



**SAKARYA ÜNİVERSİTESİ**  
**Bilgisayar ve Bilişim Bilimleri Fakültesi**  
**Bilgisayar Mühendisliği Bölümü**

# **Mikroişlemcili Sistemler ve Laboratuvarı**

## **7.HAFTA**

# Amaçlar

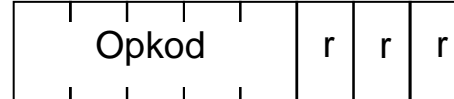
- 8051 mikrodenetleyicisinin adresleme yöntemlerini kavramak
- Komut türleri ve adresleme modları arasındaki ilişkileri kavramak

# Adresleme Yöntemleri

- Adresleme modu, bir bellek konumuna ya da bir veriye erişimin nasıl olacağını belirtir.
- Doğrudan kullanılan komut uzunluğunu etkiler.
- Kullanılan komutlara bağlı olarak bilginin farklı yollarla hedefe gitmesine olanak sağlar.
- 8051 mikrodenetleyicisinde kullanılan 8 farklı adresleme yöntemi şunlardır.
  - Kaydedici adresleme
  - Doğrudan adresleme
  - Dolaylı adresleme
  - İvedi adresleme
  - Bağlı (Koşullu) adresleme
  - Mutlak adresleme
  - Uzun adresleme
  - İndisli adresleme yöntemi

# Kaydedici Adresleme

- 8051 mikrodenetleyicisinde R0'dan R7'ye kadar 8 tane genel amaçlı kaydedici vardır.
- Kaydedici adreslemede
  - Komutu oluşturan en yüksek değerlikli 5 bit yapılacak işlevi
  - En düşük değerlikli 3 bit ise R0 ile R7 arasındaki hangi kaydedicinin kullanılacağını gösterir.



Assembly	Açıklama
ADD A, R7	;R7 kaydedicisinin içeriğini Akümülatöre ekle

00101 111

komutun makine dilindeki karşılığı 00101111B'dir

Kullanılan kaydedici R7

ADD işlemini gösteren Opkod

# Doğrudan Adresleme

- Doğrudan adresleme yöntemi, dahili alt RAM (lower RAM) ve SFR alanına erişmek için kullanılır.
- Doğrudan adresleme yönteminde komutlar 2 bayt uzunluğundadır.
  - İlk bayt opkod'u (gerçekleştirilecek işlemi),
  - ikinci bayt adres bilgisini gösterir.
- Doğrudan adresleme yöntemi adresleri örtüşen üst RAM ile SFR bölgeleri birbirinden ayrılmasını sağlar.
- Bu iki alandan SFR bölgesine doğrudan adresleme yöntemi kullanılarak erişilebilir.

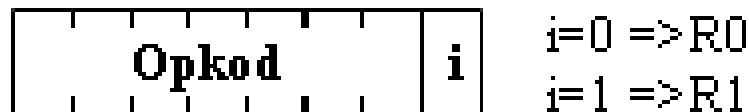


# Doğrudan Adresleme

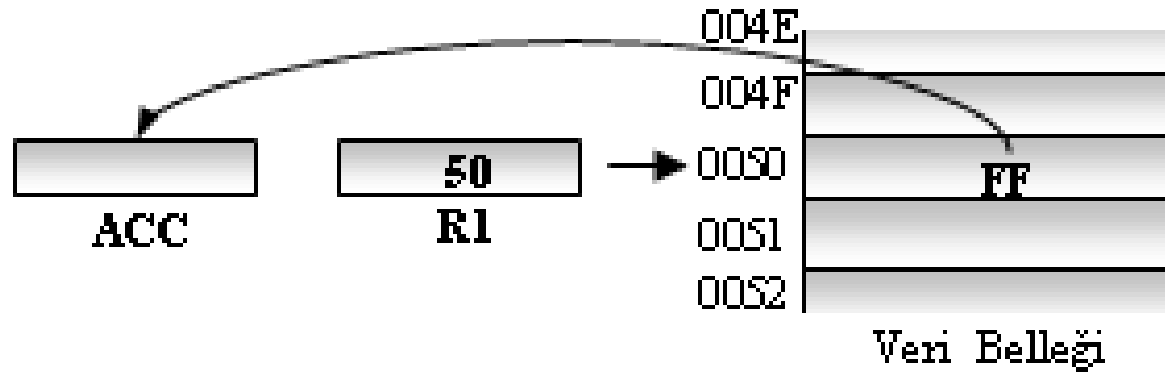
Assembly	Açıklama
MOV P1,A	;Aküyü Port 1'e kopyala
MOV A,70h	;70h adresinin içeriğini Aküye kopyala
MOV A,80h	;SFR bölgesine erişilir, 80h Port 0'ın adresidir. ;P0'daki bilgi Aküye kopyala

