubu
- Dallarına Grubu (Atlama, Çağırma ve Geri Dönme)
- Giriş/Çıkış Grubu
- Temel CPU Denetimi Grubu

Z80 komutları bir, iki, üç veya dört bayt uzunlukta olabilmektedir. Birden fazla bayttan oluşan komutlar ard arda gelen bellek yerlerinde depolanmaktadır.

## Z80'de Adresleme Biçimleri

Z80 komutlarının çoğu, CPU'nun iç kayıtlayıcılarında, bellekte veya giriş/çıkış portlarında olan veri üzerinde işlemlerde bulunurlar. Adresleme, komut içinde verinin adresinin nasıl belirlendiğini belirtir. Z80'de kullanılan adresleme biçimleri aşağıda verilmektedir :

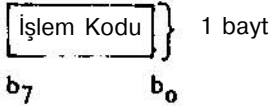
- hemen Adresleme : Bu tür adreslemede, veri komutun içindedir. Yani bellekte işlem kodunu takip eden bayt ve baytlar veri olarak alınır.



Örnek : LD B, FFH\* (8 bitlik veri)

LDBC. FFFFH (16 bitlik veri)

- Değiştirilmiş Sayfa Sıfır Adreslemesi : Z80, bir baytlık özel bir çağırma komutuna sahiptir. Bu çağırma komutunun adresi belleğin sayfa sıfırdaki 8 yerinden biri olabilmektedir.



(çağırmanın yapılacağı adres 00b<sub>5</sub>b<sub>4</sub>b<sub>3</sub>000'dir)

Örnek : RST p (p şu değerlerden birisi olabilir : 00H, 08H, 10H, 18H, 20H, 28H, 30H, 38H.)

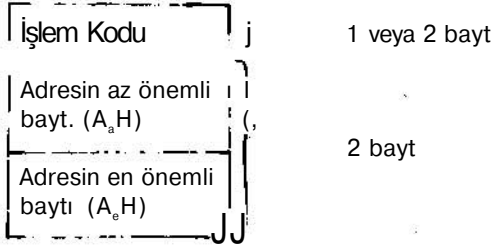
— Bağımlı (Göreceli Adresleme) Bu tür adreslemede işlem kodundan sonraki 1 baytlık veri, atlamanın o andaki

\* H verilen sayının 16'lık sayı sistemine göre olduğunu belirtir. B verilen sayının 2'li sayı sistemi ne (binary) göre olduğunu belirtir.

programın bulunduğu yerden ne kadar uzağa yapılacağı-  
nı bildirir. İşaretlenmiş 2'nin tümleyeni şeklinde olan bu  
verinin A+2'ye ilave edilmesiyle atlamanın adresi bulu-  
nur. (A, işlem kodunun bulunduğu bellek yerinin adresi-  
dir.)

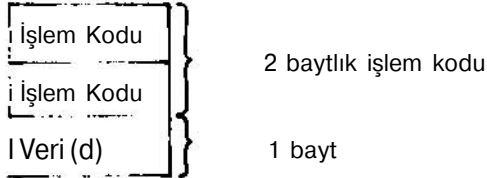


Örnek : JR e (e işaretlenmiş 2'nin tümleyeni şeklindedir.)  
- Doğrudan Adresleme : Bu tür adreslemede, işlem ko-  
dundan sonra gelen 2 bayt adresin kendisini verir.



(Adres: A<sub>0</sub>A<sub>1</sub>H)

Örnek : JP nn (nn, OH ile FFFFH arasında bir sayıdır.)  
- İndisli Adresleme : Bu tür adreslemede, işlem kodunu  
takip eden 1 baytlık verinin indis kayıtlıcaısmm (IX  
veya IY) içeriğine ilâve edilmesiyle bellekteki yeni adres  
bulunmuş olur. Veri, işaretlenmiş 2'nin tümleyeni şek-  
lindedir. İndis kayıtlıcaısının hangisi olduğu işlem kodu  
tarafından belirlenir.



örnek : ADD A, (IX + d)

- Kayıtlıcaı Adresleme : Bu tür adreslemede komut,  
verinin yerleştirilmiş olduğu kayıtlıcaı ya da kayıtlıcaı  
çiftini belirler.

Örnek : LD D, E

- İşleneni İçinde Olan Adresleme (Implied Addressing):  
Bu tür adreslemede işlem kodu, verinin veya veriden biri-  
nin hangi CPU kayıtlıcaısında bulunduğunu belirtir.

örnek : SUB B

- Kayıtlıcaı Dolaylı Adresleme : Bu tür adreslemede  
komut, verinin bellekte yerleştirilmiş olduğu yerin adre-  
sinin bulunduğu kayıtlıcaı çiftini belirler.

Örnek : LD A, (BC)

- Bit Adresleme : Bu tür adreslemede komutlar, herhan-  
gi bir bellek yerinin veya CPU kayıtlıcaısmm belirlen-

mesini kayıtlıcaı adresleme, kayıtlıcaı dolaylı adresle-  
me veya indisli adreslemeyle, bu bulunan 8 bitten birinin  
belirlenmesini de işlem kodunda belirlenen 3 bitle yapar-  
lar.

