

EGE ÜNİVERSİTESİ
EGE MESLEK YÜKSEKOKULU

SAYISAL ELEKTRONİK

(DERS NOTU)

HAZIRLAYANLAR

Yar.Doç.Dr. MUSTAFA ENGİN Öğr. Gör.Dr. DİLŞAD ENGİN

İZMİR 2009

İÇİNDEKİLER

SAYISAL VE ANALOG ÇOKLUKLAR	11
GİRİŞ	11
İKİLİK SAYILAR VE MANTIK DÜZEYLERİ	12
Dalga Biçimleri	15
SAYISAL DALGALAR	16
ZAMANLAMA DİYAGRAMI	16
Veri İletimi.....	16
SAYI SİSTEMLERİ VE SAYISAL KODLAR	19
ONLUK SAYILAR.....	19
İKİLİK SAYILAR	20
İkilik - Onluk Dönüştürme	21
Onluk - İkilik Dönüştürme	22
Ondalıklı Sayılar	24
İKİLİK ARİTMETİK	26
İkilik Toplama	26
İkilik Çıkarma.....	27
İkilik Çarpma.....	28
İkilik Bölme	28
İşaretili Sayıların Gösterimi	29
İşaretili Sayılarla aritmetik İşlemler	32
ONALTILIK SAYILAR.....	38
İkilik- Onaltılık Dönüşüm.....	38
Onaltılık-İkilik Dönüşüm.....	39
Onaltılık-Onluk Dönüşüm	39
Onaltılık Toplama	41
2-tümleyen yöntemiyle onaltılık çıkarma	41
İKİLİK KODLU ONLUK SAYILAR (BCD).....	43
İKO Toplama	43
SAYISAL KODLAR.....	44

Gray Kodu	44
Gray kodu arttırmalı enkoder	47
3-Artı Kod	47
TEMEL MANTIK İŞLEMLERİ	49
GİRİŞ.....	49
DEĞİL (NOT) İŞLEMİ	49
VE (AND) İŞLEMİ	50
VEYA (OR) İŞLEMİ	54
VED (NAND) GEÇİDİ	57
VED Geçidi uygulamaları	59
VEYAD (NOR) GEÇİDİ.....	60
ÖZEL VEYA GEÇİDİ (EXCLUSIVE-OR)	63
ÖZEL VEYA DEĞİL GEÇİDİ (EXNOR GATE).....	64
TÜMDEVRE LOJİK AİLELERİ.....	67
PROBLEMLER.....	68
ÖZET	69
BOOLEAN KANUNLARI VE DEMORGAN TEOREMİ	71
GİRİŞ.....	71
BOOLEAN TOPLAMA	71
BOOLEAN ÇARPMA.....	72
BOOLEAN ARİTMETİĞİNİN KANUNLARI VE KURALLARI.....	72
Boolean Aritmetiğinin Kanunları	72
Boolean Kuralları	74
DE MORGAN TEOREMİ	78
ÇARPIMLARIN TOPLAMI	78
Toplamların Çarpımı (TÇ).....	79
BİRLEŞİK MANTIK DEVRELERİNİN TASARIMI.....	81
3-Değişkenli Karnaugh Haritasının Kullanımı	81
4-Değişkenli Karnaugh haritası.....	81
TASARIM ÖRNEĞİ	88
TOPLAYICI VE KARŞILAŞTIRICI	91

TOPLAYICILAR.....	91
PARALEL TOPLAYICILAR.....	93
KARŞILAŞTIRICILAR	94
MSI Karşılaştırmacılar	97
KODÇÖZÜCÜLER VE KODLAYICILAR	101
GİRİŞ	101
KOD ÇÖZÜCÜLER (DECODERS).....	101
Temel İkili Kodçözücü.....	101
74156 ve 74138 3-e-8 Kodçözücüler.....	103
Dört Bitlik İkili Kodçözücü	105
74154 4-E-16 Kodçözücü.....	107
İKO/Onluk Kodçözücü	108
KODLAYICILAR (ENKODERS).....	110
Onludan İKO'ya kodlayıcı	111
MULTİPLEXER DEMULTİPLEXER.....	113
GİRİŞ	113
MULTİPLEXER (VERİ SEÇİCİLER).....	113
Veri Seçici / Multiplexer Uygulamaları	116
DEMULTİPLEXER	118
74154 'ün Demultiplexer olarak kullanılması	119
74156 ve 74138 'in Demultiplexer olarak kullanılması.....	120
EŞLİK BİTİ ÜRETECİ.....	121
MULTİPLEXER UYGULAMALARI	122
TUTUCULAR VE FLİP-FLOPLAR.....	125
GİRİŞ	125
SIRALI ARDIL MANTIK.....	125
TUTUCU (LATCH) DEVRELERİ	126
Geçitli S-R Tutucu	131
Geçitli D Tutucu.....	131
KENAR TETİKLİ FLİP-FLOP'LAR.....	132
S-R Flip-Flop	133

<i>D</i> flip-flop	134
<i>J-K</i> Flip-Flop	135
<i>T</i> flip-flop.....	137
ASENKRON GİRİŞLER.....	138
FLİP-FLOPLARIN BAŞARIM ÖZELLİKLERİ	139
Yayıma Gecikmesi Süresi.....	139
Kurulma (Set-up) Süresi	140
Tutma (Hold) Süresi:	140
Maksimum Saat Frekansı	140
Vuru Genişlikleri	141
Güç Tüketimi	141
SORULAR	141
555 ZAMANLAYICI VE ÜÇ DURUMLU TAMPON	143
GİRİŞ.....	143
TEK ATIMLI VURU ÜRETEÇLERİ	143
KARARSIZ MULTIVİBRATÖRLER VE ZAMANLAYICILAR	145
555 ZAMANLAYICI.....	147
555'in Tek Atımlı Modda Çalıştırılması	148
ÜÇ-DURUMLU TAMPONLAR.....	152
SORULAR	154
SAYICILAR (COUNTERS).....	155
ASENKRON SAYICILAR.....	155
2 BİT ASENKRON SAYICI	155
3 BİT ASENKRON SAYICI	157
ASENKRON ONLUK SAYICI	159
İKİLİK ASENKRON SAYICI TÛMDEVRESİ.....	160
SENKRON SAYICILAR	163
2 BİT SENKRON SAYICI.....	163
3 BİT SENKRON SAYICI.....	164
4 BİT SENKRON SAYICI.....	166

74LS163A 4 BİT İKİLİK SENKRON SAYICI	166
YUKARI/AŞAĞI SENKRON SAYICILAR.....	168
74190 YUKARI/AŞAĞI İKO (BCD) SAYICI	170
SENKRON SAYICI TASARIMI	172
SORULAR.....	177
KAYAR YAZAÇLAR.....	181
SERİ GİRİŞLİ-SERİ ÇIKIŞLI KAYAR YAZAÇLAR	184
ÜNİVERSAL KAYAR YAZAÇ	188
SORULAR.....	190
BELLEKLER VE PROGRAMLANABİLİR ELEMANLAR.....	191
YARI İLETKEN BELLEKLER.....	191
TEMEL YARIİLETKEN BELLEK DİZİMİ.....	191
BELLEK ADRES VE KAPASİTESİ.....	191
TEMEL BELLEK İŞLEMLERİ	192
RAM'LAR VE ROM'LAR.....	194
SALT OKU BELLEKLER (ROM'LAR).....	194
ROM AİLESİ.....	194
ROM'UN YAPISI	195
ROM'UN İÇ ORGANİZASYONU	196
ROM ÖRNEKLERİ	198
ÜÇ DURUMLU ÇIKIŞLAR VE YOLLAR	200
ROM erişim süresi.....	200
ÖRNEK ROM UYGULAMASI.....	201
BİLGİSAYAR UYGULAMALARINDA ROM	202
PROGRAMLANABİLİR ROM'LAR	202
PROGRAMLAMA	202
EPROM'LAR	203
UV EPROM'lar	204
EEPROM'LAR	204
ÖRNEK EPROM 27C64.....	205
OKU/YAZ DOĞRUDAN ERİŞİMLİ BELLEKLER RAM'LER	205

RAM AİLESİ	206
STATİK RAM'LER (SRAM).....	207
STATİK RAM'İN İÇ DÜZENİ.....	208
DİNAMİK RAMLER (DRAM).....	209
DRAMİN TEMEL DÜZENİ	210
ADRES ÇOĞULLAMA.....	211
BELLEK İÇERİĞİNİN TAZELENMESİ.....	212
BELLEK GENİŞLETME	215
SÖZCÜK UZUNLUĞUNU ARTIRMA	215
SATIR SAYISI ARTIRMA	217
ÖZEL BELLEK TÜRLERİ	220
İLK GİREN İLK ÇIKAR (FIFO-FIRST IN FIRST OUT) BELLEKLER.....	220
FIFO UYGULAMALARI	221
SON GİREN İLK ÇIKAR BELLEKLER.....	221
RAM yığınlar	222
CCD BELLEKLER	226
BİLGİSAYARDA KULLANILAN BELLEKLER.....	227
DİNAMİK RAMLAR.....	227
STATİK RAMLAR	228
RAMLARIN Geleceği.....	230
SONUÇ.....	230
PROGRAMLANABİLİR MANTIK AYGITLARI.....	231
PROGRAMLANABİLİR MANTIK DİZİM (PLA)	231
MANYETİK BELLEK ÇEŞİTLERİ	231
DAC VE ADC	233
SAYISALDAN ANALOĞA ÇEVİRİCİ	233
İKİLİK AĞIRLIKLI GİRİŞLİ SAD.....	233
R/2R MERDİVEN SAD	237
ÖRNEK DAC	241
SAD BAŞARIM ÖZELLİKLERİ	246

ANALOGDAN-SAYISALA ÇEVİRİCİ (ADC).....	246
Anında analog sayısal ÇEVİRİCİ.....	247
Sayısal-yokuş ADC (digital-ramp A/D)	248
İZLEYİCİ ANALOG-SAYISAL ÇEVİRİCİ	249
TEK-EĞİMLİ ANALOG-SAYISAL ÇEVİRİCİ.....	252
ÇİFT EĞİMLİ ANALOG SAYISAL ÇEVİRİCİ.....	253
ARDIŞIK YAKLAŞIM ADC.....	255
ÖRNEK ADC	256
SORULAR.....	259
TÜMDEVRE VERİ YAPRAKLARI	261
7400 DÖRTLÜ İKİ GİRİŞLİ VED GEÇİDİ.....	262
4011 DÖRTLÜ İKİ GİRİŞLİ CMOS VED GEÇİDİ	263
74LS02 DÖRTLÜ İKİ GİRİŞLİ VEYAD GEÇİDİ.....	264
4001 DÖRTLÜ İKİ GİRİŞLİ CMOS VEYAD GEÇİDİ	265
74LS04 ALTILI DEĞİL GEÇİDİ	266
74LS08 DÖRTLÜ İKİ GİRİŞLİ VE GEÇİDİ	267
74LS20 İKİLİ DÖRT GİRİŞLİ VED GEÇİDİ	268
74LS32 DÖRTLÜ İKİ GİRİŞLİ VEYA GEÇİDİ	269
74LS86 DÖRTLÜ EXOR GEÇİDİ.....	270
74HC266 DÖRTLÜ İKİ GİRİŞLİ EXNOR GEÇİDİ	Hata!
tanımlanmamış.	Yer
	işareti
74LS283 (74LS83) 4 BİT TAM TOPLAYICI	271
74HC85 4 BİT BÜYÜKLÜK KARŞILAŞTIRICI.....	273
74LS138 3-8 KODÇÖZÜCÜ/VERİ DAĞITICI	275
74HC154 4-16 KODÇÖZÜCÜ/VERİ DAĞITICI	277
74HCT147 10 GİRİŞLİ 4 BİT ÇIKIŞLI YÜKSEK GİRİŞ ÖNCELİKLİ KODLAYICI	279
74LS151 8 GİRİŞLİ VERİ SEÇİCİ	281
74LS74A DUAL D TİPİ FLİP-FLOP	283
74HCT75 DÖRTLÜ TUTUCU.....	284
LM555 Timer	286

BÖLÜM 1

SAYISAL VE ANALOG ÇOKLUKLAR

GİRİŞ

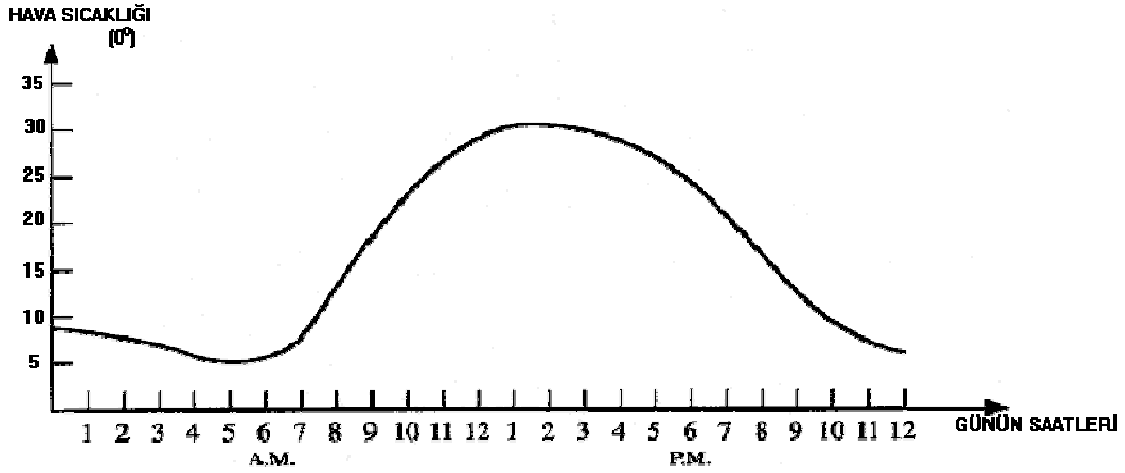
Elektronik devreleri ilgilendikleri çokluklara göre dijital (sayısal) ve analog (örneksel) olmak üzere iki ana gruba ayırabiliriz. Analog bir çoklukta değer değişimi sürekli ve kesintisiz iken, sayısal bir çokluğun değer değişimi kesiklidir ve ayrık (*discrete*) değerlerden oluşur.

Bu tanımı açmak için bir yaz günündeki ısı değişimini ele alalım. Havanın sıcaklığı birdenbire örneğin 27°C'den 28°C'ye çıkmaz, bu iki derece arasında sonsuz sayıda bütün değerleri alarak değişir. Bu değişimin grafiğini çizdiğimizde şekil-1.1'deki gibi kesintisiz ve sürekli bir eğri elde ederiz. Analog büyüklüklere diğer örnekler, zaman, basınç, uzaklık ve sestir.

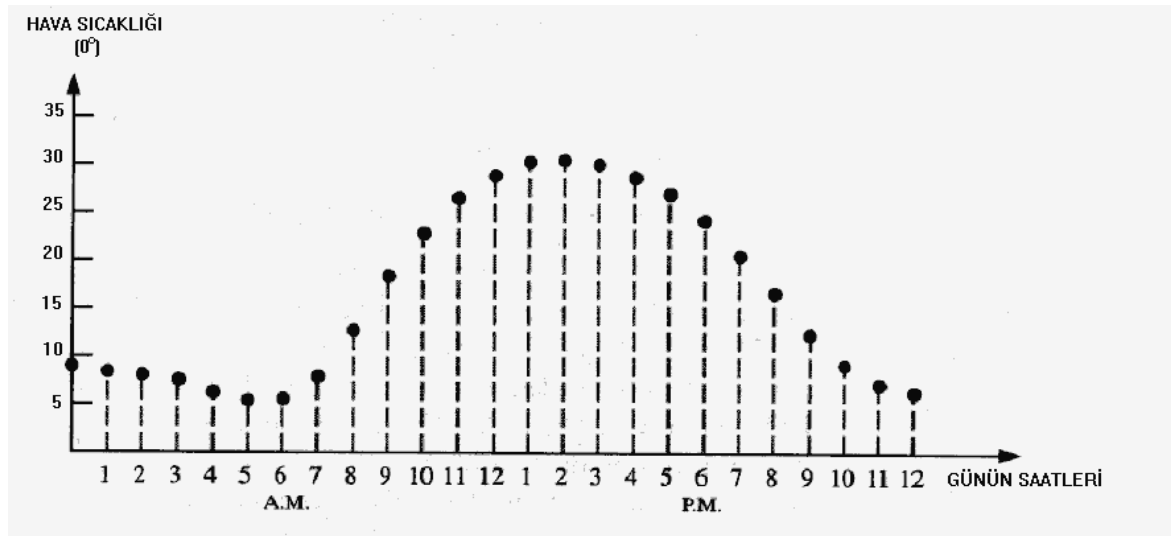
Diğer bir yöntem olarak ısıyı sürekli gözlemek yerine saat başlarında ölçerek şekil-1.2'deki gibi örnekleyebiliriz. Bu grafik henüz bir dijital gösterim değildir ama dönüşüm işleminin büyük kısmı tamamlanmıştır. Her örnek değer dijital bir kodla belirlendiğinde analog-dijital dönüşüm tamamlanmış olur. Elektronikte dijitalin analoga göre belirgin üstünlükleri vardır. En başta dijital bilgi analog bilgidен daha etkin ve güvenli olarak işlenebilir ve iletilebilir. Ayrıca bilginin saklanması gerektiğinde dijital bilginin büyük bir üstünlüğü vardır. Örneğin müzik dijitalleştirildiğinde, çok daha yoğun biçimde depolanıp büyük bir hassasiyetle yeniden üretilebilir ve analog biçime dönüştürülebilir.

Analog bir elektronik sisteme örnek olarak bir anons devresini verebiliriz. Analog doğal ses dalgaları mikrofon yardımıyla ses işareti denilen küçük analog gerilimlere dönüştürülür. Bu gerilim sesin genliği ve frekansı ile değişir ve yükselteç ile güçlendirildiğinde de bu özelliklerini yitirmez. Yükselteç yardımıyla yeterince güçlendirilen ses işareti hoparlöre uygulanarak yeniden ses dalgalarına dönüşmesi sağlanmış olur. Dijital ve analog işaretlerin birlikte kullanıldığı bir sisteme en tanınmış örnek CD çalardır. Şekil-1.3'teki basitleştirilmiş diyagram temel ilkeyi göstermektedir. Dijital formdaki müzik CD'den laser diyot yardımıyla okunur ve bu kod dizileri D/A dönüştürücüye aktarılarak ses işaretine çevrilir. Dönüştürücü çıkışında elde edilen ses işareti analog yükselteç ile güçlendirilerek

hoparlöre iletilir. Müzik CD'ye kaydedilirken burada açıklanan işlemin tersi, A/D dönüştürme kullanılmıştır.



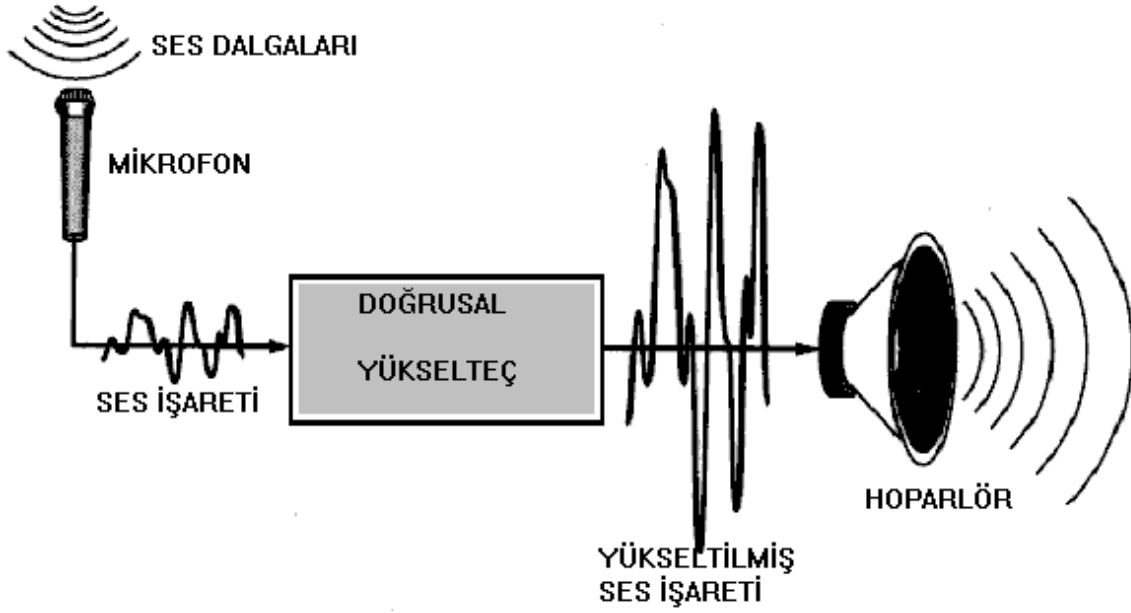
Şekil 1-1 Analog işaret örneği



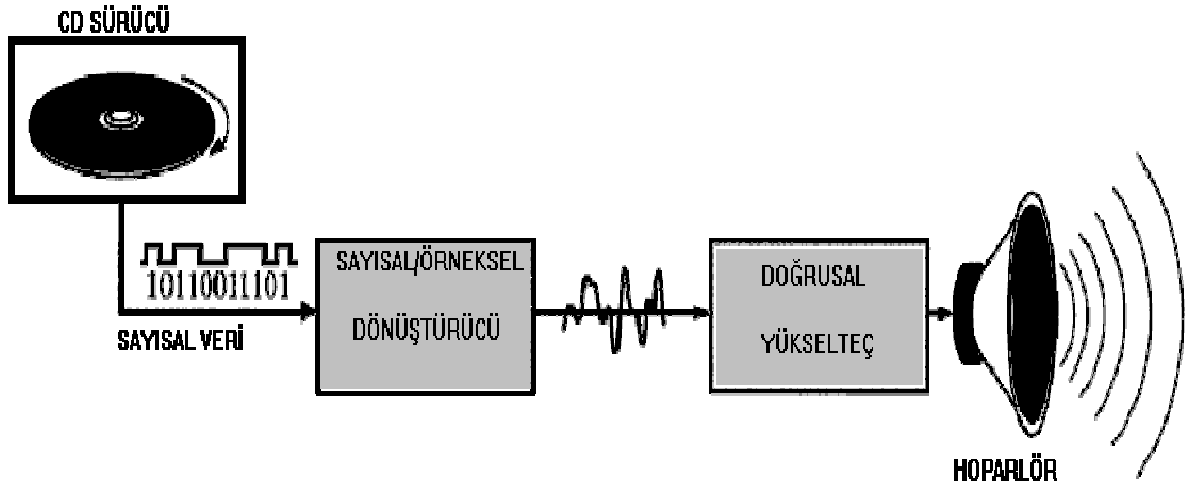
Şekil 1-2 Sayısal işaret örneği.

