

LOJİK KAPILAR (GATES)

- ❖ ‘Değil’ veya ‘Tümleme’ Kapısı (NOT Gate)
- ❖ ‘Ve’ Kapısı (AND Gate)
- ❖ ‘Veya’ Kapısı (OR Gate)
- ❖ ‘Vedeğil’ Kapısı (NAND Gate)
- ❖ ‘Veyadeğil’ Kapısı (NOR Gate)
- ❖ ‘Özelveya’ Kapısı (EXOR Gate)
- ❖ ‘Özelveyadeğil’ Kapısı (EXNOR Gate)
- ❖ Lojik İfadelerden Lojik Devrelerin Elde Edilmesi

LOJİK KAPILAR

Lojik devrelerin en temel elemanı, lojik kapılardır. Kapılar, lojik değişkenlerin değerlerini (1 veya 0) giriş olarak alırlar, bu değerler üzerinde işlem yaparlar ve lojik çıkış üretirler.

Kapılar transistör, diyot, direnç, kondansatör gibi devre elemanlarından oluşur. Düşük enerji tüketimi, az yer kaplaması, ek bağlantı içermemesi, ekonomik olması gibi nedenlerle entegre devre olarak piyasadan temin edilebilirler.

Temel olarak 7 kapı vardır bunlar;

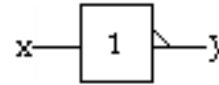
‘ve’ (AND), ‘veya’ (OR), ‘değil’ (NOT),

‘vedeğil’ (NAND), ‘veyadeğil’ (NOR),

‘özelveya’ (EXOR) ve ‘özelveyadeğil’ (EXNOR) kapılarıdır.

‘Değil’ veya ‘Tümleme’ Kapısı (NOT Gate)

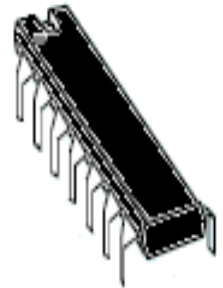
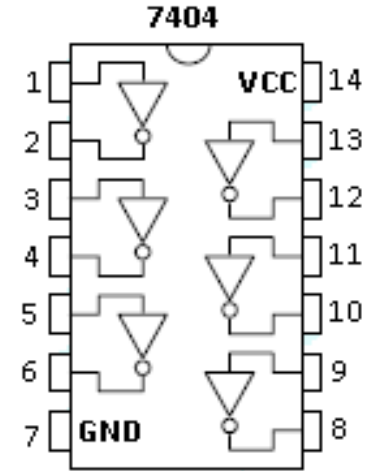
Girişine gelen lojik sinyalleri tersler. Yani, girişine lojik 1 değeri geldiyse çıkışı 0, 0 değeri geldiyse çıkışı 1 olur. Aşağıdaki sembollerle gösterilir;



Değil veya tümleyen kapısının doğruluk tablosu;

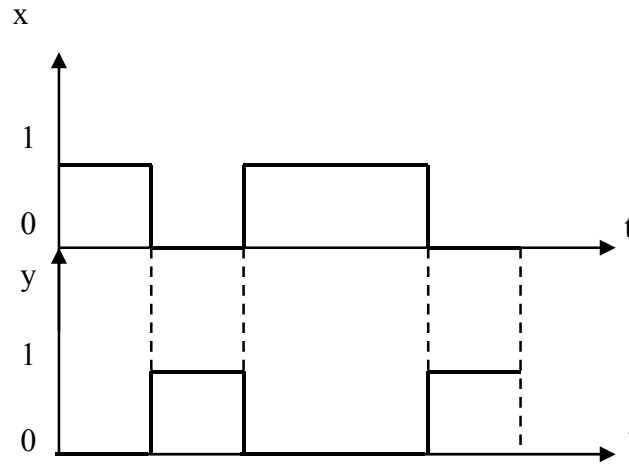
x	y
0	1
1	0

lojik ifadesi $y = x'$ veya $y = \bar{x}$ olarak ifade edilir.