Örnek : SET b, r (r A, B, C, D, E, H, L kayıtlıcaıların-  
dan birini, b'de bitin bu kayıtlıcaı içindeki yerini belir-  
tir.)

#### A.2 Z80'de Giriş/Çıkış Grubu Komutları

Z80'deki giriş/çıkış grubu komutları 8080'deki giriş/çı-  
kış grubu komutlarının hepsini kapsamaktadır. Z80 bu-  
nun yanısıra 8080'de bulunmayan bazı giriş/çıkış komut-  
larına da sahiptir. Bunlar Z80'in giriş/çıkış işlemlerinde  
8080'den bir derece daha üstün olmasına neden olmakta-  
dır.

Z80'nin giriş/çıkış grubu komutlarının kısa adları (mne-  
monic), onların sembolik açıklamaları ve etkiledikleri  
bayraklar aşağıdaki tabloda gösterilmektedir :

Komutun Kısa Adı (Mnemonic)	Komutun Yaptığı işlem	Etkilenen Bayraklar				İşlemKodu (H)
		S	Z	P/V	C	
IN A, (n)	A - M	.	.	.	.	DB
INr.(C)	r←(C)	E	E	p	.	ED
INI	(HL)-(C) B - B - 1 HL - HL + 1	X	E	X	X	ED A2
INIR	<HL>-(C) B - B - 1 HL - HL + 1 ve B = 0 olana kadar tekrarla	X	1	X	X	ED <b>ta</b>
IND	(HL)-(C) B - B - 1 HL - HL - 1	X	E	X	X	ED AA
INDR	(HL)-(C) B - B - 1 HL - HL - 1 ve B = 0 olana kadar tekrarla	X	1	X	X	ED BA
OUT (n), A	(n) - A	.	.	.	.	D3
OUT(C), r	(C) - r	.	.	.	.	ED
OUTI	(C)-(HL) B - B - 1 HL - HL + 1	X	E	X	X	ED A3
OTIR	<C>-(HL) B - B - 1 HL - HL + 1 ve B = 0 olana kadar tekrarla	X	1	X	X	ED B3
OUTD	(C)-(HL) B - B - 1 HL - HL - 1	X	E	X	X	ED AB
OTDR	(C)-(HL) B - B - 1 HL - HL - 1 ve B = 0 olana kadar »krarh	X	1	X	X	ED BB

Not : Yukarıdaki tabloda kullanılan semboller şu anlamları taşımaktadır.

.

1 = Bayrak birleşmektedir.

X = Bayrağın durumu belli değildir.

E = Bayrak işlemin sonucuna göre etkilenmektedir.

\* = Eğer B-1 sıfır sıfır bayrağı (Z) birleşmekte, aksi takdirde Z sıfır olmaktadır.

P = P/V bayrağı işlem sonucunun eşlik durumuna göre etkilenmektedir.

r = A, B, C, D, E, H, L kay Ulayıcılarından herhangi biri  
n = 8 bitlik port adresi

(\_\_\_) = Parantezin içinde bulunan kayıtlayıcıların veya adreslenmiş bellek yerinin içeriğini verir. (Onlarda depolanmış veriyi gösterir.)

Bütün Z80 giriş/çıkış komutlarında kullanılan port adresleri 8 bitlidir.

Yukarıda sayılan giriş/çıkış komutları dört gruba ayrılabilir :

1) Port adresini doğrudan kullanan komutlar (Doğrudan Adreslemeyi kullanan komutlar) IN A, (n) ve OUT(n), A. Bu komutlar, komutta adresi verilmiş olan port ile bir keç (A) arasındaki 8 bitlik veri aktarımını gerçekleştirirler.

2) Tek baytlık ve kayıtlayıcı dolaylı adreslemeyi kullanan komutlar : IN r, (C) ve OUT (C), r. Bu komutlar, belirtilen kayıtlayıcı ile adresi C kayıtlayıcısında verilmiş olan port arasındaki 8 bitlik veri aktarımını gerçekleştirirler. Port adresi sürekli olarak C kayıtlayıcısında bulunmaktadır. Bu port adresi program belleğinin bir parçası değildir. Bu nedenle o, bir değişken olarak kullanılabilmekte ve programa bir esneklik getirebilmektedir.

3) Blok giriş/çıkış komutları: INI, OUTI, IND ve OUTD. INI ile OUTI komutları, HL kayıtlayıcı çiftinde adresi verilen bellek yeri ile C kayıtlayıcısında adresi verilen port arasındaki veri aktarımını gerçekleştirirler. Bunun yanı sıra her iki komutta HL'deki adresi birer arttırır ve B kayıtlayıcısındaki sayıcının değerini birer azaltırlar. IND ve OUTD komutları tek fark dışında INI ve OUTI komutlarına aynı işlevi görürler: IND ve OUTD komutları HL'deki adresi birer arttırmazlar, tam tersi birer azaltırlar.

4) Tekrarlanan blok giriş/çıkış komutları : INIR, OTIR, INDR ve OTDR. INIR ve OTIR komutları, INI ve OUT, komutlarının işlevlerini B'deki sayıcının değerinin sıfıra düşmesine kadar tekrarlar (yaparlar). INDR ve OTDR komutlarına IND ve OUTD komutları arasında da böyle bir benzer ilişki vardır.

