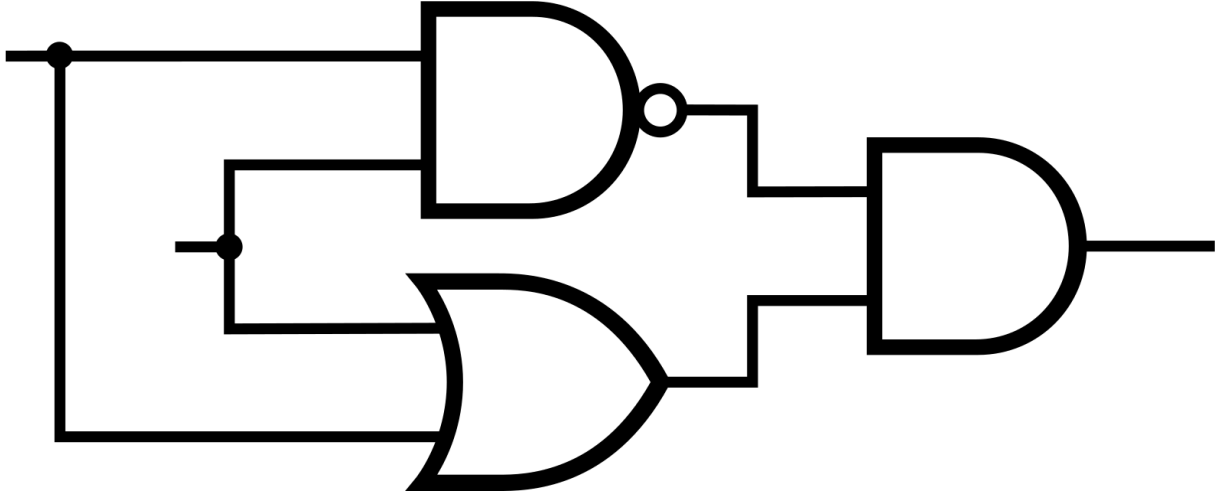


METE HOCA LOGICSHIELD



LOJİK KAPI ARDUINO MODÜLÜ EĞİTİM ve KULLANIM KİTABI

(Son güncelleme: 2 Haziran 2022)

www.metehoca.com

METE HOCA

DİĞER METE HOCA ÜRÜNLERİ

ARDUINO ÖNCESİ TEMEL ELEKTRONİK EĞİTİM ve DENEY SETİ

ARDUINO'da iyi olmak için öncelikle ELEKTRONİK'te iyi olmak gerekir.

Arduino Öncesi Temel Elektronik Eğitim ve Deney Seti, Arduino'ya doğru ve güçlü bir şekilde başlamak için gereken temel elektronik bilgisini önemli detayları atlamadan, gereksiz teknik bilgilerden arındırarak ve olabildiğince eğlenceli bir şekilde vermeyi hedefleyen bir settir.

Oldukça kapsamlı hazırlanmış bu set ile elektrik ve elektroniğin temellerini **METE HOCA** farkıyla hızlı ve basit bir şekilde öğrenecek, Arduino eğitimlerinin çok önemli olmasına rağmen çoğu zaman göz ardı edilen elektronik devre bileşenleri konusunda çok daha bilinçli olabileceksiniz.

ARDUINO'YA GÜÇLÜ BAŞLANGIÇ EĞİTİM ve PROJE SETİ

ÖZGÜN ve KULLANIŞLI projeler yapabilmek için ARDUINO'yu doğru öğrenmek gerekir.

Arduino'ya Güçlü Başlangıç Eğitim ve Proje Seti, bayrağı öncül set olan Arduino Öncesi Temel Elektronik Eğitim ve Deney Seti'nin bıraktığı yerden devralıyor ve geleceğin tasarımcılarını Arduino ile tanıştırıyor.

METE HOCA'nın "gereksiz bilgilerden arındırılmış basit ve eğlenceli anlatım" felsefesini devam ettiren bu set ile Arduino'yu en temelden, ilginç kuruluş hikâyesinden başlayarak keşfedeceksiniz.

ARDUINO HEADER ETİKETLERİ

Klon Arduino kullanırken yapılan yanlış bağlantılar projemizin çalışmamasından, olası bir kısa devre nedeniyle Arduino board'umuzun, proje modüllerinin, hatta bilgisayarımızın bozulmasıyla sonuçlanabiliyor.

Orijinal Arduino'larda bulunan baskılar yüksek pin header yapılarının getirdiği hatalı bağlantıları asgariye indiriyor. Klon Arduino board'lar ise ucuzluk adına bu özelliğe sahip değil ve hatalara davetiye çıkarıyorlar.

METE HOCA ARDUINO HEADER ETİKETİ serisi özellikle yeni öğrenen küçüklerin kolaylıkla yapabilecekleri bu hataların önüne geçebilmenizi sağlıyor. Header etiketleri Arduino Uno, Arduino Mega ve Protoshield için hazırlandı.

METE HOCA LOGICSHIELD

LogicShield, dijital elektroniğin mantıksal işleyişini ve kodlama yapısını anlamak için hayati önemde olan **lojik kapı**ların öğrenciler tarafından deneyimlenmesini sağlayan bir Arduino shield'dir. Arduino Uno ve Mega'da kullanılabilir.

Tüm hakları saklıdır.

Mayıs 2022, Mete K. Atay

www.metehoca.com

İÇİNDEKİLER

Mete Hoca LogicShield Nedir?	2
Kullanırken Nelere Dikkat Edilmelidir?	3
Lojik Kapı Nedir?	4
Lojik Kapı Çeşitleri	5
AND (VE) Kapısı	6
OR (VEYA) Kapısı	7
NOT (DEĞİL) Kapısı	8
XOR (ÖZELVEYA) Kapısı	9
NAND (VEDEĞİL) Kapısı	10
NOR (VEYADEĞİL) Kapısı	11
XNOR (ÖZELVEYADEĞİL) Kapısı	12
BUFFER (TAMPON) Kapısı	13
Lojik Kapıların Toplu Doğruluk Tablosu (Truth Table)	14
Lojik Kapıların Bir Arada Kullanılmaları	15
Lojik Kapı Alıştırmaları	17
Arduino IDE'de Lojik İfade Kullanımı	19

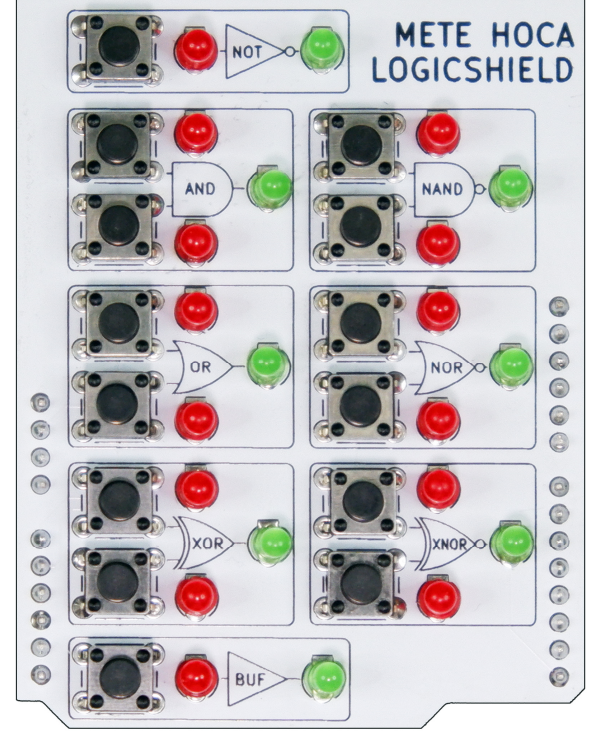
METE HOCA LOGICSHIELD NEDİR?

Mete Hoca LogicShield, dijital elektroniğin temellerini ve kodlama yapısının mantıksal işleyişini anlamak için hayati önemde olan **lojik kapıların** öğrenciler tarafından deneyimlenmesini sağlayan bir **Arduino** modülüdür.

Arduino dünyasında **shield** olarak anılan bu modüller Uno'nun pinlerine üstten takılacak şekilde tasarlanmışlardır ve Uno'ya kablo bağlantısına gerek kalmadan belirli işlevler eklenebilmesini sağlarlar. LogicShield de lojik kapıların nasıl çalıştıklarının uygulamalı olarak görülebilmesini sağlayan bir Arduino Uno shield'i olarak tasarlandı.

LogicShield Arduino ve kodlama eğitimi verilen kurslarda öğrencilerin mantıksal yapıları, dijital elektroniğin temellerini ve kodlamanın algoritma mantığını anlayabilmeleri için görsel ve fiziksel etkileşimli uygulama olarak kullanılabilir.

Küçük yaştaki çocuklar bile lojik kapıların çalışma mantığını kavrayabilecek beceriye sahiptir. Detayları akıllarında tutamaları ve isimlerini bilmeseler bile lojik kapıların birbirlerinden farklı şekilde çalıştığını kavrayabilirler.



NASIL KULLANILIR?