Örnek:

Değil kapısının girişine x sinyali uygulandığında, çıkışındaki sinyal (y) aşağıdaki gibidir;



‘Ve’ Kapısı (AND Gate)

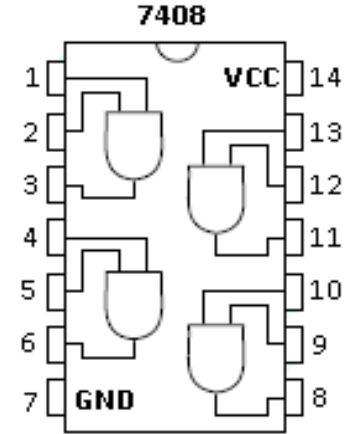
Mantıksal çarpma olarak bilinir. Bir AND kapısı iki veya daha fazla girişe sahip olabilir, tek çıkışı vardır. Aşağıdaki sembollerle gösterilir;



AND kapısının doğruluk tablosu;

x	y	z
0	0	0
0	1	0
1	0	0
1	1	1

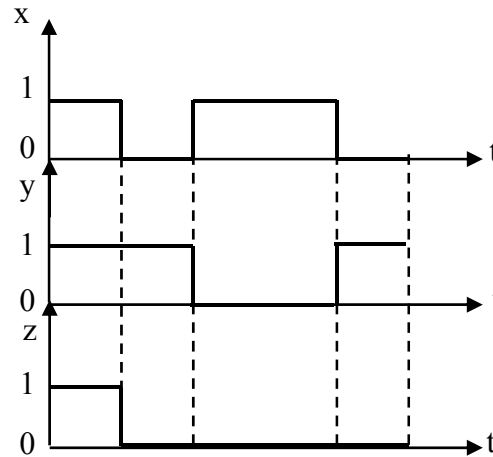
lojik ifadesi $z = x.y$ veya $z = xy$



AND kapısının bütün girişleri 1 iken çıkışı lojik 1 seviyesindedir, herhangi bir giriş lojik 0 ise çıkış 0'dır.

Örnek:

2 girişli bir AND kapısının girişlerine x ve y sinyalleri uygulandığında çıkışındaki sinyal (z) aşağıdaki gibidir;

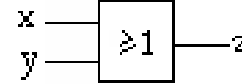


Not: AND işlemi, ikili çarpım ile aynı işleve sahiptir.

Yani, $0.0 = 0$, $0.1 = 0$, $1.0 = 0$, $1.1 = 1$

‘Veya’ Kapısı (OR Gate)

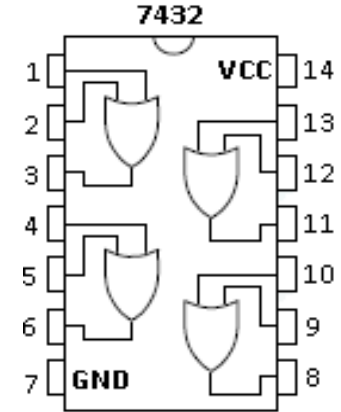
OR kapısı mantıksal toplama olarak da bilinir. Aşağıdaki sembollerle gösterilir;



OR kapısının doğruluk tablosu;

x	y	z
0	0	0
0	1	1
1	0	1
1	1	1

lojik ifadesi $z = x + y$

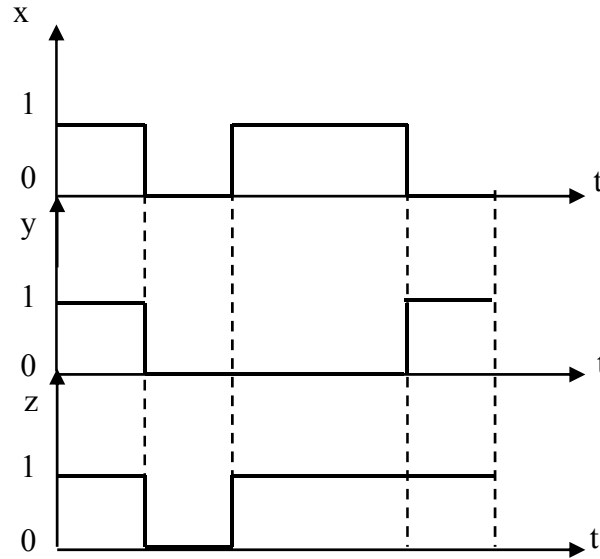


İkili toplama işlemine benzerdir ancak her iki giriş de 1 olduğunda sonuç 1'dir, oysa ikili toplamada toplam 0, elde 1'dir.