İKİLİK SAYILAR VE MANTIK DÜZEYLERİ

Sayısal elektronikte olası yalnız iki durum vardır: YÜKSEK yada DÜŞÜK. Bu iki durum akım şiddetleri, yanık yada sönmük lambalar, açık yada kapalı anahtarlar olarak yada en yaygın biçimiyle iki değişik gerilim değeri ile gösterilirler. Dijital sistemlerde kod dediğimiz ve bu iki durumun kombinasyonlarından oluşan diziler, sayıları, simgeleri, alfabetik karakterleri ve diğer bilgi türlerini göstermekte kullanılırlar. Bu iki durumlu sayı sistemine İKİLİK (BINARY) denir ve bu sistem 0 ve 1'den başka sayı içermez.

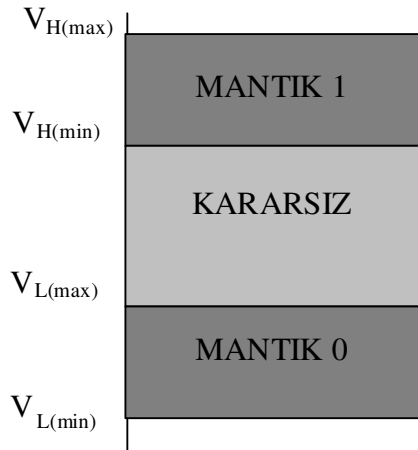


Şekil 1-3 Analog işaretin işlenmesi.



Şekil 1-4 Sayısal işaretin işlenmesi.

İkili sistemde kullanılan iki sayı yani 1 ve 0, *BIT* olarak adlandırılırlar. 1 ve 0 ları göstermek için kullanılan gerilim aralıklarına mantık düzeyi denir. $V_{H(max)}$ ile $V_{H(min)}$ arasında kalan gerilim değerleri mantık 1, $V_{L(max)}$ ile $V_{L(min)}$ arasında kalan gerilim değerleri de mantık 0 bitini ifade eder. $V_{H(min)}$ ile $V_{L(max)}$ sınırları arasında kalan gerilim değerleri belirsizlik ifadesidirler ve iki düzey arasında gerekli tampon aralığını sağlarlar.

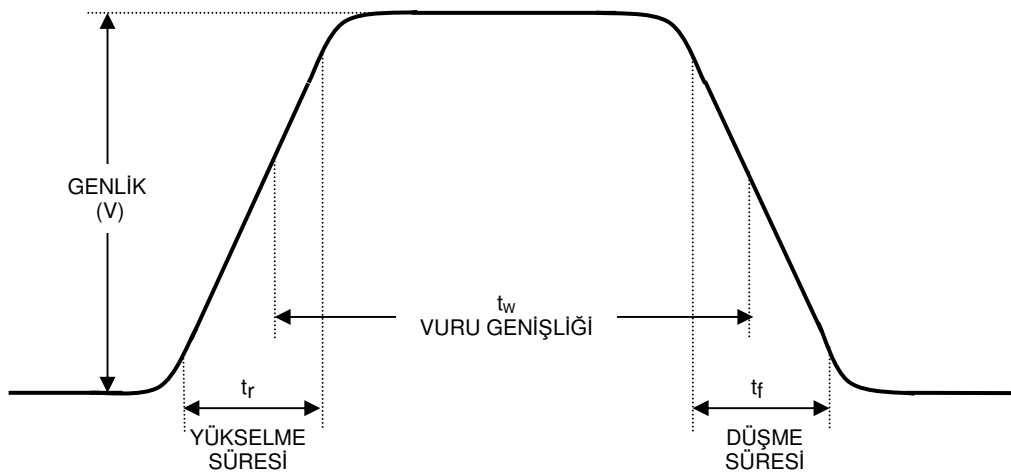


Şekil 1-5 Mantık değerlerin gerilim seviyeleri.

Sayısal dalga biçimleri YÜKSEK ve DÜŞÜK arasında gidip gelen gerilimlerden oluşurlar. Sık karşılaşılan temel kavramları açıklayalım;

VURU (PULSE)

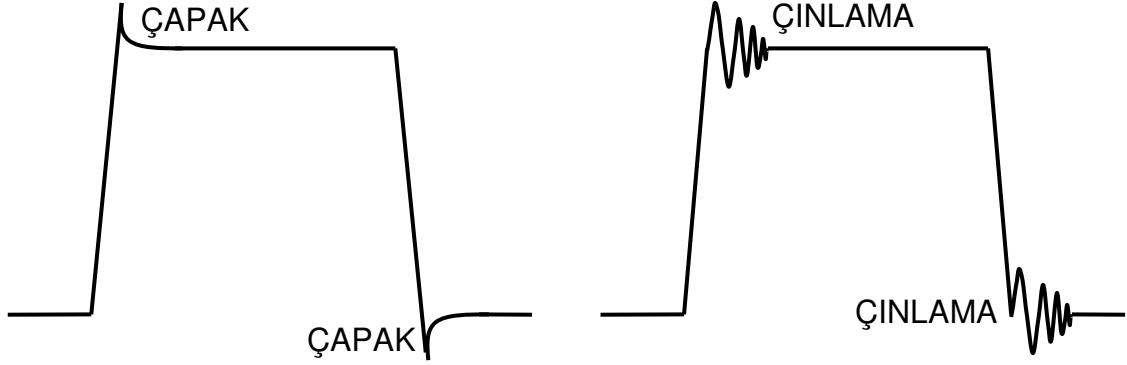
İki kenarı bulunan ve durumlar arası gidip gelen sayısal işaretin her bir adımına vuru denir. Vuruyu oluşturan kenarlardan birincisine **yükselen (önder) kenar** (*rising or leading edge*), ikincisine ise **düşen (izleyen) kenar** (*falling or trailing edge*) denir. Bir vurunun ideal olması için durum geçişlerinin sıfır sürede gerçekleşmesi gerekir ve bu duruma uygulamada hiçbir zaman ulaşamaz. Vurunun DÜŞÜK'ten YÜKSEK'e geçmesi için gereken zamana **yükselme süresi** (*rising time*), tersi için gereken süreye de **düşüş süresi** (*fall time*) denir. Bu süreler vurunun tepe genliğinin 10% ve 90% değerleri arasında ölçülür. **Vuru genişliği** genliğin 50% değerleri arasındaki süre ile ölçülür.



Şekil 1-6 Vurunun özellikleri.

Çapak ve çınılama (overshoot, undershoot and ringing), istenmeyen ama

genellikle oluşan bu bozulmalardan ilki olan çapak, devrenin yada ölçme aletinin sığasal etkisi nedeniyle oluşur ve normal değerleri kısa süreli aşan gerilim sıçramalarına neden olur. Vurumun yükselen ve düşen kenarlarında oluşan çınlama aslında küçük bir salınımdır ve devredeki kapasitans ile endüktanstan kaynaklanır. Çınlama, çapak bileşenlerini de içerir ve kısa sürede söner.



Şekil 1-7 Vuruda istem dışı oluşan çınlama ve çapaklar.

DALGA BİÇİMLERİ

Sayısal sistemlerde karşılaşılan çoğu dalga biçimi vuru dizilerinden oluşmuştur ve periyodikliklerine göre adlandırılırlar. Eğer dalga biçimi belirli bir aralıkta kendini yiniyorsa periyodik vuru olarak adlandırılır. frekans, *hertz* olarak yinelenme hızıdır. Şekil-1.8'de periyodik olan bir sayısal işaret, Şekil-1.9'da ise periyodik olmayan bir sayısal işaret gösterilmiştir. Periyodik dalgada tüm vuruların periyotları eşittir, periyodik olmayan işaretle ise her vurunun periyodu farklıdır. Sayısal sistemlerde her iki işaret türünde kullanılır ve birbirine göre üstünlüğü yoktur. Bir dalga biçiminin sıklığı periyodu ile ters orantılıdır. Sıklık ile periyot arasındaki bağıntıyı şu eşitliklerle gösterebiliriz:

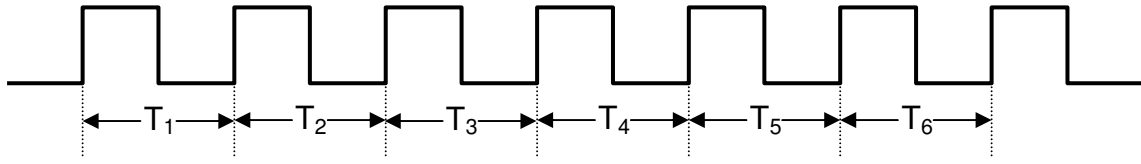
$$f = 1 / T = T^{-1}$$

$$T = 1 / f = f^{-1}$$

Periyodik bir dalga biçiminin önemli bir diğer özelliği de **görev süresidir (duty cycle)**. Görev süresi, vuru (t_w) genişliğinin periyoda olan oranının yüzdelik ifadesidir ve şu şekilde gösterilir:

$$\text{Görev süresi \% (D)} = \frac{t_w}{T} \times 100$$

Şekil-1.8'de periyodik dalganın görev süresi %50'dir. Periyodik olmayan dalganı ise görev süresi her vuru için farklıdır. Bu tür işaretlerin görev süresi hesaplanamaz. İstenildiğinde belirli bir kısmının görev süresi hesaplanabilir.



$$\text{Periyot} = T_1 = T_2 = T_3 = T_4 = T_5 = T_6$$

$$\text{Frekans} = \frac{1}{T}$$

Şekil-1.8 Periyodik dalga şekli.



Şekil-1.9 Periyodik olmayan dalga şekli.

SAYISAL DALGALAR

Sayısal sistemlerde işlenen bilgiler bit dizilerini temsil eden sayısal işaret biçimleri olarak üretilir ve iletilir. İşaretin YÜKSEK olması ikilik 1 verisini, DÜŞÜK olması da ikilik 0 bilgisini gösterir. Sıralı bitlerin her birisi, *bit süresi* denilen belirli bir zaman aralığını kaplar.

Saat (clock): Çoğu sayısal sistemde bütün dalga biçimleri saat denilen temel bir işaretle eş zamanlanırlar. Saat; vuruları arasındaki süre bir bit süresine eşit olan ve periyodik dalga biçimli bir işarettir.

ZAMANLAMA DİYAGRAMI

Zamanlama diyagramı, bütün dalga biçimlerinin zamana göre ilişkilerini ve birbirlerine göre nasıl değiştiklerini gösteren bir grafiktir ve çok sayıda sayısal işareti içerebilir. Bu diyagramlar yardımıyla bir bakışta bütün dalga biçimlerinin durumları (YÜKSEK yada DÜŞÜK) ve diğerleri ile ilişkileri görülebilir. Yanda dört dalga biçimi içeren bir zamanlama diyagramı görülüyor. Bu diyagramdan, örneğin, her üç dalga biçiminin de (A, B ve C) bit süresi 7 boyunca YÜKSEK olduğunu ve bit süresi 7 bitince hep birlikte DÜŞÜK duruma geçtikleri kolayca görülmektedir.

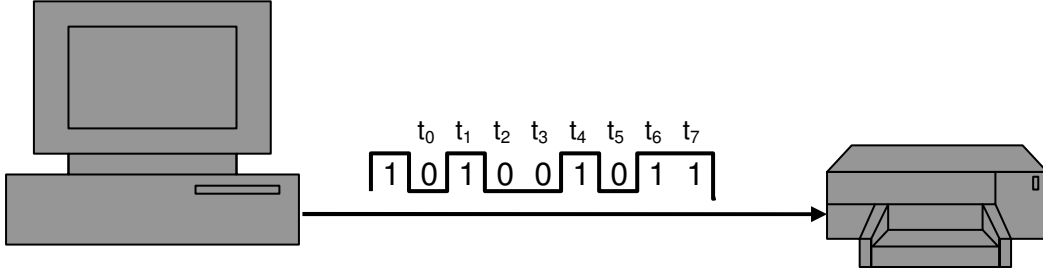
VERİ İLETİMİ

Bir tür bilgi taşıyan bit gruplarına **veri (data)** denir. Bir işlem gerçekleştirilebilmesi için, dijital dalga biçimlerinden oluşan ikilik verinin sistemler arasında iletilmesi gereklidir. Veri iletimi seri ve paralel olmak üzere iki türlü yapılmaktadır.

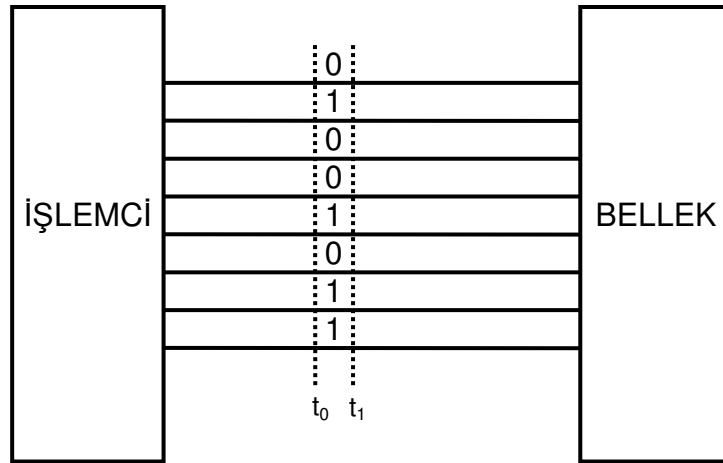
Seri iletimde bitler bir iletken hat üzerinden ardarda gönderilirler. Bu iletim türüne örnek olarak bilgisayardan yazıcı/basıcıya olan veri akışını verebiliriz.

Paralel iletimde ise küçük veri paketleri aynı anda ayrı iletkenler üzerinden gönderilirler. Her bit için bir hat gerekli olduğundan daha masraflıdır ama bir iş

süresinde gönderilebilen bit sayısı paralel hat sayısı kadar fazla olduğundan hızı çok daha yüksektir. Bu tür iletme örnek bilgisayarın mikroişlemcisi ile bellek arasındaki veri akışını verebiliriz.



a. Seri veri iletimi



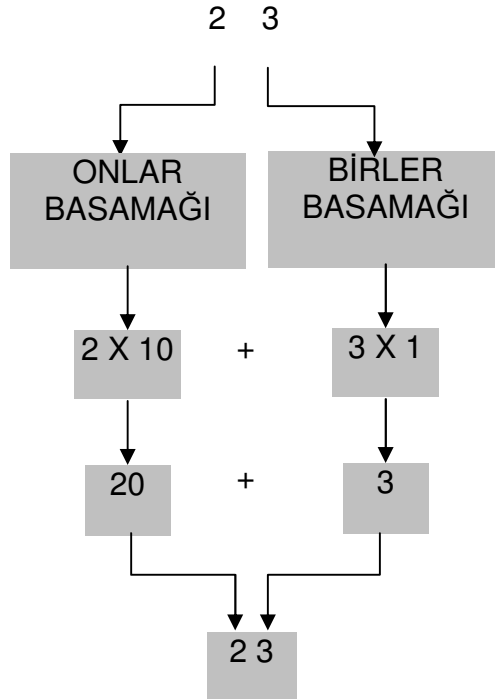
b. Paralel veri iletimi

Şekil 1-8 Seri ve paralel veri iletimi.

BÖLÜM 2

SAYI SİSTEMLERİ VE SAYISAL KODLAR

ONLUK SAYILAR



Onluk sayı sisteminde kullanılan her rakam (0 - 9) belli bir çokluğu gösterir. Buna karşın basamak değerleri değişik olduğundan gerekli basamaklara gerekli rakamları koyarak istediğimiz her çokluğu ifade edebiliriz. 9'a kadar olan çoklukları bir basamakta gösterebiliriz. Eğer dokuzdan yüksek bir değeri belirtmemiz gerekirse bir yada daha fazla basamak ekleyebiliriz. Kullanılan 10 değişik rakam olduğundan on-tabanlı sistem de denilir. Onluk 23 sayısını çarpanlarına ayıralım; 3 rakamının ağırlığı 1'dir , 2 rakamı bu basamakta 10 ağırlığındadır. Her rakam bulunduğu basamağın *ağırlığına* bağlı bir değer gösterir. Onluk sistemde basamakların ağırlığı en sağ basamakta $10^0=1$ den başlar ve sola doğru 10 un pozitif kuvvetlerini alarak artar.

$$\dots\dots 10^5 10^4 10^3 10^2 10^1 10^0$$

Kesirli sayılarda da basamakların ağırlığı sağa doğru 10'un negatif kuvvetleri ile azalır.

$$\dots\dots 10^1 10^0 \cdot 10^{-1} 10^{-2} 10^{-3} \dots\dots\dots$$

İKİLİK SAYILAR

İkilik sayı sistemi ile de dilediğimiz çokluğu gösterebiliriz. Yalnızca iki rakam içerdiği için onluk sistemden daha basittir. İkilik sistemde yalnızca iki rakam bulunduğu için iki-tabanlı sayı sistemi olarak da adlandırılır.

Onluk sistemde sayarken sıfırdan başlar ve dokuzaya dek tek basamakla gideriz. Kullanabileceğimiz rakamlar bitince bir basamak artırır ve en küçük rakamı (1) bu basamağa koyarak saymayı sürdürürüz. Yeni basamaktaki rakamı artırarak bütün kombinasyonları bitirip 99 a gelince bir basamak daha artırıp devam ederiz.

ONLUK SAYI	İKİLİK SAYI			
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1
10	1	0	1	0
11	1	0	1	1
12	1	1	0	0
13	1	1	0	1
14	1	1	1	0
15	1	1	1	1

Çizelge 2-1 4 bit ikilik sayıların onluk karşılıkları.

Yalnızca iki değişik rakam bulunması dışında ikilik sistemdeki sayma işlemi de aynı yapıdadır. Saymaya başlayalım: 0, 1. İki rakamı da kullandık. Şimdi basamak arttırmalıyız. 10, 11... bir basamak daha 100, 101, 110, 111. Şimdi dördüncü basamağa gerek duyuyoruz sonra beş, altı. Görüldüğü gibi aynı çokluğu belirtmek için ikilik sistemde onluk sistemden daha fazla basamak gerekmektedir.

İkilik sistemde belli bir sayıda basamakla gösterilebilecek en büyük onluk sayı şu şekilde hesaplanır:

$$\text{En büyük onluk sayı} = 2^n - 1$$

Burada n kullanılan bit sayısıdır. Örnek olarak beş bitlik bir ikilik sayı ile gösterilebilecek en yüksek onluk değeri hesaplayalım:

$$2^5 - 1 = 32 - 1 = 31$$

Altı bitle gösterilebilecek en yüksek değer de;

$$2^6 - 1 = 64 - 1 = 63$$

olarak bulunur.

İKİLİK - ONLUK DÖNÜŞTÜRME

Bir ikilik sayının onluk eşdeğeri, her basamaktaki bitin, o basamağın ağırlığıyla çarpılıp, sonra bütün çarpımların toplanmasıyla bulunur. En sağdaki bit en az önemli bit (*least significant bit* - *LSB*), en soldaki bit ise en önemli bit (*most significant bit* - *MSB*) olarak adlandırılır. LSB'nin ağırlığı, $2^0 = 1$ dir. MSB'nin ağırlığı ise sayının boyuna bağlıdır.

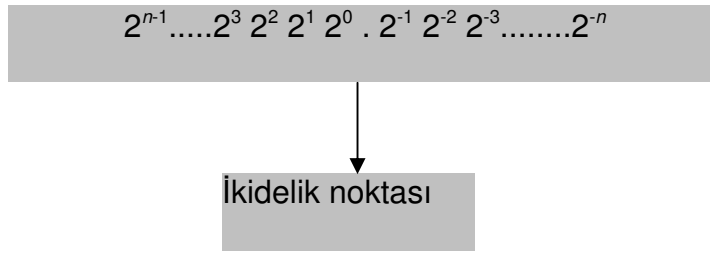
ÖRNEK 2.1:

10100101 ikilik sayının onluk karşılığını bulun.

Çözüm: Her bitin ağırlığı belirlenir, değeri bir olan bitler ağırlığı ile çarpılır ve çarpımlar toplanarak onluk karşılığı elde edilir.

Ağırlığı	2^8	2^7	2^6	2^5	2^4	2^3	2^1	2^0	
İkilik sayı	1	0	1	0	0	1	0	1	
=	1×256	+	1×64			+	1×8	+	1×1
=	256	+	64			+	8	+	1
=	329								

İkilik sistemde kesirli sayılar da gösterilebilir. Burada da aynı onluk sistemdeki gibi ondalık noktasının sağına doğru azalan negatif kuvvetler ile basamak ağırlığı düşer.

**ÖRNEK 2.2:**

101.00101 ikilik sayının onluk karşılığını bulun.

Çözüm: Her bitin ağırlığı belirlenir, değeri bir olan bitler ağırlığı ile çarpılır ve çarpımlar toplanarak onluk karşılığı elde edilir.

Ağırlığı	2^2	2^1	2^0	.	2^{-1}	2^{-2}	2^{-3}	2^{-4}	2^{-5}
İkilik sayı	1	0	1	.	0	0	1	0	1
=	4×1	+	1×1	+			0.125×1	+	0.03125×1
=	4	+	1	+			0.125	+	0.03125
=	5.15625								

ONLUK - İKİLİK DÖNÜŞTÜRME**Ağırlıklar toplamı yöntemi:**

Verilen onluk sayının ikilik karşılığını bulmada kullanılan yöntemlerin ilki *ağırlıklar toplamı*dır. Bu yöntemde, verilen onluk sayının değerini verecek bit grubu belirlenir. İkilik sistemde basamakların değerleri ya sıfırdır yada basamak ağırlığına eşittir. Buradan yola çıkarak dönüştürülecek onluk sayının değerinden küçük en büyük ağırlığa sahip olan basamağa 1 yazılır. Geri kalan miktar için de aynı işlem yapılır. Onluk sayının tam değerine ulaşana dek işlem sürdürülerek dönüşüm tamamlanır.