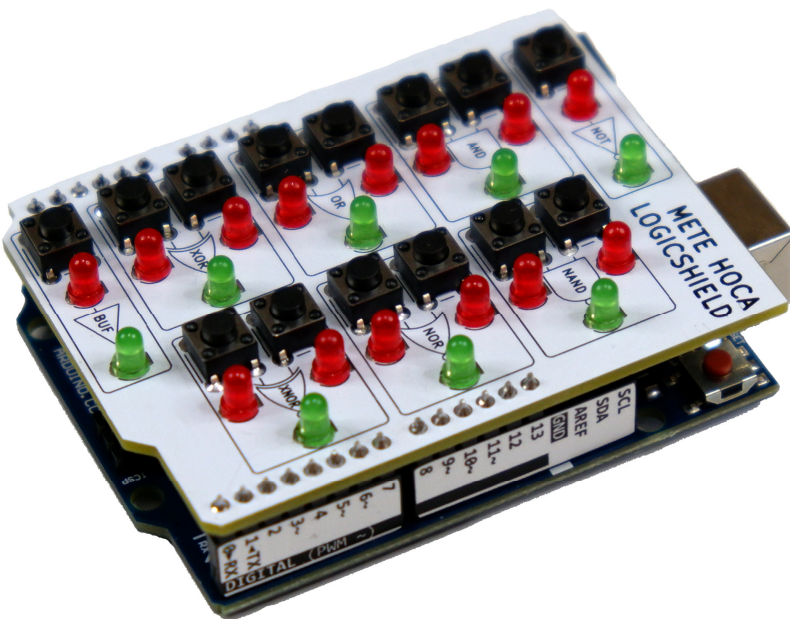
LogicShield'i kullanabilmek için Arduino Uno üzerine yerleştirmek ve Mete Hoca web sitesindeki LogicShield sayfasında bulunan linkten gerekli Arduino sketch kodunu indirip Arduino'ya yüklemek yeterlidir. Kullanmak için Arduino'nun USB ile bilgisayara bağlı olması gerekli değildir, Uno'ya adaptör veya pil ile güç vererek kullanılabilir.

LogicShield üzerinde 8 farklı lojik kapı simülatörü bulunuyor. Her bir lojik kapı devresi bir çerçeve içine alınmıştır. Bu lojik kapılar birbirinden tamamen bağımsız olarak çalışırlar ve giriş-çıkışları birbirlerine bağlı değildir.

Her bir lojik kapının girişinde buton ve kırmızı renkli gösterge LED'leri yer alır. Butonlar lojik kapıların girişlerine sinyal, yani 0 veya 1 göndermek için kullanılır.

Her bir girişin butonuna basıldığında bitişindeki kırmızı LED yanar ve butona basıldığını gösterir. İki girişe sahip lojik kapıların iki girişine birden 1 göndermek için iki buton da basılı tutulur. Butonlara basılmadığı zamanlarda lojik kapı girişlerine 0 gönderilmiş olur.

Lojik kapıların çıkışlarında ise yeşil renkli gösterge LED'leri yer alır. Bu LED'ler lojik kapı çıkışı 1 olduğunda yanar, 0 olduğunda söner.

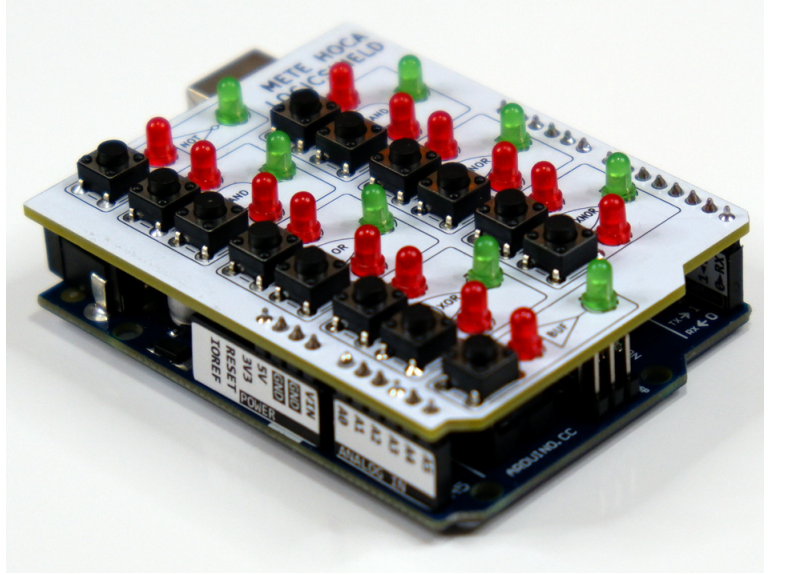


KULLANIRKEN NELERE DİKKAT EDİLMELİDİR?

LogicShield her elektronik cihaz gibi kısa devrelere karşı hassastır ve çalışırken üzerine iletken bir şeyler düşürülmesi veya sıvı teması olduğunda bağlı olduğu Arduino ile birlikte bozulabilir.

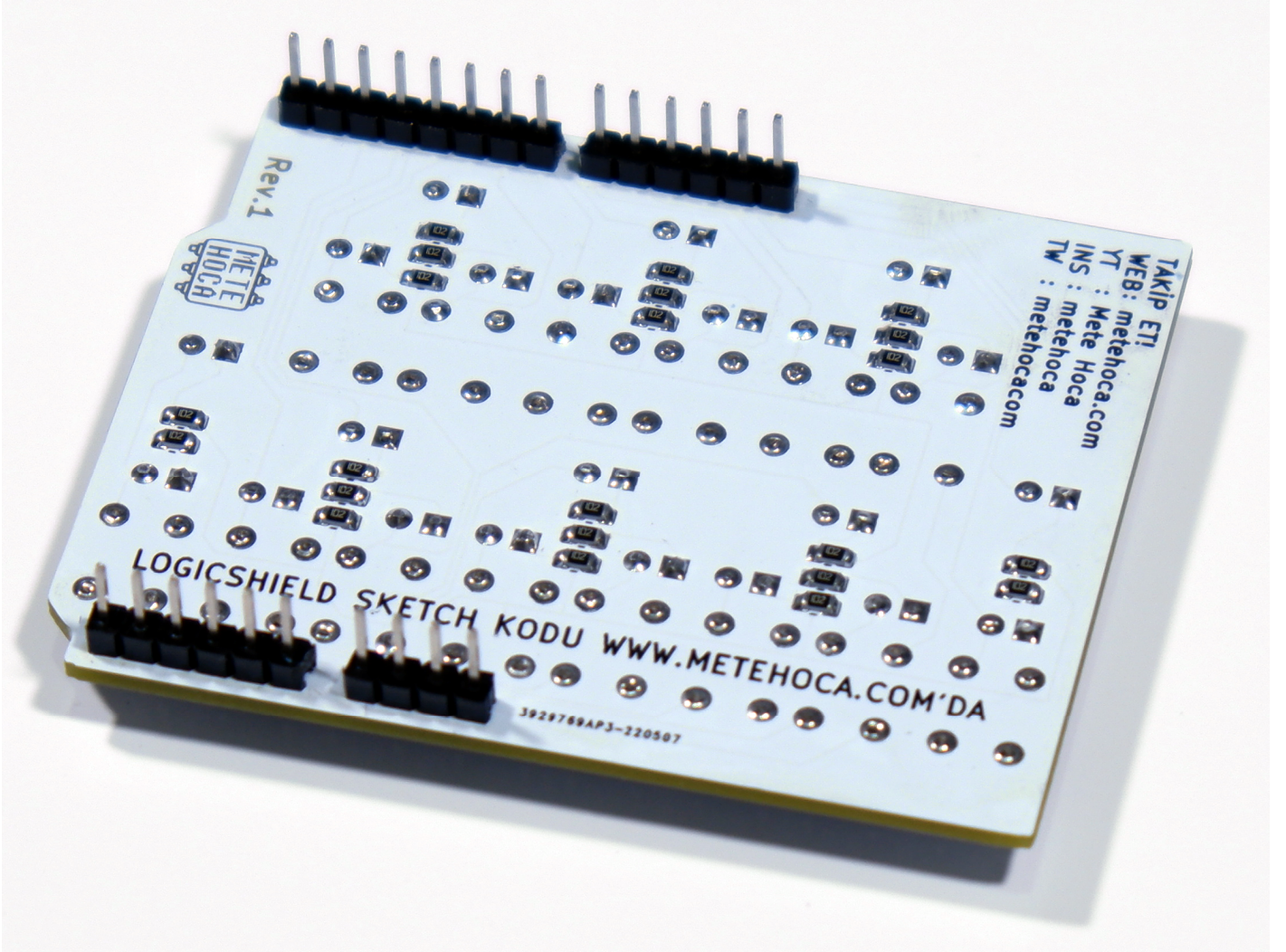
Arduino board ve LogicShield asla metal yüzeyler üzerinde kullanılmamalı, üzerine iletken herhangi bir şey değdirilmemelidir.

Her Arduino shield gibi LogicShield'in de altında Arduino Uno'ya bağlanması için pinler yer alır. LogicShield'i Arduino'ya takarken ve çıkarırken pinlerinin eğilmemesine ve Arduino'ya doğru şekilde bağlanmasına dikkat edilmelidir.



LogicShield'i Arduino'ya bağlarken pin sıraları arasındaki boşluğa dikkat edilmeli, Arduino üzerindeki doğru boşluğa denk geldiklerinden emin olunmalıdır.

Hatalı bağlantı Arduino ve LogicShield'in bozulmasına neden olabilir.



LogicShield kullanılmadığı zamanlarda Arduino üzerinden sökülmeli ve pinlerinin eğilmemesi için paketinde gelen köpük üzerine yerleştirilerek saklanmalıdır.

LOJİK KAPI NEDİR?

