

# BİÇİMSEL Dil İler ve Otomat a Teorisi

VII'yi ara

Kl enekuram II

İZZET FATİH ŞENTÜRK



# Kl ~~ene~~Teorem

- Düzenli ifadeler veya sonlu otomatik veya geçiş grafikleri ile tanımlanabilir herhangi bir dil, üç yöntem ile tanımlanabilir • Kanıt üPart 1: Bir FA ile tanımlanabilir herhangi bir dil, bir TG ile tanımlanabilir.

üPart 2: Bir TG ile tanımlanabilir herhangi bir dil  
RE ile tanımlanabilir ir

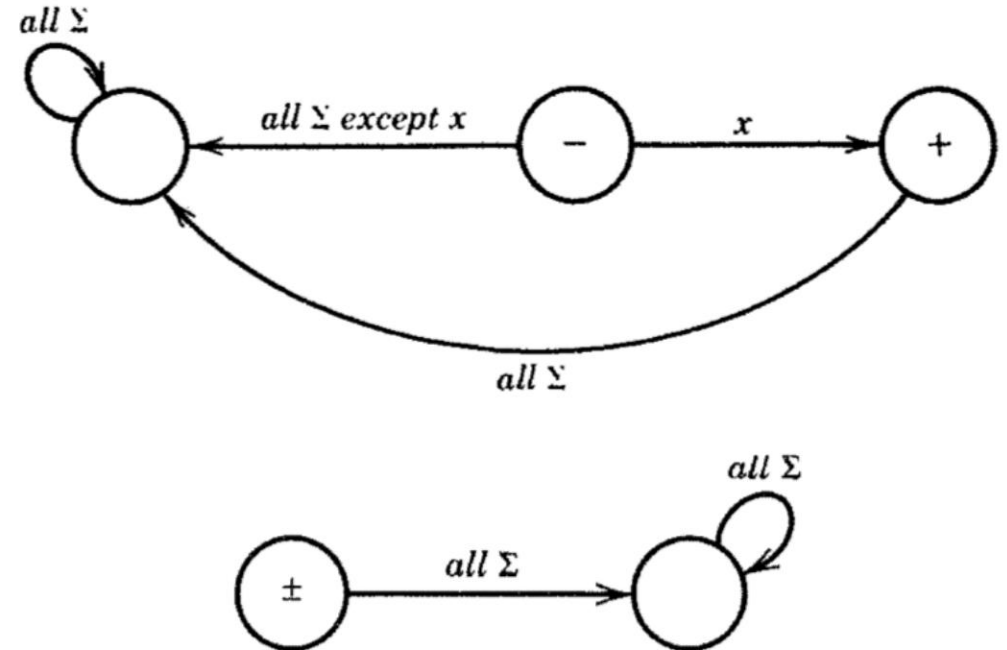
- Bölüm 3: RE ile tanımlanabilir herhangi bir dil, FA ile tanımlanabilir ir.

## Kanıt , Bölüm 3: RE'leri FA'ya Dönüştürme

- Bu, tümtörenin en zor kısmıdır • Her RE, belirli bir kural ile tekrar tekrar uygulanmasıyla  $\Sigma$  ve  $\Lambda$  alfabesinin harflerinden oluşturulabilir: • Toplama, birleştirme ve kapatma
- Bir RE oluşturduğumuzda, aynı zamanda aynı dili kabul eden bir FA da oluşturuyor olabiliriz.

### 3. Kısımların Kanıtı, 1. Kural

- Kural 1: Alfabenin herhangi bir harfini kabul eden bir FA vardır. Yalnızca  $\Lambda$  ifadesini kabul eden bir FA vardır.
- Kural 1'in Kanıtı: Eğer  $x, \Sigma$  içindeyse o zaman FA yalnızca  $x$  ifadesini kabul eder
- Kural 1'in Kanıtı: Kabul eden bir FA sadece  $\Lambda$

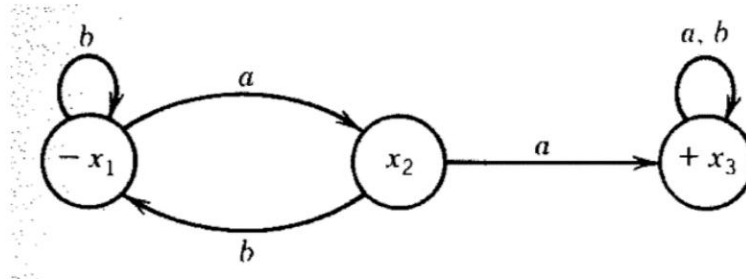


### 3. Kısım Kanıt 1, 2. Kural

- RE  $r_1$  tarafından tanımlanan dil i kabul eden FA1 adlı bir FA varsa ve RE  $r_2$  tarafından tanımlanan dil i kabul eden FA2 adlı bir FA varsa, tanımlanan dil i kabul eden FA3 olarak adlandıracağımız bir FA vardır. RE tarafından  $(r_1 + r_2)$   
 $\rightarrow$  ! Birlik!
- İki eski makineden yeni makinenin nasıl yapıldığını göstererek Kural 2'yi ispatlayacağız.
- Genel kriterleri belirtmeden önce bunları bir örnek üzerinde göstereceğiz.

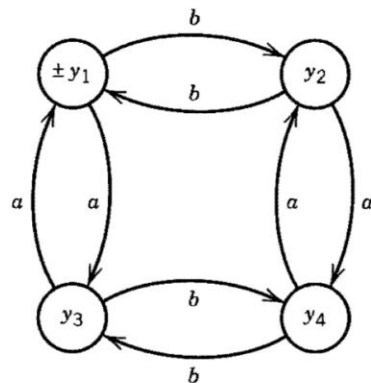
### 3. Kısımların 1, 2. Kural

- FA1:  $\Sigma = \{ab\}$  üzerindeki tüm kelimelerin dilini içeren bir yarı otomatik a



	$a$	$b$
$-x_1$	$x_2$	$x_1$
$x_2$	$x_3$	$x_1$
$+x_3$	$x_3$	$x_3$

- FA2: ÇİFT-ÇİFT (a'nın çift sayısı ve b'nin çift sayısı)



	$a$	$b$
$\pm y_1$	$y_3$	$y_2$
$y_2$	$y_4$	$y_1$
$y_3$	$y_1$	$y_4$
$y_4$	$y_2$	$y_3$

### 3. Kısım Kanıt 1, 2. Kural

- FA3: aa'sı olan veya EVEN-EVEN içinde olan ve hiçbir özel l iği olmayan diğer tüm dizeyi reddeden tüm kelimelerin dil i • Yeni makinenin dil i, bu ikisinin birleşimidir

Dil l er

- Bu yeni makinedeki durumları gerektiği kadar  $z_1, z_2, z_3 \dots$  olarak adlandıracağız.
- Bu makineyi geçiş tablosu ile tanımlayacağız •

FA1a1 onda çalışıyorsa girdinin nerede olacağını ve FA1'de çalışıyorsa girdinin nerede olacağını takip edeceğiz

tekbasına FA2

### 3. Kısım Kanıt 1, 2. Kural

- Öncelikle bir  $z_1$  başlangıç durumuna ihtiyacımız var •  $z_1, x_1$  (FA1'de çalıyor) veya  $1$ 'i (FA2'de çalıyor) birleştirir • FA3 makinesindeki tüm durumları çift anlamlıya çıkar • Hem FA1 hem de FA2'de çalıyor ve ikisini de takip ediyoruz  
aynı anda oynuyorlar
- Giriş harfi  $a$  okunursa hangi yeni durumlar oluşabilir?



### 3. Kısım Kanıt 1, 2. Kural

- Giriş harf i a okunursa hangi yeni durumlar oluşabilir? • FA1 için makineyi  $x_2$  durumuna getirir • FA2 için makineyi  $y_3$  durumuna getirir
- FA3'te a harf i makineyi  $x_2$  veya  $y_3$  anlamına gelen  $z_2$  durumuna getirir.