Giriş/çıkış komutlarının daha iyi anlaşılabilmesi amacıyla, aşağıda bu komutlarla ilgili çok basit program örnekleri verilmektedir. (Bu örneklerde zamanlama göz önünde bulundurulmayacaktır.)

Örnekler:

1) Birikeci (A'yı) giriş portu 5'ten (adres 5H olan giriş portundan) yükle

— Doğrudan adreslemeyle

IN A, (5)

- Kayıtlayıcı dolaylı adreslemeyle

LD C, 5

IN A, (C)

2) Birikeçteki (A'daki) veriyi çıkış portu 55H'e gönder.

— Doğrudan adreslemeyle

OUT (55H), A

- Kayıtlayıcı dolaylı adreslemeyle

LD C, 55H

OUT (C), A

3) Giriş portu 15H'ten F5AOH ile adreslenmiş bellek yerini yükle

— Doğrudan adreslemeyle

IN A, (15H) ; VERİYİ ELDE ET

LD (F5AOH), A ; VERİYİ DEPOLA

— Kayıtlayıcı dolaylı adreslemeyle

LD C.15H ; PORT ADRESİNİ

ELDE ET

IN A, (C) ; VERİYİ ELDE ET

LD (F5AOH), A ; VERİYİ DEPOLA

— Blok giriş/çıkış komutlarıyla

LD C.15H ; PORT ADRESİNİ

ELDE ET

LD HL.F5AOH ; BELLEK YERİNİN

ADRESİNİ ELDE

ET

INI ; VERİYİ ELDE

ET VE DEPOLA

4) 1800H ile adreslenmiş bellek yerindeki veriyi çıkış portu F8H'e gönder.

— Doğrudan adreslemeyle

LD A, (1800H) ; VERİYİ ELDE ET

OUT (F8H), A ; VERİYİ GÖNDER

— Kayıtlayıcı dolaylı adreslemeyle

LD C.F8H ; PORT ADRESİNİ

ELDE ET

LD A, (1800H) ; VERİYİ ELDE ET

LUT (C), A ; VERİYİ GÖNDER

— Blok giriş/çıkış komutlarıyla

LD C.F8H ; PORT ADRESİNİ

ELDE ET

LD HL, 1800H ; BELLEK YERİNİN

ADRESİNİ ELDE

ET

OUTI ; VERİYİ ELDE ET

VE GÖNDER

5) 1800H ile 1808H arasındaki bellek yerlerini giriş portu 6'dan yükle

— Doğrudan adreslemeyle

LD HL.1800H ; VERİNİN BEL-

LEKTEKİ BAŞ-

LANGIÇ ADRESİ-

Nİ ELDE ET

LD B,9 ; SAYICININ DEĞE-

RİNİ 9'A GETİR

DEVAM: IN A, (6) ; VERİYİ ELDE ET

LD (HL), A ; VERİYİ DEPOLA

INC HL

D) NZ DEVAM

— Blok giriş/çıkış komutlarıyla

LD HL.1800H ; VERİNİN BEL-

LEKTEKİ BAŞ-

LANGIÇ ADRESİ-

Nİ ELDE ET

LD B,9 ; SAYICININ DEĞE-

RİNİ 9'A GETİR

LD C,6 ; PORT ADRESİNİ  
ELDE ET  
DEVAM : INI ; VERİYİ ELDE ET  
VE DEPOLA

JR NZ, DEVAM  
- Tekrarlanan blok giriş/çıkış komutlarıyla  
LD HL, 1800H ; VERİNİN BEL-  
LEKTEKİ BAŞ-  
LANGIÇ ADRESİ-  
Nİ ELDE ET

LD B,9 ; SAYICININ DEĞE-  
RİNİ 9'A GETİR

LD C,6 ; PORT ADRESİNİ  
ELDE ET

INIR ; VERİYİ ELDE ET  
VE DEPOLA (SA-  
YICI SIFIR OLUN-  
CAYA KADAR)

6) Bellekte bulunan 1800H ile 1808H arasındaki veriyi  
çıkışı; portu 6'ya gönder

- Doğrudan adreslemeyle  
LD HL, 1800H ; VERİNİN BEL-  
LEKTEKİ BAŞ-  
LANGIÇ ADRESİ-  
Nİ ELDE ET

LD B,9 ; SAYICININ DEĞE-  
RİNİ 9'A GETİR  
DEVAM : LD A, (HL) ; BELLEKTEN VE-  
RİYİ ELDE ET  
OUT (6), A ; VERİYİ ÇIKIŞ  
PORTUNA GÖN-  
DER

DJNZ DEVAM  
- Blok giriş/çıkış komutlarıyla  
LD HL, 1800H ; VERİNİN BEL-  
LEKTEKİ BAŞ-  
LANGIÇ ADRESİ-  
Nİ ELDE ET