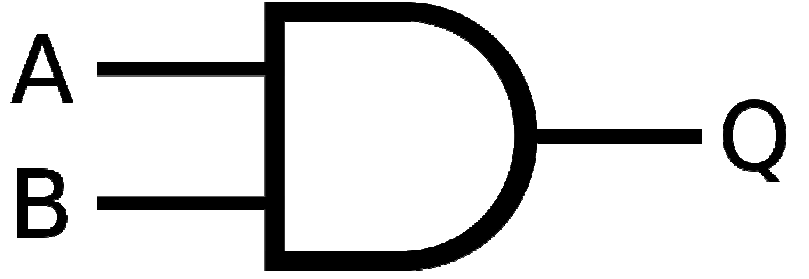
Dijital elektroniğin temeli olan **lojik (mantıksal) kapılar** (İngilizce: **Logic Gate**) temelde giriş ve çıkış terminallerinden oluşan elektronik yapılardır ve girişlerine uygulanan dijital sinyalleri tasarlandıkları mantıksal yapıya göre işleyerek çıkış üretirler.

Bilgisayarlarımız ve akıllı telefonlarımızın mikroişlemcileri, depolama üniteleri ve hesap makineleri lojik kapıların belirli kombinasyonları kullanılarak tasarlanırlar. Lojik kapıları öğrenmek, dijital elektronik ve kodlamanın temellerini iyi anlamak için çok önemlidir.

Lojik kapılar günümüzde paket halinde entegre devreler halinde satılırlar da, transistör ve diyot gibi temel elektronik bileşenlerinin belirli kombinasyonlarda bir araya getirilmeleriyle üretilirler.

Dijital elektronik ve Arduino uygulamalarından aşına olduğumuz **1, 0, HIGH, LOW, true** ve **false** terimleri **binary (ikili)** sayı sistemi üzerine inşa edilmiş elektrik sinyalleri tanımlarlar. 1, HIGH veya true devrede elektriksel gerilimin var olduğunu ifade ederken, 0, LOW veya false ise devrede herhangi bir elektriksel gerilim olmadığı anlamına gelir.

Lojik kapılar işte bu sinyalleri işleyerek tasarlandıkları mantığa göre çıkışlar üretirler ve kullanıldıkları cihazın belirli şartlar sağlandığında doğru işlemleri başlatmaları veya durdurmalarına imkân verirler. Lojik kapılar devre şemalarında gösterilirken girişleri A, B, C... ve çıkışları Q harfi ile gösterilir.



Bir örnekle açıklamak gerekirse; bilgisayarda yazı yazarken büyük harf yazmak istediğimizde bunu iki farklı yöntemle yapabiliriz. Birinci yöntem SHIFT tuşunu basılı tutarak harfe basmaktır. İkinci yöntem ise CAPS LOCK tuşuna basarak büyük yazmayı açtıktan sonra istediğimiz harfe basmak.

Klavye üzerindeki elektronik devre hangi tuşlara basıldığını kontrol eder ve bunları belirli bir mantıkla işledikten sonra bilgisayara gerekli tuşu gönderir. Büyük A harfi yazmak istediğimizde ihtiyacımız olan mantıksal yapı şu cümleyle tanımlanabilir; (CAPS LOCK açıkta **VEYA** SHIFT tuşu basılıysa) **VE** "a" tuşu basılıysa bilgisayara **A** gönder.

Daha net anlaşılması için yapıyı mantıksal bir düzene göre ayırabiliriz;

CAPS LOCK açıkta

VEYA

VE

"a" tuşu basılıysa

=>

bilgisayara **A** gönder.

SHIFT tuşu basılıysa

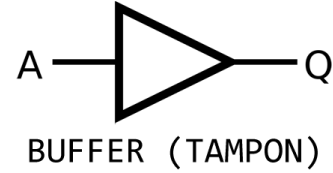
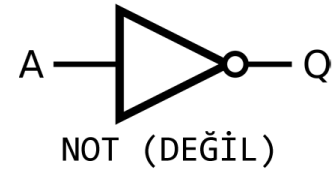
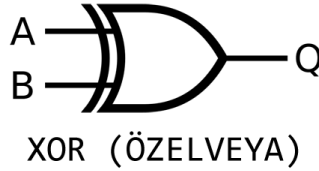
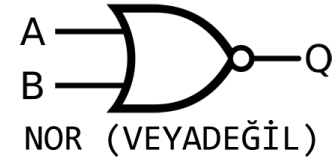
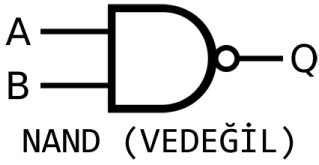
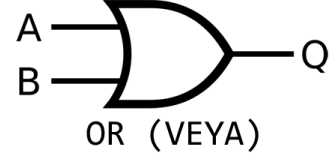
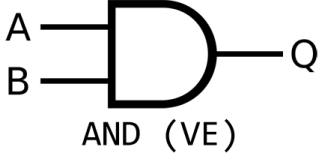
Yukarıdaki mantıksal yapıyı incelediğimizde bilgisayara büyük A tuşunun gönderilmesi için temelde iki şartın sağlanması gerekiyor; Öncelikle CAPS LOCK açık olacak veya SHIFT tuşu basılı olacak, ardından a tuşu da basılı olacak.

CAPS LOCK'un açık olması veya SHIFT tuşunun basılı olması durumlarının hangisinin gerçekleştiği fark etmeksizin bilgisayara A tuşu gönderilir.

LOJİK KAPI ÇEŞİTLERİ

Dijital elektronikte 8 temel lojik kapı bulunuyor. Klavye örneğinde üç temel lojik kapının ikisinden bahsetmiş olduk; **VE (AND)** ve **VEYA (OR)**. Üçüncü temel lojik kapımız ise **DEĞİL (NOT-INVERTER)** kapısıdır.

Diğer lojik kapılar bu üç temel lojik kapının belirli kombinasyonlarla bir araya getirilmeleri ile oluşturulur ve temel amaçları sık ihtiyaç duyulan mantık ihtiyaçlarını karşılamaktır. Bu kapılar **VEDEĞİL (NAND)**, **VEYADEĞİL (NOR)**, **ÖZELVEYA (Exclusive OR / XOR)** **ÖZELVEYADEĞİL (Exclusive NOR / XNOR)** ve **TAMPON (BUFFER)** olarak sıralanabilir.



ÖZELVEYA lojik kapısının Türkçe olarak **DIŞARMALI** veya **DIŞLAMALI** olarak adlandırıldıklarını da duymak mümkün. Bu iki terim aslında Exclusive OR'un daha doğru olan karşılığıdır, ancak uzun yıllardır eğitim sistemlerinde ÖZELVEYA adıyla kullanıldıklarından dolayı bu kullanım daha yaygındır.

Lojik kapılar ile ilgili projeler veya kaynakların büyük kısmının İngilizce olduğunu düşünecek olursak İngilizce isimlerini kullanmak daha pratik olur. Bundan sonra kapılara İngilizce isimleri ile hitap edeceğiz.

Tüm lojik kapıları bir tabloda görmek istersek;

LOJİK KAPI	TÜRKÇE ADI	İŞLEVİ
AND	VE	Girişlerin tamamı 1 ise çıkış 1 olur.
OR	VEYA	Girişlerin herhangi biri 1 ise çıkış 1 olur.
NOT	DEĞİL	Giriş ne ise çıkış tersi olur.
XOR	ÖZELVEYA	Girişler birbirinden farklı ise çıkış 1 olur.
NAND	VEDEĞİL	Girişlerin tamamı 1 ise çıkış 0 olur.
NOR	VEYADEĞİL	Girişlerin herhangi biri 1 ise çıkış 0 olur.
XNOR	ÖZELVEYADEĞİL	Girişler birbirinden farklı ise çıkış 0 olur.
BUFFER	TAMPON	Giriş ne ise çıkış aynısı olur.

AND (VE) KAPISI

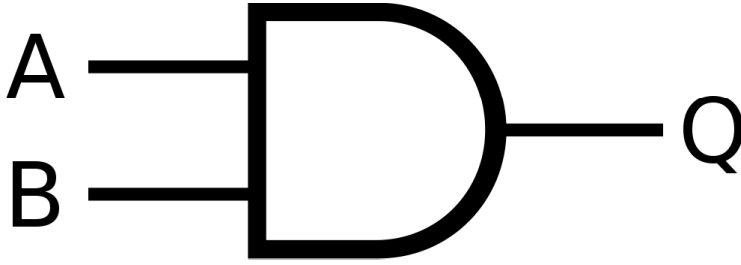
AND kapısı üç temel lojik kapıdan biridir ve tüm girişleri 1 olduğu zaman çıkışı 1 olur. AND kapılarının çıkışının 1 olabilmesi ancak tüm girişlerinin 1 olması durumunda gerçekleşir.

AND kapısı bir işlemi başlatmak için gerekli olan alt işlemlerin tamamının tamamlanmış olması gerektiği durumlarda kullanılır.

Matematiksel gösterimi $Q = A \cdot B$ şeklindedir. Yani giriş değerlerinin çarpımı çıkış değerini belirler. Dolayısıyla giriş değerlerinin tamamı 1 olmadan çıkış 1 olamaz.

AND kapısının devre sembolü ve doğruluk tablosunu aşağıdaki resimde görebiliriz.

DEVRE SEMBOLÜ



DOĞRULUK TABLOSU

GİRİŞ		ÇIKIŞ
A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

Resimde gösterilen A ve B, AND kapısının girişleridir ve iki giriş birden 1 olunca Q ile gösterilen çıkış 1 olur. Girişlerden herhangi biri 0 olduğunda çıkış 0'a döner.