	$a$	$b$
$-x_1$	$x_2$	$x_1$
$x_2$	$x_3$	$x_1$
$+x_3$	$x_3$	$x_3$

	$a$	$b$
$\pm y_1$	$y_3$	$y_2$
$y_2$	$y_4$	$y_1$
$y_3$	$y_1$	$y_4$
$y_4$	$y_2$	$y_3$

$$\pm z_1 = x_1 \quad \text{or} \quad y_1$$

$$z_2 = x_2 \quad \text{or} \quad y_3$$

### 3. Kısım Kanıt 1, 2. Kural

- $z_1$ 'deyse keble harf ini okuyorsak
  - FA1 için makineyi  $x_1$  durumuna getirir (  $x_1$  durumundan) • FA2 için makineyi  $y_2$  durumuna getirir (  $y_1$  durumundan)

	$a$	$b$
$-x_1$	$x_2$	$x_1$
$x_2$	$x_3$	$x_1$
$+x_3$	$x_3$	$x_3$

$$\pm z_1 = x_1 \quad \text{or} \quad y_1$$

$$z_2 = x_2 \quad \text{or} \quad y_3$$

$$z_3 = x_1 \quad \text{or} \quad y_2$$

	$a$	$b$
$\pm y_1$	$y_3$	$y_2$
$y_2$	$y_4$	$y_1$
$y_3$	$y_1$	$y_4$
$y_4$	$y_2$	$y_3$

- FA3 için geçiş tablosunun başlangıcı

	$a$	$b$
$\pm z_1$	$z_2$	$z_3$

### 3. Kısım Kanıt 1, 2. Kural

- $z_2$ 'deyse  $a$  harf ini okursak • FA1 için makineyi  $x_3$  durumuna (son durum) getirir • FA2 için makineyi  $y_1$  durumuna getirir
- Eğer  $z_2$ 'deyse  $b$  harf ini okuyorsak
  - FA1 için makineyi  $x_1$  durumuna getirir • FA2 için makineyi  $y_4$  durumuna getirir

	$a$	$b$
$-x_1$	$x_2$	$x_1$
$x_2$	$x_3$	$x_1$
$+x_3$	$x_3$	$x_3$

	$a$	$b$
$\pm y_1$	$y_3$	$y_2$
$y_2$	$y_4$	$y_1$
$y_3$	$y_1$	$y_4$
$y_4$	$y_2$	$y_3$

$$+z_4 = x_3 \quad \text{or} \quad y_1$$

$$z_5 = x_1 \quad \text{or} \quad y_4$$

	$a$	$b$
$\pm z_1$	$z_2$	$z_3$
$z_2$	$z_4$	$z_5$

- Makine FA1 veya FA2 tarafından kabul , FA3 tarafından kabul için yeterli idir

### 3. Kısım Kanıt 1, 2. Kural

- $z_3$ 't eysekvea harf ini okuyorsak
  - FA1 için makineyi  $x_2$  durumuna getirir • FA2 için makineyi  $y_4$  durumuna getirir
- $z_3$ 't eysekveb harf ini okuyorsak
  - FA1 için makineyi  $x_1$  durumuna getirir • FA2 için makineyi  $y_1$  durumuna getirir

	$a$	$b$
$-x_1$	$x_2$	$x_1$
$x_2$	$x_3$	$x_1$
$+x_3$	$x_3$	$x_3$

	$a$	$b$
$\pm y_1$	$y_3$	$y_2$
$y_2$	$y_4$	$y_1$
$y_3$	$y_1$	$y_4$
$y_4$	$y_2$	$y_3$

$$z_6 = x_2 \quad \text{or} \quad y_4$$

	$a$	$b$
$\pm z_1$	$z_2$	$z_3$
$z_2$	$z_4$	$z_5$
$z_3$	$z_6$	$z_1$

### 3. Kısım Kanıt 1, 2. Kural

- $z_4$ 't eysekvea harf ini okursak • FA1 için  
makineyi  $x_3$  durumuna (son durum) getirir • FA2 için  
makineyi  $y_3$  durumuna getirir
- $z_4$ 't eysekveb harf ini okursak • FA1 için  
makineyi  $x_3$  durumuna (son durum) getirir • FA2 için  
makineyi  $y_2$  durumuna getirir

	$a$	$b$
$-x_1$	$x_2$	$x_1$
$x_2$	$x_3$	$x_1$
$+x_3$	$x_3$	$x_3$

	$a$	$b$
$\pm y_1$	$y_3$	$y_2$
$y_2$	$y_4$	$y_1$
$y_3$	$y_1$	$y_4$
$y_4$	$y_2$	$y_3$

$$+z_7 = x_3 \quad \text{or} \quad y_3$$

$$+z_8 = x_3 \quad \text{or} \quad y_2$$

	$a$	$b$
$\pm z_1$	$z_2$	$z_3$
$z_2$	$z_4$	$z_5$
$z_3$	$z_6$	$z_1$

### 3. Kısım Kanıt 1, 2. Kural

If we are in  $z_5$  and we read an  $a$ , we go to  $x_2$  or  $y_2$ , which we shall call  $z_9$ .

If we are in  $z_5$  and we read a  $b$ , we go to  $x_1$  or  $y_3$ , which we shall call  $z_{10}$ .

$$z_9 = x_2 \quad \text{or} \quad y_2$$

$$z_{10} = x_1 \quad \text{or} \quad y_3$$

If we are in  $z_6$  and we read an  $a$ , we go to  $x_3$  or  $y_2$ , which is our old  $z_8$ .

If we are in  $z_6$  and we read a  $b$ , we go to  $x_1$  or  $y_3$ , which is  $z_{10}$  again.

If we are in  $z_7$  and we read an  $a$ , we go to  $x_3$  or  $y_1$ , which is  $z_4$  again.

If we are in  $z_7$  and we read a  $b$ , we go to  $x_3$  or  $y_4$ , which is a new state,  $z_{11}$ .

$$+z_{11} = x_3 \quad \text{or} \quad y_4$$

If we are in  $z_8$  and we read an  $a$ , we go to  $x_3$  or  $y_4 = z_{11}$ .

If we are in  $z_8$  and we read a  $b$ , we go to  $x_3$  or  $y_1 = z_4$ .

If we are in  $z_9$  and we read an  $a$ , we go to  $x_3$  or  $y_4 = z_{11}$ .

If we are in  $z_9$  and we read a  $b$ , we go to  $x_1$  or  $y_1 = z_1$ .

If we are in  $z_{10}$  and we read an  $a$ , we go to  $x_2$  or  $y_1$ , which is our last new state,  $z_{12}$ .

$$+z_{12} = x_2 \quad \text{or} \quad y_1$$

If we are in  $z_{10}$  and we read a  $b$ , we go to  $x_1$  or  $y_4 = z_5$ .

If we are in  $z_{11}$  and we read an  $a$ , we go to  $x_3$  or  $y_2 = z_8$ .

If we are in  $z_{11}$  and we read a  $b$ , we go to  $x_3$  or  $y_3 = z_7$ .

If we are in  $z_{12}$  and we read an  $a$ , we go to  $x_3$  or  $y_3 = z_7$ .