LD B,9 ; SAYICININ DEĞE-  
RİNİ 9'A GETİR

LD C,6 ; PORT ADRESİNİ  
ELDE ET

DEVAM : OUTI ; VERİYİ ELDE ET  
VE ÇIKIŞ PORTU-  
NA GÖNDER

JR NZ, DEVAM  
- Tekrarlanan blok giriş/çıkış komutlarıyla  
LD HL, 1800H ; VERİNİN BEL-  
LEKTEKİ BAŞ-  
LANGIÇ ADRESİ-  
Nİ ELDE ET

LD B,9 ; SAYICININ DEĞE-  
RİNİ 9'A GETİR

LD C,6 ; PORT ADRESİNİ  
ELDE ET

OTIR ; BELLEKTEKİ VE-  
RİYİ ELDE ET VE  
ÇIKIŞ PORTUNU  
GÖNDER (SAYICI  
SIFIRLUNCAYA  
KADAR)

Görüldüğü gibi "tekrarlanan blok giriş/çıkış" komutları  
sürekli olan bir veri aktarımını gerçekleştirirler.  
Bu durumda, aktarımlar arasında bir zamanlama yapmak  
söz konusu olamamaktadır. Bu nedenle bu komutlar, çev-  
re elemanlarının CPU ile aynı hızda çalışmadığı durum-

larda veya zamanlamanın clonanims.il olarak ele alınıp  
çözümlemediği clurumlarda kullanılamazlar.

ZSO Paralel Giriş/Çıkış Devresi (PIO), CPU ile aynı hızda  
çalışmaktadır. PIO kullanarak, çevresel elemanlara yapı-  
labilecek veya onlardan gelebilecek veri aktarımlarının  
zamanlaması yapılabilmektedir.

### A.3 Z80 Paralel Giriş/Çıkış Devresi (PIO)

PIO'nun biricak x(isterimi Şekil 4'te, PIO'nun blok şema-  
sı da Şekil 5'te gösterilmektedir. PIO'da 8'şer bitlik iki  
tane port bulunmaktadır. (Port A ve Port S). Bu portlar-  
ının blok şeması da Şekil 6'da gösterilmektedir. PIO'nun  
kesme ile ilgili özelliklerine ve yeteneklerine bu ya/ıda  
yer verilmeyecektii .

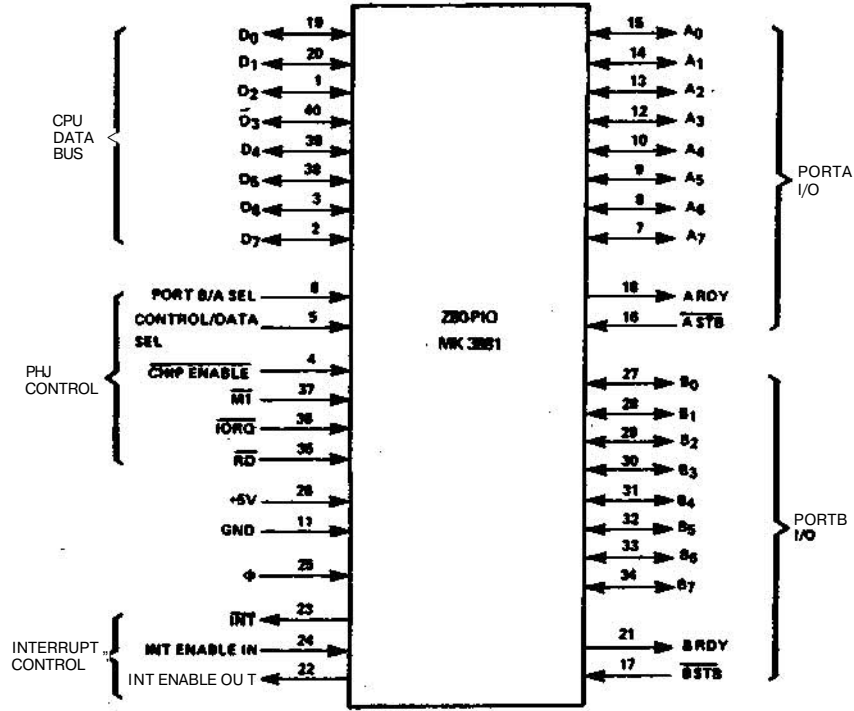
Her PIO, dön giriş port adresini ve dört çıkış port adre-  
sini kaplar.

Eğer PIO'da C/D SEL A-j'e ve U/A SEL A<sub>0</sub>'a bağlanmış  
ise port adresleri aşağıdaki gibi olacaktır :