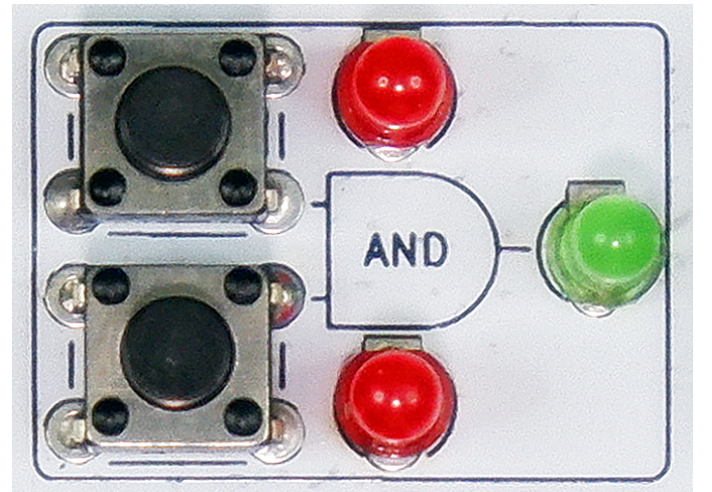
Genellikle iki girişe sahip olsalar da üç veya dört girişe sahip AND kapıları ile karşılaşmak mümkündür.

AND kapısını LogicShield ile deneyimlediğimizde yeşil renkli çıkış LED'inin sadece iki butona birden basıldığında yandığını görürüz.

Diğer tüm buton kombinasyonlarında çıkış LED'i sönük kalacaktır.

AND kapısı genelde dijital elektronikte bir veri sinyali giriş veya çıkışını kapatıp açmakta kullanılır. Kapının bir girişine kontrol edilecek sinyal girilir.

Diğer giriş ise ENABLE olarak adlandırılır ve veri aktarımına izin verildiği zaman bu girişe 1 verilir. Böylece veri sinyali çıkışa ulaşabilir.



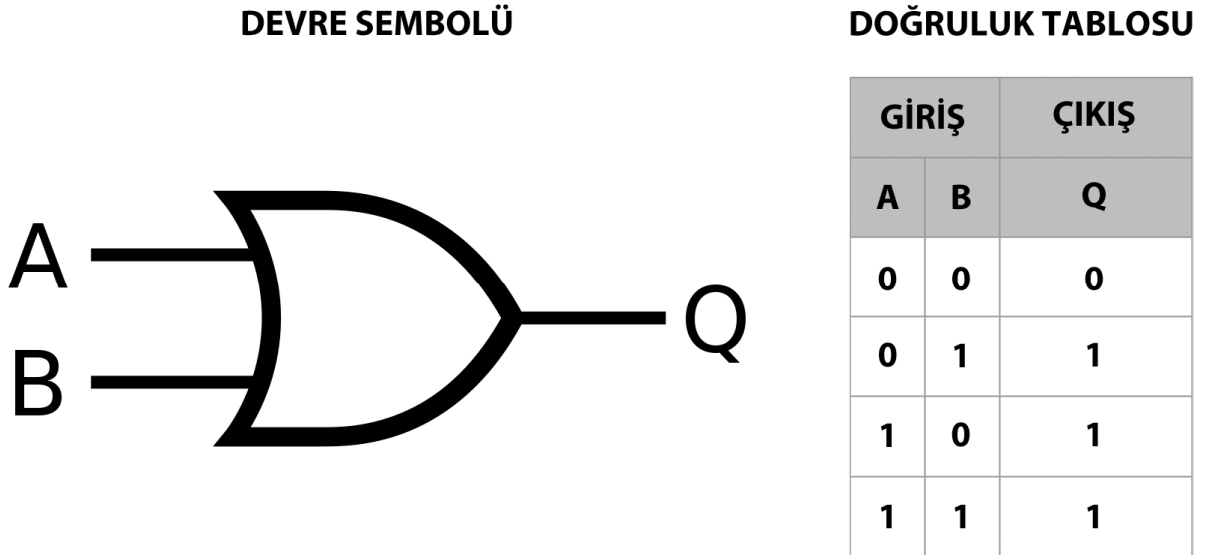
OR (VEYA) KAPISI

OR kapısı üç temel lojik kapıdan bir diğeridir ve girişlerinden herhangi biri 1 olduğu zaman çıkışı 1 olur.

OR kapısı bir işlemi başlatmak için gerekli olan alt işlemlerin en az birinin tamamlanmış olması gerektiği durumlarda kullanılır.

Matematiksel gösterimi $Q = A + B$ şeklindedir. Yani giriş değerlerinin toplamı çıkış değerini belirler. Dolayısıyla giriş değerlerinin herhangi biri 1 olduğunda çıkış da 1 değerine ulaşır.

OR kapısının devre sembolü ve doğruluk tablosunu aşağıdaki resimde görebiliriz.



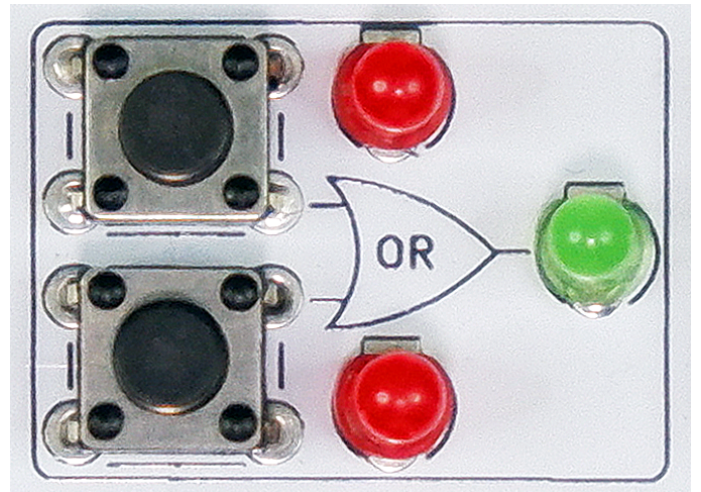
Resimde gösterilen A ve B, OR kapısının girişleridir ve iki girişten en az biri 1 olunca Q ile gösterilen çıkış 1 olur. Girişlerin ikisi birden 0 olduğunda çıkış 0'a döner.

Genellikle iki girişe sahip olsalar da üç veya dört girişe sahip OR kapıları ile karşılaşmak mümkündür.

OR kapısını LogicShield ile deneyimlediğimizde yeşil renkli çıkış LED'inin butonlardan herhangi birine veya ikisine birden basıldığında yandığını görürüz.

Butonlardan hiçbiri basılı değilse çıkış LED'i sönük kalacaktır.

OR kapısı dijital elektronikte en yaygın olarak hata kontrolü için kullanılır. Sistem üzerindeki hata sensör devrelerinden herhangi biri etkinleştğinde OR kapısının çıkışı 1 olur ve sistem durdurulur.



NOT (DEĞİL) KAPISI

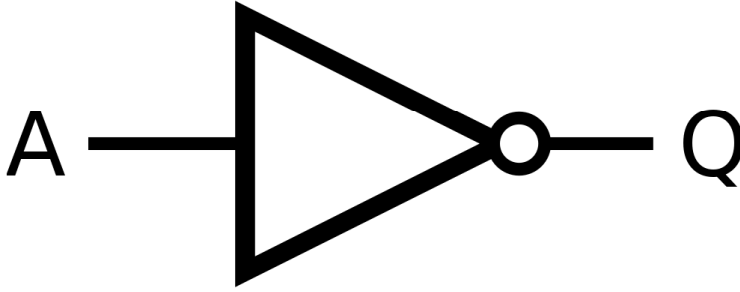
NOT kapısı üç temel lojik kapının sonuncusu ve belki de en sık kullanılanıdır. Bir giriş ve bir çıkıştan oluşur ve çıkışı girişinin tam tersi olur.

Güncel teknik kullanımda genelde INVERTER (Tersleyici) olarak adlandırılan NOT kapısı bir işlemin sonucunun tam tersine ihtiyaç duyulan durumlarda kullanılır.

Matematiksel gösterimi $Q = \bar{A}$ şeklindedir. Çıkış değeri, giriş değerinin terslenmiş halidir. Diğer bir deyişle giriş 0 olduğunda çıkış 1, giriş 1 olduğunda çıkış 0 olur.

NOT kapısının devre sembolü ve doğruluk tablosunu aşağıdaki resimde görebiliriz.

DEVRE SEMBOLÜ



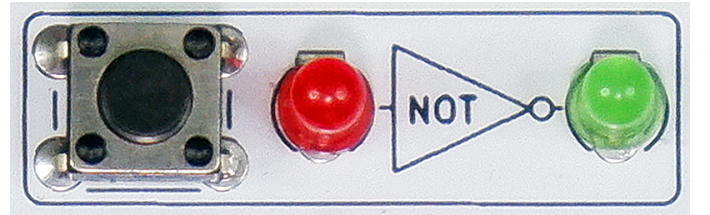
DOĞRULUK TABLOSU

GİRİŞ	ÇIKIŞ
A	Q
0	1
1	0

Resimde gösterilen A, NOT kapısının girişidir ve bu giriş 1 olunca Q ile gösterilen çıkış 0 olur. Giriş 0 olduğunda ise çıkış 1'e döner.

NOT kapısını LogicShield ile deneyimlediğimizde yeşil renkli çıkış LED'inin butona basılmadığında yandığını görürüz.

Butona bastığımızda çıkış LED'i söner. Yani NOT kapısı girişe uygulanan sinyalin tam tersini üretir.



NOT kapısı dijital elektronikte ters çalışan, yani algılama yaptığında 1 yerine 0 üreten sensör çıkışlarını normalleştirmekte, yani herhangi bir algılama yapıldığında çıkıştan 0 yerine 1 vermesi istendiğinde kullanılır.