OR kapısının girişlerinden en azından biri lojik 1 seviyesine sahipse çıkış 1'dir.

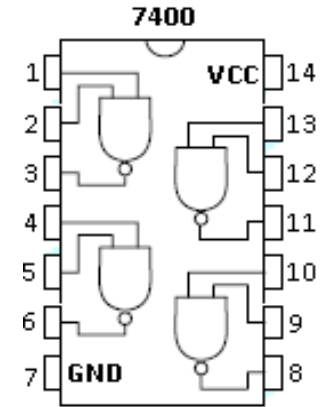
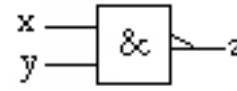
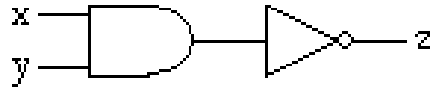
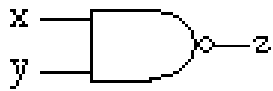
Örnek:

OR kapısının girişlerine x ve y sinyalleri uygulandığında çıkışı (z) aşağıdaki gibidir;



‘Vedeğil’ Kapısı (NAND Gate)

Üniversal kapı olarak kullanılırlar. Yani NAND kapısı kullanılarak AND, OR veya NOT kapıları elde edilebilir. AND kapısının çıkışına NOT kapısının bağlanmış hali olarak da düşünülebilir. Aşağıdaki sembollerle gösterilir;



NAND kapısının doğruluk tablosu;

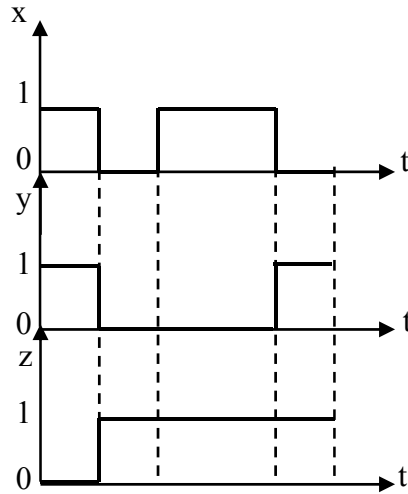
x	y	z
0	0	1
0	1	1
1	0	1
1	1	0

Lojik ifadesi $z = (x.y)'$ veya $z = \overline{x.y}$

NAND kapısının tüm girişleri 1 iken çıkışı 0, herhangi bir girişi 0 değerini alırsa çıkışı 1 olur.

Örnek:

NAND kapısının girişlerine x ve y sinyalleri uygulandığında çıkışındaki sinyal (z) aşağıdaki gibidir;

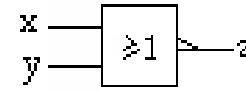
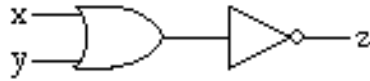


OR işlemiyle NAND işlemi arasında bir benzerlik olduğu söylenebilir. OR kapısında bir ya da daha fazla sayıda giriş 1 iken çıkış 1'dir, NAND'de ise bir veya daha fazla sayıda giriş 0 iken çıkış 1'dir.



‘Veyadeğil’ Kapısı (NOR Gate)

NAND kapısı gibi üniversal bir kapıdır. OR kapısının çıkışına NOT kapısının bağlanmış hali olarak da düşünülebilir. Aşağıdaki sembollerle gösterilir;



NOR kapısının doğruluk tablosu;