# Dolaylı Adresleme

- Tanımlanan bir değişkenin adresinin değiştirilmesi, hesaplanması ya da tekrar değiştirilmesi işlemlerinde dolaylı adresleme yöntemi kullanılmaktadır.
- Adresleme yöntemlerinin en güçlüsüdür.
- Bu adreslemede kaynak veya hedefin adresi komutun içerisinde açık olarak verilmez.
- Verinin gerçek adresini tutmak için R0 ve R1 kaydedicileri “işaretçi” olarak kullanılır.
- Bu kaydediciler bilginin RAM’de yazılacağı veya okunacağı adresi içermektedirler.



# Dolaylı Adresleme



**Assembly**

**Açıklama**

`MOV A,@R1`

*;Alt RAM'deki 50h adresinin içeriği (FFh) Aküye aktar*



# İvedi Adresleme

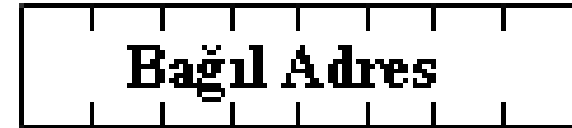
- DPTR'nin kullanıldığı istisnai durum dışında 2 bayt uzunluğundaki komutlardan oluşur.
- Bilginin geçici olarak komut içerisine yüklenmesi yüksek komut hızı sağlar.



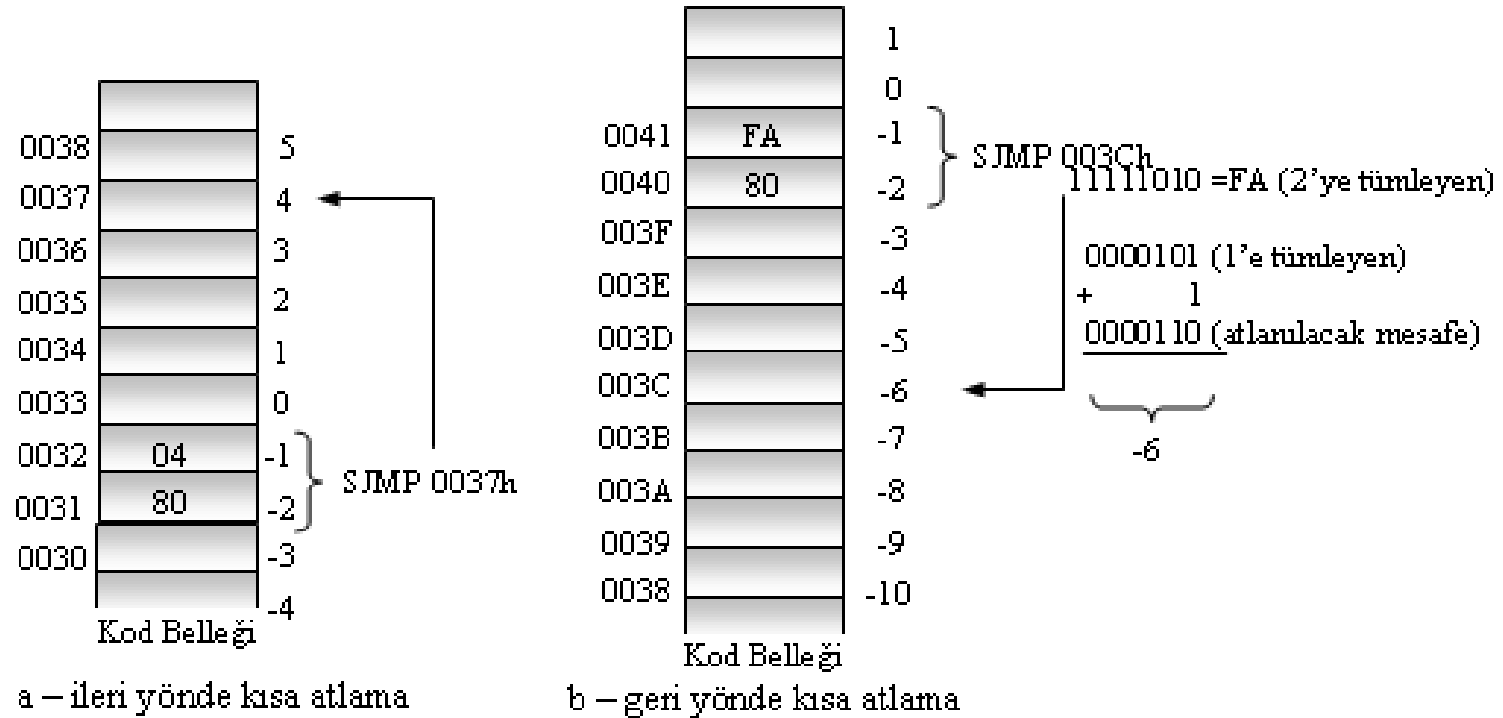
Assembly	Açıklama
MOV A, #12	<i>;Akü'ye 12 değerinin atılması</i>
MOV R0, #10h	<i>;10h bilgisini R0 saklayıcısına yükle</i>
MOV DPTR, #2000H	<i>;2000h bilgisini DPTR'ye yükle, 3 bayt'lık komut</i>