XOR (ÖZELVEYA) KAPISI

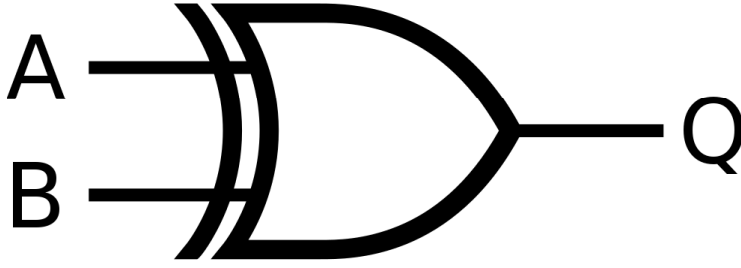
XOR kapısı temel lojik kapılardan farklı ve sık gereksinim duyulan bir durum üzerine geliştirilmiştir. Girişleri birbirinden farklı olduğu zaman çıkışı 1 olur.

XOR kapısı, OR kapısının iki giriş birden 1 olduğunda da 0 çıkış veren sürümü olarak düşünülebilir. Bir işlemin girişlere bağlı alt işlemler arasında fark olduğu zaman yürütülmesi istendiğinde kullanılır.

Matematiksel gösterimi $Q = A \oplus B$ şeklindedir. Yani giriş değerleri arasındaki eşitsizlik çıkış değerini belirler. Girişler arasında eşitlik varsa çıkış 0, bir eşitsizlik oluştuğunda ise çıkış 1 olur.

XOR kapısının devre sembolü ve doğruluk tablosunu aşağıdaki resimde görebiliriz.

DEVRE SEMBOLÜ



DOĞRULUK TABLOSU

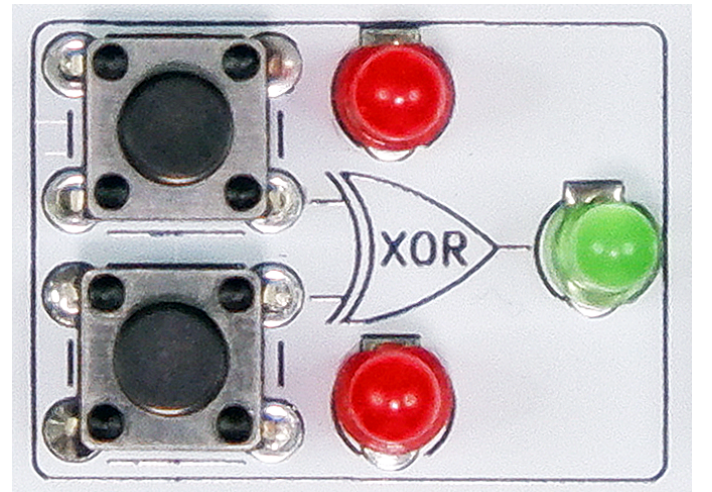
GİRİŞ		ÇIKIŞ
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

Resimde gösterilen A ve B, XOR kapısının girişleridir ve iki giriş birbirinden farklı olduğunda Q ile gösterilen çıkış 1 olur. Girişlerin ikisi birden 0 veya ikisi birden 1 olduğunda çıkış 0'a döner.

XOR kapısını LogicShield ile deneyimlediğimizde yeşil renkli çıkış LED'inin butonlardan sadece birine basıldığında yandığını görürüz.

Butonlardan hiçbiri basılı değilse veya ikisi birden basılıysa çıkış LED'i sönmüş kalacaktır.

XOR kapısı dijital elektronikte parite hesaplama adı verilen anlık veri doğrulama sistemlerinde basit hata kontrolcüsü olarak ve şifreleme devrelerinde ara uygulama aşaması olarak kullanılır.



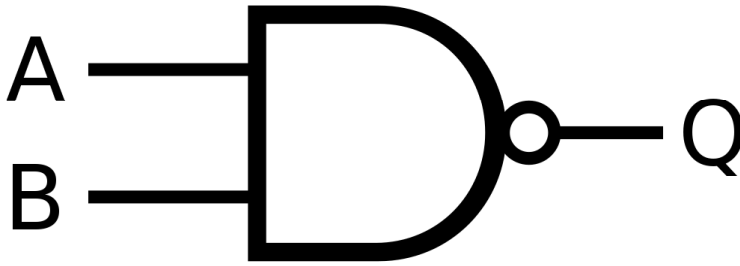
NAND (VEDEĞİL) KAPISI

NAND kapısı temel lojik kapılardan biri olan AND kapısının çıkışına bir NOT kapısı eklenmiş, diğer bir deyişle AND kapısının çıkışı terslenmiş sürümüdür. İki girişi birden 1 olduğunda çıkışı 0 olur, diğer her durumda çıkışı 1 olur.

Matematiksel gösterimi $Q = \overline{A \cdot B}$ şeklindedir. Yani çıkış değeri girişlerin çarpımının terslenmesi ile elde edilir.

NAND kapısının devre sembolü ve doğruluk tablosunu aşağıdaki resimde görebiliriz.

DEVRE SEMBOLÜ



DOĞRULUK TABLOSU

GİRİŞ		ÇIKIŞ
A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

Resimde gösterilen A ve B, NAND kapısının girişleridir ve iki giriş birden 1 olunca Q ile gösterilen çıkış 0 olur. Girişlerden herhangi biri 0 olduğunda çıkış 1'e döner.

Genellikle iki girişe sahip olsalar da üç veya dört girişe sahip NAND kapıları ile karşılaşmak mümkündür.

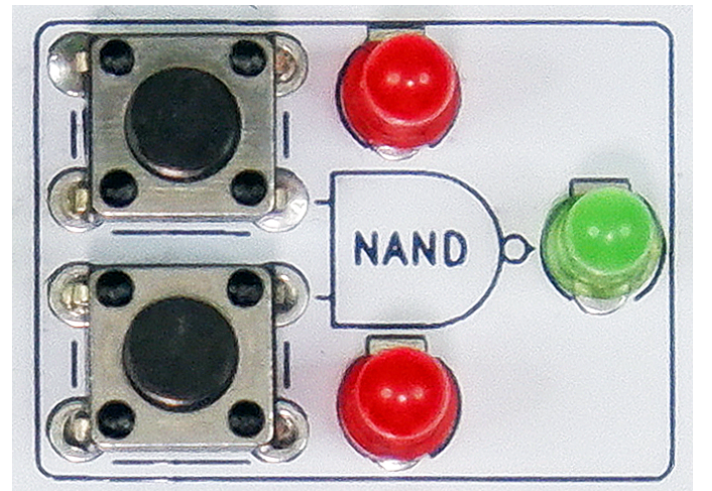
NAND kapısını LogicShield ile deneyimlediğimizde yeşil renkli çıkış LED'inin sadece iki butona birden basıldığında söndüğünü görürüz.

Diğer tüm buton kombinasyonlarında çıkış LED'i yanık kalacaktır.

NAND kapısı genelde dijital elektronikte sensör temelli hata kontrolü ve alarm sistemlerinde kullanılır.

Bir sistemin çalıştığını gösteren sensörler 1 üretirler ve tüm sensörler olumlu sinyal gönderdiğinde NAND kapısının çıkışı 0 olur ve bu sistemde bir eksiklik olmadığı anlamına gelir.

NAND kapısı elektronikte transistör veya MOSFET'lerle en kolay şekilde üretilen lojik kapı olduğu için çoğu zaman diğer kapıların yerine tercih edilirler.



NOR (VEYADEĞİL) KAPISI

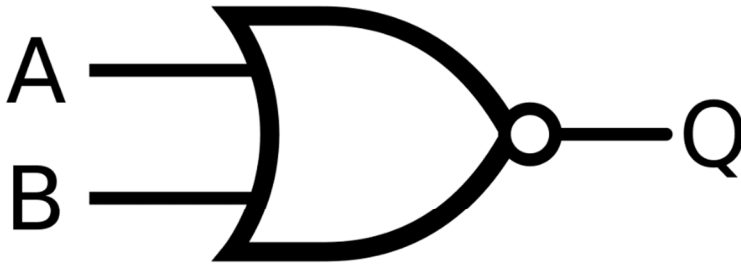
NOR kapısı temel lojik kapılardan biri olan OR kapısının çıkışına bir NOT kapısı eklenmiş, diğer bir deyişle OR kapısının çıkışı terslenmiş sürümüdür.

NOR kapısı bir işlemi durdurmak için önemli olan alt işlemlerin herhangi birinin ortaya çıktığı durumlarda kullanılır.

Matematiksel gösterimi $Q = \overline{A+B}$ şeklindedir. Yani giriş değerlerinin toplamının tersi çıkış değerini belirler. Dolayısıyla giriş değerlerinin herhangi biri 1 olduğunda çıkış da 0 değerini alır.

NOR kapısının devre sembolü ve doğruluk tablosunu aşağıdaki resimde görebiliriz.

DEVRE SEMBOLÜ



DOĞRULUK TABLOSU

GİRİŞ		ÇIKIŞ
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

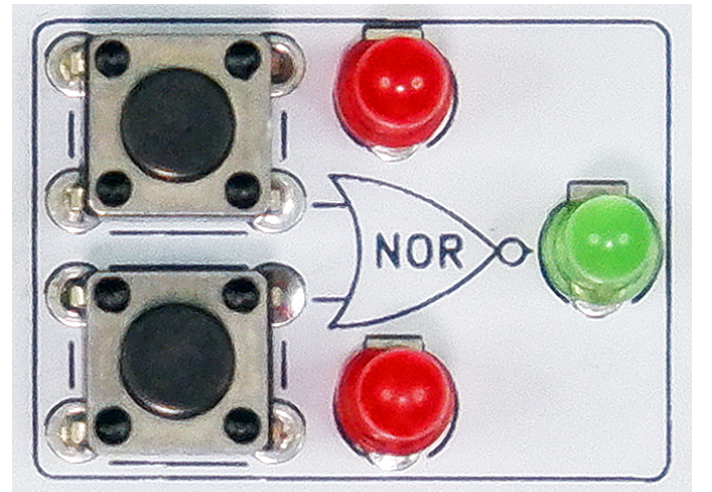
Resimde gösterilen A ve B, NOR kapısının girişleridir ve iki girişten herhangi biri 1 olunca Q ile gösterilen çıkış 0 olur. Girişlerin ikisi birden 0 olduğunda çıkış 1'e döner.