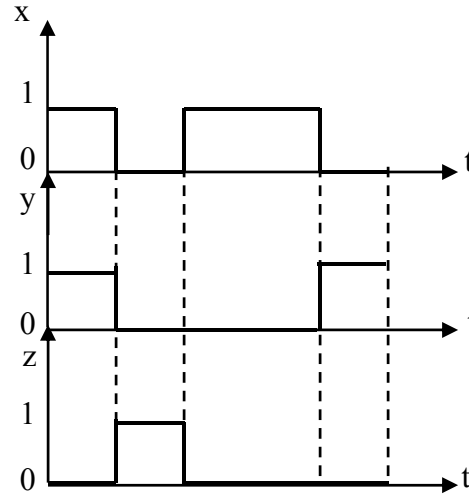
x	y	z
0	0	1
0	1	0
1	0	0
1	1	0

Lojik ifadesi $z = (x + y)'$ veya $z = \overline{x + y}$

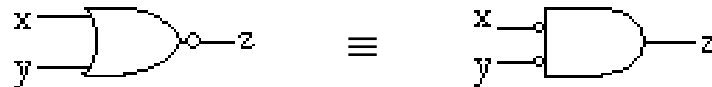
NOR kapısının tüm girişleri 0 ise çıkışı 1, aksi halde çıkışı 0'dır.

Örnek:

NOR kapısının girişlerine x ve y sinyalleri uygulandığında çıkışındaki sinyal (z) aşağıdaki gibidir;



AND işlemiyle NOR işlemi arasında bir benzerlik olduğu söylenebilir. AND'te tüm girişler 1 değerini aldığı anda çıkışı 1'dir, NOR'da ise tüm girişler 0 değerini aldığı anda çıkış 1'dir.



‘Özelveya’ Kapısı (EXOR Gate)

EXOR kapısı, AND ve OR kapıları kullanılarak elde edilebilir. Farklılık kapısı olarak da anılır. Yerine getirdiği işlevin öneminden dolayı temel lojik kapılar gibi kullanılırlar. Aşağıdaki sembollerle gösterilir;



EXOR kapısının doğruluk tablosu;

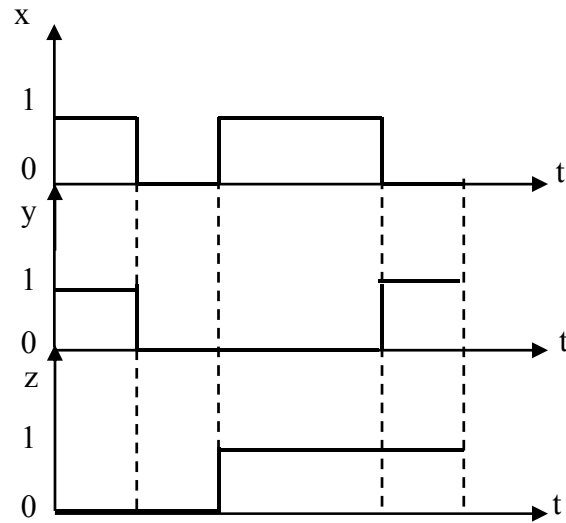
x	y	z
0	0	0
0	1	1
1	0	1
1	1	0

lojik ifadesi $z = x \oplus y$ veya $z = x' y + x y'$

- ❖ Çok sayıda girişe sahip EXOR kapısının çıkışı, girişlerindeki 1'lerin tek sayıda olması durumunda lojik 1 seviyesinde olacaktır.
- ❖ Birleşme özelliği vardır: $a \oplus (b \oplus c) = (a \oplus b) \oplus c = a \oplus b \oplus c$

Örnek:

EXOR kapısının girişlerine x ve y sinyalleri uygulandığında çıkışındaki sinyal (z) aşağıdaki gibidir;



‘Özelveyadeğil’ Kapısı (EXNOR Gate)

EXNOR kapısı da EXOR kapısı gibi, AND ve OR kapıları kullanılarak elde edilebilir. Eşitlik kapısı olarak da anılır. Yerine getirdiği işlevin öneminden dolayı temel lojik kapılar gibi kullanılırlar. Aşağıdaki sembollerle gösterilir;



EXNOR kapısının doğruluk tablosu;