Örneğin onluk 9 sayısı ikilik ağırlıklar toplamı olarak şöyle gösterilebili; Bulunan ağırlıkları taşıyan basamaklara 1 değerlerine sıfır yazarak,

$$9=2^3+2^0,$$

burada bulunmayan 2^2 , 2^1 ağırlıklarının çarpanı sıfır, olanların çarpanı 1 olarak yazıldığında ikilik karşılığı elde edilir.

$$9=1001$$

Şimdi de 47 sayısını ikilik olarak yazalım;

İkilik sistemdeki basamak ağırlıkları 1, 2, 4, 8, 16, 32, 64, 128, 256, 512, 1024, 2048,... olarak sıralanır. Burada 47 sayısına sığabilen en küçük basamak 6. sıradaki 32 dir. Buna göre bulunacak ikilik eşdeğer altı basamaklı olacaktır. Geriye

$$47 - 32 = 15 \text{ kalır.}$$

Buna sığan en büyük basamak 4. sıradaki 8'dir. Geriye kalan;

$$15 - 8 = 7 \text{ 'dir}$$

Sırasıyla 3., 2. ve 1. basamaklar da 4, 2 ve 1 ağırlıklarıyla doldurulur. Ağırlıklarını kullandığımız ikilik basamaklara 1 diğerlerine sıfır yazarsak 47'nin ikilik karşılığı

$$(101111)_2$$

olarak elde edilir.

Örnek 2. 3:

Aşağıda verilen onluk sayıları ikiliye dönüştürün.

a. $12 = 8 + 4 = 2^3 + 2^2$ **1100**

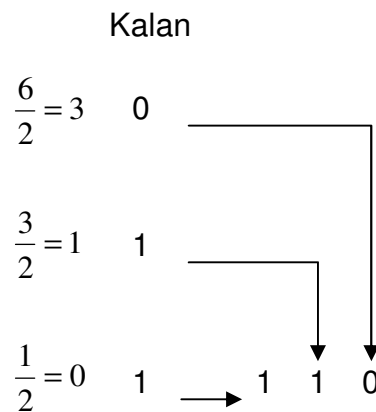
b. $25 = 16 + 8 + 1 = 2^4 + 2^3 + 2^0$ **11001**

c. $58 = 32 + 16 + 8 + 2 = 2^5 + 2^4 + 2^3 + 2^1$ **111010**

d. $82 = 64 + 16 + 2 = 2^6 + 2^4 + 2^1$ **1010010**

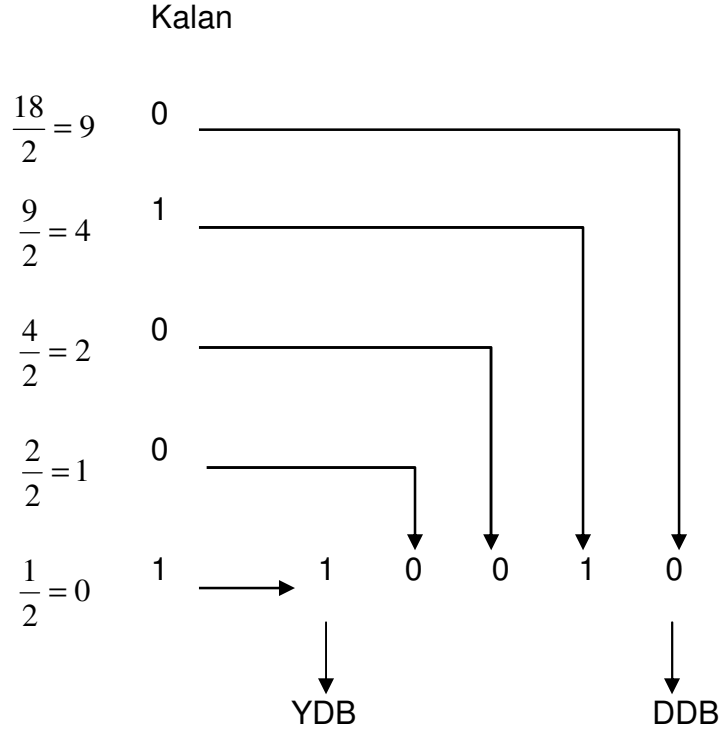
Sürekli 2' ye bölme yöntemi:

Onluk tamsayıları ikiliğe dönüştürmede kullanılan başka bir yöntem de tekrar tekrar ikiye bölmekten oluşan *sürekli 2' ye bölme yöntemi*dir. Bu yöntemde onluk sayı bölümün tamsayı kısmı 0 çikana dek ikiye bölünür. Her bölmeden sonra **kalan** ikilik sayıyı oluşturur. Örnek olarak 6'yı bu yöntemle ikilik olarak yazalım.



Örnek 2.4:

18'i 2'ye bölme yöntemi ile ikiliğe dönüştürün.

**ONDALIKLI SAYILAR****Ağırlıklar toplamı yöntemi**

Yöntem kesirli onluk sayılara da uygulanabilir. Örnek olarak 0.625 sayısını ikilik olarak yazalım:

$$0.625 = 0.5 + 0.125 = 2^{-1} + 2^{-3}$$

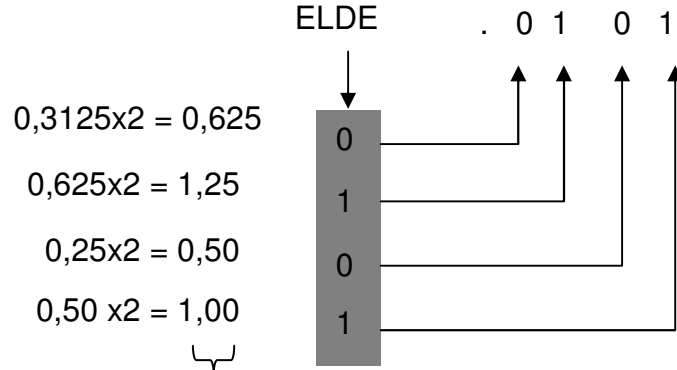
2^{-1} ve 2^{-3} basamaklarına 1 yazarak $0.625 = 0.101$ bulunur.

Sürekli 2 ile çarpma yöntemi

Bu yöntem kesirli sayıların dönüşümünde de küçük değişikliklerle kullanılır. İlk önce verilen kesirli sayı ikiyle çarpılarak sonucun ondalıklı bölümü yeniden ikiyle çarpılır. Bu işleme kesirli kısım sıfırlanana dek yada istenildiği kadar devam edilir. En sonunda sonuçların tamsayılarına bakılır. Taşınan basamakların yada eldelerin oluşturduğu ikilik bit dizisi aranan ikilik karşılığı oluşturur.

Örnek 2.5:

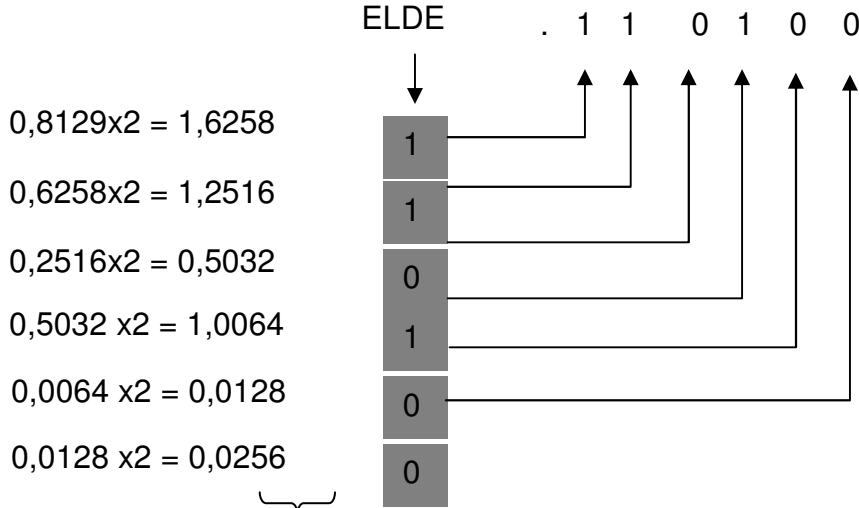
0,3125 sayısını ikilik tabana dönüştürün.



İstenilen sayıda uzatılır yada kesirli bölüm hep 0 olunca bitirilir.

Örnek 2.6:

0,8129 sayısını ikilik tabana dönüştürün.



İstenilen sayıda uzatılır yada kesirli bölüm hep 0 olunca bitirilir. Bu örnekte devam edildiğinde arka arkaya 7 sıfırdan sonra en az anlamlı bit 1 olmaktadır. Bu kadar yüksek hassasiyet istenmediği durumda kesirli kısmın sıfıra en yakın olduğu aşamada, yani 4. basamaktan sonra işlem bırakılır.

Örnek 2.7:

45,8129 sayısını ikilik tabana dönüştürün.

Sayı tam ve kesirli olmak üzere iki kısımdan oluşmaktadır. Dönüşüm ayrı ayrı yapılır, işlem sonunda birleştirilir. Sürekli 2'ye bölme yöntemi ile tamsayı kısmı dönüştürülür ve yukarıdaki örnekte yapıldığı gibi, sürekli 2 ile çarpma yöntemi ile de kesirli kısmın dönüşümü yapılır.

$$45_{10} = 101101_2$$

ve

$$0,8129_{10} \approx 0,1101_2$$

$$45,8129 \approx 101101,1101_2$$

İKİLİK ARİTMETİK

Bütün bilgisayarlarda ve çoğu diğer dijital sistemlerde ikilik aritmetik kullanılır. Dijital sistemleri anlayabilmek için ikilik toplama, çıkarma, çarpma ve bölme işlemlerini bilmek gerekir.

İKİLİK TOPLAMA

İkilik sayı bitlerini toplamının dört temel kuralı şunlardır:

$$0 + 0 = 0 \quad \text{toplama 0, elde 0}$$

$$0 + 1 = 1 \quad \text{toplama 1, elde 0}$$

$$1 + 0 = 1 \quad \text{toplama 1, elde 0}$$

$$1 + 1 = 10 \quad \text{toplama 0, elde 1}$$

ELDE BİTİ

1	+ 0 + 0 = 01
1	+ 1 + 0 = 10
1	+ 0 + 1 = 10
1	+ 1 + 1 = 11

İlk üç kuralda sonuç toplam tek bitten oluşmaktadır. Son kuralda ise biri elde biri de toplam olmak üzere iki bit vardır. İkilik sayılar toplandığında elde (varsa) bir soldaki basamağa eklenir.

Örnek 2.8:

$$(11) + (01) = ?$$

Sağ sütunda toplam $1 + 1 = 0$ ve elde 1 olarak bulunur. Elde orta sütun toplamına katılır ve $1 + 1 + 0 = 0$ ve elde 1 sonucu bulunur. Bu yeni elde de sol sütun toplamına eklenerek $1 + 0 + 0 = 1$ olarak toplamın son biti de yazılır. Elde 1 olduğunda üç bitlik bir toplama yapılması gerekir. Bu durumda toplama kuralları şu şekilde uygulanır:

$$\begin{array}{r}
 \begin{array}{|c|} \hline 1 \\ \hline \end{array} & \begin{array}{|c|} \hline 1 \\ \hline \end{array} & \\
 0 & 1 & 1 \\
 + 0 & 0 & 1 \\
 \hline
 1 & 0 & 0
 \end{array}$$

İKİLİK ÇIKARMA

İkilik çıkarmanın dört kuralı şöyle sıralanır:

$$0 - 0 = 0$$

$$1 - 1 = 0$$

$$1 - 0 = 1$$

$$10 - 1 = 1$$

Borç alındığı için aslında sonuç -1 'dir.

İkilik aritmetikte 0 dan 1 çıkarılırken bir soldaki basamaktan borç alınması gerekir. Borç 0'dan 1'i çıkarmak gerektiğinde alınır. Soldaki basamaktan borç alınınca çıkarma yapılan sütunun değeri 10_2 olur. Böylece buraya ikinci kural uygulanabilir.

Örnek 2.9:

101 'den 011 'i çıkarın.

$$\begin{array}{r}
 1 \quad 0 \quad 1 \\
 \underline{0 \quad 1 \quad 1} \\
 0 \quad 1 \quad 0
 \end{array}$$

İkinci kolonda 0'dan 1 çıkmaz, böyle durumlarda onluk sistemde olduğu gibi bir soldaki kolondan borç alınır. En soldaki kolonda da bu olay gerçekleşir ise çıkan sonuç negatif olur.

İKİLİK ÇARPMA

İkili çarpmanın dört temel kuralı vardır:

$$0 \times 0 = 0$$

$$0 \times 1 = 0$$

$$1 \times 0 = 0$$

$$1 \times 1 = 1$$

İkili sayıların çarpımı onluk sayılarınki ile aynı biçimdedir. Basamaklar birer birer çarpılır elde edilen ara toplamlar bir sola kaydırılarak yazılır. Bu ara toplamların toplamı çarpımı verir.

Örnek 2.10:

$$\begin{array}{r} 11 \\ \times 11 \\ \hline 11 \\ 11 \\ \hline \end{array}$$

$$\begin{array}{r} \text{Ara} \\ \text{çarpımlar} \left\{ \begin{array}{r} 11 \\ \times 11 \\ \hline 11 \\ + 11 \\ \hline 1001 \end{array} \right. \end{array}$$

$$\begin{array}{r} \text{Ara} \\ \text{çarpımlar} \left\{ \begin{array}{r} 111 \\ \times 101 \\ \hline 111 \\ 000 \\ + 111 \\ \hline 100011 \end{array} \right. \end{array}$$

İKİLİK BÖLME

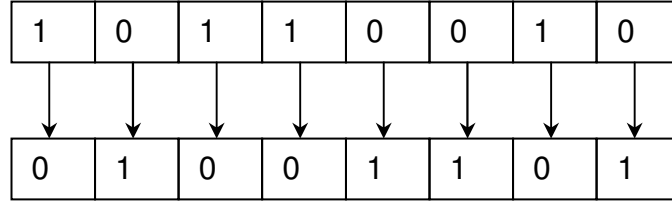
İkili sayılarda bölme onluk sayılardakiyle aynı biçimdedir.

$$\begin{array}{r} 110 \overline{)11} \\ \underline{11} \\ 000 \end{array}$$

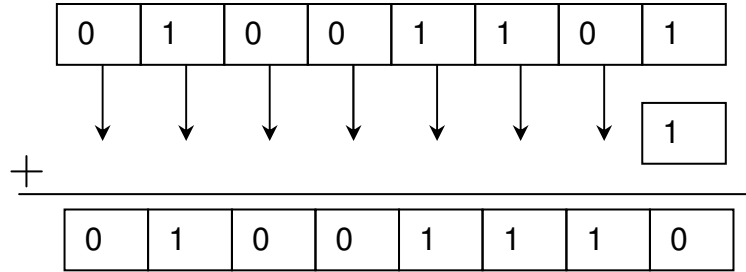
$$\begin{array}{r} 110 \overline{)10} \\ \underline{10} \\ 010 \\ \underline{10} \\ 000 \end{array}$$

1 VE 2 TÜMLEYEN KAVRAMLARI

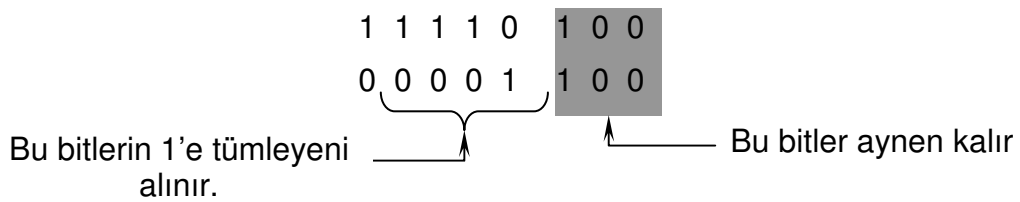
İkili sayılarda tümleyen kavramı önemlidir, çünkü negatif sayıların gösterimini sağlar ve 2-tümleyen, bilgisayarların negatif sayılarla işlem yapabilmesi için kullanılır.

İkilik sayının 1-tümleyenin bulunması

İkilik bir sayının 1-tümleyeni bütün bitleri ters çevrilerek kolayca bulunur.

İkilik sayının 2-tümleyenin bulunması

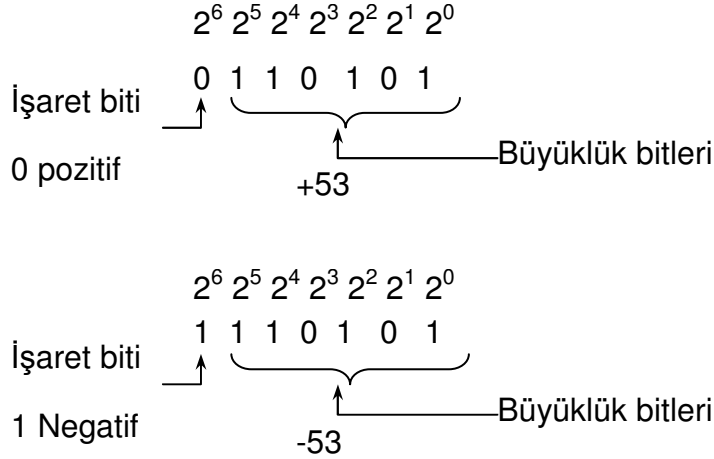
İkilik sayıların 2-tümleyeni, 1-tümleyenin LSB sine 1 eklenerek bulunur. İkilik bir sayının 2-tümleyeni bulmak için kullanılan başka bir yöntem de şudur: En sağ bittin (LSB) başlanarak sola doğru bütün bitler ilk 1 de dahil olmak üzere aynen yazılır. Geri kalan bitlerin 1-tümleyeni alınır.

**İŞARETLİ SAYILARIN GÖSTERİMİ**

İkilik sayı sisteminde negatif sayıları belirtmek için ek bir simge yoktur. Bu aritmetikte sayının işaretini belirtmek için sayı değerine ek bir bit kullanılır. İşaretili sayıların ikilik olarak yazılmasında işaret-büyük, 2-tümleyen ve 1-tümleyen sistemleri kullanılır. İkilik bir sayının en sol biti (MSB), sayının pozitif mi negatif mi olduğunu belirten işaret bitidir. Bu bit "0" pozitif, "1" negatif olduğunu göstermektedir.

İşaret-Büyüklik Sistemi

İşaretli bir sayı bu sistemde gösterildiğinde en sol bit işaret, geri kalan bitler de büyüklük için kullanılır. Büyüklük, bildiğimiz ikilik sistem ile gösterilir. Örnek olarak onluk + 53 ve -53 sayısı bu sistemde aşağıda gösterildiği gibi yazılır. + 53 ile - 53 arasındaki tek gösterim farkı işaret bitidir. Çünkü büyüklük bitleri standart ikilik sistemdedir.



İşaret-büyüklik sisteminde gösterilen bir negatif sayının büyüklük bitleri, sayının pozitif karşılığı ile aynıdır ama bundan farklı olarak işaret biti 1 dir.

1-tümleyen sistemi

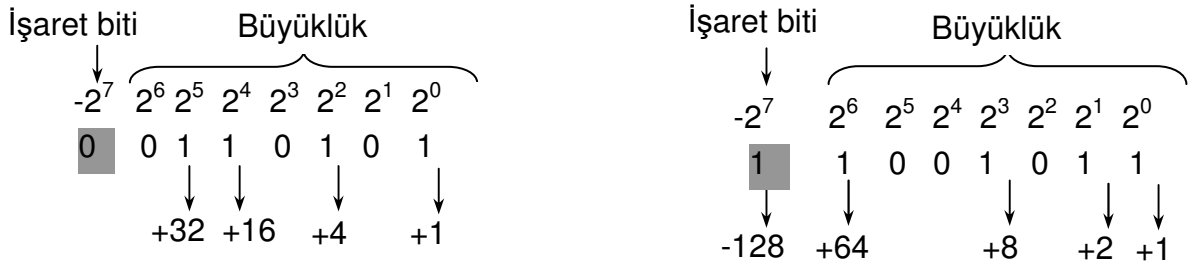
Bu sistemde pozitif sayılar aynı işaret-büyüklik sistemindeki gibi gösterilir. Negatif sayılar ise pozitif karşılığın 1-tümleyeni ile gösterilir. Örneğin - 53, + 53 sayısının 0110101 1-tümleyeni 1001010 olarak gösterilir.

Bu sistemde pozitif sayılar ağırlıklar toplamına göre değerlendirilir. Negatif sayılarda ise işaret bitinin ağırlığına negatif değer verilir ve diğer bit ağırlıkları bu değerle toplandıktan sonra sonuca 1 eklenir.

2.1.1.1 2-TÜMLEYEN SİSTEMİ

Pozitif sayıların gösterimi bu sistemde de aynıdır. Negatif sayılar ise pozitif karşılığın 2-tümleyeni olarak gösterilir. Örnek olarak yine -53 sayısını verelim. Bu sistemde +53 0110101 olarak, -53 ise 1001011 olarak gösterilir.

2-tümleyen sisteminde pozitif ve negatif sayıların onluk değerleri, 1 bulunan basamakların ağırlıkları toplanarak bulunur. Bilgisayarlarda işaretli sayıların işlenmesinde en yaygın kullanılan sistem 2-tümleyendir.

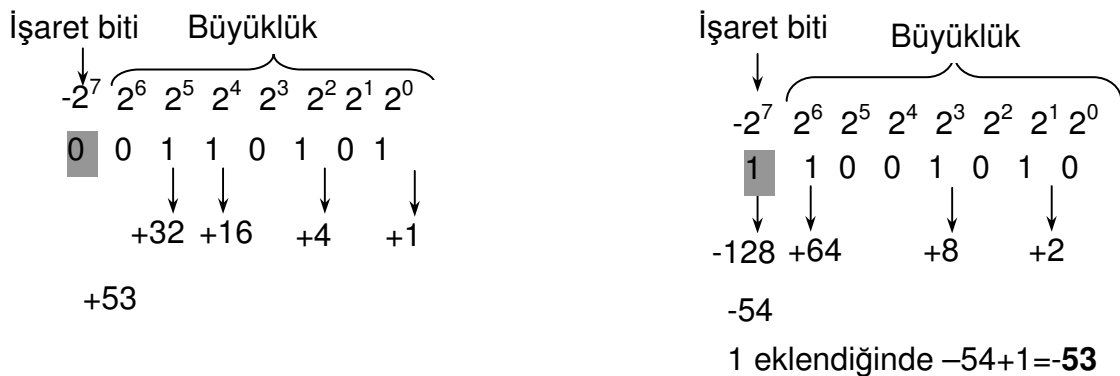


Bu örnekleri kullanarak, 2-tümleyen sisteminin 1-53 sistemine yeğlenme nedenlerinden birini söyleyebiliriz: Sayının pozitif yada negatif olmasından bağımsız olarak, yalnızca basamak ağırlıklarının toplanmasıyla onluk değer bulunabilmektedir. İşaret-büyükük sisteminde ağırlıkların toplanması ve işaret bitinin denetimi olmak üzere iki aşama vardır. 1-tümleyen sisteminde bunlara ek olarak eğer sayı negatifse toplama 1 eklenmesi gerekir. Ayrıca 1-tümleyen sisteminde sıfırın iki ayrı karşılığı vardır. 00000000 ve 11111111. Çoğu bilgisayar sistemlerinde 2-tümleyen sisteminin yeğlenme ve kullanılma nedeni aritmetik işlemlerin bu ortamda daha kolay yapılmasıdır.