Genellikle iki girişe sahip olsalar da üç veya dört girişe sahip NOR kapıları ile karşılaşmak mümkündür.

NOR kapısını LogicShield ile deneyimlediğimizde yeşil renkli çıkış LED'inin butonlardan herhangi birine veya ikisine birden basıldığında söndüğünü görürüz.

Butonlardan hiçbiri basılı değilse çıkış LED'i yanık kalacaktır.

NOR kapısı dijital elektronikte genelde shift register ve multiplexer gibi seri veri işleme ve dağıtma ihtiyaçlarında kullanılır.



XNOR (ÖZELVEYADEĞİL) KAPISI

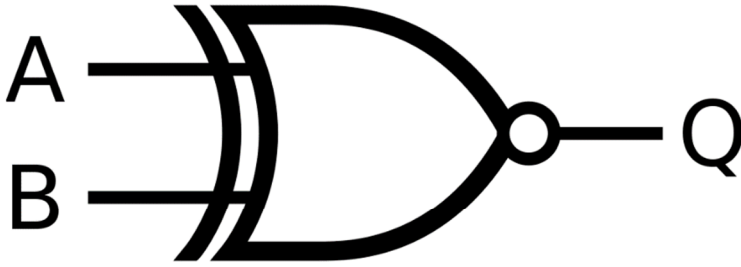
XNOR kapısına XOR kapısının çıkışına bir NOT kapısı bağlanmış sürümü diyebiliriz. Girişleri birbirinin aynı olduğunda çıkışı 1, farklı olduğunda ise 0 olur.

XOR kapısı giriş sinyalleri arasında bir fark olup olmadığını kontrol etmekte kullanılır.

Matematiksel gösterimi $Q = \overline{A \oplus B}$ şeklindedir. Yani giriş değerleri arasındaki eşitsizlik çıkış değerini belirler. Girişler arasında eşitlik varsa çıkış 1, bir eşitsizlik oluştuğunda ise çıkış 0 olur.

XNOR kapısının devre sembolü ve doğruluk tablosunu aşağıdaki resimde görebiliriz.

DEVRE SEMBOLÜ



DOĞRULUK TABLOSU

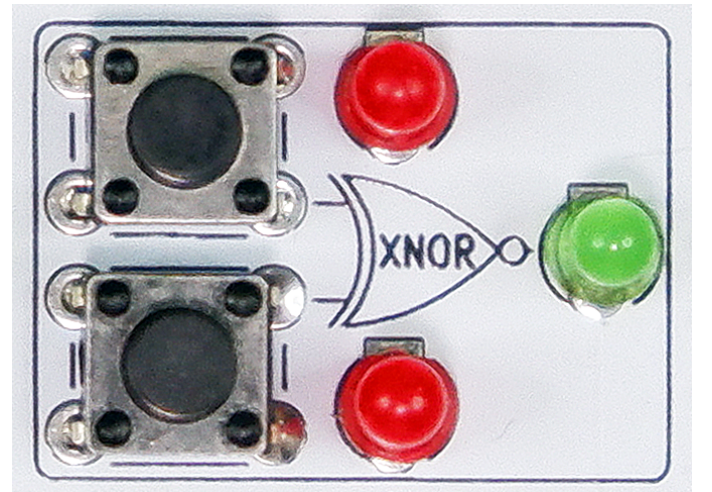
GİRİŞ		ÇIKIŞ
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1

Resimde gösterilen A ve B, XNOR kapısının girişleridir ve iki giriş birbirinden farklı olduğunda Q ile gösterilen çıkış 0 olur. Girişlerin ikisi birden 0 veya ikisi birden 1 olduğunda çıkış 1'e döner.

XNOR kapısını LogicShield ile deneyimlediğimizde yeşil renkli çıkış LED'inin butonlardan sadece birine basıldığında söndüğünü görürüz.

Butonlardan hiçbiri basılı değilse veya ikisi birden basılıysa çıkış LED'i yanık kalacaktır.

XNOR kapısı dijital elektronikte parite hesaplama adı verilen anlık veri doğrulama sistemlerinde basit hata ayıklama cihazı olarak kullanılır.



BUFFER (TAMPON) KAPISI

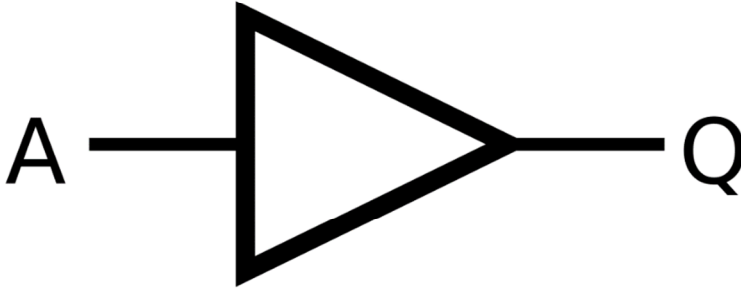
BUFFER kapısı giriş sinyalinin aynısını çıkış olarak veren bir kapıdır. Yani giriş sinyalini hiçbir şekilde işlemez, aynı şekilde iletir.

İlk bakışta hiçbir işe yaramıyormuş gibi görünen BUFFER kapısı çıkışı çok fazla lojik kapıya bağlanacak lojik kapılar için sinyal yükseltici olarak kullanılır. Çünkü genelde lojik kapıların çıkış güçleri çok sayıda lojik kapı girişini besleyebilecek kadar güçlü değildir. Bu durumda çıkışa BUFFER bağlanır ve sinyal seviyesinin düşmesinin önüne geçilir.

Matematiksel gösterimi $Q = A$ şeklindedir. Çıkış değeri, giriş değerinin aynısıdır. Diğer bir deyişle giriş 0 olduğunda çıkış 0, giriş 1 olduğunda çıkış 1 olur.

BUFFER kapısının devre sembolü ve doğruluk tablosunu aşağıdaki resimde görebiliriz.

DEVRE SEMBOLÜ

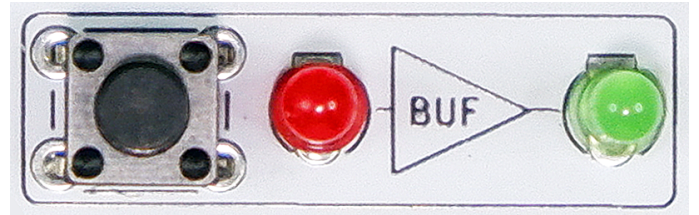


DOĞRULUK TABLOSU

GİRİŞ	ÇIKIŞ
A	Q
0	0
1	1

Resimde gösterilen A, BUFFER kapısının girişidir ve bu giriş 1 olunca Q ile gösterilen çıkış 1 olur. Giriş 0 olduğunda ise çıkış 0'a döner.

BUFFER kapısını LogicShield ile deneyimlediğimizde yeşil renkli çıkış LED'inin butona basıldığında yandığını, basılmadığında da sönük kaldığını görürüz.

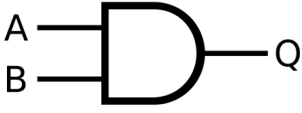


Yani BUFFER kapısı girişe uygulanan sinyalin aynısını üretir.

BUFFER kapısı dijital elektronikte sinyal güçlendirme ve devreler arası dijital izolasyon sağlamakta aracı olarak kullanılır.

LOJİK KAPILARIN TOPLU DOĞRULUK TABLOSU (TRUTH TABLE)

AND



GİRİŞ		ÇIKIŞ
A	B	Q
0	0	0
0	1	0
1	0	0
1	1	1

OR



GİRİŞ		ÇIKIŞ
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	1

XOR



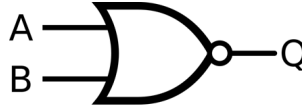
GİRİŞ		ÇIKIŞ
A	B	Q
0	0	0
0	1	1
1	0	1
1	1	0

NAND



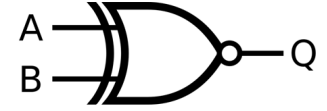
GİRİŞ		ÇIKIŞ
A	B	Q
0	0	1
0	1	1
1	0	1
1	1	0

NOR



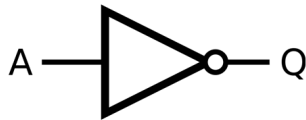
GİRİŞ		ÇIKIŞ
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	0

XNOR



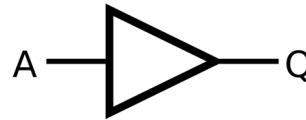
GİRİŞ		ÇIKIŞ
A	B	Q
0	0	1
0	1	0
1	0	0
1	1	1

NOT



GİRİŞ	ÇIKIŞ
A	Q
0	1
1	0

BUFFER

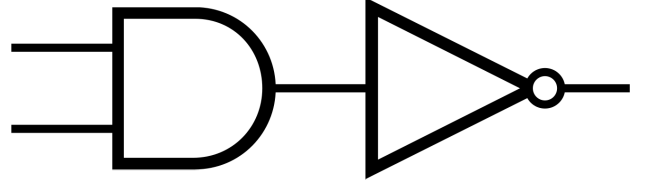


GİRİŞ	ÇIKIŞ
A	Q
0	0
1	1

LOJİK KAPILARIN BİR ARADA KULLANILMALARI

Lojik kapıların tamamını gördük ve nasıl çalıştıklarını öğrendik. Şimdi sıra lojik kapıları birbirine bağlayarak karmaşık mantık ihtiyaçlarını gidermeye geldi.