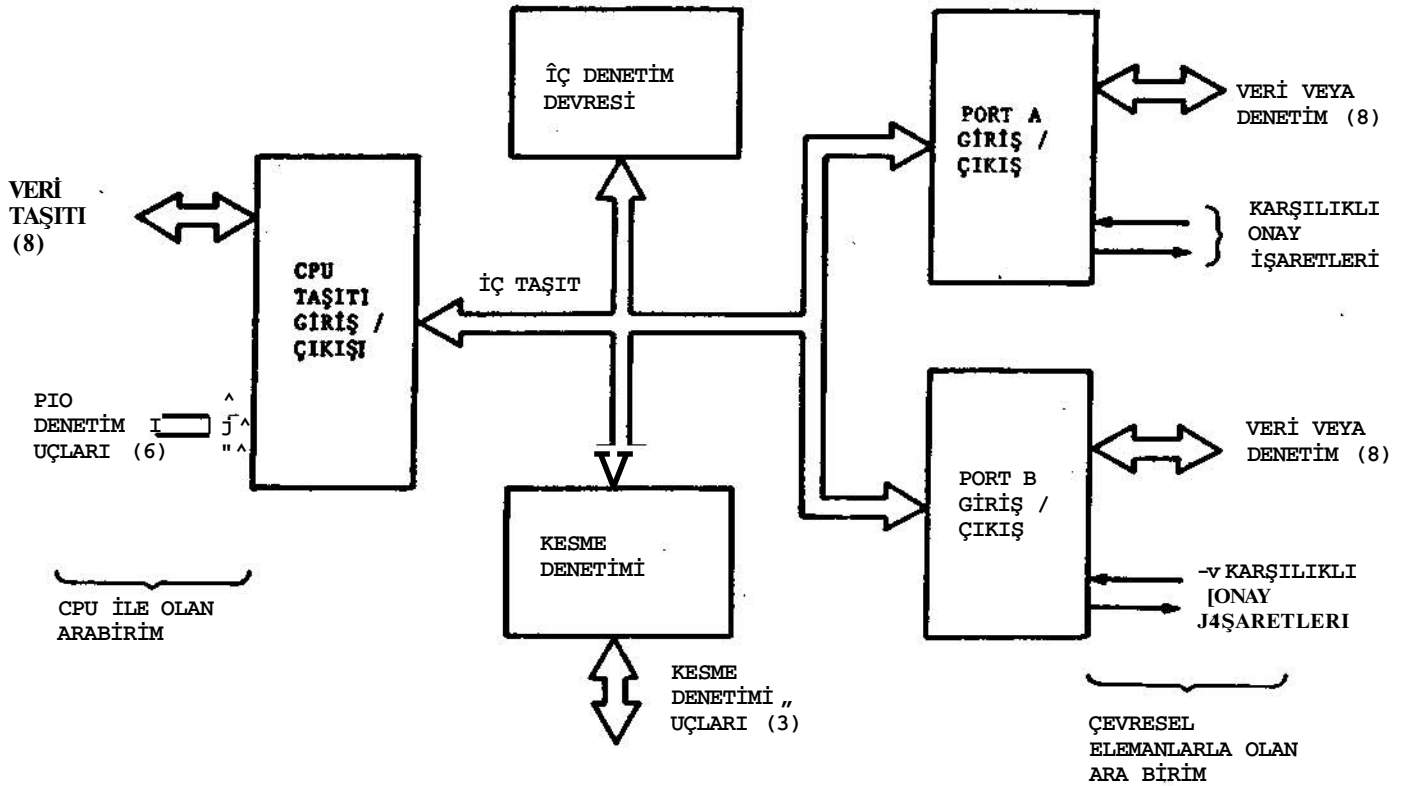
	Port Adresleri (PORTAD ile uuaşladığı Düşünülürse)
-Vireslenmiş Ka> Ulayıcılar	
Veri Kayıtlayıcısı A	PORTAD
Veri kayıtlayıcısı L>	PORTAD+1
Vnetim Kayıtlayıcısı A	PORTAD + 2
Denetim Kayıtlayıcısı B	PORTAD+3

Denetim Kayıtlayıcısı A veya B'ye gönderilen verinin  
D7D6D5D4D3D2D1D0 şeklinde olduğu kabul edilirse  
PIO Denetim Kayıtlayıcılarının adreslemesi de aşağıdaki  
şekilde gösterildiği gibi ifade edilebilir.

Kayıtlayın	Adreslemesi
\IOD DENETİMİ	D3 = D2 = D1 = D0 = 1
GİRİŞ/ÇIKIŞ DENETİMİ	MOD EDENETİMİ MOD3Ü SEÇTİKTEN SONRAKİ SÖZCÜK
MASKELEME DENETİM KAYITLAYICISI	03 = 0, 02 = 0! = D <sub>0</sub> = 1
KESMEYİ MASKELEME KAYITLAYICISI KAYITLAYICISI	MASKELEME DENETİM KAYITLAYICISINA D <sub>4</sub> = 1 İLE GİRİLDİKTEN * SONRAKİ SÖZCÜK
KESMEYE İZİN VERME	D3 = D2 = 0 . D ^ D ^ I
KESME VEKTÖRÜ	D <sub>0</sub> = 1