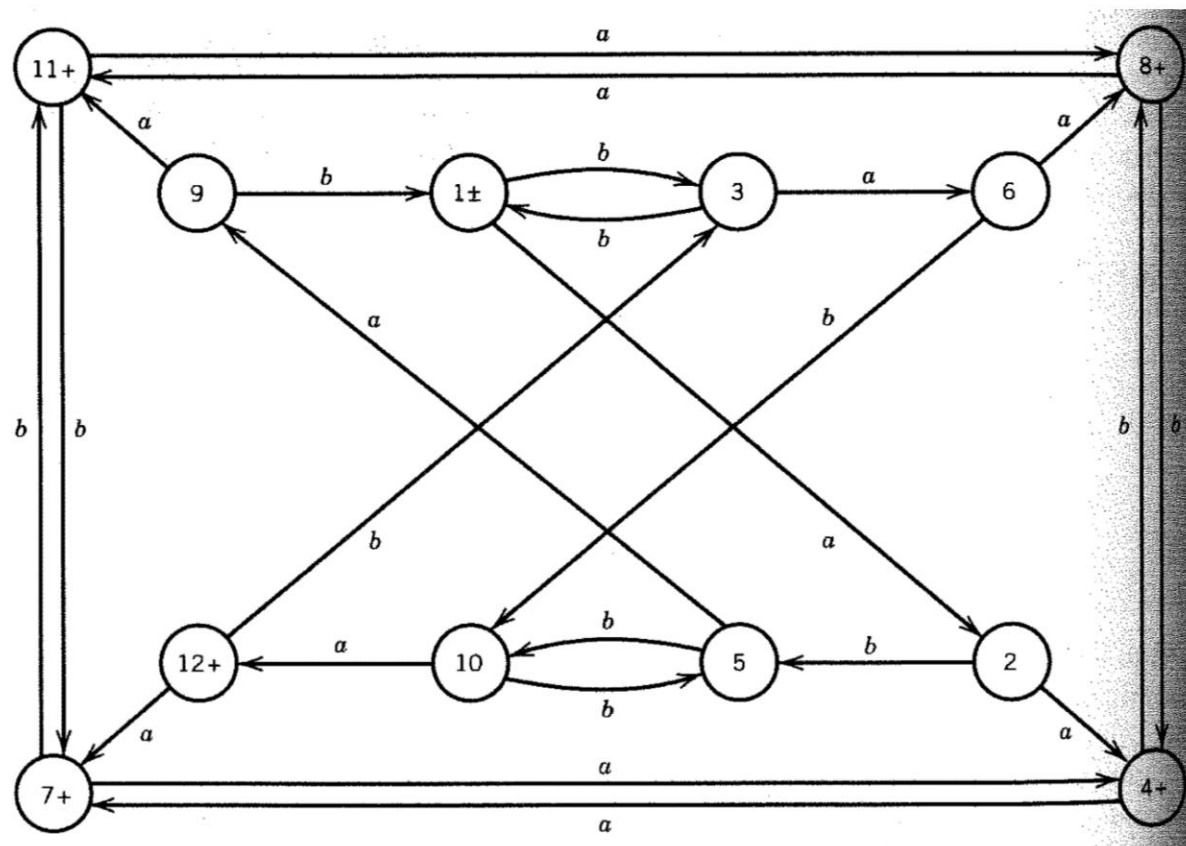
If we are in  $z_{12}$  and we read a  $b$ , we go to  $x_1$  or  $y_2 = z_3$ .

	$a$	$b$
$\pm z_1$	$z_2$	$z_3$
$z_2$	$z_4$	$z_5$
$z_3$	$z_6$	$z_1$
$+z_4$	$z_7$	$z_8$
$z_5$	$z_9$	$z_{10}$
$z_6$	$z_8$	$z_{10}$
$+z_7$	$z_4$	$z_{11}$
$+z_8$	$z_{11}$	$z_4$
$z_9$	$z_{11}$	$z_1$
$z_{10}$	$z_{12}$	$z_5$
$+z_{11}$	$z_8$	$z_7$
$+z_{12}$	$z_7$	$z_3$

### 3. Kısım Kanıt 1, 2. Kural

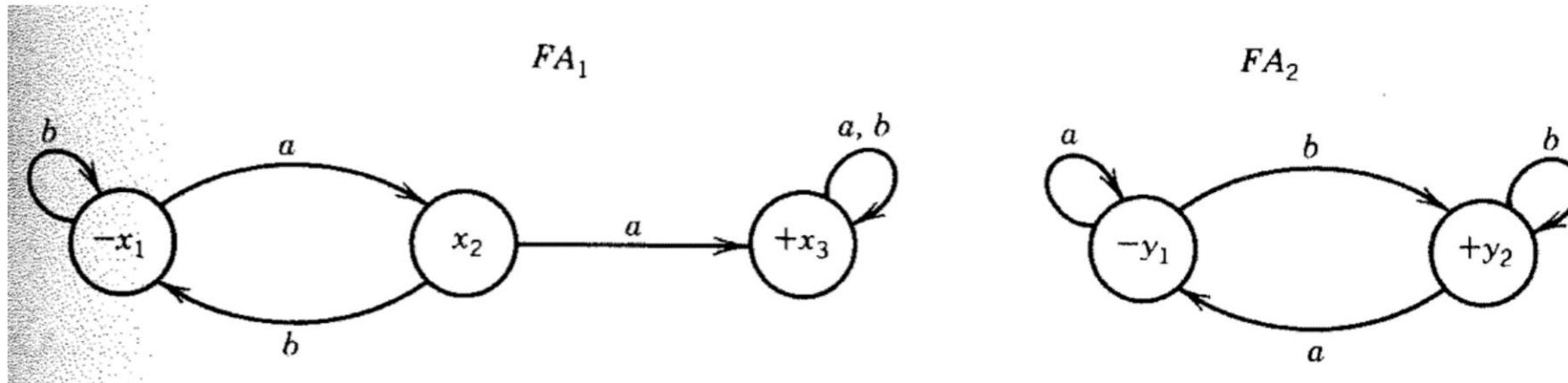
- FA3

	<i>a</i>	<i>b</i>
$\pm z_1$	$z_2$	$z_3$
$z_2$	$z_4$	$z_5$
$z_3$	$z_6$	$z_1$
$+z_4$	$z_7$	$z_8$
$z_5$	$z_9$	$z_{10}$
$z_6$	$z_8$	$z_{10}$
$+z_7$	$z_4$	$z_{11}$
$+z_8$	$z_{11}$	$z_4$
$z_9$	$z_{11}$	$z_1$
$z_{10}$	$z_{12}$	$z_5$
$+z_{11}$	$z_8$	$z_7$
$+z_{12}$	$z_7$	$z_3$



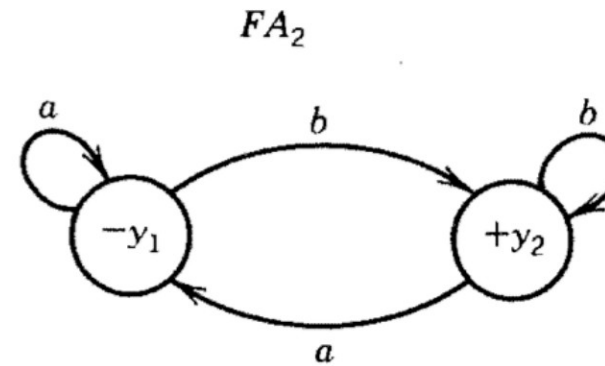
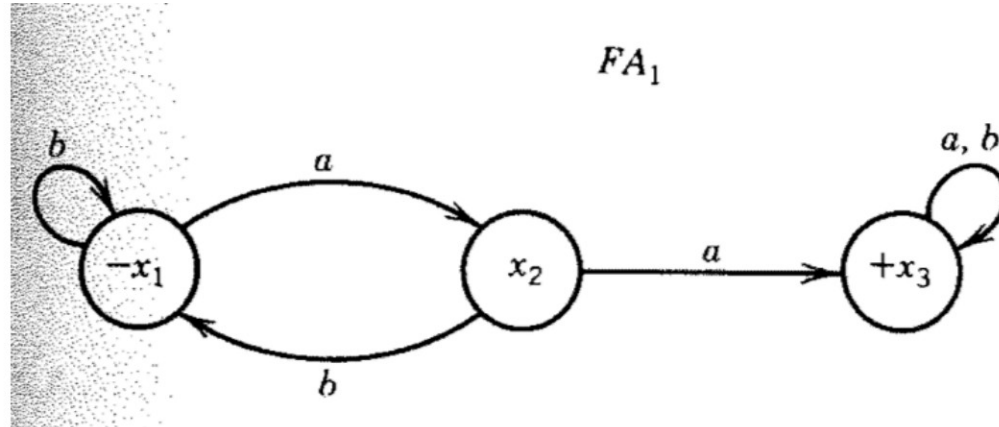
# Örnek

- FA1 , içindeki t a bul unan t ümkel imel eri kabul eder •
- FA2 , b il e bit en t ümkel imel eri kabul eder