Örneklere sekiz bitlik sayıların kullanılmasının nedeni, 8-bit gruplamanın bilgisayar sistemlerinde standart olmasıdır. Bu gruplar **bayt** olarak adlandırılmışlardır. Sekiz bit kullanılarak 256 değişik sayı gösterilebilir. İki bayt birleştirilerek 16 bitlik bir kod elde edildiğinde 65,536 değişik sayı gösterilebilir. 32 bitlik bir dizi ile 4.295×10^9 ayrı sayı elde edilebilir. Elde edilebilecek en yüksek birleşim sayısı 2^n eşitliğiyle bulunur. 2-tümleyen sisteminde n -bitle gösterilebilecek sayı aralığı,

$$-(2^{n-1}) \text{ ile } +(2^{n-1}-1)$$

sınırları ile belirlidir. Örneğin dört bitle 2-tümleyen sistemde $-(2^3) = -8$ ile $2^3 - 1 = +7$ arası sayılar gösterilebilir. Benzer biçimde sekiz bitle -128 ile +127, onaltı bitle de -32,768 ile +32,767 sayı belirlenebilir.



İŞARETLİ SAYILARLA ARİTMETİK İŞLEMLER

Bilgisayar ve mikroişlemci temelli sistemlerde işaretli sayılar için en yaygın gösterim 2-tümleyen sistemi olduğundan, bu bölümdeki örnekler tümüyle bu sistemi kapsamaktadır. Anlatılacak işlemler gerekirse diğer gösterim sistemlerine de uygulanabilir.

Toplama

Toplamada **artan** ve **arttıran** adlı iki sayı toplanarak **toplam** ve **elde** olmak üzere iki sonuç elde edilir. İki ikilik sayı toplandığında oluşabilecek dört durum vardır:

1. Her iki sayı pozitifdir.
2. Pozitif sayı negatif sayıdan büyüktür.
3. Negatif sayı pozitif sayıdan büyüktür.
4. Her iki sayı negatiftir.

Örnek 2.11:

Aşağıdaki toplama işlemlerini yapın.

- a. Her iki sayı pozitif

$$\begin{array}{r} 01011101 \\ + 00011011 \\ \hline 01111000 \end{array} \qquad \begin{array}{r} 93 \\ + 27 \\ \hline 120 \end{array}$$

Toplam pozitifdir ve ikilik olarak doğrudur (gerçektir).

- b. Pozitif sayı negatif sayıdan büyük

Sonuçta çıkan elde gözardı edilir ve böylece sonuç pozitif ve ikilik olarak gerçek olur.

$$\begin{array}{r} 01011101 \\ + 11011000 \\ \hline 100110101 \end{array} \qquad \begin{array}{r} 93 \\ + -40 \\ \hline 53 \end{array}$$

Oluşan elde atılır. →

- c. Negatif sayı pozitif sayıdan büyük

$$\begin{array}{r} 00011101 \\ + 11011000 \\ \hline 11110101 \end{array} \qquad \begin{array}{r} 29 \\ + -40 \\ \hline -11 \end{array}$$

Toplam negatiftir ve 2-tümleyen olarak gerçektir.

d. Her iki sayı da negatif

$$\begin{array}{r}
 11011101 \quad -35 \\
 + 11011000 \quad + -40 \\
 \hline
 110110101 \quad -75
 \end{array}$$

Oluşan elde atılır. → 1

Çıkan elde gözardı edildiğinde çıkan sonuç negatiftir ve 2-tümleyen olarak gerçektir. Bilgisayarlarda negatif sayılar 2-tümleyen formunda saklanır ve görüldüğü gibi toplama işlemi çok basittir: *İki sayı toplanır ve elde çıkarsa gözardı edilir.*

$$\begin{array}{r}
 01011101 \quad 93 \\
 + 01011000 \quad + 88 \\
 \hline
 10110101 \quad 181
 \end{array}$$

İşaret biti hatalı →

Büüklük hatalı →

Taşma Durumu İki sayı toplandığında çıkan sonucu gösterebilmek için gereken bit sayısı toplanan sayılardaki bit adedini aşarsa, yanlış işaret biti ile belirlenen taşma durumu oluşur. Taşma yalnızca her iki sayı da pozitif yada negatifse oluşur. Taşma durumunu 8-bitlik bir örnekle gösterelim:

181 sayısını ikilik olarak gösterebilmek için sekiz bit gereklidir. Sayılarda yedişer büyüklük biti olduğu için (biri tanesi işaret biti) sonucun işaret bitine taşmayı gösteren bir elde gelir. İşaret biti çıkması gerekenden farklı ise, işaret biti ve büyüklük bitleri hatalıdır. Bu hata taşma ile oluşabilir veya iki negatif sayının toplanmasında oluşur. İki'den fazla sayıyı toplamak ancak bu sayıları ikişer ikişer sırayla toplamakla olur. İlk iki sayı toplandıktan sonra bunların toplamına üçüncü sayı, yeni toplama da dördüncü sayı eklenerek işlem sürdürülür.

Örnek 2.12:

01000100, 00011011, 00001110 ve 00010010 sayılarını toplayın.

$$\begin{array}{r}
 01000100 \quad 68 \\
 + 00011011 \quad + 27 \\
 \hline
 01011111 \quad 95 \\
 + 00001110 \quad + 14 \\
 \hline
 01101101 \quad 109 \\
 + 00010010 \quad + 18 \\
 \hline
 01111111 \quad 127
 \end{array}$$

İlk iki sayı toplanır

Üçüncü sayı toplama eklenir

dördüncü sayı toplama eklenir

ÇIKARMA

Çıkarma, toplamanın özel bir durumudur. Örneğin +9 dan (**eksilen**) +6 yı (**eksilten**) çıkarmakla, -6 ile +9 u toplamak aynı sonucu verir. Çıkarma işlemi, *eksiltenin işareti değiştirilip eksilen ile toplanmasıyla* gerçekleştirilir. Çıkarma işleminin sonucuna **fark** denilir. Pozitif yada negatif bir sayının işareti, sayının **2-tümleyeni alınarak değiştirilir**. Örnek olarak pozitif 00000100 (+4) sayısının 2-tümleyeni alınınca elde edilen 11111100 sayısı $-128+64+32+16+8+4 = -4$ onluk değerini verir. Diğer bir örnek olarak 11101101 (-19) sayısının 2-tümleyeni alırsak bulacağımız 00010011 sayısı $16+2+1 = 19$ onluk değerini verir. İşaretsiz iki sayıyı çıkarmak için eksiltenin 2-tümleyeni alınır ve varsa elde biti gözardı edilir.

Örnek 2.13:

Aşağıdaki çıkarma işlemlerini yapın.

$$a. 00001000 - 00000011 = 00001000 + 11111101 = 100000101$$

Elde atılır ve sonuç: 00000101, $8 - 3 = 8 + (-3) = 5$

$$b. 00001100 - 11110111 \quad 12 - (-9) = 12 + 9 = 21$$

$$c. 11100111 - 00010011 \quad -25 - (+19) = -25 + (-19) = -44$$

$$d. 10001000 - 11100010 \quad -120 - (-30) = -120 + 30 = -90$$

ÇARPMA

Çarpma işlemindeki sayılar, çarpılan, çarpan ve çarpım olarak adlandırılırlar. Çoğu bilgisayarda çarpma işlemi de çıkarma gibi toplama işlemi kullanılarak yapılır. Doğrudan toplama ve kısmi çarpımlar, toplama kullanılarak çarpma yapma da kullanılan temel yöntemlerdir.

Doğrudan toplama yönteminde çarpılan, çarpanın sayısına eşit sayıda kendisiyle toplanır. Örneğin yukarıdaki çarpma $8 + 8 + 8 = 24$ olarak gerçekleştirilir.

$$\begin{array}{r} 3 \text{ Çarpılan} \\ \times 8 \text{ Çarpan} \\ \hline 24 \text{ Çarpım} \end{array}$$

Bu yöntemin sakıncası, çarpanın büyümesi durumunda işlem süresinin çok uzamasıdır. Eğer örneğin 350 ile 75' i çarpmak istersek 350 kendisi ile 75 kez toplanmalıdır. Çarpma işlemini ifade ederken kere terimini kullanmamızın nedeni de budur herhalde.

Kısmi çarpımlar yöntemi daha tanıdık gelecektir çünkü elle çarpma yaparken hep kullandığımız yöntemin ta kendisidir. Çarpılan sayı çarpanın en sağ basamağından (enaz önemli bit - LSB) başlanıp sola doğru her basamağıyla birer

birer çarpılır. Bu çarpımların sonucuna kısmi çarpım denir ve her biri sola doğru bir basamak kaydırılarak sırayla toplanır.

Çarpımın işareti çarpan ve çarpılanın işaretlerine bağlıdır.

- İşaretler aynı ise çarpım pozitifdir.
- İşaretler farklı ise çarpım negatifdir.

İkilik sayılar çarpılırken gerçek (tümlememiş) durumda olmalıdırlar.

Örnek 2.14:

İşaretsiz 01001101 (çarpılan) ve 00000100 (çarpan) sayılarını çarpın.

$$\begin{array}{r}
 01001101 \\
 + 01001101 \\
 \hline
 10011010 \\
 + 01001101 \\
 \hline
 11100111 \\
 + 01001101 \\
 \hline
 100110100
 \end{array}$$

Her iki sayı da pozitif ve tümlememiş durumda olduğuna göre çarpım pozitif olacaktır. Çarpanın onluk değeri 4 olduğuna göre çarpılan dört kez kendisi ile toplanır.

Çarpan ve çarpılanın işaret bitlerinin aynı olup olmadığına bakılır. Bütün negatif sayılar tümlememiş durumda olmalıdır. Bilgisayar sistemlerinin çoğunda negatif sayılar 2-tümleyen olarak saklandığından negatif sayıları gerçek ikilik duruma getirmek için 2-tümleyen işlemi kullanılarak dönüşüm yapılır. Enaz önemli bit LSB den başlayarak kısmi çarpımlar yapılır. Çarpan biti 1 olduğunda kısmi çarpım çarpılana eşit olur. Çarpan biti 0 iken kısmi çarpım sonucu sıfırdır. Her kısmi çarpım sola doğru bir kaydırılarak yazılır.

Çıkan her yeni kısmi çarpım, önceki kısmi çarpımlar toplamına eklenerek son çarpım bulunur. Başlangıçta belirlenen işaret negatifse çarpımın 2-tümleyeni alınır. Pozitif sonuçlar için çarpım gerçektir. İşaret biti çarpıma eklenir.

Örnek 2.15:

İşaretsiz 01010011 (çarpılan) ve 11000101 (çarpan) ikilik sayılarını çarpın.

1010011	Çarpılan
X0111011	Çarpan
<hr/>	
1010011	1. bitin çarpımı
+ 1010011	2. bitin çarpımı
<hr/>	
11111001	Ara toplam
+ 0000000	3. bitin çarpımı
<hr/>	
011111001	Ara toplam
+ 1010011	4. bitin çarpımı
<hr/>	
1110010001	Ara toplam
+ 1010011	5. bitin çarpımı
<hr/>	
100011000001	Ara toplam
+ 1010011	6. bitin çarpımı
<hr/>	
1001100100001	Ara toplam
+ 0000000	7. bitin çarpımı
<hr/>	
1001100100001	Sonuç (Çarpım)

Çarpılanın işaret biti 0 çarpanın işaret biti 1 olduğu için sonucun işaret biti 1 (negatif) olacaktır. Çarpanı gerçek duruma getirmek için 2-tümleyeni alınır.

1001100100001 2-tümleyeni 0110011011111 olur.

Başlangıçta sonucun işaret biti 1 olarak belirlendiğine göre çarpımın 2-tümleyeni alınır ve işaret biti eklenir. 1001100100001 sayısının 2-tümleyeni 0110011011111 olur. İşaret biti de eklenince işaretli çarpım sonucu

101100110111111 olur.

Bölme

Bölme işlemindeki sayılar **bölen**, **bölünen** ve **bölüm** olarak adlandırılırlar. Bölme işlemi bilgisayarlarda çıkarma işlemi kullanılarak yapılır. Çıkarma işlemi toplama ile yapıldığına göre bölme işlemi de toplama ile yapılabilir. Bölme işleminin sonucu, bölüm, bölünenin içinde kaç tane bölen olduğunu belirtir. Yani bölen, bölünenden, bölüm sayısı kez çıkarılabilir. Örnek olarak 21 sayısını 7 sayısına bölelim;

$$21 - 7 = 14 - 7 = 7 - 7 = 0$$

Bu basit örnekte, sıfır kalan elde edilene dek bölen bölünenden üç kez çıkarılmıştır. Buna göre bölüm 3'tür.

Bölümün işareti bölen ve bölünenin işaretlerine bağlıdır.

- işaretler aynı ise bölüm pozitifdir.
- işaretler farklı ise bölüm negatifdir.

İki ikilik sayı bölünürken her iki sayı da gerçek (tümlememiş) durumda olmalıdır.

Bölme işlemi yapılırken şu sıra izlenir;

- Bölen ve bölünen sayıların işaretlerinin aynı olup olmadığına bakılarak sonucun işaretinin ne olacağı belirlenir. Ayrıca bölüm yazacı sıfırlanır.
- 2-tümleyen toplama kullanılarak bölen bölünenden çıkarılıp ilk kısmi kalan bulunur ve bölüme 1 eklenir. Eğer bu kısmi kalan pozitifse 3. aşamaya geçilir. Eğer sonuç sıfır yada negatifse bölme tamamlanmıştır.
- Bölen kısmi kalandan çıkarılarak bölüme 1 eklenir. Eğer sonuç pozitifse işlem sürdürülür. Sonuç sıfır yada negatifse bölme tamamlanmıştır.

Örnek 2.16:

01100100 sayısını 00011001 sayısına bölün.

$$\begin{array}{r} 01100100 \text{ Bölünen} \\ + 11100111 \text{ Bölenin 2-ye tümlenmiş hali} \\ \hline 01001011 \text{ 1. ara kalan} \end{array}$$

Bölüm kaydedicisi 1 artırılır.

$$00000000+1=00000001$$

$$\begin{array}{r} 01001011 \text{ 1. ara kalan} \\ + 11100111 \text{ Bölenin 2-ye tümlenmiş hali} \\ \hline 00110010 \text{ 2. ara kalan} \end{array}$$

Bölüm kaydedicisi 1 artırılır.

$$00000001+1=00000010$$

$$\begin{array}{r} 00110010 \text{ 2. ara kalan} \\ + 11100111 \text{ Bölenin 2-ye tümlenmiş hali} \\ \hline 00011001 \text{ 3. ara kalan} \end{array}$$

Bölüm kaydedicisi 1 artırılır.

$$00000010+1=00000011$$

$$\begin{array}{r} 00011001 \text{ 3. ara kalan} \\ + 11100111 \text{ Bölenin 2-ye tümlenmiş hali} \\ \hline 00000000 \text{ 4. ara kalan} \end{array}$$

Bölüm kaydedicisi 1 artırılır.

$$00000011+1=0000100$$

Her iki sayı da pozitif olduğuna göre bölüm pozitif olacaktır. Bölümün yazılacağı kaydedici 00000000 durumuna getirilir.

2-tümleyen toplama yöntemi ile (eldelerin atıldığını unutmadan) bölen bölünenden çıkarılır:

Bölüm=00000100

ONALTILIK SAYILAR

Onaltılık sayı sisteminin tabanı onaltıdır, yani onaltı ayrı karakter (basamak) içerir. Sayısal sistemlerin çoğunda ikilik veriler dört ve katları sayıda bit içeren gruplar olarak işlendiğinden onaltılık sistemin kullanılması çok uygun olmaktadır. Çünkü her onaltılık basamak 4-bitlik bir ikilik sayıya karşılık gelmektedir. Bu sistemde kullanılan karakterlerin on tanesi nümerik altı tanesi ise alfabetiktir. A, B, C, D, E, ve F harfleri kullanılarak yazılan sayılar başlangıçta garip gelebilir ama aslında bütün sayı sistemleri bir dizi simgeden başka bir şey değildir. Bu simgelerin hangi çokluğu belirttiğini öğrendikten sonra simgelerin biçimlerinin bir önemi yoktur. F'ye kadar saydıktan sonra saymaya nasıl devam edilecek? Aynı ikilik ve onluk sistemlerdeki gibi bir basamak eklenir ve bütün simgeler sırayla yeniden sayılır:

E, F, 10, 11, 12, 13, 14, 15, 16, 17, 18, 19, 1A, 1B, 1C, 1D, 1E, 1F, 20, 21, 23, 24, 25, 26, 27, 28, 29, 2A, 2B, 2C, 2D, 2E, 2F, 30,

İki onaltılık basamakla FF_{16} yani onluk 255 sayısına dek sayılır, daha büyük sayılar için daha fazla basamak eklenir. Örneğin 100_{16} onluk 256'ya eşittir. Dört basamakla yazılabilen en büyük sayı $FFFF_{16}$ yani onluk 65,535'tir.

İKİLİK- ONALTILIK DÖNÜŞÜM

İkilik bir sayının onaltılık sisteme çevrilmesi çok kolaydır. Sayı en sağ bittten başlanarak dört bitlik gruplara ayrılır ve her bir grup, karşılığı olan onaltılık basamakla gösterilir.

Örnek 2.17:

1100101001010111_2 ve 111111000101101001_2 ikilik sayılarını onaltılık olarak yazın.

Birinci sayıdaki rakam adedi 4'ün katı olduğu için herhangi bir ek işleme gerek kalmadan dönüşüm yapılır. İkinci sayıda ise soldaki grubu dört bite tamamlamak için sola doğru sıfır(lar) eklenir.

1 1 0 0 1 0 1 0 0 1 0 1 0 1 1 1
 C A 5 7

Sonradan eklenen sıfırlar

0 0 1 1 1 1 1 1 0 0 0 1 0 1 1 0 1 0 0 1
 3 F 1 6 9

ONALTILIK-İKİLİK DÖNÜŞÜM

Onaltılık bir sayıyı ikilik olarak yazmak için ikilik-onaltılık dönüşüm işlemi tersine çevrilir yani her bir onaltılık simgenin yerine dört bitlik ikilik karşılığı yazılır.

Örnek 2.18:

$10A4_{16}$, $CF8E_{16}$ ve 9742_{16} onaltılık sayılarını ikilik olarak yazın.

(1 0 A 4)₁₆ (C F 8 E)₁₆ (9 7 4 2)₁₆
 (0001 0000 1010 0100)₂ (1100 1111 1000 1110)₂ (1001 0111 0100 0010)₂

En soldaki sıfırlar atılır.

(100010100100)₂

(1100111110001110)₂

(1001011101000010)₂

Sayının ikilik eşiti yazılırken soldaki sıfırların çıkarılması uygun olur.

ONALTILIK-ONLUK DÖNÜŞÜM

Onaltılık bir sayının onluk sisteme çevrilme yollarından birincisi, önce ikilik oradan da onluk sisteme geçmektir.

Örnek 2.19:

$1C_{16}$ ve $A85_{16}$ sayılarını onluk sistemde gösterin.

$$\begin{array}{c} (A\ 8\ 5)_{16} \\ \downarrow \quad \downarrow \quad \downarrow \\ \underbrace{1010} \quad \underbrace{1000} \quad \underbrace{0101} \end{array}$$

$$(101010000101)_2$$

$$\begin{aligned} &= 1X2^{11} + 0X2^{10} + 1X2^9 + 0X2^8 + 1X2^7 + 0X2^6 + \\ &\quad 0X2^5 + 0X2^4 + 0X2^3 + 1X2^2 + 0X2^1 + 1X2^0 \\ &= 2693_{10} \end{aligned}$$

$$\begin{array}{c} (1\ C)_{16} \\ \downarrow \quad \downarrow \\ \underbrace{0001} \quad \underbrace{1100} \end{array} \quad \text{En soldaki sıfırlar atılır.}$$

$$(11100)_2$$

$$\begin{aligned} &= 1X2^4 + 1X2^3 + 1X2^2 + 0X2^1 + 0X2^0 \\ &= 28_{10} \end{aligned}$$

Diğer bir yöntem de, onaltılık sayının her basamağındaki sayının onluk değerinin o basamağın ağırlığıyla çarpılıp bu çarpımların toplanmasıdır.

Örnek 2.20:

E_{16} ve $B2F8_{16}$ sayılarını onluk sisteme dönüştürün.

$$E_{16} = (E \times 16) + (5 \times 1) = (14 \times 16) + (5 \times 1) = 224 + 5 = 229_{10}$$

$$B2F8_{16} = (B \times 96) + (2 \times 256) + (F \times 16) + (8 \times 1)$$

$$= (11 \times 4096) + (2 \times 256) + (15 \times 16) + (8 \times 1)$$

$$= 45,056 + 512 + 240 + 8 = 45816_{10}$$

2.1.1.2 ONLUK-ONALTILIK DÖNÜŞÜM

Onluk sayıyı sürekli 16'ya bölerek kalanlardan onaltılık sayı elde edilebilir. İlk kalandan en az önemli sayı DDS (LSD), sonuncudan da en önemli sayı YDS (MSD) elde edilir. Tamsayısı sıfır olunca işlem biter.

ONALTILIK TOPLAMA

Onaltılık sayılarla toplama basamak değerlerinin 0 ile 15 arasında değiştiği akılda tutulursa doğrudan doğruya onluk sistemdeki gibi yapılabilir. Toplama yaparken şu kurallara uyulmalıdır.