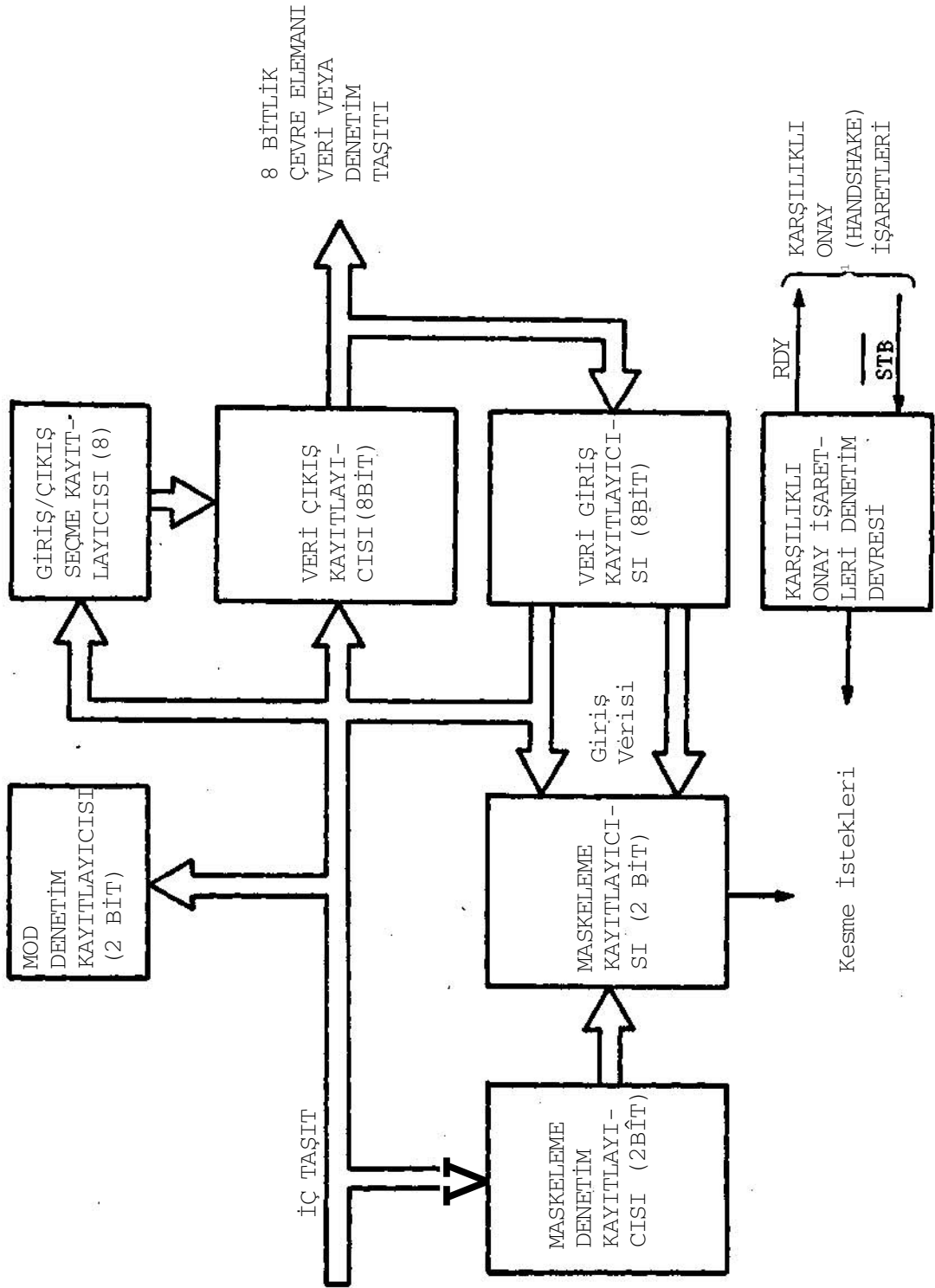


Şekil : 4 I/O'unun li.cak dostu-rim'



Şekil : 5 PIO'nun Blok Şemdsı





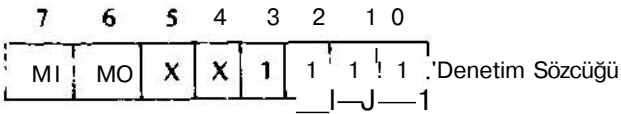
Şekil :6 PIO'nun l'ortlarının Blok Şeması

## PIO Mod Denetimi

PIO'da çalışma modu, PIO'ya aşağıda belirtilen şekil-  
bir denetim sözcüğünün yazılmasıyla belirlenir.

Modun Belirlenmesi :

MI	MO	Mod
0	0	Çıkış
0	1	Giriş
1	0	Çift yön
1	1	Bit Denetimi



Mod 3 seçildiği zaman, onu takibeden sözcük giriş/çıkış  
denetimi kayıtlayıcısını düzenleyecektir.

G/Ç7	G/Ç6	G/Ç5	G/Ç4	G/Ç3	G/Ç2	G/Ç1	G/Ç0
------	------	------	------	------	------	------	------

Giriş/Çıkış Denetimi Kayıtlayıcısı

G/Ç 1 ise bu bit giriş olur.

G/Ç 0 ise bu bit çıkış olur.

PIO'nun mod'ları, onlara karşılık gelen anlamlar ve de-  
netim sözcükleri aşağıda gösterildiği gibidir.

PIO Modları	Anlamları	Denetim Sözcüğü	
		İkili (Binary)	Hex
0	Çıkış	00001111	0F
1	Giriş	01001111	4F
2	Çift yön	10001111	8F
3	Denetim	11001111	CF

Not : Bit 4 ve bit 5 belirli bir amaç için kullanılmamak-  
tadır. Bu iki bit herhangi bir değeri alabilir.