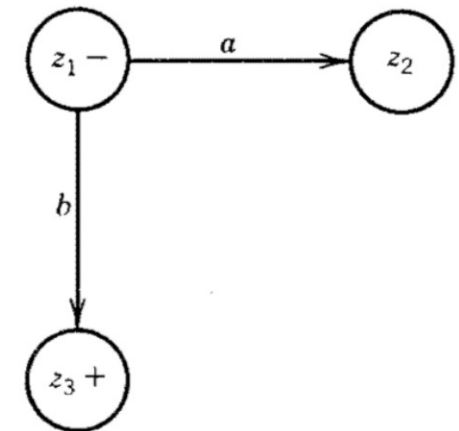
# Example



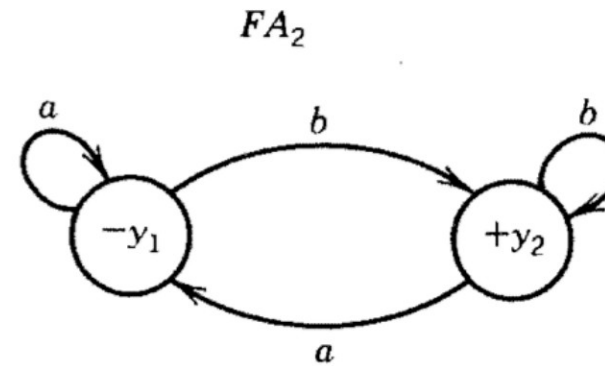
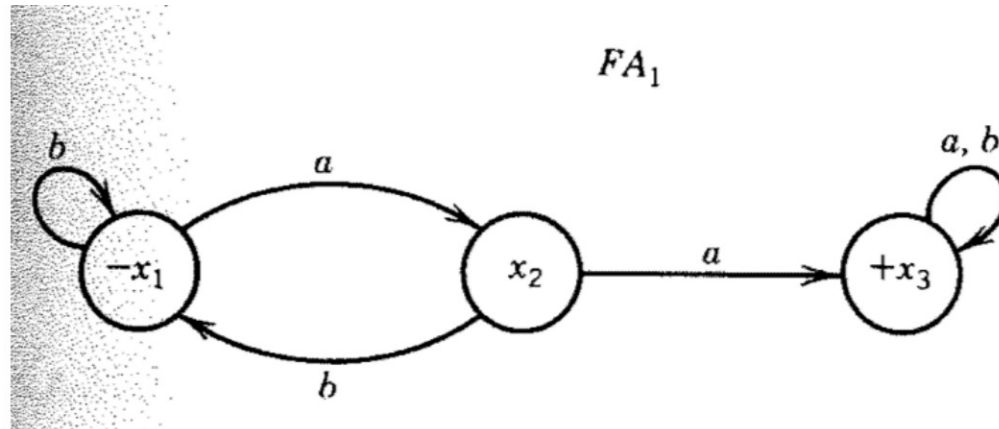
$$-z_1 = x_1 \quad \text{or} \quad y_1$$

In  $z_1$  if we read an  $a$ , we go to  $x_2$  or  $y_1 = z_2$

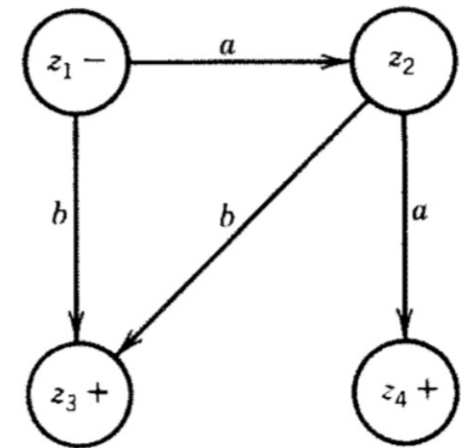
In  $z_1$  if we read a  $b$ , we go to  $x_1$  or  $y_2 = z_3$ , which is a final state since  $y_2$  is.



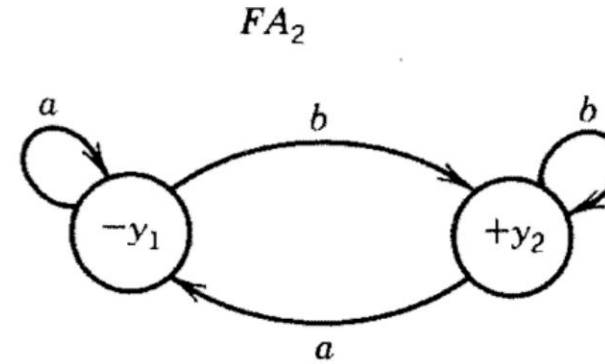
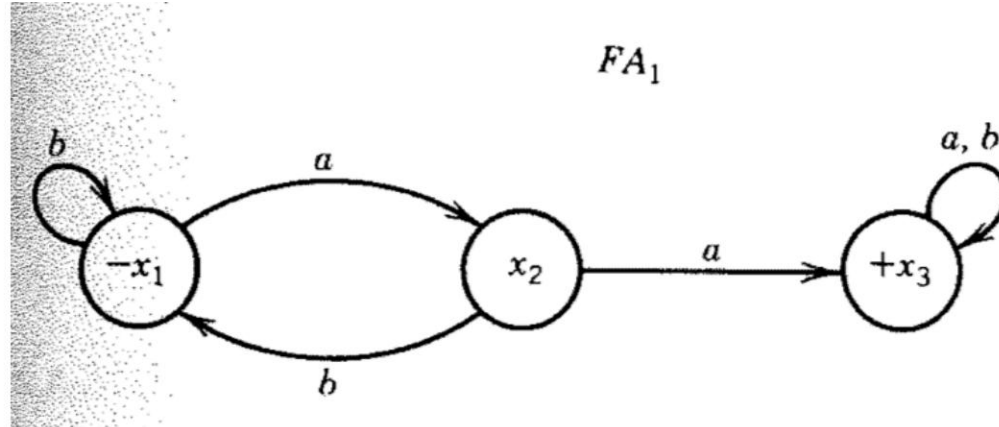
# Example



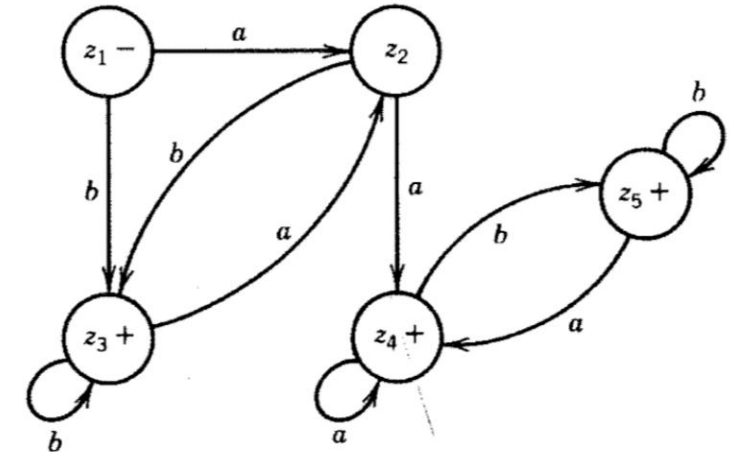
In  $z_2$  if we read an  $a$ , we go to  $x_3$  or  $y_1 = z_4$ , which is a final state because  $x_3$  is.  
 In  $z_2$  if we read a  $b$ , we go to  $x_1$  or  $y_2 = z_3$ .



# Örnek



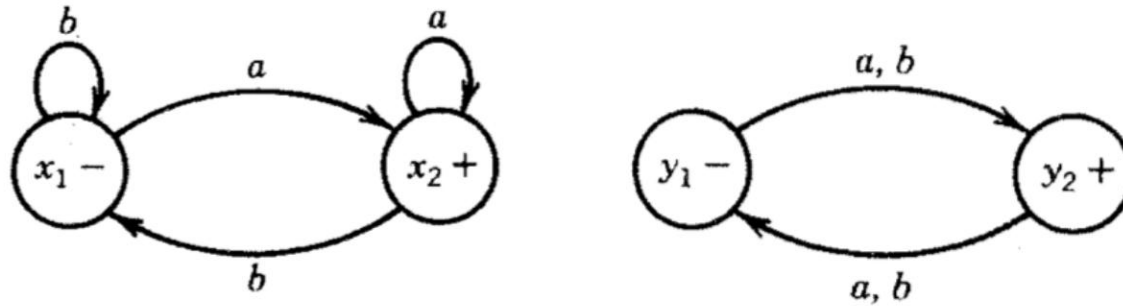
In  $z_3$  if we read an  $a$ , we go to  $x_2$  or  $y_1 = z_2$ .  
 In  $z_3$  if we read a  $b$ , we go to  $x_1$  or  $y_2 = z_3$ .  
 In  $z_4$  if we read an  $a$ , we go to  $x_3$  or  $y_1 = z_4$ .  
 In  $z_4$  if we read a  $b$ , we go to  $x_3$  or  $y_2 = z_5$ , which is a final state.  
 In  $z_5$  if we read an  $a$ , we go to  $x_3$  or  $y_1 = z_4$ .  
 In  $z_5$  if we read a  $b$ , we go to  $x_3$  or  $y_2 = z_5$ .