Her zaman bütün sayıların onluk değerlerini gözönüne alın. Örneğin

$$5_{16} = 5_{10} \text{ ve } C_{16} = 12_{10}$$

İki sayının toplamı 15_{10} yada daha azsa, buna karşılık gelen onaltılık sayıyı bu sütunun toplamı olarak alta yazın. Eğer bu iki dijitin toplamı 15_{10} 'ten büyükse, 16_{10} 'dan büyük olan kısmını toplam olarak sütunun altına yazın ve soldaki sütuna elde 1 taşıyın.

Örnek 2.21:

Aşağıdaki verilen sayıları toplayın.

a. $23_{16} + 16_{16} = ?$

sağ sütun: $3_{16} + 6_{16} = 3_{10} + 6_{10} = 9_{10} = 9_{16}$

sol sütun: $2_{16} + 1_{16} = 2_{10} + 1_{10} = 3_{10} = 3_{16}$

toplam: 39_{16}

b. $2B_{16} + 84_{16} = ?$

sağ sütun: $B_{16} + 4_{16} = 11_{10} + 4_{10} = 15_{10} = F_{16}$

sol sütun: $2_{16} + 8_{16} = 2_{10} + 8_{10} = 10_{10} = A_{16}$

toplam: AF_{16}

c. $DF_{16} + AC_{16} = ?$

sağ sütun: $F_{16} + C_{16} = 15_{10} + 12_{10} = 27_{10} = 1B_{16}$ (elde var)

sol sütun: $1_{16} + D_{16} + A_{16} = 1_{10} + 13_{10} + 10_{10} = 24_{10} = 18_{16}$

toplam: $18B_{16}$

2-TÜMLEYEN YÖNTEMİYLE ONALTILIK ÇIKARMA

Onaltılık bir sayı ikilik bir sayıyı gösterebildiğine göre, ikilik bir sayının 2-tümleyenini de gösterebilir. Örneğin 11001001_2 onaltılık karşılığı $C9_{16}$ 'dir. Bu ikilik sayının 2-tümleyeni 00110111 onaltılık olarak yazılırsa 37_{16} elde edilir. Daha önceden de bildiğimiz gibi bir sayının 2-tümleyeni, toplama işlemi kullanarak çıkarma yapmamızı sağlar. Bu yöntem onaltılık sayılarda da geçerlidir.

Örnek 2.22:

- a.
- $84_{16} - 2A_{16}$
- işlemini yapın.

$2A_{16} = 00101010_2$ sayısının 2-tümleyeni alınır $11010110_2 = D6_{16}$ ve 84_{16} ile toplanır.

$$\begin{array}{r} 84_{16} \\ + D6_{16} \\ \hline \text{Elde atılır} \leftarrow 1 \quad 5A_{16} \end{array}$$

Oluşan elde atılır kalan sonuç gerçektir.

- b.
- $C3_{16} - 0B_{16}$
- işlemini yapın.

$0B_{16} = 00001011_2$ sayısının 2-tümleyeni alınır $11110101_2 = F5_{16}$ ve $C3_{16}$ ile toplanır.

$$\begin{array}{r} C3_{16} \\ + F5_{16} \\ \hline \text{Elde atılır} \leftarrow 1 \quad B8_{16} \end{array}$$

Oluşan elde atılır kalan sonuç gerçektir.

- c.
- $A5_{16} - B8_{16}$
- işlemini yapın.

$B8_{16} = 10111000_2$ sayısının 2-tümleyeni alınır $01001000_2 = 48_{16}$ ve $A5_{16}$ ile toplanır.

$$ED_{16} = 11101101_2 \rightarrow 00010011_2 = -13_{16}$$

$$\begin{array}{r} A5_{16} \\ + 48_{16} \\ \hline \text{Elde yok} \leftarrow 0 \quad ED_{16} \end{array}$$

Elde yoktur ve sonuç ikiye tümlenmiş haldedir. Sonucun ikiye tümleyeni alınır ve önüne eksi işareti konur.

İKİLİK KODLU ONLUK SAYILAR (BCD)

İKO her onluk basamağı bir ikilik kodla göstermenin bir yoludur. İKO sisteminde yalnızca on kod grubu bulunduğu için onluk ve İKO sistemleri arasında dönüşüm çok kolaydır. Onluk sistemde okuyup yazmayı sayıciydiğimiz için İKO ikilik sistemlerle aramızda çok uygun bir ara aşamadır. En yaygın olarak kullanıldığı yerler tuş takımları ve dijital göstergelerdir.

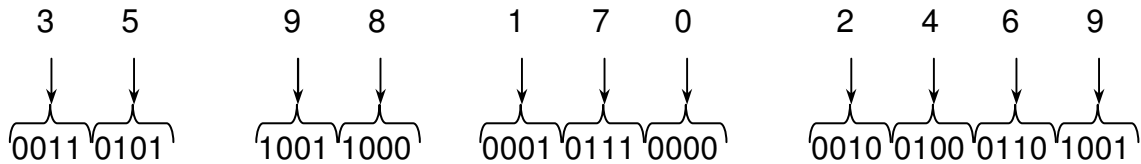
8421 kodu

8421 bir tür İKO'dur. İkilik kodlu onluk demek, 0'dan 9'a kadar olan sayılardan her birinin, ikilik bir kodla gösterilmesi demektir. 8421 adlandırması dört bitin ikilik ağırlıklarını belirtir ($2^3, 2^2, 2^1, 2^0$). 8421 kodu ile onluk sistem arasında dönüşümün kolaylığı, bu sistemin en büyük üstünlüğüdür. Bütün bilinmesi gereken, aşağıdaki tabloda gösterilmiştir. 8421 kodu dört bitten oluşur, bu bitler ile 16 değer kodlanabilir fakat BCD (İKO) bunlardan 10 tanesini kullanır. Diğerleri geçersiz kodlardır. Onluk sayıyı İKO olarak kodlamak için kolayca onluk sayının rakamları ayrı ayrı dörder bit olarak kodlanır. Soldaki sıfırlar asla atılmaz.

ONLUK	0	1	2	3	4	5	6	7	8	9
İKO	0000	0001	0010	0011	0100	0101	0110	0111	1000	1001

Örnek 2.23:

35, 98, 170, 2469 sayılarını İKO olarak kodlayın.



İKO TOPLAMA

İKO nümerik bir koddur aritmetik işlemler yapılabilir. Toplama yapılırken bazı kurallara uyulmalıdır. Bu kuralları şöyle sıralayabiliriz.

- İKO sayıları ikilik toplamanın kurallarını uygulayarak yapın.
- Eğer 4 bit toplam 9 ve 9'dan küçük ise sonuç doğrudur.
- Eğer 4 bit toplam 9'dan büyük ise İKO sonuç elde etmek için toplama 6 (0110) eklenir. Bu işlem sonrası elde oluşur ise bir soldaki dördü gruba aktarılır.

Örnek 2.24:

Aşağıdaki toplama işlemlerini yapın.

$$\begin{array}{r}
 1001 \\
 + 0100 \\
 \hline
 1101 \\
 + 0110 \\
 \hline
 0011
 \end{array}$$

Geçersiz İKO sayı
6 ekle doğru sayıyı bul.
Geçerli İKO sayı

0001
↓
1

0011
↓
3

Oluşan elde bir sola aktarılır.

$$\begin{array}{r}
 0011 \\
 + 0100 \\
 \hline
 0111
 \end{array}
 \qquad
 \begin{array}{r}
 0010 \ 0011 \\
 + 0001 \ 0101 \\
 \hline
 0011 \ 1000
 \end{array}$$

$$\begin{array}{r}
 1000 \ 0110 \\
 + 0001 \ 0011 \\
 \hline
 1001 \ 1001
 \end{array}
 \qquad
 \begin{array}{r}
 0100 \ 1101 \ 0000 \\
 + 0100 \ 1001 \ 0111 \\
 \hline
 1001 \ 0110 \ 0111
 \end{array}$$

$$\begin{array}{r}
 1001 \\
 + 1001 \\
 \hline
 0010 \\
 + 0110 \\
 \hline
 1000
 \end{array}$$

Elde oluştu yanlış sonuç.
6 ekle doğru sayıyı bul.
Geçerli İKO sayı

0001
↓
1

0011
↓
3

Oluşan elde bir sola aktarılır.

SAYISAL KODLAR

Şu ana kadar öğrendikleriniz dışın bir çok kod vardır. Bunların bazıları nümerik bazıları ise alfanümerik kodlardır. Nümerik kodlarla sadece sayılar kodlanabilir, alfanümerik kodlarla ise semboller, komutlar ve sayılar kodlanabilir. Bu bölümde İKO'ya göre daha az kullanılan nümerik kodlardan Gray (Yansıtılmış) kodu ve 3-artı kodlarını öğreneceksiniz. Alfanümerik kodlardan en yaygın kullanılan ASCII kodunu inceleyeceğiz.

GRAY KODU

Gray kodu rakamların ağırlığı olmayan ve aritmetik olmayan bir koddur. En önemli özelliği bir durumdan diğer duruma geçerken sadece bir bit değer değiştirebilir. Bu

özelliği bazı uygulamalarda özellikle bir durumdan diğer duruma geçerken birden fazla bitin değişmesi ile oluşan hatalı veya şüpheli sonucun istenmediği uygulamalarda kullanılmasına olanak sağlamıştır. Mil dönme hareketinin belirleyen arttırmalı kodlayıcılarda bu kod kullanılmaktadır. Tablo-2.3'te ikilik, onluk ve gray kodu karşılıkları gösterilmiştir. İkilik sayılarda olduğu gibi gray kodu da istenilen bit adedi ile kodlanabilir.

ONLUK	İKİLİ	GRAY	ONLUK	İKİLİ	GRAY
0	0000	0000	8	1000	1100
1	0001	0001	9	1001	1101
2	0010	0011	10	1010	1111
3	0011	0010	11	1011	1110
4	0100	0110	12	1100	1010
5	0101	0111	13	1101	1011
6	0110	0101	14	1110	1001
7	0111	0100	15	1111	1000

Tablo-2.3

2.1.1.3 İKİLİK-GRAY DÖNÜŞÜM

Gray kodunun hafızamızda kalması zordur. Onun yerine gerektiğinde ikilikten dönüşüm yapmak daha kolaydır. Gray kodunda aritmetik işlem yapılamaması da bunu zorunlu hale getirmektedir. Aşağıdaki kurallar uygulanarak dönüşüm yapılır.

- En yüksek değerli bit (MSB), en soldaki bit, ikilik ile Gray'de aynıdır.
- Soldan sağa giderken yan yana olan iki bit topla ve gray biti olarak yaz. Oluşan eldeyi at.

ÖRNEK 1: 10110_2 sayısını gray koduna dönüştürün.

Adım 1: MSB'yi ikilikten aynen Gray'e yaz.

1	0	1	1	0	İKİLİK
↓					
1					GRAY

Adım 2: Soldan başlayarak ilk iki biti topla, graye soldan ikinci bit olarak yaz.

$$\begin{array}{cccc}
 \boxed{1 + 0} & 1 & 1 & 0 & \text{İKİLİK} \\
 \downarrow & & & & \\
 1 & 1 & & & \text{GRAY}
 \end{array}$$

Adım 3: Soldan başlayarak ikinci ve üçüncü biti topla, graye soldan üçüncü bit olarak yaz.

$$\begin{array}{cccc}
 1 & \boxed{0 + 1} & 1 & 0 & \text{İKİLİK} \\
 & \downarrow & & & \\
 1 & 1 & 1 & & \text{GRAY}
 \end{array}$$

Adım 4: Soldan başlayarak üçüncü ve dördüncü biti topla, graye soldan dördüncü bit olarak yaz.

$$\begin{array}{cccc}
 1 & 0 & \boxed{1 + 1} & 0 & \text{İKİLİK} \\
 & & \downarrow & & \\
 1 & 1 & 1 & 0 & \text{GRAY}
 \end{array}$$

Adım 5: Soldan başlayarak dördüncü ve beşinci biti topla, graye soldan beşinci bit olarak yaz.

$$\begin{array}{cccc}
 1 & 0 & 1 & \boxed{1 + 0} & \text{İKİLİK} \\
 & & & \downarrow & \\
 1 & 1 & 1 & 0 & 1 & \text{GRAY}
 \end{array}$$

GRAY-İKİLİ DÖNÜŞÜM

İkili-gray dönüşümüne benzer bir kural uygulanarak dönüşüm gerçekleştirilir. Dönüşümün kurallarını aşağıdaki gibi özetleyebiliriz.

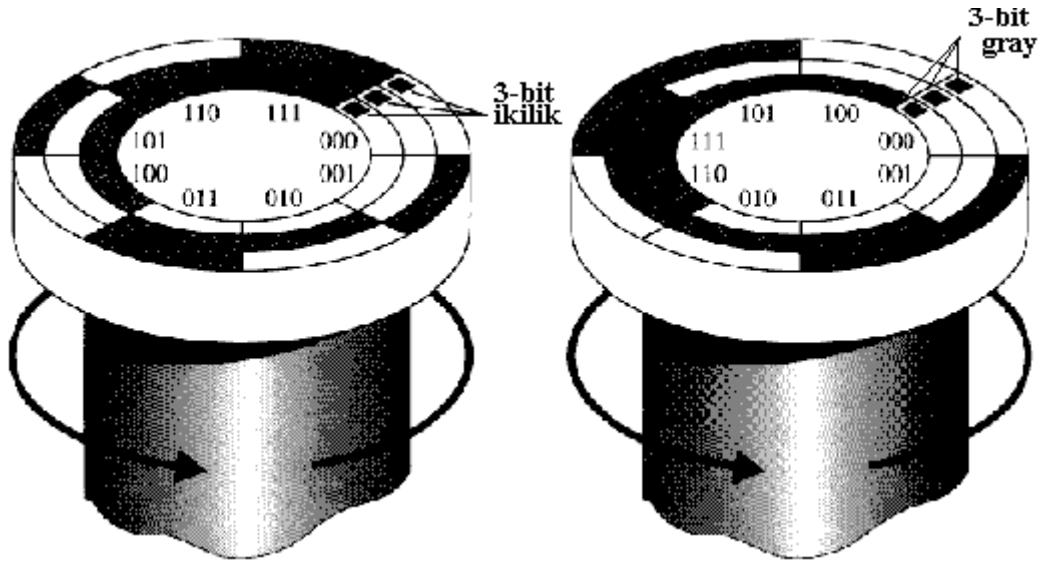
- En yüksek değerli bit (MSB), en soldaki bit, ikili ile grayde aynıdır.
- İkili'nin en soldaki biti ile Gray kodunun soldan ikinci biti toplanarak ikiliye ikinci bit olarak yazılır. Elde varsa atılır.

Örnek: Aşağıdaki dönüşümü yapın.

$$\begin{array}{cccccccc}
 1 & 0 & 1 & 0 & 1 & 1 & 1 & 1 \\
 \downarrow & \nearrow & \downarrow & \nearrow & \downarrow & \nearrow & \downarrow & \nearrow \\
 1 & 1 & 0 & 0 & 1 & 0 & 1 & 0
 \end{array}$$

GRAY KODU ARTTIRMALI ENKODER

Şekil-2.1'de 3 bitlik arttırmalı enkoderin prensip çizimleri ikilik ve gray olarak verilmiştir. İkilik olanda siyahtan beyaza geçişler birden fazla bitte aynı anda olduğundan hatalı değer elde edilmesi olasılığı yüksektir. Gray kodlu olan enkoder de ise geçiş aynı anda sadece 1 bittedir. Beyaz bölgelerde çıkıştan mantık 0 alınır, siyah bölgelerden mantık 1 alınır.



Şekil-2.1

3-ARTI KOD

İKO koda benzer bir koddur, sadece İKO'nun 4 bitlik ikilik kodlanmış sayısına 3 eklenerek elde edilir. Bitlerin pozisyonundan kaynaklanan herhangi bir ağırlıkları yoktur. Tablo-2.4'te 15'e kadar İKO ve 3-ARTI kodlamalar verilmiştir. Bu kod sisteminde hiçbir zaman tüm bitler aynı anda sıfır olmaz. En büyük özelliği budur.

ONLUK	İKO	3-ARTI	ONLUK	İKO	3-ARTI
0	0000	0011	8	1000	1011
1	0001	0100	9	1001	1100
2	0010	0101	10	0001 0000	0100 0011
3	0011	0110	11	0001 0001	0100 0100
4	0100	0111	12	0001 0010	0100 0101
5	0101	1000	13	0001 0011	0100 0110
6	0110	1001	14	0001 0100	0100 0111
7	0111	1010	15	0001 0101	0100 1000

Tablo-2.4

BÖLÜM 3

TEMEL MANTIK İŞLEMLERİ

GİRİŞ

Mantık temel anlamıyla belli koşullar sağlandığında belli bir önermenin doğru olduğunu söyleyen bir bilim dalıdır. Örneğin, "ışık yanıyor" önermesi ya doğrudur yada yanlış. "Ampul bozuk değil" ve "düğme açık" önermeleri de aynı şekildedir. Günlük yaşamda karşılaşılan çoğu sorun, işlem ve durum önerme yada mantık işlevi (logic function) olarak ifade edilebilir. Bunlar doğru/yanlış yada evet/hayır durumları olduğuna göre iki durum (1 ve 0) içeren dijital devreler bu durumlara rahatlıkla uygulanabilir.

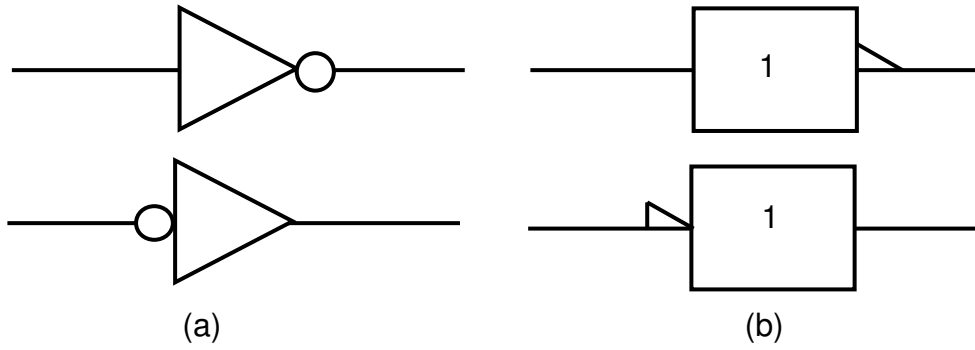
DEĞİL (NOT) İŞLEMİ

DEĞİL işlemi bir mantık değerini tersine dönüştürür. Giriş YÜKSEKse çıkış DÜŞÜKTÜR. Eğer giriş DÜŞÜKse çıkış YÜKSEK olur. DEĞİL işlemi evirici (inverter) denilen dijital mantık devresi ile gerçekleştirilir. Evirici mantık sembolü Şekil-3.1'de gösterilmiştir. Şekil-3.1(a)'da üçgen yapılı sembolü (b)'de ise kare yapılı sembolü verilmiştir. Üçgen yapıda değil deliği önce çıkışa konmuştur, bunun anlamı çıkışın etkin durumunun DÜŞÜK seviye olduğudur. Girişe konan delik ise girişin etkin durumunun DÜŞÜK seviye olduğunu belirtir. Her iki sembolü bilmek ileriki çalışmalarınızda yarar sağlayacaktır.

DEĞİL geçidinin doğruluk tablosu Tablo-3.1'de verilmiştir. Giriş YÜKSEK ise çıkış DÜŞÜK, giriş DÜŞÜK ise çıkış YÜKSEKTİR.

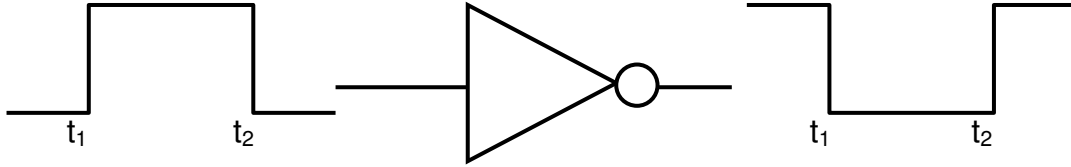
Giriş	Çıkış
DÜŞÜK (0)	YÜKSEK (1)
YÜKSEK (1)	DÜŞÜK (0)

Tablo-3.1 DEĞİL geçidinin doğruluk tablosu.



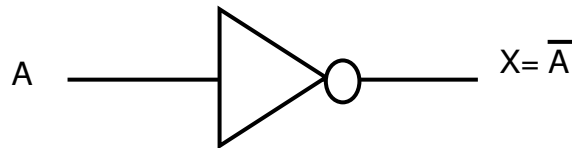
(Şekil-3.1 DEĞİL geçidinin mantık sembolleri)

DEĞİL geçidi girişine bir vuru uygulanırsa bu vurunun tersi çıkıştan elde edilir. Şekil-3.2'de giriş ve çıkış vuruları verilmiştir.



Şekil-3.2 Değil geçidini girişine uygulanan vuru ve çıkışından elde edilen vuru.

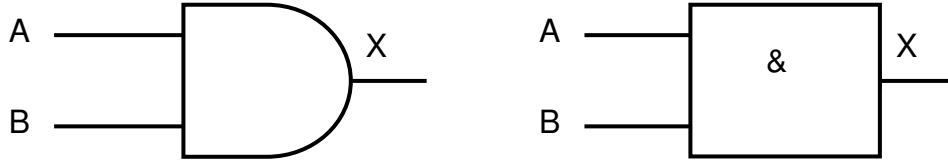
DEĞİL geçidi girişine uygulanan ikili sayının 1-e tümleyenini alır. Girişine A uygulanırsa çıkışından A'nın değili elde edilir.



Şekil-3.3 DEĞİL mantık işlevi

VE (AND) İŞLEMİ

VE işlemi, ancak bütün girişleri YÜKSEK olduğunda çıkışı YÜKSEK olan bir işlemdir. Girişlerden en az biri DÜŞÜK olursa çıkış DÜŞÜK olur. Bu mantık işlem, VE geçidi (AND gate) denilen mantık devresiyle gerçekleştirilir. Şekil-3.4'te VE geçidinin kullanılan mantık gösterimleri verilmiştir.



Şekil-3.4 VE geçidinin simgeleri. Bu tabloda sadece iki girişli bir VE geçidinin değerleri verilmiştir. Üç girişli VE geçidinin doğruluk tablosu ise Tablo-3.3'de verilmiştir. Bu iki tabloyu kullanarak daha fazla sayıda girişi bulunan VE geçidinin doğruluk tablosunu elde edebilirsiniz.