PIO'nun mod'ları kısaca şu şekilde özetlenebilir:

1) Mod 0 - ÇIKIŞ MODU : Port çıkış kayıtlayıcısına  
veriyi yazmak, verinin port veri taşıtında hazır olmasını  
sağlar. RDY (Hazır) işaretinin O'dan 1'e çıkması verinin  
hazır olduğunu belirtir. RDY, çevre elemanından verinin  
kabul edildiğini belirtir. STB işaretinin gelmesine kadar  
1'de kalır. Eğer kesmeye izin verilmiş ise STB 'nin O'dan  
1'e geçişi kesmeye neden olmaktadır.

2) Mod 1 - GİRİŞ MODU : Çevre elemanı veya eleman-  
ları, veriyi port giriş kayıtlayıcısına STB işareti ile koyar.  
STB 'nin O'dan Ve geçişi (STB'nin artan kenarı) kesme-  
ye neden olur (kesmeye izin verilmişse) ve RDY'yi sıfır-

lar. CPU veriyi okuduğu zaman verinin kabul edildiğini  
göstermek için RDY 1'e gider.

3) Mod 2 - ÇİFT YÖN MODU : Bu mod, 4 karşılıklı  
onay (handshake) işaretinin hepsini kullanır. Onun için  
veri giriş ve çıkışı sadece port A'dan yapılır. Port A'nın  
RDY ve STB işaretleri çıkış denetimi için, port L'nin  
RDY ve STB işaretleri ise giriş denetimi için kullanılır.

4) Mod 3 - DENETİM KODU : Bu mod RDY ve STB  
işaretlerini kullanmaz. Bu mod her bir bitin tek başını  
anlam taşıdığı durum vedenetim uygulamaları için amaç-  
lanmıştır. Mod 3 seçildikten sonraki PIO'ya (giriş/çıkış  
denetimi kayıtlayıcısına) gönderilen denetim sözcüğü port  
veri bit'lerinin yönlerini belirler. Daha önce de belirtildi-  
ği gibi, bu sözcük içindeki herhangi bir bit yerindeki " 1 "  
buna karşılık gelen taşıtı giriş yapar. Benzer şekilde " 0 "  
da o bit'e karşılık gelen taşıtı çıkış yapar.

PIO'nun Programlanması

Güç kaynağı yeni açıldığı zaman PIO giriş moduna (Mod  
1'e) girer. Aynı zamanda bütün kesmeler kesmeye izin ve-  
rilmez durumuna getirilir ve denetim işaretleri sıfırlanır.

PIO'nun programlanmasında şunlar gözönünde bulundu-  
rulur :

1) Çalışma Modunu belirlemek için, uygun denetim söz-  
cüğü Mod Denetimi Kayıtlayıcısına yazılır. Kesme .dene-  
timi ile giriş/çıkış mod'u bilgisinin de gönderilmesi gere-  
kebilir.

2) Eğer Mod 3'e girildiyse, Mod 3'ün seçildiğini bildiren  
denetim sözcüğünden hemen sonra, giriş/çıkış denetimi  
kayıtlayıcısına bir denetim sözcüğü gönderilerek giriş/çı-  
kış bacaklarının yönü belirlenir.

PIO'nun programlanmasının daha iyi anlaşılabilmesi için,  
aşağıda bunlarla ilgili basit program örneklerine yer veri-  
lecektir. (Bu örneklerde kesme ile ilgili program örnekle-  
rine yer verilmemektedir.)

örnekler :

1) B portunu çıkış portu yapmak

```
LD A,00001111B ;B PORTUNU ÇIKIŞ YAP
OUT (PIODNB),A
```

2) B portunu giriş portu yapmak

```
LD A,01001111B ;B PORTUNU GİRİŞ YAP
OUT (PIODNB),A
```

3) A portunun çift yönlü port olarak kullanılması

```
LD A.10001111B ; A PORTUNU ÇİFT YÖNLÜ
YAP
```

```
OUT (PIODNA*),A
```

4) Portun Denetim Portu olarak kullanılması

```
- LD A.11001111B ; A PORTUNU DENETİM
PORTU YAP
```

```
OUT (PIODNA),A
```

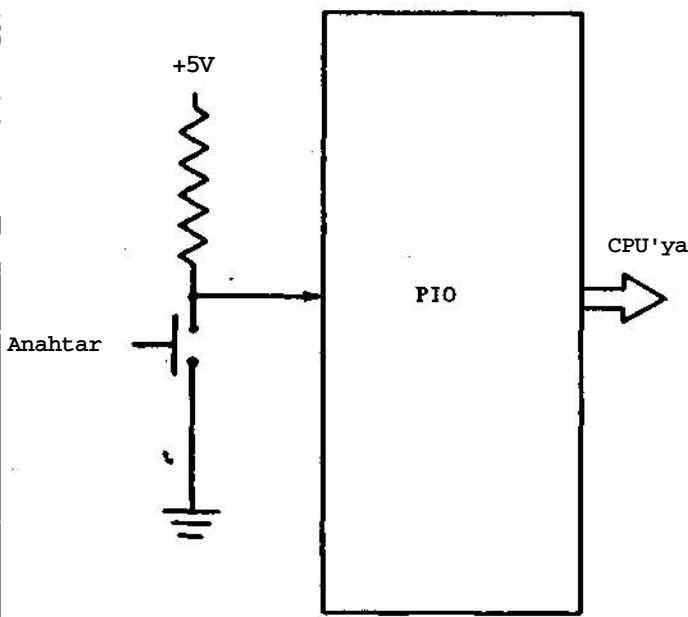
```
LD A.FFH ; BÜTÜN BİTLERİ GİRİŞ YAP
```

```
OUT (PIODNA),A
```