•  $z_6 = x_2$  veya  $y_2$  çıkmıyor, neden?

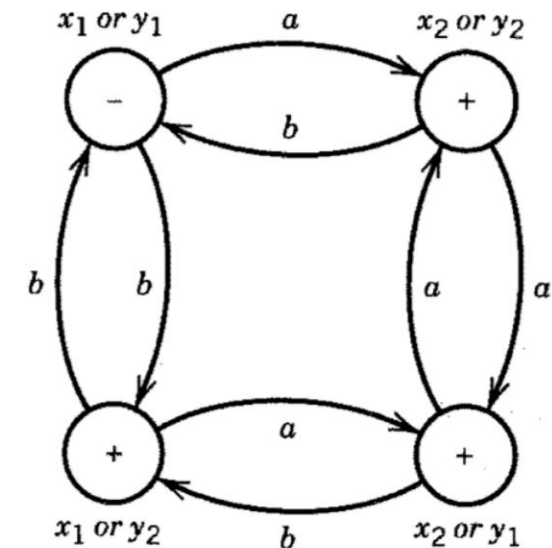
# Örnek

- FA1 , a ile biten tüm kelimeleri kabul eder
- FA2 , harfleri tek sayılı olan tüm kelimeleri kabul eder



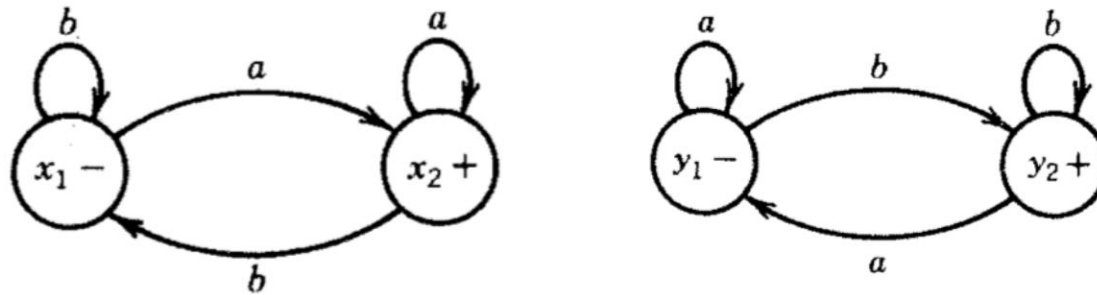
- FA3 , tek sayılı harflerle sahip olan veya a ile biten tüm kelimeleri kabul eder.

- + durumu olan tek durum- 'dir.  
durum

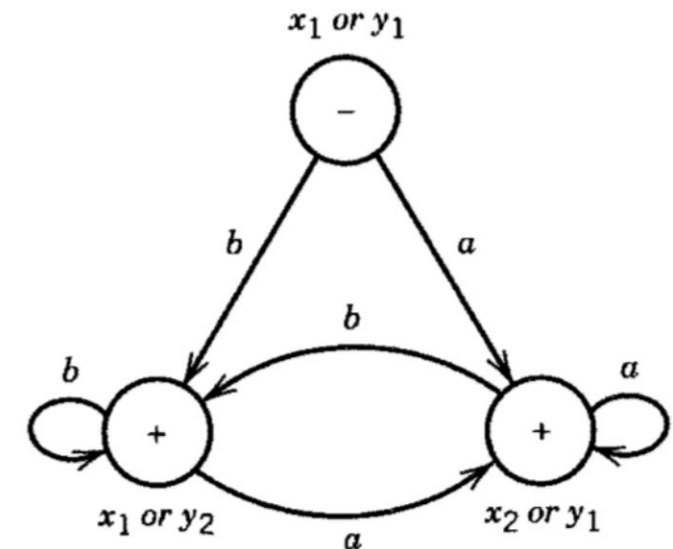


# Örnek

- FA1 , a ile biten tüm kelimeleri kabul eder
- FA2 , b ile biten tüm kelimeleri kabul eder



- FA3 , a veya b ile biten tüm kelimeleri kabul eder ( $\wedge$  hariç tüm kelimeler) • **x2 veya y2 durumuna ulaşılabilir**

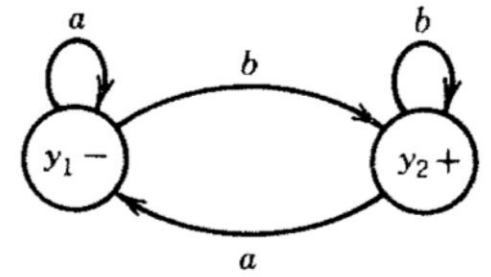
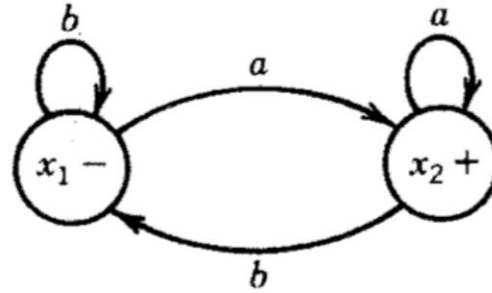


# Birliği Üretmek İçin Alternatif Bir Prosedür makine

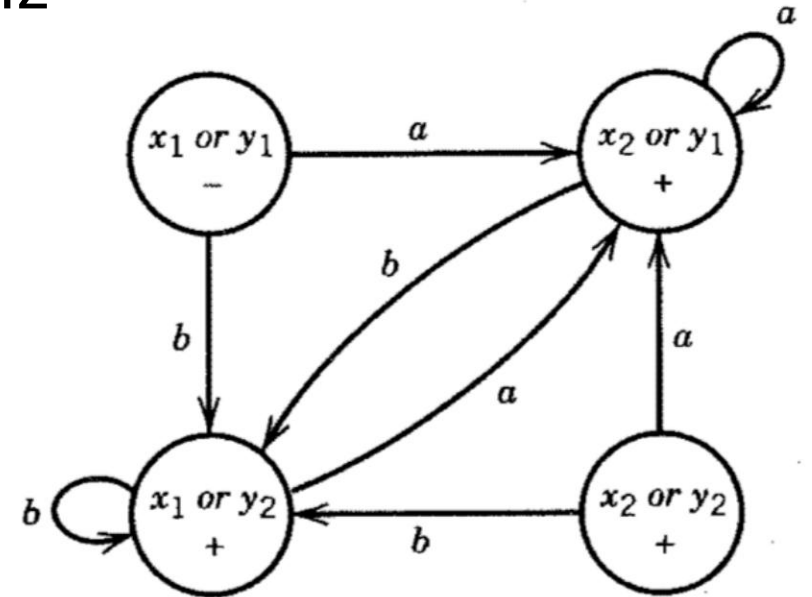
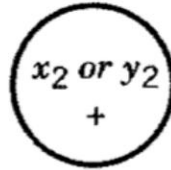
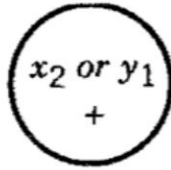
- FA1'in  $x_1, x_2, \dots$  durumlarına sahip olmasına izin verin
  - FA2'nin  $y_1, y_2, \dots$  durumlarına sahip olmasına izin verin
  - Birleştirme makinesini (FA3) başlatma için tüm  $x_1$  veya  $y_1$  durumlarına sahip olacak şekilde tanımlayabiliriz. FA3'teki durumların sayısı her zaman FA1 ve FA2'deki durum sayısının çarpımı olacaktır
  - FA3'teki her durumu için herhangi bir sırayla kenarlarını ve kenarlarını çizebiliriz
- 
- Daha önce yaptığımız şey, şu durumlarda yeni durumları yaratmaktır:  
gerektiği

# Bir İki Üretmek İçin Alternatif Bir Prosedür makine

- Önceki örneğin dört olası durumu a başlıyabiliriz.