Şekil-3.5'te iki girişli bir VE geçidinin girişine verilen işaretler uygulanmıştır. Çıkışı doğruluk tablosundan yararlanılarak şekilde görüldüğü gibi çizilmiştir. Giriş işaretleri ile çıkış işaretinin aynı zaman aralıkları için çizilmesine o elemanın zamanlama diyagramı adı verildiğini birinci bölümde öğrenmiştiniz. Zamanlama diyagramı yeterli sayıda girişi olan osilaskoplardan veya lojik analizörlerden görüntülenebilir.

GİRİŞLER		ÇIKIŞ
A	B	X
0	0	0
0	1	0
1	0	0
1	1	1

Tablo-3.2 VE geçidinin doğruluk tablosu.

Doğruluk tablosu yapılırken bulunması gereken toplam kombinasyon sayısı şu formülle bulunabilir.

$$N=2^n$$

Bu formülde;

$$N=\text{toplam kombinasyon sayısı.}$$

$$n=\text{giriş sayısı.}$$

İki girişli eleman için $N=2^2=4$

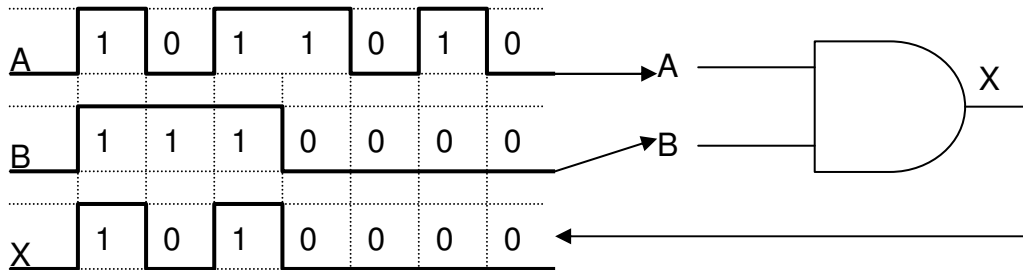
Üç girişli eleman için $N=2^3=8$

Dört girişli eleman için $N=2^4=16$

Şekil-3.5'te iki girişli VE geçidi girişlerine A ve B işaretleri uygulanmıştır. Çıkış işareti her zaman aralığında iki giriş işaretine bakılarak ve doğruluk tablosu kullanılarak elde edilir. t_1 zaman aralığında A ve B girişine gelen her iki işaret YÜKSEKtir, doğruluk tablosuna bakıldığında bu kombinasyon VE geçidinin girişlerine geldiğinde çıkışın YÜKSEK olduğu görülür. Çıkış işareti bu zaman aralığında YÜKSEK olarak çizilir. t_2 zaman aralığında A işareti DÜŞÜK B işareti YÜKSEKtir, doğruluk tablosu bu durum için çıkışın DÜŞÜK olduğunu göstermektedir, çıkış DÜŞÜK olarak çizilir. t_3 zaman aralığında A ve B işaretlerinin her ikisi YÜKSEKtir bu durumda çıkış YÜKSEK olacaktır. t_4 zaman aralığında A YÜKSEK, B DÜŞÜK seviyededir. Tablo bu durum için çıkışın DÜŞÜK seviyede olması gerektiğini göstermektedir. Bu zaman aralığı için çıkış DÜŞÜK olarak çizilir. t_5 zaman aralığında A ve B işaretlerinin her ikisi DÜŞÜK seviyededir. Tabloda çıkış için DÜŞÜK seviye gösterildiğinden çıkış DÜŞÜK olarak çizilir.

GİRİŞLER			ÇIKIŞ
A	B	C	X
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1

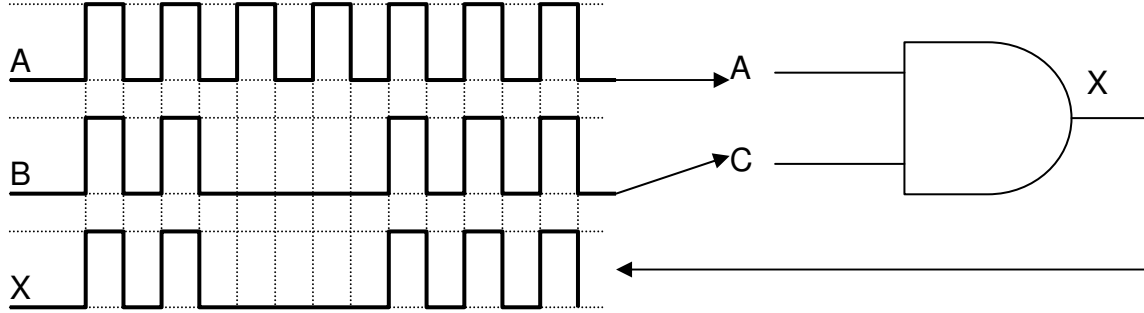
Tablo-3.3 Üç girişli VE geçidinin doğruluk tablosu.



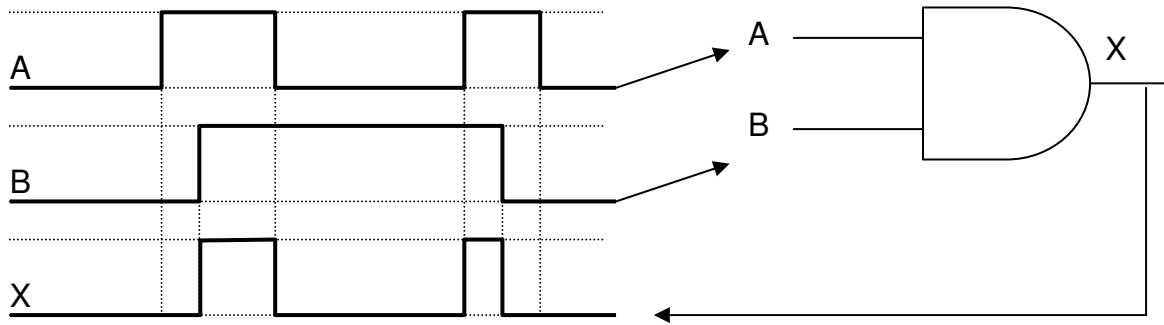
Şekil-3.5

Şekil-3.6'daki zamanlama diyagramında yine iki girişli bir VE geçidi girişine A

olarak adlandırılan bir saat işareti, B olarak adlandırılan bir kontrol işareti uygulanmıştır. A saat işareti B kontrol işaretinin izin verdiği zaman aralıklarında çıkışa aktarılır, diğer zamanlarda çıkış DÜŞÜK seviyede kalır.



Şekil-3.6



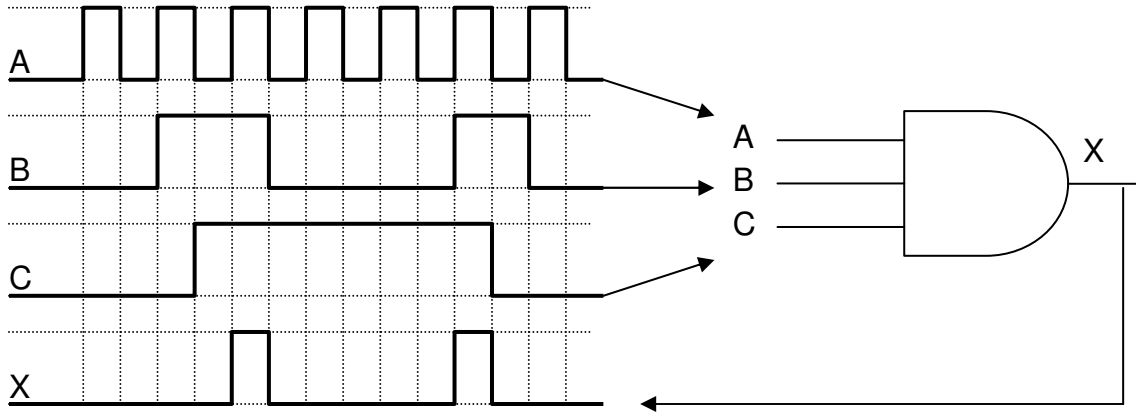
Şekil-3.7

Şekil-3.7'deki zamanlama diyagramında giriş işaretlerinin durum değiştirmeleri birbirleri ile uyumlu değildir. Bu tür durumla karşılaştığınızda işaretlerin durum değiştirme anları sizin zamanlama çizgilerini çizdiğiniz yerler olacaktır. Zaman dilimleri eşit olmayacaktır. Çıkıştan elde edilen işaret de periyodik olmayacaktır.

Şekil-3.8'de 3 girişli VE geçidinin zamanlama diyagramı görülmektedir. Diğerlerinden farkı giriş sayısının fazla olması ve buna bağlı olarak zaman çizgilerinin fazla olmasıdır.

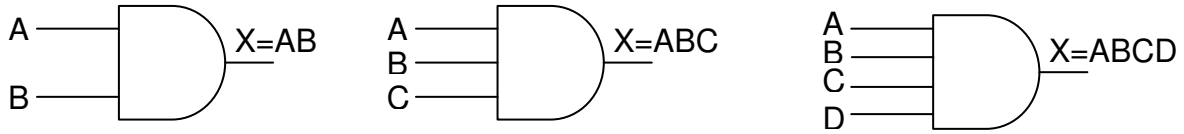
VE geçidinin mantık işlevi doğruluk tablosu incelendiğinde ikili çarpma ile aynı olduğu görülür. Kısaca **VE geçidi ikili çarpma** işlemi yapar diyebiliriz. Çıkış fonksiyonunu şöyle yazılabilir:

$X=A \cdot B$ nokta genellikle kullanılmaz, $X=AB$ olarak yazılır.



Şekil-3.8

Giriş sayısı arttıkça çarpımdaki harf sayısı da artacaktır. Şekil-3.9'da iki, üç, dört girişli VE geçidinin çıkış fonksiyonları yazılmıştır.

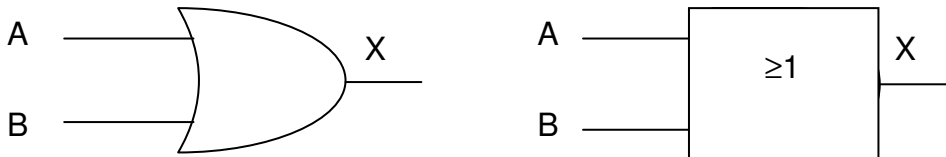


Şekil-3.9 VE geçidinde giriş sayısı arttıkça çıkış fonksiyonundaki terim

VEYA (OR) İŞLEMİ

VEYA işlemi, yalnızca bütün girişleri DÜŞÜK olduğunda çıkışı DÜŞÜK olan bir işlemdir. Girişlerden en az biri YÜKSEK olursa çıkış YÜKSEK olur. Bu mantık işlemi, **VEYA geçidi** (*or gate*) denilen mantık devresiyle gerçekleştirilir. VEYA geçidinin kullanılan her iki sembolü Şekil-3.10'da verilmiştir. Her iki sembolü derste kullanabilirsiniz.

Şekil-3.11'de ise VEYA geçidinin girişlerine olası giriş seviyeleri uygulandığında oluşacak mantık seviyeleri gösterilmiştir. Bu geçidin çıkışının YÜKSEK olabilmesi için girişlerden birinin YÜKSEK olması yeterlidir. Çıkış ancak tüm girişler DÜŞÜK ise DÜŞÜK'tür. Tablo-3.4'te doğruluk tablosu verilmiştir.

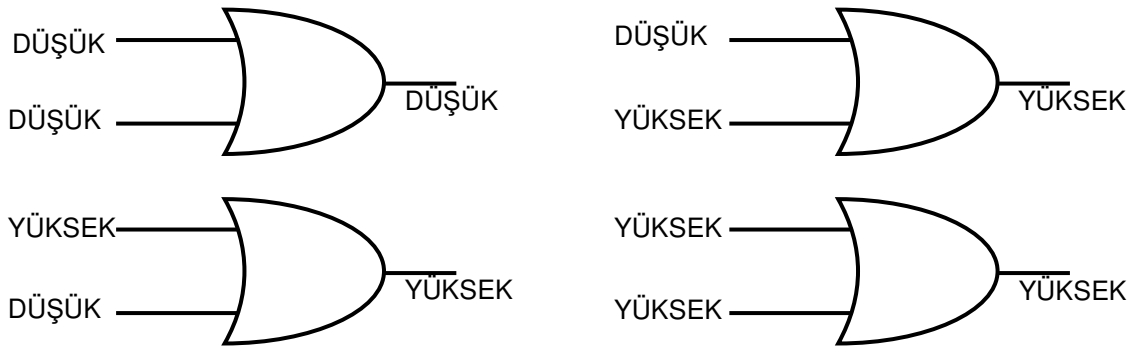


Şekil-3.10 VEYA geçidinin simgeleri.

İki VEYA geçidinin doğruluk tablosu Tablo-3.4'te verilmiştir. Giriş sayısı arttıkça kombinasyon sayısı VE geçidinde olduğu gibi artacaktır.

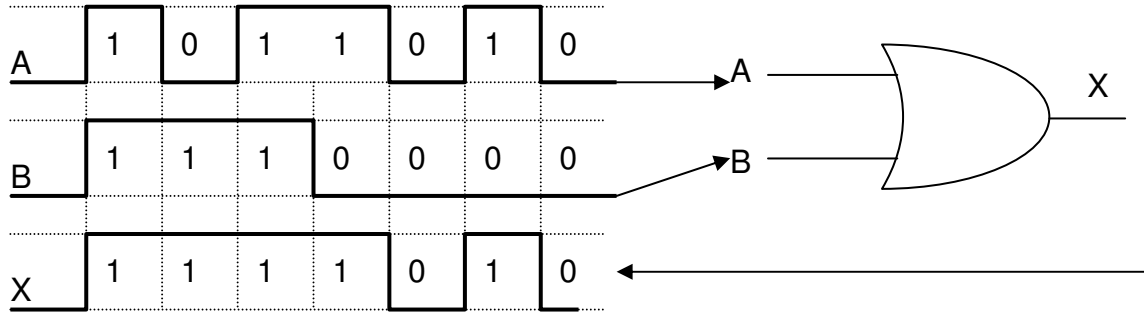
GİRİŞLER		ÇIKIŞ
A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

Tablo-3.4 VEYA geçidinin doğruluk tablosu.

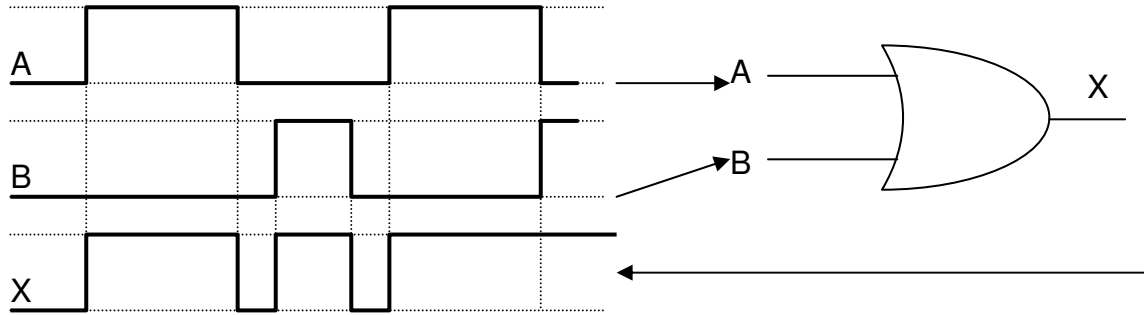


Şekil-3.11

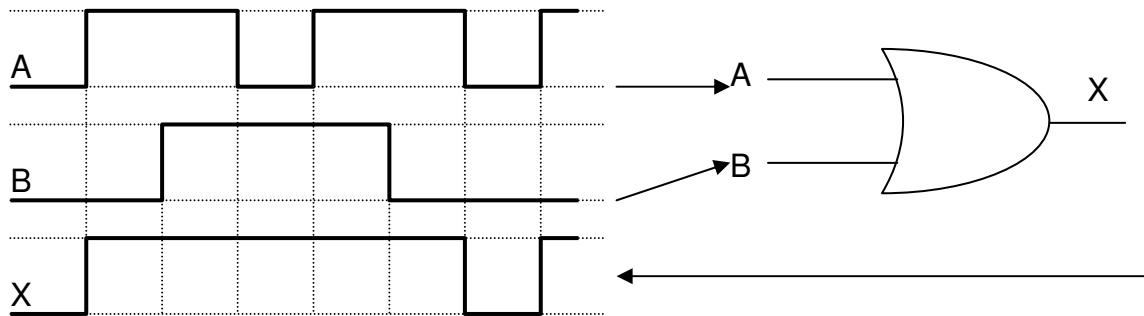
Şekil-3.12’de iki girişli VE geçidi girişlerine A ve B işaretleri uygulanmıştır. Çıkış işareti her zaman aralığında iki giriş işaretine bakılarak ve doğruluk tablosu kullanılarak elde edilir. t_1 zaman aralığında A ve B girişine gelen her iki işaret YÜKSEKtir, doğruluk tablosuna bakıldığında bu kombinasyon VEYA geçidinin girişlerine geldiğinde çıkışın YÜKSEK olduğu görülür. Çıkış işareti bu zaman aralığında YÜKSEK olarak çizilir. t_2 zaman aralığında A işareti DÜŞÜK B işareti YÜKSEKtir, doğruluk tablosu bu durum için çıkışın YÜKSEK olduğunu göstermektedir, çıkış YÜKSEK olarak çizilir. t_3 zaman aralığında A ve B işaretlerinin her ikisi DÜŞÜKTÜR, bu durumda çıkış DÜŞÜK olacaktır. t_4 zaman aralığında A YÜKSEK, B DÜŞÜK seviyededir. Tablo bu durum için çıkışın YÜKSEK seviyede olması gerektiğini göstermektedir. Bu zaman aralığı için çıkış YÜKSEK olarak çizilir.



VEYA geçidi girişine gelen işaretlerin yüksek seviyelerini toplamak amacıyla kullanılır. Şekil-3.13'teki devre bu uygulamaya bir örnektir.

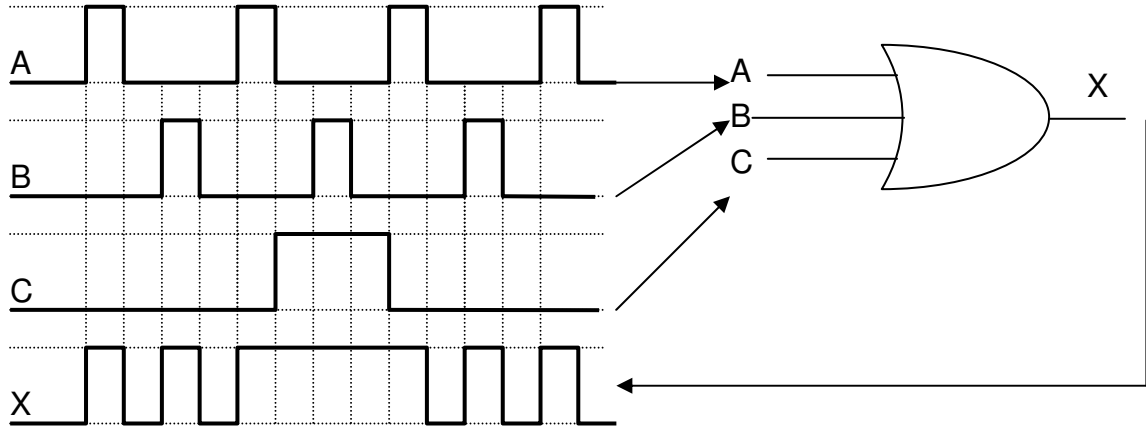


Şekil-3.13



Şekil-3.14

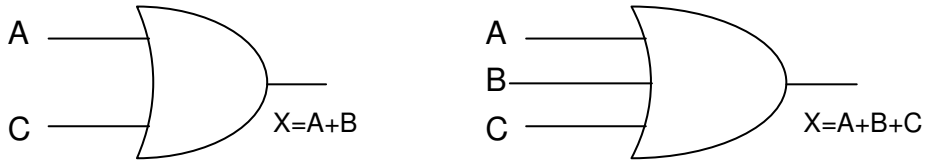
Şekil-3.14'te VEYA geçidinin girişlerine farklı işaretler uygulandığında elde edilen çıkış işaretine başka bir örnek verilmiştir. Şekil-3.15'te üç girişli VEYA geçidi girişlerine gösterilen A, B, C işaretleri uygulanmış ve çıkışından gösterilen işaret elde edilmiştir. Giriş sayısının artması zaman çizgilerinin sayısını arttırmak dışında çıkış işaretinin çizim kurallarını değiştirmemiştir.



VEYA geçidi boolean toplama işlemi yapar. İkili toplama boolean toplamadan farklıdır. Boolean toplamada $1+1=1$ 'dir, oysa ikili toplamada bu toplamdan elde oluşur ve toplam 10_2 olur. VEYA geçidinin çıkış fonksiyonunu şöyle yazabiliriz:

$$X=A+B$$

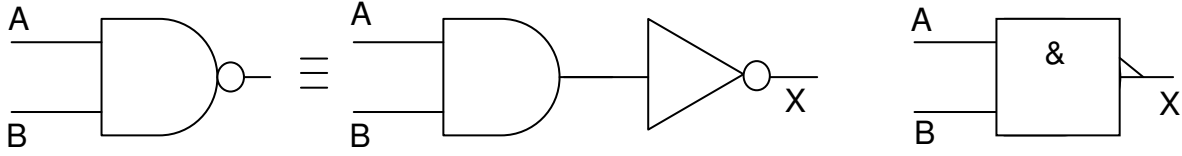
Giriş sayısı arttıkça toplama giriş sayısı kadar ekleme yapılacaktır. Şekil-3.16'da iki, üç, ve dört girişli VEYA geçitlerinin çıkış fonksiyonları verilmiştir.



Şekil-3.16

VED (NAND) GEÇİDİ

VED geçidi en yaygın olarak kullanılan bir geçittir, bunun nedeni üniversal geçit olmasıdır. Bu geçit ile diğer mantık işlemleri yapılabilir. Bu tür geçitlere üniversal geçit adı verilir. VED geçidi işlev olarak daha önce tanıdığınız VE ve DEĞİL geçitlerinin birleşimidir. Şekil-3.17'de eşdeğer devresi ile kullanılan iki farklı sembolü gösterilmiştir. VED geçidinin çıkışı sadece her iki giriş YÜKSEK olduğunda DÜŞÜK olur. Diğer durumlarda geçidin çıkışı YÜKSEK tir. Tablo-3.5'te VED geçidinin doğruluk tablosu verilmiştir. Dikkat edilirse VE geçidinin tam tersi bir işlem gerçekleştirir.