# Bağlı Adresleme

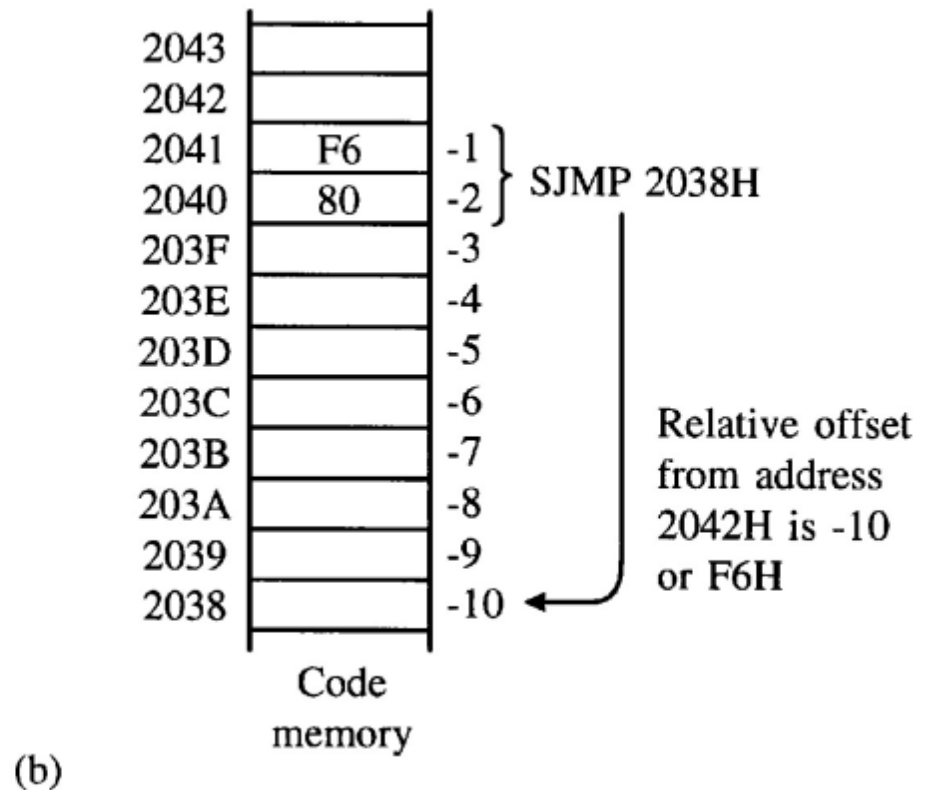
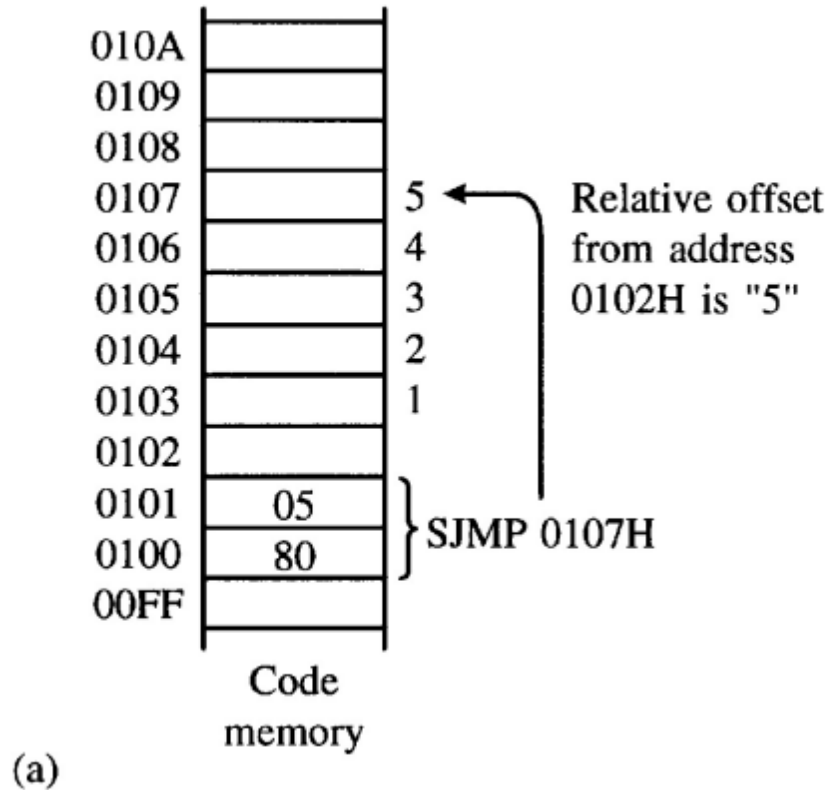
- Sadece atlama komutları ile birlikte kullanılır.
- Komutlar 1 bayt opkod ve 1 bayt adres bilgisi olmak üzere toplam 2 bayt uzunluğundadır.
- Adres bilgisi 8-bit ile ifade edildiği için maksimum +127 (ileri yön) ve -128 (geri yön) aralığında bir atlama işlemi gerçekleştirilir.