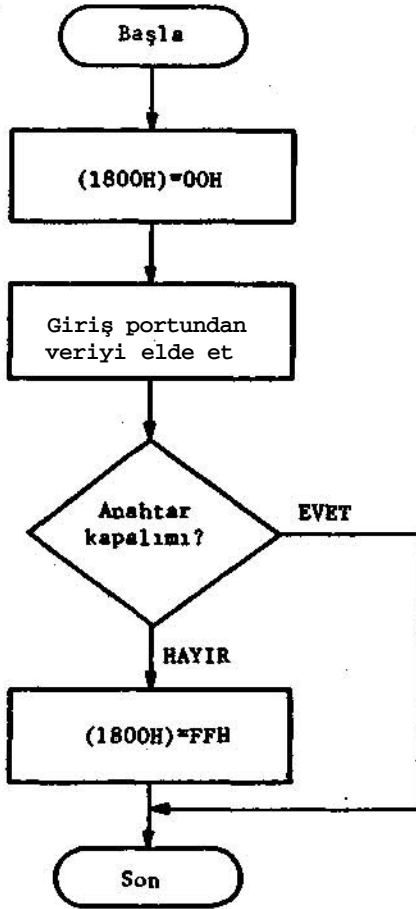
```
- LD A,11001111B ; B PORTUNU DENETİM
PORTU YAP
```

```
OUT (PIODNB),A
```

```
LD A.OOH ; BÜTÜN BİTLERİ ÇIKIŞ YAP
```



Şekil : 7-a



Şekil : 7-b

```

OUT (PIODNB).A
LD A,110011113 ; APORTUNU DENETİM
PORTU YAP

OUT (PIODNA).A
LD A.000011110B ; TAŞITIN 1., 2,3. BİTLERİ-
Ni GİRİŞ, 0., 4., 5., 6., 7.
BİTLERİNİ ÇIKIŞ YAP

OUT (PIODNA).A
5) 35H'in (00110101B) çıkış portu A'ya gönderilmesi
LD A.OFH ; APORTUNU ÇIKIŞ YAP
OUT (PIODNA).A
LD A,35H ; 35H'i (0011010113i) BİRİ-
KECE YÜKLE
OUT (PIOVRA).A ; BİRİKEÇTEN ÇIKIŞAĞÖN-
DER
  
```

• i PIO giriş portu A'nın 0. bitine bir anahtar bağlanmıştır. Şekil 7a'dan da görüldüğü gibi, bu anahtar açıkken C.

bite 1(+5V) kapalıyken 0(0V) gelmektedir. Öyle bir program yazılması istenmektedirki, bu program bu anahtarın açık mı yoksa kapalı mı olduğunu test etsin ve anahtar açıksa belleğin 1800H ile adreslenmiş yerini FFH'le, kapalıysa 00H'le doldursun.

Bunu sağlayan programın akış diagramını Şekil 7B'de gösterilmektedir.

Bu program aşağıdaki gibidir.

```

LD A.01001111B ; APORTUNU GİRİŞ YAP
OUT (PIODNA).A
LD HL.1800H
LD (HL),00H ; İŞARET =00H
IN A.(PIOVRA) ; ANAHTARIN DURUMUNU
OKU
AND 01H ; ANAHTAR KAPALI MI (0)?
)R Z.SON ;EVET
LD (HL),FFH ; HAYIR. İŞARET = FFH
SON: HALT
  
```

#### KAYNAKLAR :

- 1) MicrucompuK T Comproanın D;ii;ı Hook.Moslrk Corporation, 1979
- 2) Z80 Asscmly LanguuKC Programmin!;, I;incc A. I.venthal, Ad;im Osborne and Associates, Inc.. I 979
- 3) Z-80Microprocessor, Pro.~i-ımmiiKiin(l Iucrfacing, Joseph C. Nichols, F.li/abcth A. Nirlolsand Petri R.Rony, Nanotran, Inc., 1979.
- 4) ODTÜ Elektrik Mühendisliği Böl. Mustafa Parlar Yaz Okulu "Bilgisayar Donanımı ve Mikroişlemci Sistemleri" Kurs Notları. Hasan Güran, Uğur Halıcı, Aykut Dalgıç, (28 Haziran - 2 Temmuz 1982)
- 5) The Z-80 Microcomputer Handbook, U'lliam liardcn, JR., Howard W. Sams and Co., Inc., 1978.
- 6) Microcomputer- BasedDesignJohn B. Peatman,McC.iau -II ili. Inc., 1977.

\* PIODNB = Denetim Kayıtlayıcısı B'nin adresi  
 PIODNA = Denetim Kayıtlayıcısı A'ran adresi  
 PIOVRA = Veri Kayıtlayıcısı A'nın adresi