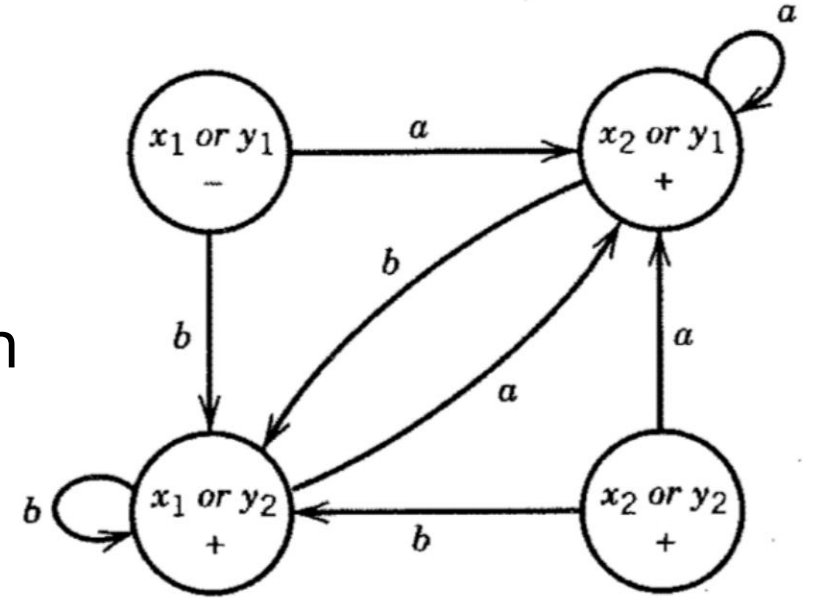


- Bu dört durumun her biri için iki kenar çizeceğiz



# Bir İki Üretmek İçin Alternatif Bir Prosedür Makine

- Bu, FA1 + FA2 birleşiminde için tamamen olası bir FA'dır • Ancak sağ alt taraftaki durumunun tamamen işeyaramaz olduğunu görüyoruz • Aslında herhangi bir dizi ile başlıyan girilmez -de
- Yararsız bir duruma sahip olmak FA'nın tanımına aykırı değildir.





### 3. Kısımların Kanıtı, 3. Kural

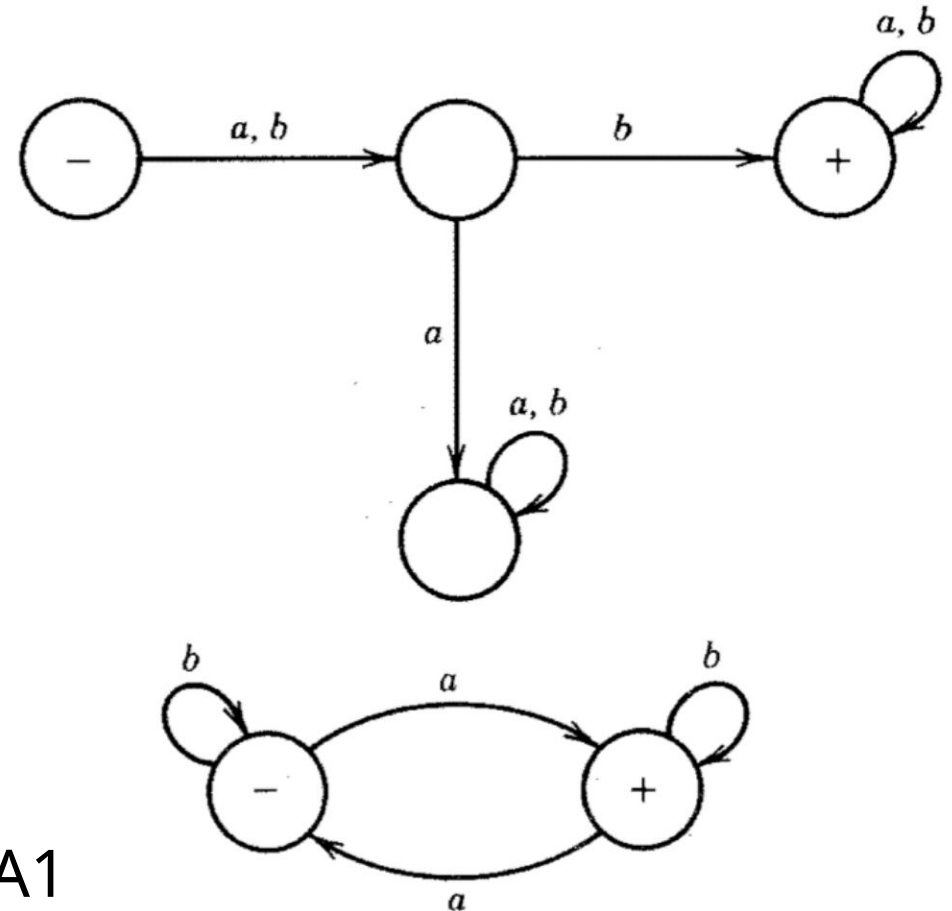
- Kural 3
  - Normal ifadesi  $r_1$  tarafından tanımlanan dil i kabul eden bir FA1 ver2 normal ifadesi  $r_2$  tarafından tanımlanan dil i kabul eden bir FA2 varsa, o zaman  $r_1 r_2$  birleştirilerek tanımlanan dil i, ürün dil i kabul eden bir FA3 vardır.

# Örnek

- L1: İkinci harfi bol an t tüm  
kelimelerin dil i

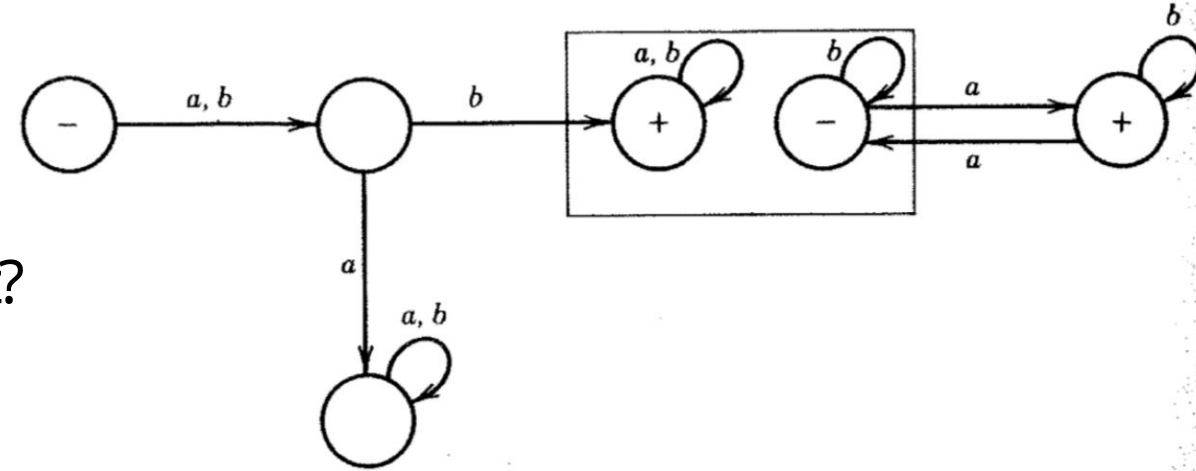
- D2: Tüm kelimelerin dil i  
a'ların t eksayısı ol an

- (ab)(abbaa) giriş dizisini dikkat eal ın • FA1  
il ebaşı ayın ve + il ebit irin • Kal an  
dizeyl eFA'ya at l ayın ve bit irin  
+