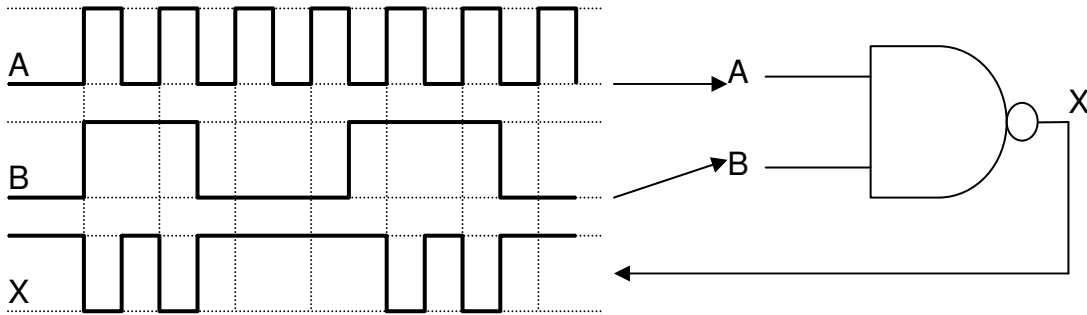


Şekil-3.18'de A ve B gibi iki farklı VED geçidi girişine uygulandığında X çıkışı doğruluk tablosu kullanılarak çizilebilir. Dikkat edilmesi gereken nokta her iki giriş 1 ise çıkış 0, aksi durumlarda çıkış 1 olur.

GİRİŞLER		ÇIKIŞ
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

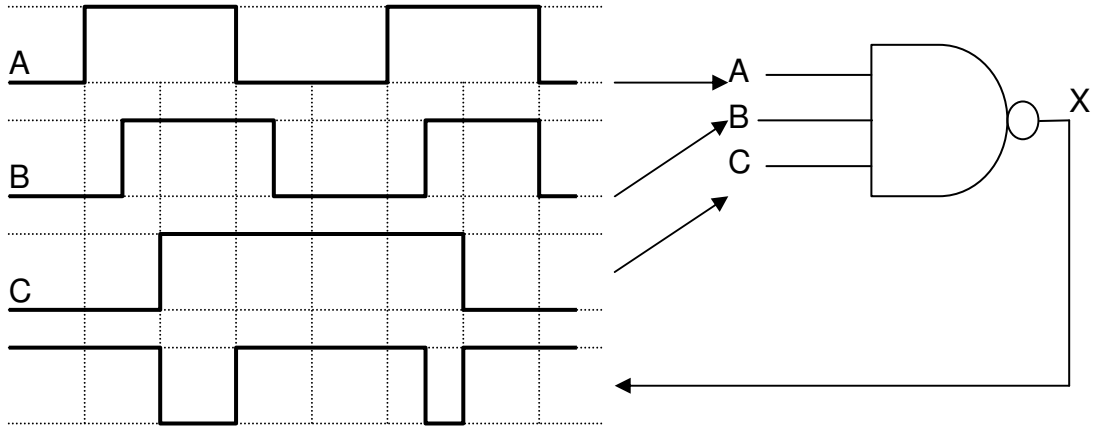
Tablo-3.5 VED geçidinin doğruluk tablosu

VED geçidi ikiden fazla girişe sahip olabilir çıkış için kural yine değişmez. Şekil-3.19'da bu duruma bir örnek verilmiştir.



Şekil-3.18

VED geçidi negatif VEYA geçidi olarak kullanılabilir. VEYA geçidinin özelliği girişlerden biri YÜKSEK ise çıkış YÜKSEKTİR. Her iki giriş DÜŞÜK ise çıkış DÜŞÜKTÜR. VED geçidinin çıkışındaki delik geçit üzerinden girişe aktarılır ise geçidin tipi değişir ve Şekil-3.20'deki görüntü elde edilir.



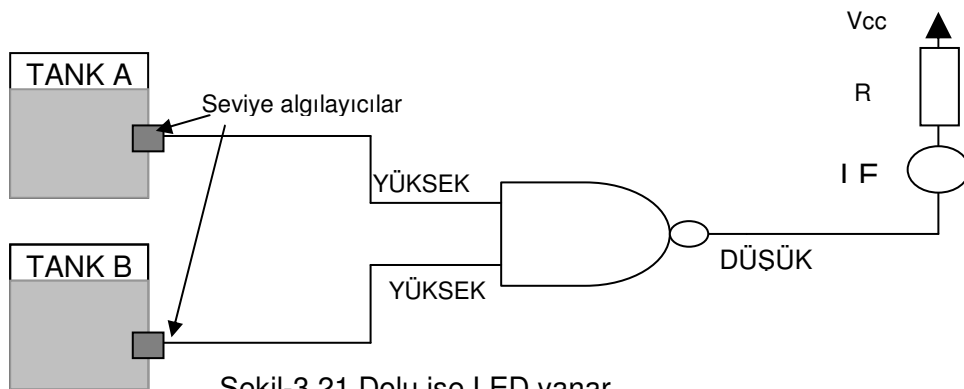
Şekil-3.19



Şekil-3.20 VED geçidinin kullanılan iki farklı simgesi

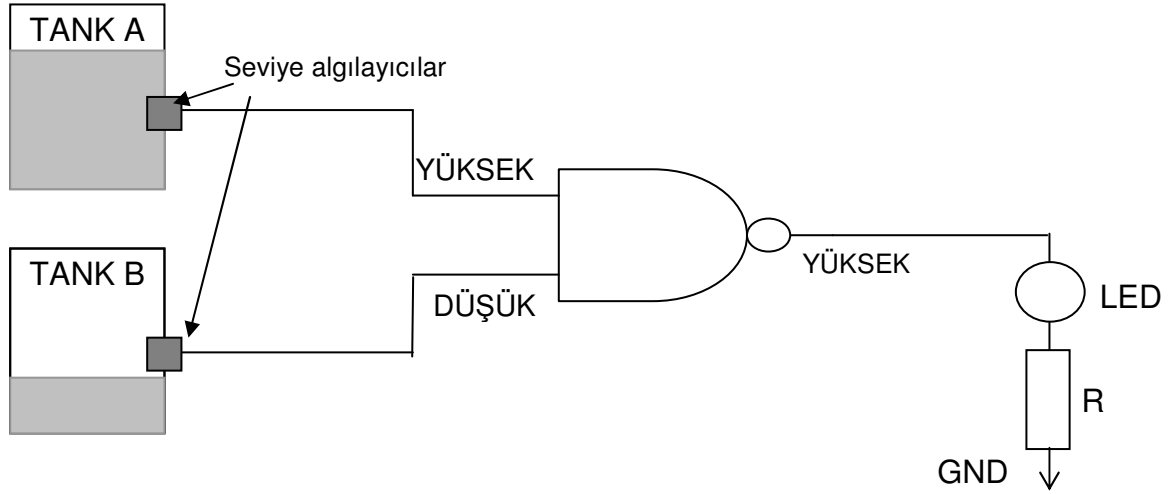
VED GEÇİDİ UYGULAMALARI

Şekil-3.21'deki seviye algılayıcılar kullanılarak tank A ve tank B'deki sıvı seviyesinin %25'in altına düşüp düşmediği belirlenecektir. Her iki tankta veya herhangi birinde seviye istenen seviyenin altına düştüğünde LED sönecektir.



Şekil-3.21 Dolu ise LED yanar.

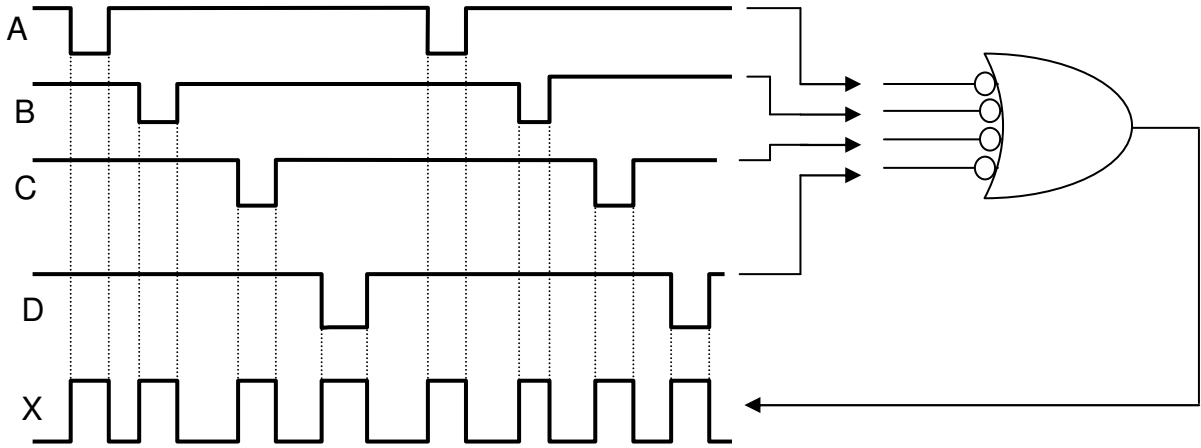
Aynı sistemde dolu durumda LED'in yanması yerine herhangi birinde %25'in altına düştüğünde uyarı LED'inin yanmasını istiyorsak devrenin Şekil-3.22'de olduğu gibi değiştirilmesi gerekir.



Şekil-3.22 %25 altına düşerse LED yanar

Şekil-3.23'te 4 girişli VED geçidi girişine verilen işaretler uygulanmıştır, çıkış işareti aynı şekilde gösterilmiştir. Giriş sayısı arttıkça çıkış dalga şeklini çizmek zorlaşır. Fakat VED geçidinin fonksiyonu uygulandığında iki girişli VED geçidi ile aynı kolaylıkla çıkış işareti çizilebilir.

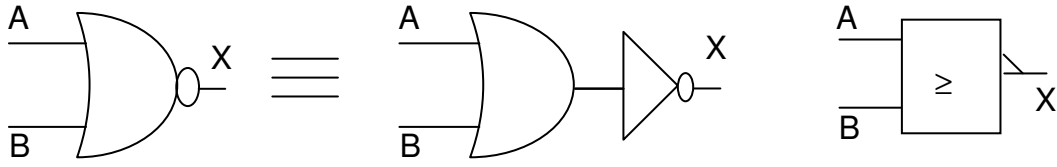
$$X = \overline{AB} = \overline{A} + \overline{B}$$



Şekil-3.23 4 girişli VED geçidine uygulanan giriş dalga şekilleri ve çıkış dalga şekli.

VEYAD (NOR) GEÇİDİ

VEYAD kelimesi VEYA ile DEĞİL kelimelerinin birleşimidir, adı gibi VEYAD geçidi VEYA geçidi ile DEĞİL geçidinin birleşimi ile elde edilir. VED geçidi gibi üniversal geçittir. Kullanılan sembolleri ve eşdeğeri şekil-3.24'te gösterilmiştir.



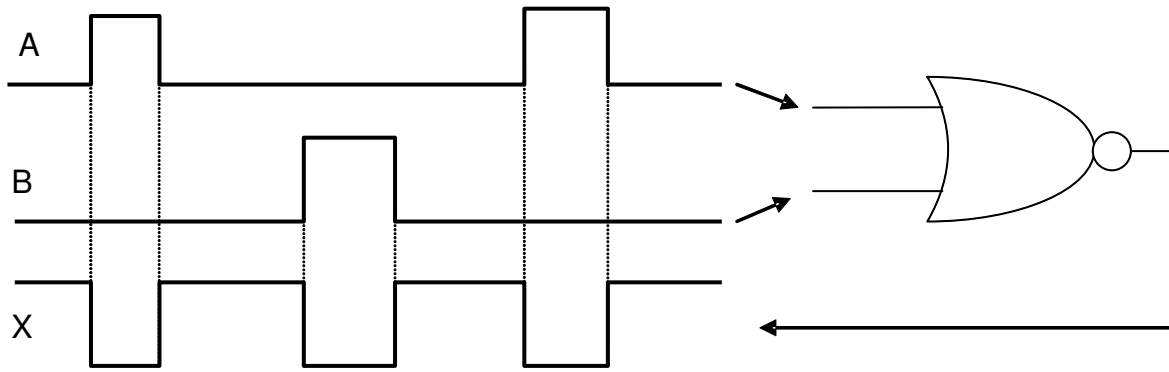
Tablo-3.6'da VEYAD geçidinin doğruluk tablosu verilmiştir. Bu geçidin çıkışının YÜKSEK olabilmesi için mutlaka her iki girişin DÜŞÜK olması gerekir. Diğer tüm giriş birleşimlerinde çıkış DÜŞÜKTÜR. Şekil-3.25'te iki girişli VEYAD geçidin girişlerine A ve B işaretleri uygulanmış ve X çıkışındaki işaret elde edilmiştir. X çıkışı çizilirken öncelikle giriş işaretlerinin değişim anlarından zaman çizgileri çizilir. Bu zaman çizgilerinde giriş işaretlerinin birleşimi doğruluk tablosundan bulunur ve çıkış işareti olarak çizilir.

VEYAD geçidi üç girişli veya daha fazla girişli olarak da üretilir. Giriş sayısının artması çıkış kuralını değiştirmez. Çıkış sadece tüm girişler DÜŞÜK olduğunda YÜKSEKTİR, diğer durumlarda DÜŞÜKTÜR. Şekil-3.26'da üç girişli bir VEYAD geçidinin girişine uygulanan A, B, C işaretleri ve bu işaretlere bağlı olarak elde edilen X işareti gösterilmiştir

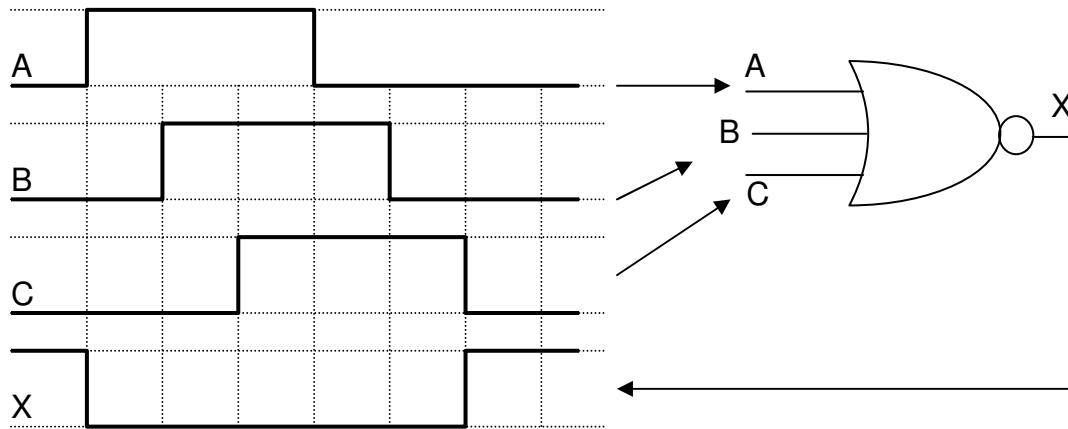
GİRİŞ		ÇIKIŞ
A	B	X
0	0	1
0	1	0
1	0	0
1	1	0

Tablo-3.6 VEYAD geçidi doğruluk tablosu

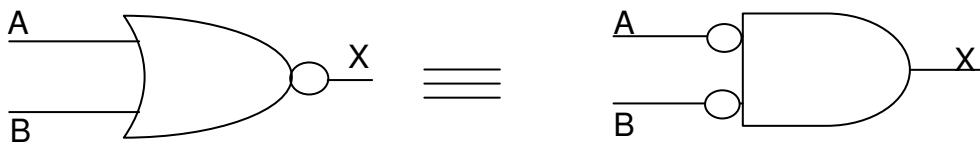
VEYAD geçidi girişleri terslenmiş VE geçidi olarak kullanılabilir. Çıkıştaki delik geçit girişlerine alındığında geçidin şekli değişir. Şekil-3.27'de gösterilen şekil elde edilir. Bu sembole girişleri terslenmiş VE geçidi veya negatif VE geçidi adı verilir. Şekil-3.28'de dört girişli negatif VE geçidi girişine A, B, C, D işaretleri uygulanmıştır. Negatif VE geçidi VEYAD geçidi ile aynı özelliğe sahip olduğundan aynı kurallar kullanılarak çıkış işareti X çizilir.



Şekil-3.25 VEYAD geçidi zamanlama diyagramı.



Şekil-3.26



Şekil-3.27 VEYAD geçidinin simgeleri.

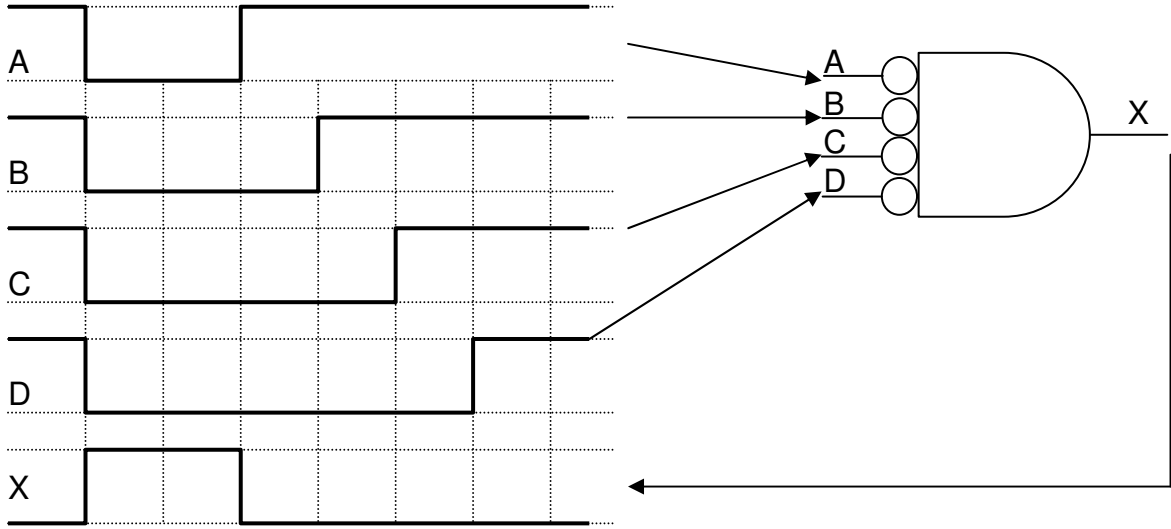
VEYAD geçidinin çıkış fonksiyonunu yazalım;

$$X = \overline{A+B} = \overline{A} \cdot \overline{B}$$

Giriş sayısı arttıkça çarpım uzayacaktır.

$$X = \overline{A+B+C} = \overline{A} \cdot \overline{B} \cdot \overline{C}$$

$$X = \overline{A+B+C+D} = \overline{A} \cdot \overline{B} \cdot \overline{C} \cdot \overline{D}$$



Şekil-3.28

ÖZEL VEYA GEÇİDİ (EXCLUSIVE-OR)

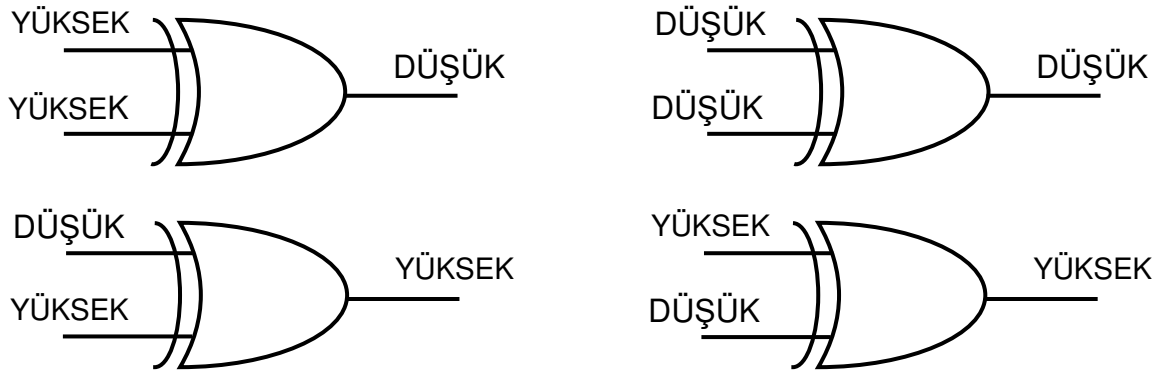
Bu geçit aslında diğer geçitlerin kombinasyonundan oluşur. Bir çok uygulamada kullanılması bu geçidin bağımsız bir sembolü olmasına neden olmuştur. Kullanılan standart sembolleri Şekil-3.29'da gösterilmiştir. Bu geçit sadece iki girişli olarak üretilir, birden fazla girişli elde etmek için diğer geçitler kullanılarak yapılabilir.



Şekil-3.29 EXOR geçidinin sembolleri.

İngilizce ve piyasada bilinen ismi EXOR'dur. Türkçede ÖZEL VEYA geçidi olarak adlandırılmaktadır. Çıkışının YÜKSEK olabilmesi için girişlerinin farklı seviyelere sahip olması gerekir. Buradan yola çıkarak bazı kitaplarda ZIT geçidi olarak da anılmaktadır.

ÖZEL VEYA geçidinin girişlerine gelebilecek olası seviyeler ve bunların sonucu çıkışında oluşabilecek seviyeler Şekil-3.30'da gösterilmiştir. Geçidin doğruluk tablosu Tablo-3.7'dedir.



Şekil-3.30 ÖZEL VEYA (EXOR) geçidinin girişine gelen seviyelere göre oluşan çıkış seviyeleri.

GİRİŞ		ÇIKIŞ
A	B	X
0	0	0
0	1	1
1	0	1
1	1	0

Tablo-3.7 ÖZEL VEYA geçidinin doğruluk tablosu

ÖZEL VEYA DEĞİL GEÇİDİ (EXNOR GATE)

ÖZEL VEYA geçidi gibi diğer geçitlerin kombinasyonundan elde edilebilir. Özel kullanım alanları olduğu için tek bir geçit gibi üretilirler ve standart sembolü vardır. Şekil-3.31'de kullanılan standart sembolleri ve girişine gelebilecek olası giriş seviyeleri ve buna karşılık gelen çıkış seviyeleri gösterilmiştir.



Şekil-3.31 (EXNOR) ÖZEL VEYA DEĞİL geçidinin simgeleri.