# Bağıl Adresleme



# Bağıl adresleme

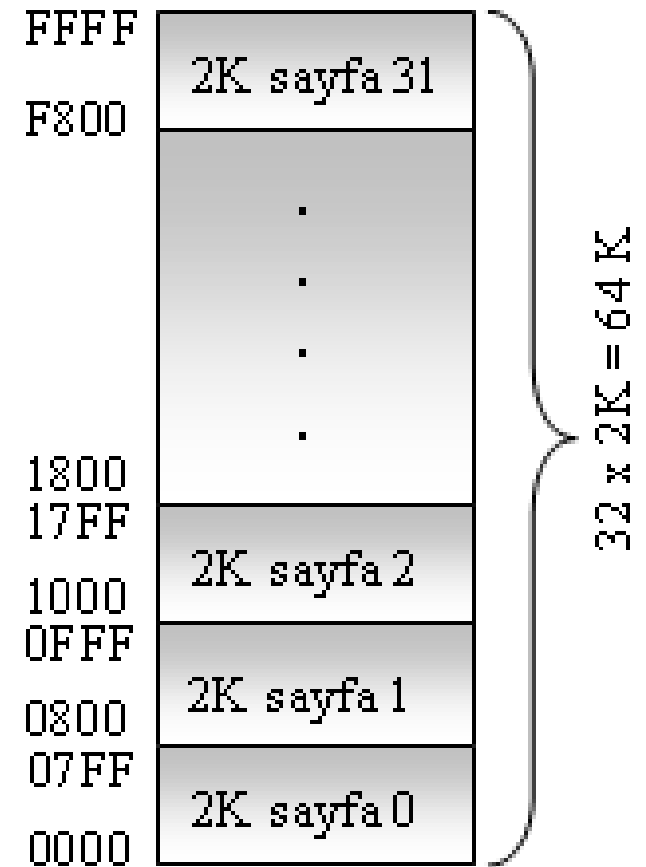
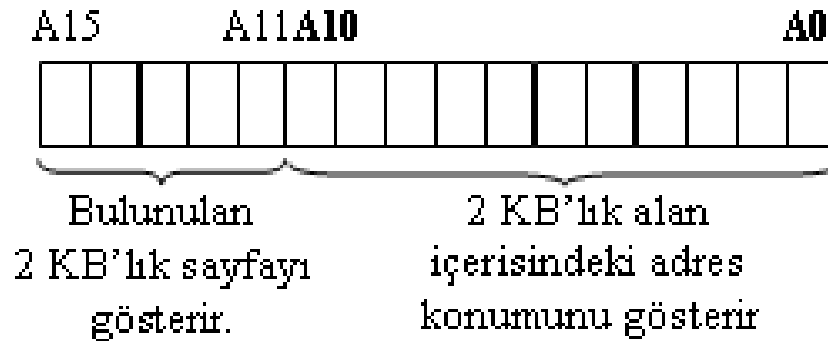