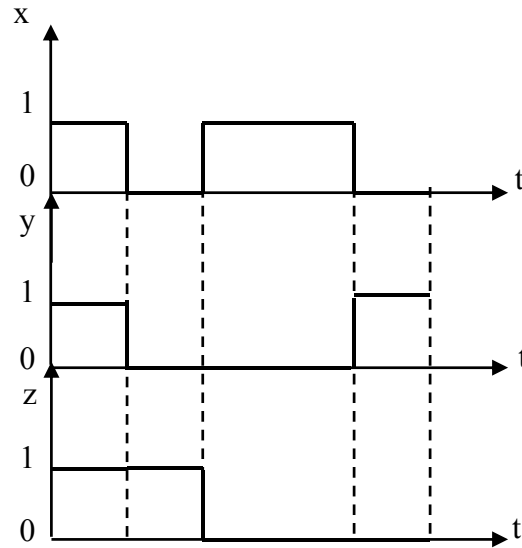
x	y	z
0	0	1
0	1	0
1	0	0
1	1	1

lojik ifadesi $z = x \otimes y$ veya $z = x.y + x'.y'$

- ❖ EXNOR kapısı, EXOR kapısının değildir. Yani $z = (x \oplus y)'$
- ❖ Çok sayıda girişe sahip EXNOR kapısının çıkışı girişlerindeki 0'ların çift sayıda olması durumunda 1'dir.
- ❖ Birleşme özelliği vardır: $a \otimes (b \otimes c) = (a \otimes b) \otimes c = a \otimes b \otimes c$

Örnek:

EXNOR kapısının girişlerine x ve y sinyalleri uygulandığında çıkışındaki sinyal (z) aşağıdaki gibidir;



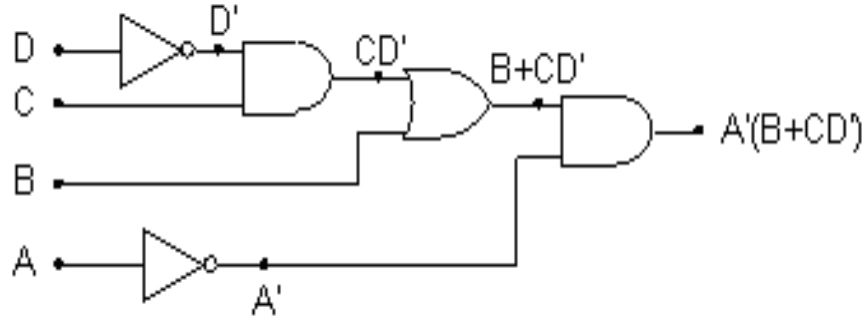
Lojik İfadelerden Lojik Devrelerin Elde Edilmesi

Lojik ifadelerden lojik devrelerin elde edilmesi işleminde, önce lojik ifadedeki değişkenlerin deęilleri alınması gerekiyorsa NOT kapıları kullanılarak bu işlem yapılır. Daha sonra çarpım ifadelerine bakılır ve bu çarpımlar AND kapıları kullanılarak gerçekleştirilir. Lojik ifadedeki toplanlar için de OR kapıları kullanılır. AND ve OR işlemlerinden sonra tümleyen işlemi gerekiyorsa yine NOT kapısı kullanılır ve elde edilen devre parçaları bir araya getirilir. Önemli olan şey, lojik ifadedeki parantezler ve çarpım işlemlerinin toplam işlemlerine göre öncelikli olduğudur.

Örnek:

$F = A' \cdot (B + C.D')$ lojik ifadesini temel kapılar kullanarak gerçekleştirelim.

İlk olarak parantez içindeki terimlere bakılır; bir çarpma ve bir de toplama işlemi mevcuttur. Önce çarpma işlemi AND kapısı kullanılarak gerçekleştirilir (D nin tümleyenini aldıktan sonra) daha sonra da toplama işlemi OR kapısı kullanılarak gerçekleştirilir. Parantezin içindeki ifade elde edildikten sonra parantezin dışındaki ifadelere bakılır. Parantezin dışındaki A nın tümleyeni, NOT kapısı kullanılarak elde edilir. Daha sonra da elde edilen devre parçaları AND kapısı kullanılarak bir araya getirilir.



Verilen bir lojik devrenin lojik ifadesini elde etmek için de, devrenin çıkışını oluşturan kapıya kadar olan tüm kapıların çıkışları bulunur ve bu çıkışlar bir araya getirilir.