NAND, NOR ve XNOR kapılarını anlatırken bu kapıların AND, OR ve XOR kapılarının çıkışlarına NOT kapısı eklenerek üretildiklerinden bahsetmiştik. Yandaki resimde bir AND kapısının çıkışına NOT kapısı eklenerek elde edilen NAND kapısı görülüyor.



Bu yöntem farklı kombinasyonlarla kullanılarak çok farklı mantıksal çözümler üretilebilir. Birkaç örnekle açıklayalım.

Bu kitabın en başında lojik kapıların ne olduklarından bahsederken bir klavye örneği vermiştik. Büyük A basmayı hedefleyen o örneği hatırlayalım;

CAPS LOCK açıkta

VEYA

VE

"a" tuşu basılıysa

=>

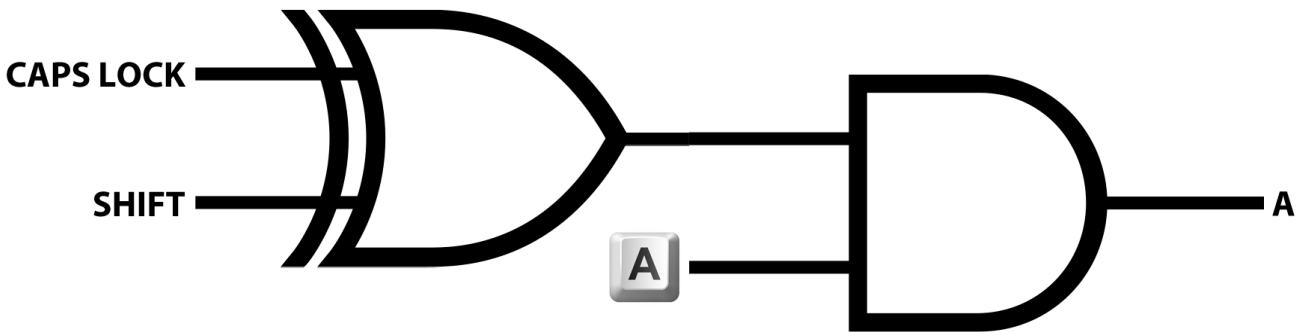
bilgisayara **A** gönder.

SHIFT tuşu basılıysa

Bu örneği dikkatlice incelersek her zaman doğru çalışmayabileceğini fark edebiliriz. En başta CAPS LOCK ve SHIFT tuşlarını denetleyen lojik kapıyı OR (VEYA) olarak seçersek hem CAPS LOCK açık, hem de SHIFT basılı olduğu zaman da 1 üretir. Bu durumda sonuç büyük A yerine küçük a olur.

Bu sorunu çözmemiz gerekiyor. CAPS LOCK veya SHIFT tuşlarının **yalnızca birinin** basılı olduğu bir durumda 1 üretecek bir kapıya ihtiyacımız var. Yani hiçbirini basılı olmadığı ve ikisi birden basılı olduğunda 0 üretmeli.

İşte XOR kapısı burada yardımımıza koşuyor. Aşağıdaki lojik devreyi inceleyelim;



XOR kapısının çıkışının girişleri birbirinden farklı olduğu zaman 1 olduğunu ve diğer durumlarda 0 olduğunu biliyoruz. Tam da ihtiyacımıza uygun bir fonksiyon. Geriye XOR kapısının çıkışını klavyedeki A tuşu ile birleştirerek büyük A basmayı kontrol etmemiz kalıyor. Bunu da AND kapısı ile yapıyoruz.

Lojik devremiz çalışıyor: CAPS LOCK veya SHIFT tuşlarının yalnızca biri basılı ve klavyeden A tuşu da basılıysa büyük A gönder.

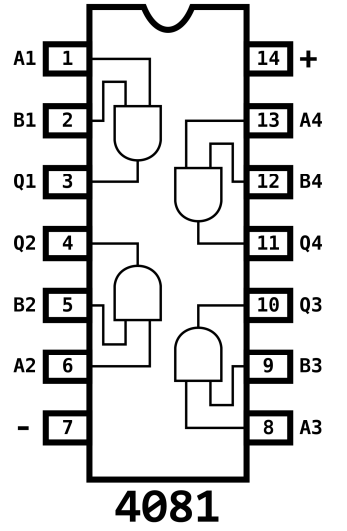
Matematiksel olarak göstermek istersek; **BÜYÜK A = (CAPS LOCK \oplus SHIFT) . A TUŞU**

4 GİRİŞLİ AND KAPISI YAPALIM

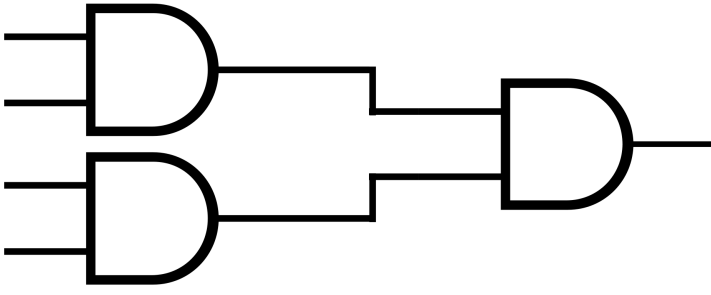
Yeni bir örnekle devam edelim. İhtiyacımız olan şey 4 butona birden basıldığında yanan bir lamba. Elimizde de içinde 4 tane AND kapısı bulunan 4081 kodlu bir çip var. 4 butona birden basıldığında çıkış vermesi gerektiğini düşündüğümüzde AND kapısı işimize yarıyor.

Lojik kapıların genellikle 2 girişli olduğunu, ancak daha fazla girişe sahip kapıların da üretildiğini öğrenmiştik. Bu kez bize 4 girişe sahip bir adet AND kapısı gerekiyor.

Yandaki resimde görüldüğü gibi çipimizin sahip olduğu AND kapıları sadece 2 girişli. Bu çipi kullanarak 4 girişli bir AND kapısı yapabilir miyiz?



Demek ki yine kapıları belirli bir kombinasyonda birbirine bağlamamız gerekiyor. Butonlarımızı ikişer ikişer birer AND kapısının girişlerine bağladığımızda elimizde iki çıkış kalıyor. Bu çıkışları üçüncü bir AND kapısının girişine bağladığımızdaysa sadece 4 girişi birden 1 olduğunda çıkışı 1 olan bir AND kapısı elde edebiliyoruz.



Soldaki resimdeki lojik devreyi incelediğimizde sol tarafta bulunan AND kapılarının çıkışlarının 1 olabilmesi için girişlerinin 1 olması gerektiğini görürüz.

Bu iki kapının çıkışı da 1 olduğunda sağda görülen AND kapısının girişlerinin ikisi birden 1 olur ve çıkışı 1 olur.

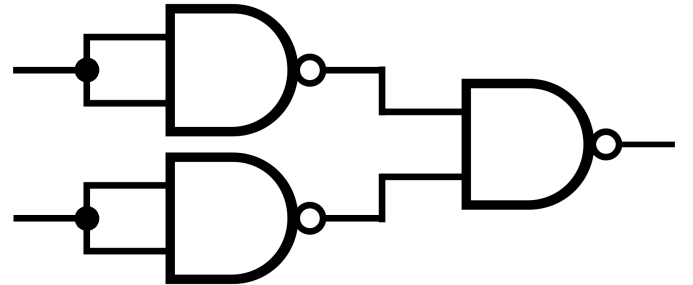
Böylece 3 adet 2 girişli AND kapısı kullanarak 4 girişli bir AND kapısı elde etmiş olduk. 4081 çipimizin boşta kalan dördüncü AND kapısını da projemizin farklı bir yerinde kullanabiliriz.

NAND KAPILARIYLA OR KAPISI YAPALIM

Peki, NAND kapılarından OR kapısı yapmaya ne dersiniz? NAND ve OR mantık olarak tamamen ilgisiz görünen iki lojik kapı olarak görünebilir. Aşağıda 3 adet NAND kapısıyla nasıl OR kapısı elde edileceğini görüyoruz.

Bu son örnek üzerine aklımıza şu soru gelebilir. Neden farklı kapılarla başka bir kapı elde edelim ki? Doğrudan o kapıyı edinip kullanabiliriz.

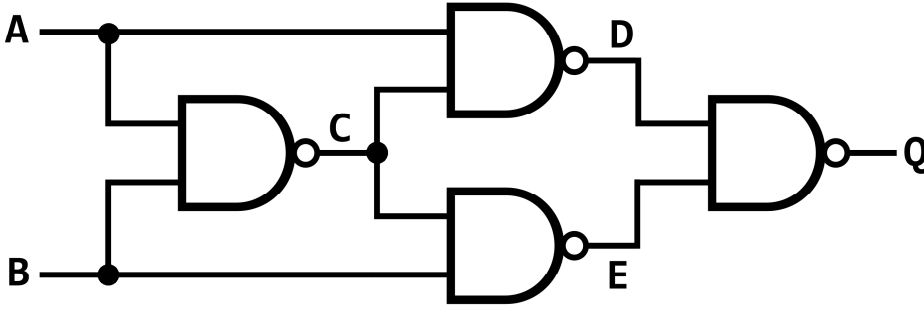
Böyle ihtiyaçların doğma sebebi aslında maliyettir. Devremizde 1 adet OR ve bir adet NAND kapısına ihtiyacımız varsa 4093 kodlu ve içinde 4 adet NAND kapısı barındıran tek bir çip kullanarak bu iki kapıyı üretebiliriz.