# Mutlak Adresleme

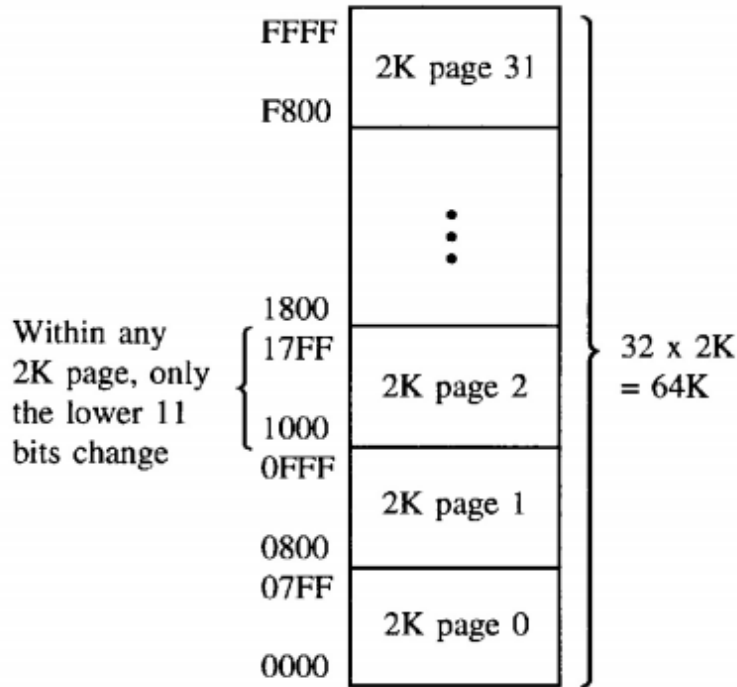
- Sadece ACALL ve AJMP komutları ile kullanılır.
- 2 bayt uzunluğundadır ve kod bellek içerisinde 2 KBaytlık bir alanı adresleyebilirler.
- Maksimum 64 KBayt olan kod bellek 2 KBaytlık 32 bölmeğe ayrılabilir
- Hangi bölmenin seçileceğini program sayacı (PC) belirlemektedir.



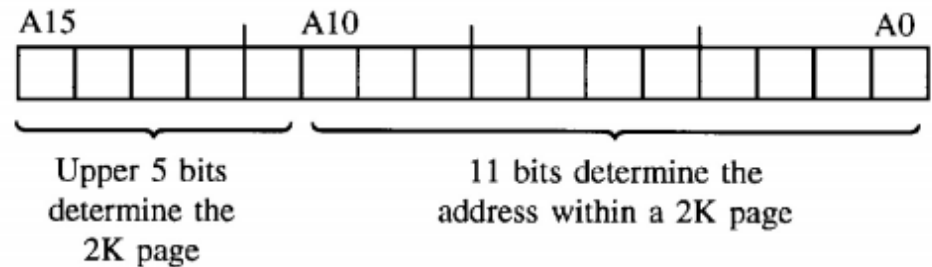
# Mutlak Adresleme



# Mutlak adresleme



(a)



(b)

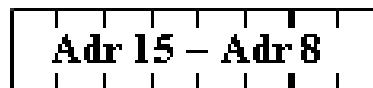
# Mutlak adresleme

- ORG 0
- MOV A,#55H
- BACK: MOV P1,A
- ACALL GECIKME
- CPL A
- SJMP BACK
- 
- GECIKME: MOV R5,#0FFH
- TEKRAR: DJNZ R5,TEKRAR
- RET
- END



# Uzun Adresleme

- Yalnızca 3 bayt'lık LCALL ve LJMP komutları kullanılır.
- 16-bit hedef adres bulunabilir.
- $2^{16} = 64$  KBaytlık adres aralığında atlama işlemi gerçekleştirilebilir.



# Uzun adresleme

- ORG 0
- BACK: MOV A,#55H
- MOV P1,A
- LCALL GECIKME
- MOV A,#0AAH
- MOV P1,A
- LCALL GECIKME
- SJMP BACK
- ORG 300H
- GECIKME: MOV R5,#0FFH
- TEKRAR: DJNZ R5,TEKRAR
- RET
- END

# Sıralı Adresleme

- Bellekte bulunan sıralı bilgilere erişmek için en elverişli adresleme yöntemidir.
- Çok sayıda veriye az sayıda komut kullanarak erişmek mümkündür.
- JMP ve MOVC komutları kullanılır.
- PC veya DPTR ile akümülatörün toplamı, atlanılacak olan etkin adres bilgisini belirler.