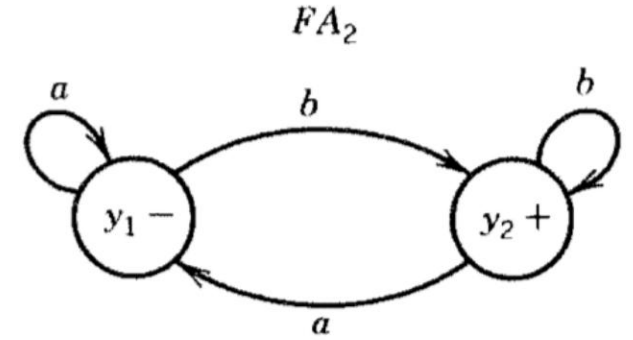
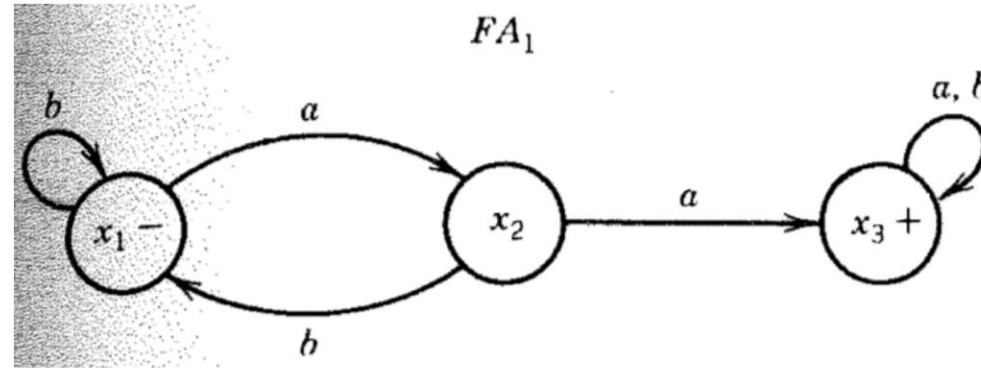


# Örnek

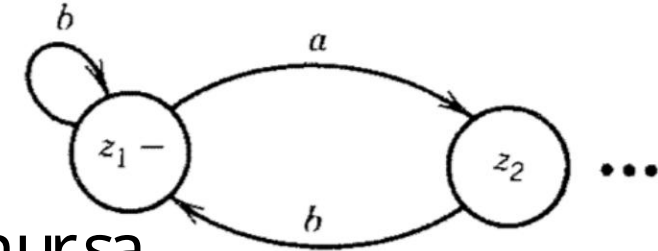
- Bu basit fikir işeyaramaz • Aynı ürün dil i için farklı bir giriş dizesi düşünün:  
ababbab
- (abab)(bab) kabul edil ir •  
(ab)(abbab) reddedil ir • Ne zaman atlayacağımızı nasıl bil ebil iriz?



# Örnek



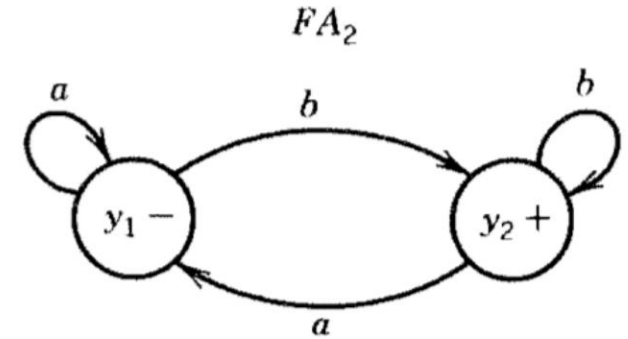
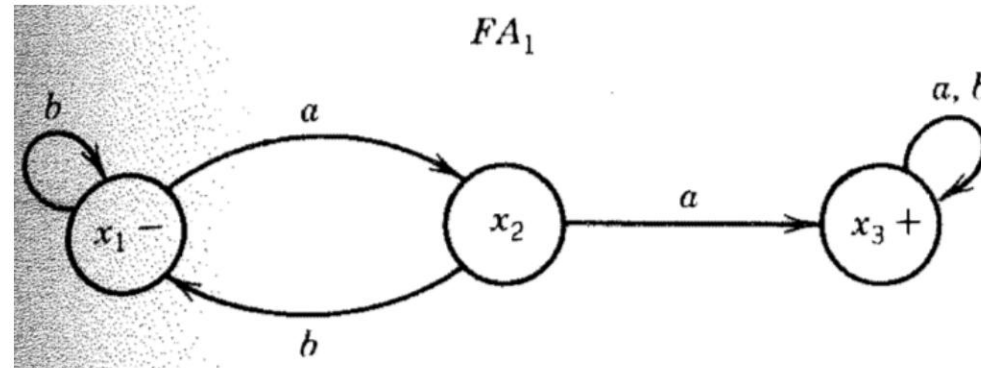
- Tamol arak  $x_1$ 'e benzeyen  $z_1$  durumuyla başlı ayın
  - Giriş dizisi yalnızca  $FA_1$ 'de çalışıyor
  - $z_1$ 'den,  $a$  b okunursa,  $x_1$ 'e dönmedir
  - $z_1$ 'den,  $a$  okunursa,  $x_2$ 'ye gitmedir
  - $z_2$ ,  $x_2$  ile aynıdır
  - $z_2$ 'den  $a$  okunursa,  $z_3$ 'e gitmedir
  - $z_3$ ,  $x_3$  ile aynıdır



- $x_3$ 'ün çift kimliği vardır
  - Ya  $FA_1$ 'de son duruma geldiğimiz anlamına gelir
  - Yoksa geçiyoruz

$$z_3 = \begin{cases} x_3, \text{ and we are still running on } FA_1 \\ \text{or} \\ y_1, \text{ and we have begun to run on } FA_2 \end{cases}$$

# Örnek



- z3't e yaz vebir a okuyoruz , üç seçeneğimiz var

We are back in  $x_3$  continuing to run the string on  $FA_1$

or

we have just finished on  $FA_1$  and we are now in  $y_1$   
beginning to run on  $FA_2$

or

we have looped from  $y_1$  back to  $y_1$  while already running on  $FA_2$

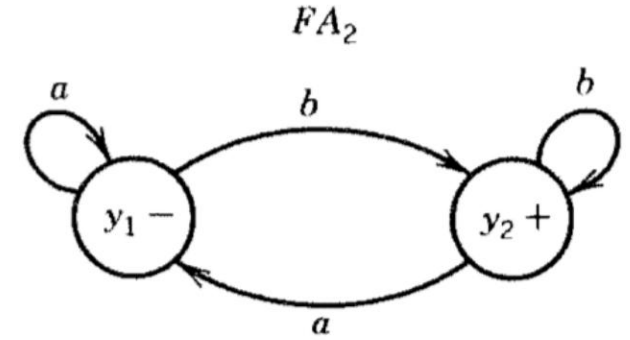
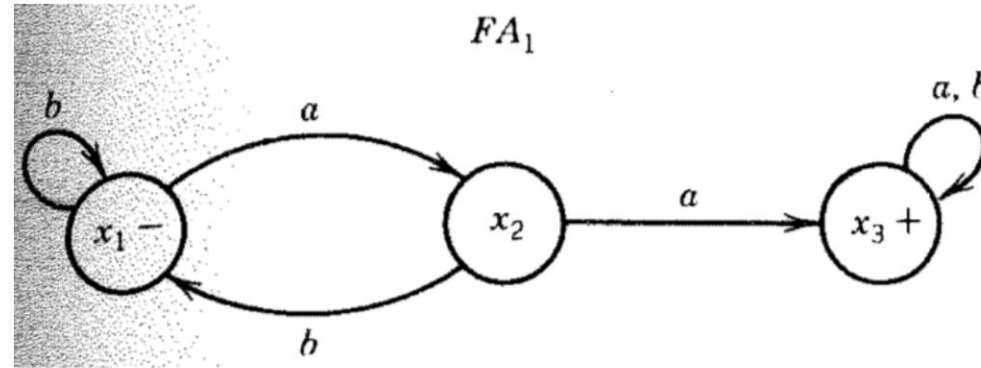
=  $x_3$  or  $y_1$

(because being in  $y_1$  is the same whether we are  
there for the first time or not)

=  $z_3$

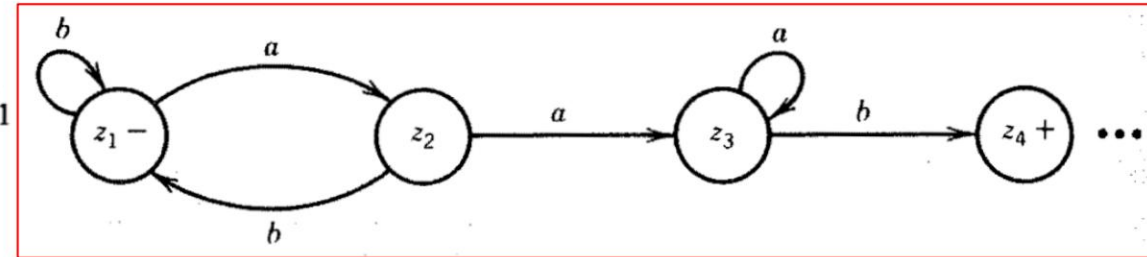
• a'yı okumak bizi z3't en z3'e geri götürür