NAND kapıları dijital elektronik dünyasının en sık kullanılan lojik kapısıdır. Bu kapılar farklı kombinasyonlarla bir araya getirilerek OR, AND, XOR ve NOT kapıları kolaylıkla üretilebilir ve farklı ihtiyaçlar için kullanılabilir.

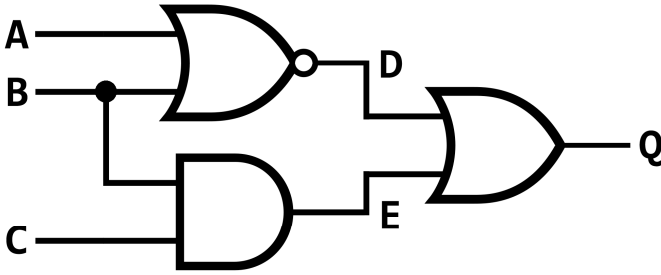
LOJİK KAPI ALIŞTIRMALARI

1. Aşağıdaki lojik devreyi inceleyin ve yanındaki doğruluk tablosunu doldurun. C, D ve E sütunları ara işlemlerde not düşmek için kullanılır. Elde ettiğiniz doğruluk tablosu (A-B-Q) size tanıdık geldi mi?



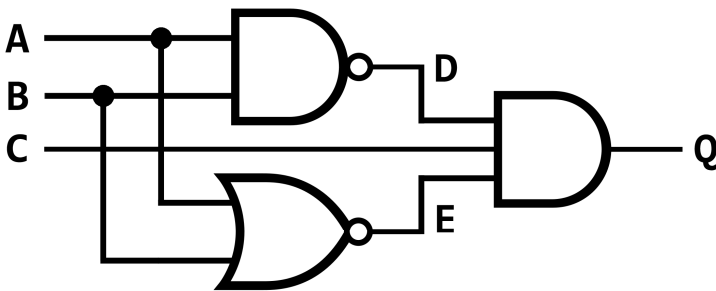
A	B	C	D	E	Q
0	0				
0	1				
1	0				
1	1				

2. Aşağıdaki lojik devreyi inceleyin ve yanındaki doğruluk tablosunu doldurun.



A	B	C	D	E	Q
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

3. Aşağıdaki lojik devreyi inceleyin ve yanındaki doğruluk tablosunu doldurun.



A	B	C	D	E	Q
0	0	0			
0	0	1			
0	1	0			
0	1	1			
1	0	0			
1	0	1			
1	1	0			
1	1	1			

ALİŞTIRMALARIN ÇÖZÜMLERİ

1. Bu tür çoklu lojik kapı devrelerini çözmek için öncelikle iki girişe birden bağlı olan kapılardan başlamalıyız. Bu durumda en solda yer alan ve çıkışı C olan NAND kapısı ile başlamalı ve doğruluk tablosunda hazır olarak verilen A ve B giriş değer kombinasyonlarını kullanarak C sütununu doldurmeliyiz. Çünkü çıkışı D ve E olan NAND kapılarının birer girişleri C'den veri alıyor.

Sonrasında çıkışı D olan NAND kapısı için A ve C sütunundaki verileri, çıkışı E olan NAND kapısı için de B ve C sütunundaki verileri kullanarak D ve E sütunlarını doldurmeliyiz. Son olarak elimizde girişleri D ve E olan son bir NAND kapısı kalır ve böylece Q sütununu doldurabiliriz.

A	B	C	D	E	Q
0	0	1	1	1	0
0	1	1	1	0	1
1	0	1	0	1	1
1	1	0	1	1	0

İlk alıştırmada sorduğumuz ikinci sorunun cevabı ise XOR. A, B ve Q sütunlarına baktığımızda XOR kapısının doğruluk tablosunu görüyoruz. Yani bu alıştırmadaki lojik devre NAND kapıları ile XOR kapısının nasıl oluşturulacağını gösteriyor.

2. Bu alıştırmada üç farklı lojik kapı kullanılmış ve A ve B girişlerinin yanında ek olarak bir de C girişine sahip. Farklı kapılardan oluşması hata yapılma ihtimalini artırıyor. Bu yüzden ezberimiz yeterince iyi değilse doğruluk tablolarını kullanarak çalışmalıyız.

Yine önceki alıştırmada bahsettiğimiz gibi iki girişe birden bağlı olan kapılardan başlamalıyız. Bu devrede NOR veya AND kapısının herhangi birinden başlayabiliriz, çünkü ikisi de girişlere bağlı.

NOR kapısı ile başlayacak olursak A ve B girişlerine göre D sütununu doldurmeliyiz. AND kapısı için B ve C sütunlarına göre E sütununu doldurmeliyiz.

Son olarak da bizi en sağda yer alan OR kapısı bekliyor. Bu kapının girişleri olan D ve E sütunlarını zaten doldurmuştuk. Bu sütunlara göre son mantıksal işlemlerimizi de yapıyor ve lojik devremizin çıkışı olan Q sütununu dolduruyoruz.

A	B	C	D	E	Q
0	0	0	1	0	1
0	0	1	1	0	1
0	1	0	0	0	0
0	1	1	0	1	1
1	0	0	0	0	0
1	0	1	0	0	0
1	1	0	0	0	0
1	1	1	0	1	1

3. Son alıştırmamız yine daha önce karşılaşmadığımız bir yapı içeriyor: 3 girişli bir AND kapısı. 3 girişli AND kapısı öğrendiğimiz 2 girişli olandan farklı çalışmıyor. Çıkışının 1 olabilmesi için tüm girişlerinin birden 1 olması gerekiyor.

Devreyi incelediğimizde C girişinin sadece AND kapısına bağlı olduğunu ve diğer kapıları etkilemediğini fark etmeliyiz. NAND ve NOR kapıları aynı girişlerden, yani A ve B'den veri alıyor.

Bu iki kapının istediğimiz birinden başlayarak sütunları doldurabiliriz. NAND kapısına gelen verileri işleyerek D sütununu, NOR kapısına gelen verileri işleyerek E sütununu dolduruyoruz.

Son olarak 3 girişli AND kapısına geçebiliriz. C, D ve E sütunlarının tümünün 1 olduğu yerde Q sütununa 1 yazıp, diğer satırları da 0 ile doldurmamız yeterli.

A	B	C	D	E	Q
0	0	0	1	1	0
0	0	1	1	1	1
0	1	0	1	0	0
0	1	1	1	0	0
1	0	0	1	0	0
1	0	1	1	0	0
1	1	0	0	0	0
1	1	1	0	0	0

ARDUINO IDE'DE LOJİK İFADE KULLANIMI

Arduino ile kodlama yaparken iki dijital girişten okunan değerleri birlikte işleyerek sonuç üretmemiz gereken durumlarla sıklıkla karşılaşırız. Bu durumda **bitwise operator** adı verilen ve tek bir bit değerine göre mantıksal işlem yapan sembolleri kullanırız.

Arduino'da (ve C++ dilinde) AND, OR, XOR ve NOT mantıksal işlemlerini kullanabiliriz. Lojik kapıları deneyimlediğimiz Mete Hoca LogicShield'in Arduino sketch kodu da bu bitwise operatörleri kullanarak çalışır.

& (AND)

Kullanıldığı iki ifade de 1 ise sonuç olarak 1 üretir.

Aşağıdaki örnekte Q değişkeni, A ve B girişlerinden okunan değerlerin ikisi birden 1 olduğunda 1 olur.

```
Q = digitalRead(A) & digitalRead(B);
```

| (OR)

Kullanıldığı ifadelerden herhangi biri veya ikisi birden 1 ise sonuç olarak 1 üretir. İki ifade de 0 ise 0 üretir.

Aşağıdaki örnekte Q değişkeni, A ve B girişlerinden okunan değerlerin herhangi biri veya ikisi birden 1 olduğunda 1 olur. Q değişkeni ancak iki giriş birden 0 olduğunda 0 olur.

```
Q = digitalRead(A) | digitalRead(B);
```

^ (XOR)

Kullanıldığı ifadelerin değerleri birbirlerinden farklıysa sonuç olarak 1 üretir. Diğer durumlarda 0 üretir.

Aşağıdaki örnekte Q değişkeni, A ve B girişlerinden okunan değerler birbirlerinden farklı olduğunda 1 olur. İki değer birbirine eşit olduğunda, yani ikisi de 0 veya ikisi de 1 olduğunda Q değeri 0 olur.

```
Q = digitalRead(A) ^ digitalRead(B);
```

~ (NOT)

Kullanıldığı ifadenin değerini tersine çevirir. Yani 0 ise 1 sonucu verirken, 1 ise 0 üretir. Bu karaktere tilde adı verilir ve yazabilmek için klavyede Alt tuşunu basılı tutarken numara bölümünden 126 girmek gerekir. Alt tuşunu bıraktığımızda tilde karakteri yazılacaktır.

Aşağıdaki örnekte Q değişkeni, X değişkenindeki tüm 1'leri 0, tüm 0'ları 1 yapar.

```
Q = ~X;
```

Denklem veya giriş okuma işlemlerinde ise NOT fonksiyonu için **ünlem (!)** sembolü kullanılmalıdır.

```
Q = !digitalRead(A);
```






Projelerinizden **#projezamanı** ile bahsedin!

Twitter: **metehocacom**

Instagram: **metehoca**

Daha fazla bilgi, videolu anlatımlar ve
sorularınıza cevaplar için;

www.metehoca.com