# Sıralı/İndisli Adresleme

Adres	Kodlar
0030h	MOV A, #0
0032h	CALL BASLA
	-----
	-----
0050h	BASLA: INC A
0051h	MOVC A, @A+PC
0052h	RET
0053h	TABLO: DB 33h
0054h	DB 55h
	-----

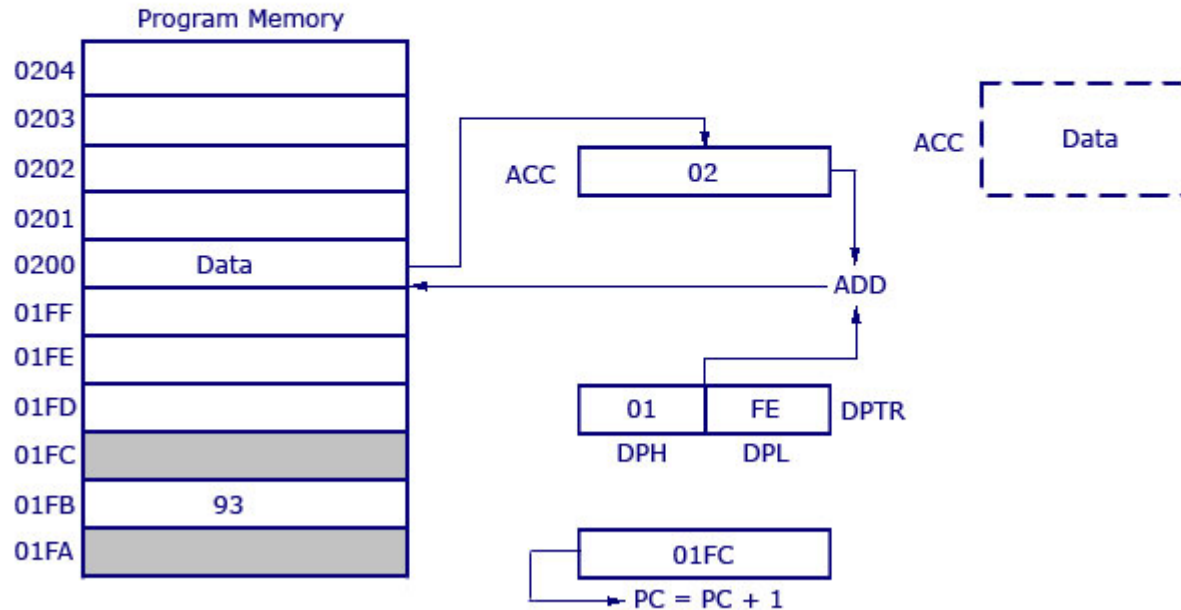
ACC	PC
0	0030
0	0032
Etkin Adres=PC+1+ACC =0051+1+1 = 0053h	
1	0050
33h	0051
33h	0052

004E	
004F	
0050	INC A
0051	MOVC A, @A+PC
0052	RET
0053	33h
0054	55h
	Kod Belleği

# İndisli adresleme

## Indexed Addressing Mode

Instruction	Opcode	Bytes	Cycles
MOVC A,@A +DPTR	93H	1	2



# İndisli adresleme

- ORG 0
- MOV DPTR, #TABLE ; the ROM location of LUT
- MOV P1, #02H ; set P1 as input
- BACK:   MOV A, P1 ; read x from port 1
- MOVC A, @A+DPTR ; find from LUT, move it to A
- MOV P2, A ; send results to P2
- SJMP BACK
- ORG 300H
- TABLE: DB 0, 1, 4, 9, 16, 25, 36, 49, 64, 81
- END

# Bölüm Soruları

- Bir mikroişlemcide çok sayıda adreslemenin olmasının avantaj ve dezavantajlarını açıklayınız?
- Uzun adresleme yerine bağıl adresleme ne zaman tercih edilmelidir?
- Adresleme ile komutların bayt miktarı arasında nasıl bir ilişki vardır?
- Tek-düze (uniform) komut ne demektir? 8051'in komut kümesi neden tek-düze komutlardan oluşamaz?