# Örnek



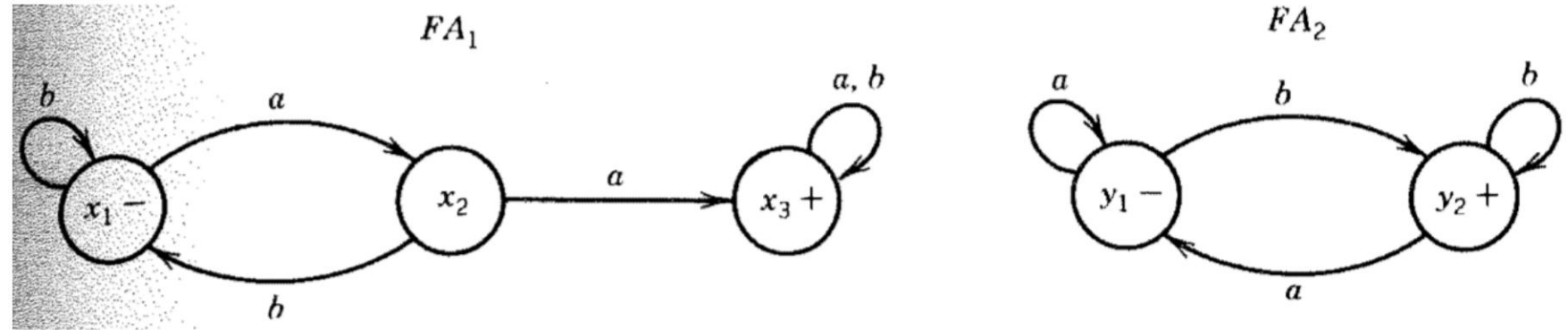
- $z_3$ 'deyiz ve a b okuyoruz, dört anlamlı olan  $z_4$ 'e gidiyoruz

$+z_4 = \begin{cases} \text{We are still in } x_3 \text{ continuing to run on } FA_1 \\ \text{or} \\ \text{we have just finished running on } FA_1 \text{ and are now in } y_1 \text{ on } FA_2 \\ \text{or} \\ \text{we are now in } y_2 \text{ on } FA_2, \text{ having reached there via } y_1 \end{cases}$   
 $= x_3 \text{ or } y_1 \text{ or } y_2$



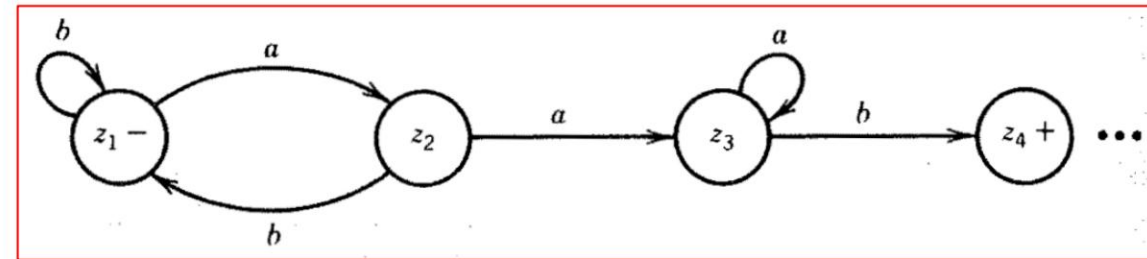
- Bir yol  $z_4$ 'te bitiyorsa, bu yol iki kısma ayrılabilir:
  - Birinci kısım  $x_1$ 'den  $x_3$ 'e
  - İkinci kısım  $y_1$ 'den  $y_2$ 'ye
  - Bu nedenle kabul edilmiştir.  $z_4$  son bir durumdur

# Örnek



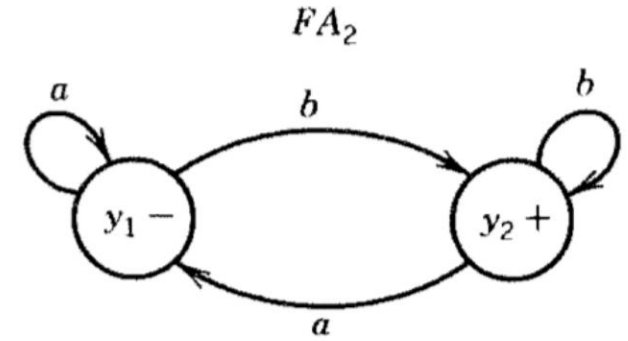
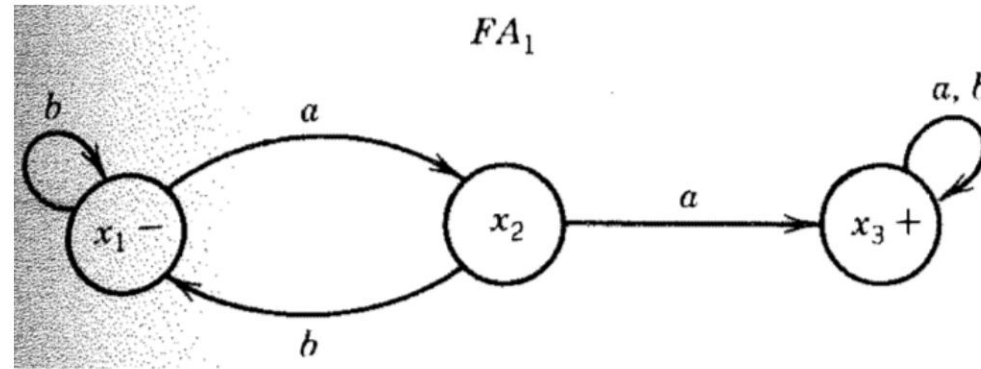
- $z_4$ 't e yazabiliriz a okuyoruz, seçeneğimiz:

{ remaining in  $x_3$  and continuing to run on  $FA_1$   
 or  
 having just finished  $FA_1$  and beginning at  $y_1$   
 or  
 having moved from  $y_2$  back to  $y_1$  in  $FA_2$   
 =  $x_3$  or  $y_1$



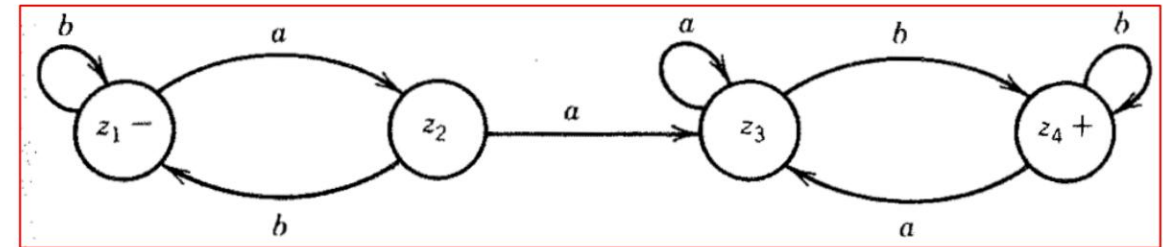
- But a model olarak  $z_3$ 'ün tanımlıdır •  
 Eğer  $z_4$ 't e yazabiliriz a okuyorsak  $z_3$ 'e geri dönüyoruz

# Örnek



- $z_4$ 'deyiz ve a b okuyoruz, seçimlerimiz:

{ remaining in  $x_3$  and continuing to run on  $FA_1$   
 or  
 having just finished  $FA_1$  and beginning at  $y_1$   
 or  
 having looped back from  $y_2$  to  $y_2$  running on  $FA_2$   
 =  $x_3$  or  $y_1$  or  $y_2$   
 =  $z_4$



- Bu  $z_4$ 'ün tanımdır • Eğer  $z_4$ 'teyiz ve a b okuyorsak  $z_4$ 'e geri döneriz



### 3. Kısım Kanıt 1, 4. Kural

- Kural 4
  - $r$  düzenli bir ifade ve  $FA1$  sonlu bir otomat ise tam olarak  $r$  tarafından tanımlanan dil kabul eder, o zaman tam olarak  $r^*$  tarafından tanımlanan dil kabul edecek olan  $FA2$  adında bir FA vardır.