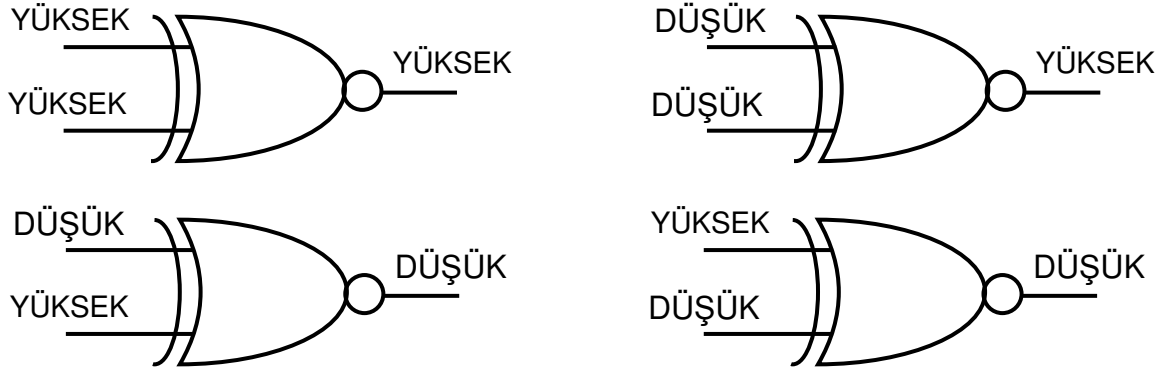
ÖZEL VEYA DEĞİL geçidi geçidinin çıkışının YÜKSEK olabilmesi için girişine gelen işaretlerin seviyelerinin aynı olması gerekir. Diğer durumlarda çıkış seviyesi DÜŞÜK olacaktır. ÖZEL VEYA DEĞİL geçidinin doğruluk tablosu Tablo-3.8'de gösterilmiştir. Girişleri aynı olduğunda YÜKSEK çıkış verdiği için EŞ geçit olarak ta adlandırılır.

GİRİŞ		ÇIKIŞ
A	B	X
0	0	1
0	1	0
1	0	0
1	1	1

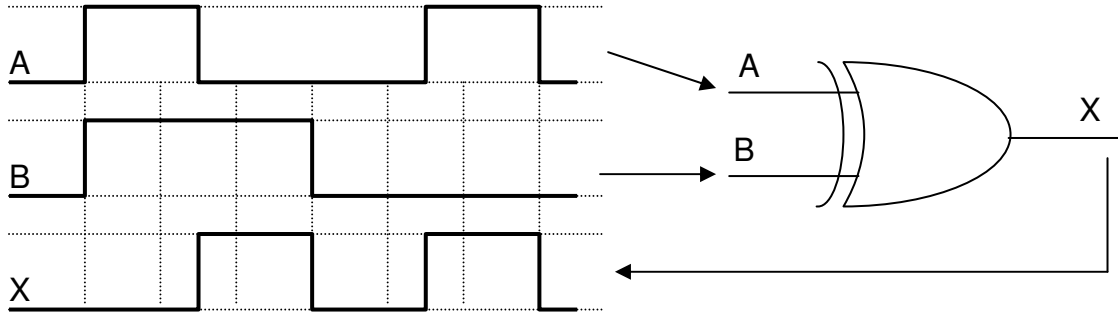
Tablo-3.8 ÖZEL VEYA DEĞİL geçidinin doğruluk tablosu

Şekil-3.33'te EXOR geçidinin girişlerine uygulanan işaretler ve bu işaretlere göre geçit çıkışı çizilmiştir. Çıkış çizilirken yine diğer geçitlerde olduğu gibi önce zaman çizgileri çizilir ve daha sonra doğruluk tablosu kullanılarak çıkış seviyeleri her bir zaman aralığı için çizilir.

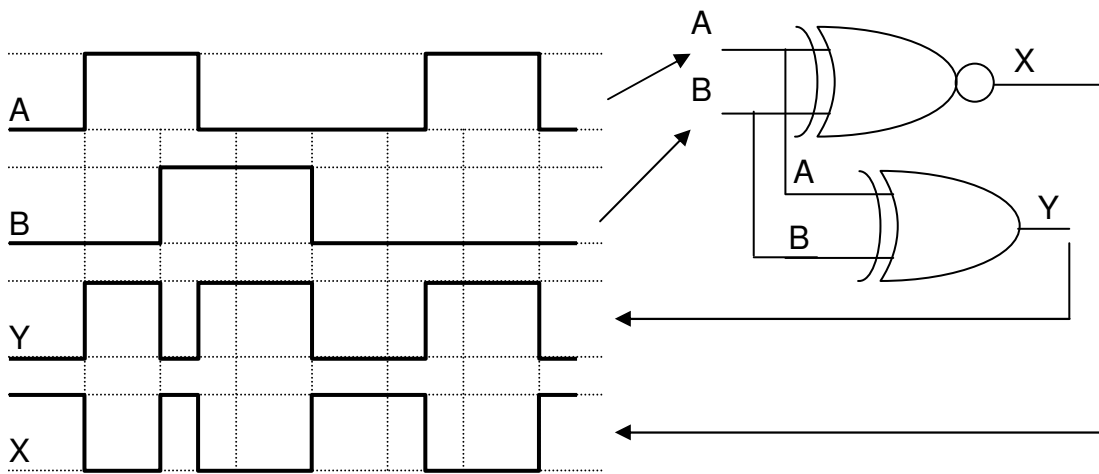
Şekil-3.34'te ise XOR ve XNOR geçitlerinin girişlerine aynı işaretler uygulanmış ve her ikisi için çıkış işaretleri çizilmiştir. Çıkış işaretlerinin birbirinin tersi olduğuna dikkat edin. XOR geçidinin diğer bir kullanım alanı ise iki ikili bitin toplanmasıdır. Şekil-3.35'te iki adet ikilinin toplanmasına örnek gösterilmiştir. Oluşan elde başka bir devre tarafından belirlenmelidir. XOR geçidi sadece toplama işlemini yerine getirecektir.



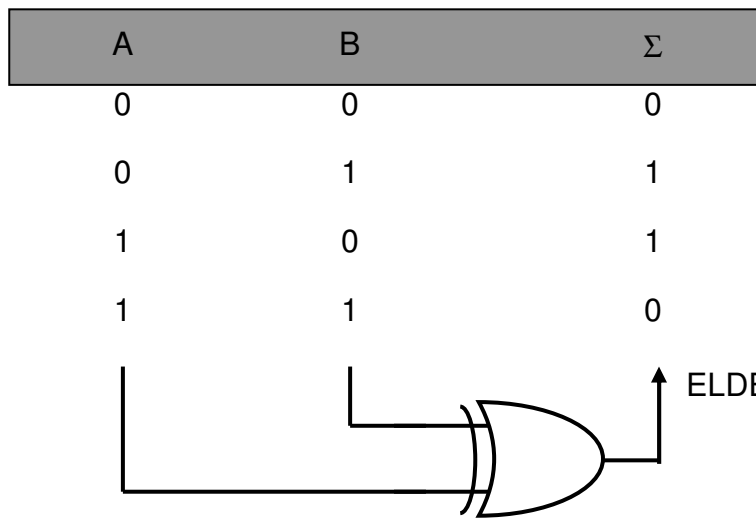
Şekil-3.32 EXNOR geçidinin girişine gelen seviyelere göre oluşan çıkış seviyeleri.



Şekil-3.33



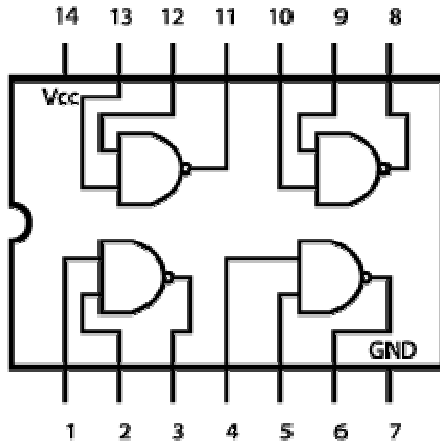
Şekil-3.34



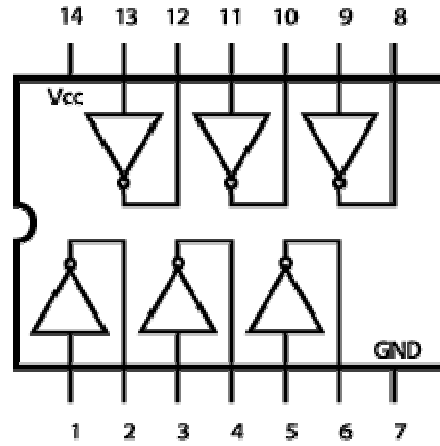
Şekil-3.35 EXOR geçidi ikilik eldesiz toplayıcı olarak kullanılabilir.

TÜMDEVRE LOJİK AİLELERİ

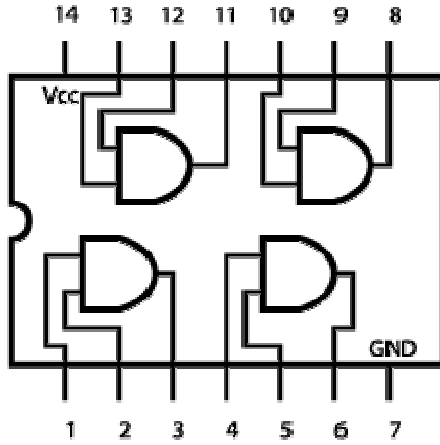
Geçitler tümdevre olarak üretilirler ve tüm devre içerisine birden fazla geçit yerleştirilebilir. Üretim teknolojisi geçitlerin işlevlerini değiştirmez fakat güç tüketimi, giriş ile çıkış arasındaki zaman gecikmesi gibi özellikleri değiştirir. Yaygın olarak kullanılan iki tür üretim teknolojisi vardır. Bunlardan birincisi TTL (bipolar transistors) ailesi, diğeri ise CMOS ailesidir. TTL ailesi 0-5 Volt ile beslenir, hatta son yıllarda 3 Voltluk olan tümdevreler üretilmektedir. CMOS ailesi ise 6 V ile 18 V arası değerlerdeki besleme gerilimi ile beslenebilir.



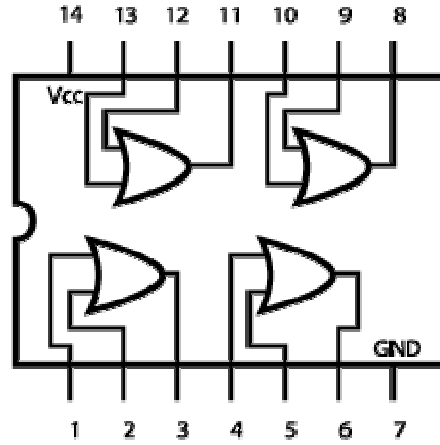
7400



7404



7408

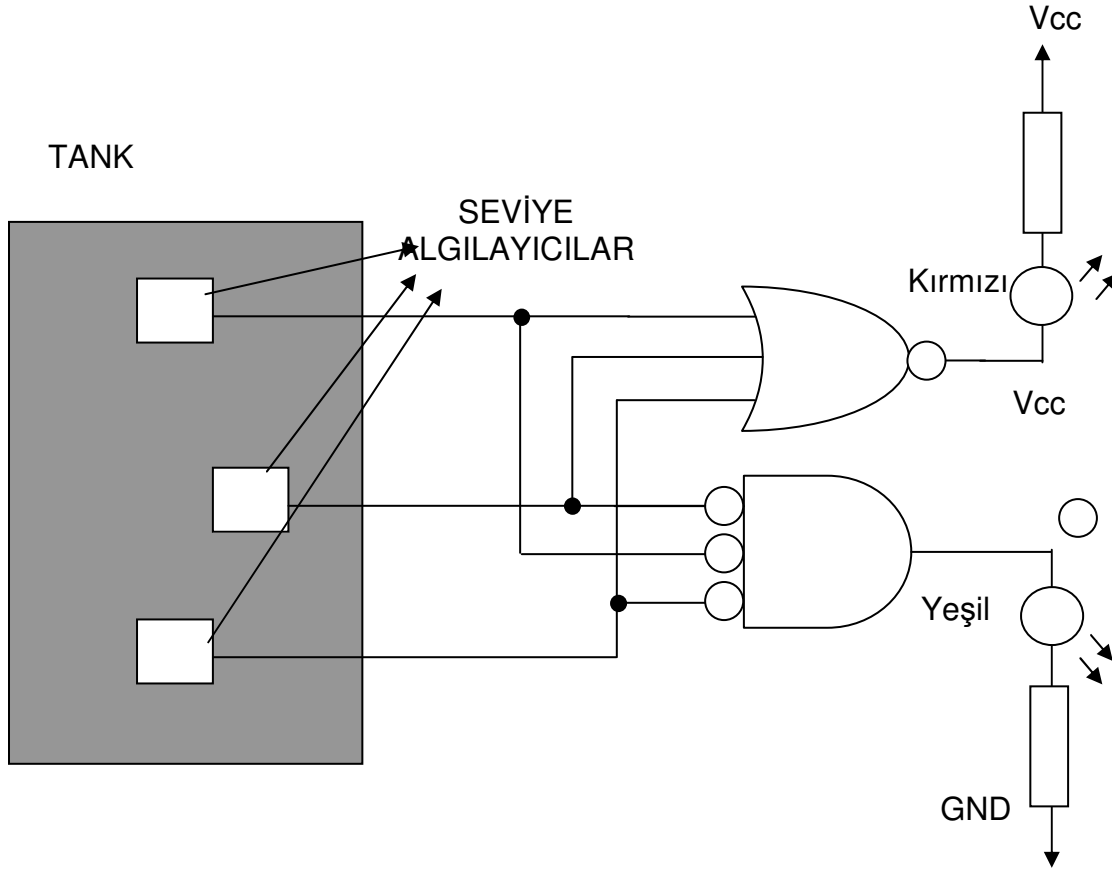


7432

Şekil-3.36 En çok kullanılan mantık tümdevreleri.

PROBLEMLER

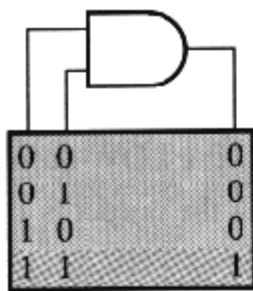
1. Şekil-3.36'daki devrede kırmızı LED'in yanması için gerekli giriş birleşimini belirleyin.



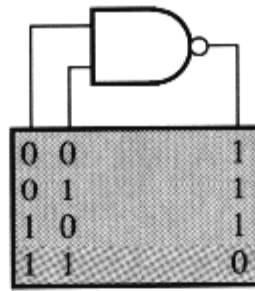
Şekil-3.36 VEYAD geçidi uygulaması.

2. Şekil-3.36'daki devrede yeşil LED'in yanması için gerekli giriş birleşimini belirleyin.

ÖZET

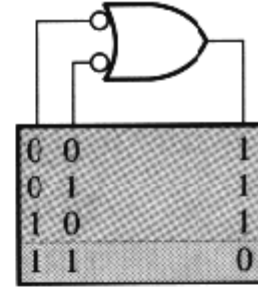


AND

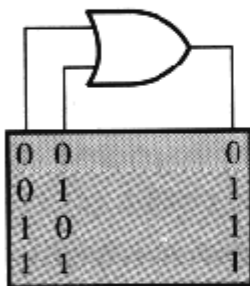


NAND

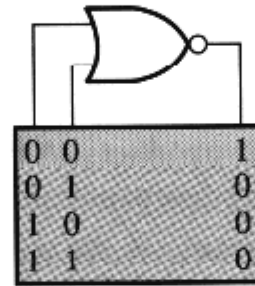
=



Negative-OR

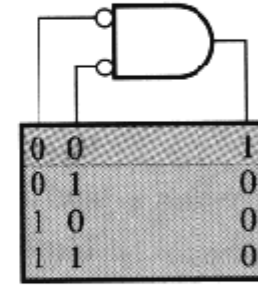


OR

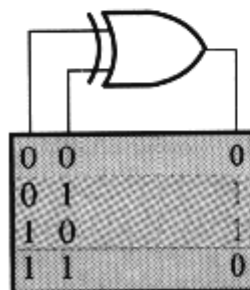


NOR

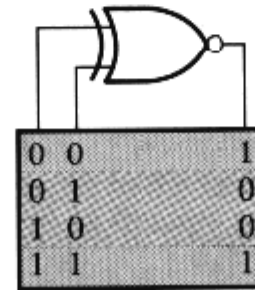
=



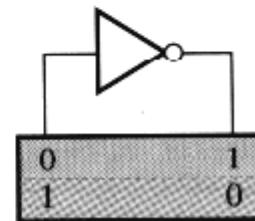
Negative-AND



Exclusive-OR



Exclusive-NOR



Inverter

BÖLÜM 4

BOOLEAN KANUNLARI VE DEMORGAN TEOREMİ

GİRİŞ

Boolean aritmetiği ikilik sayı sisteminin matematiğidir. Matematikte gördüğünüz işlemlerin tamamı boolean aritmetiğinde de vardır. Fakat değişkenlerin alabileceği sadece iki değer vardır. Değişken olarak sadece "1" ve "0" yada doğru ve yanlış vardır. Boolean aritmetiğinin 12 adet kanunu vardır, genellikle harfler kullanılarak bu kanunlar ifade edilir.

Boolean aritmetiğinde kullanılan sembollerin bazıları matematikteki ile aynı olmasına rağmen farklı anlam taşıyabilir. Harfin üzeri çizili ise bunun anlamı DEĞİL demektir, bazı kaynaklarda üst çizgi yerine ' kesme işareti kullanılır (). + işareti VEYA işlemini, • işareti VE işlemini gösterir. Nokta işareti genellikle kullanılmaz, $A \bullet B$ yerine AB kullanılır.

BOOLEAN TOPLAMA

Üçüncü konuda gördüğünüz gibi boolean toplama VEYA işlemine eşittir. Basit olarak toplamanın kuralını şöyle özetleyebiliriz.

$$0+0=0$$

$$0+1=1$$

$$1+0=1$$

$$1+1=1$$

Boolean aritmetiğinde toplama toplama dahil olan literallerin toplamıdır. Sadece VEYA işlemidir.

$$A+B$$

$$A+B'$$

$$A+B+C'$$

$$A'+B+C+D'$$

Toplam giriş literallerinden en az biri 1 olduğunda 1, aksi halde 0'dır.

BOOLEAN ÇARPMA

Daha önce tanımladığımız gibi boolean çarpma VE işlemine eşdeğerdir. Kurallarını şöyle özetleyebiliriz.

$$0.0=0 \quad 0.1=0 \quad 1.0=0 \quad 1.1=1$$

Boolean aritmetiğinde çarpım giriş literallerinin çarpımıdır.

$$A.B \quad A.B' \quad A.B.C \quad AB'CD'$$

Çarpım literallerden biri sıfır olduğunda 0, tümü 1 olduğunda 1'dir.

BOOLEAN ARİTMETİĞİNİN KANUNLARI VE KURALLARI

Matematikte işlemler için çok iyi kurallar ve kanunlar geliştirilmiştir. Bu kural ve kanunlar boolean aritmetiğine uygulanabilir.

BOOLEAN ARİTMETİĞİNİN KANUNLARI

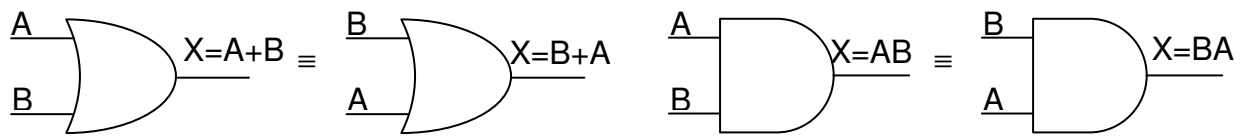
Boolean aritmetiğinin değişme, birleşme ve dağılma özellikleri vardır.

Değişme özelliği

VEYA işleminde veya VE işleminde girişlerin sırası önemli değildir.

$$A+B=B+A \quad \text{veya} \quad AB=BA$$

Şekil-4.1'de geçitler üzerinde değişme özelliği gösterilmiştir.



Şekil-4.1 Değişme özelliği.

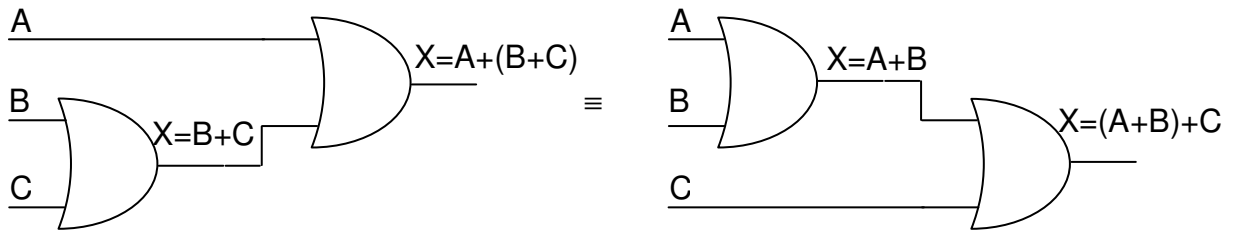
Birleşme özelliği

$$A+(B+C)=(A+B)+C$$

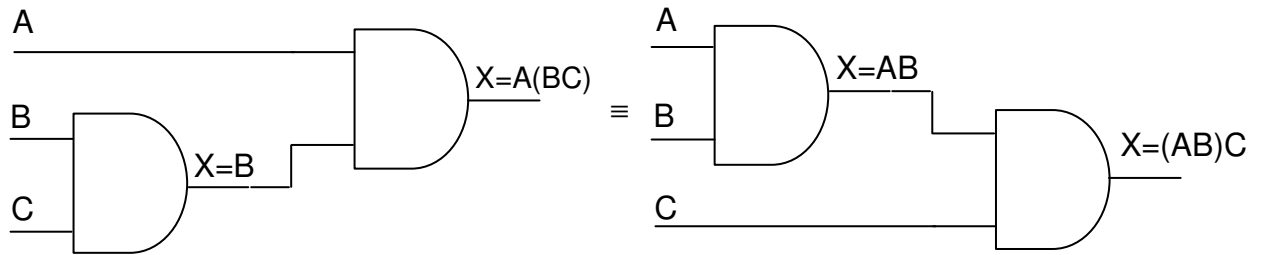
Bu özellik ikiden fazla değişkenin VEYAlanmasının sırasının önemli olmadığını gösterir. Aynı özellik VElemede de geçerlidir.

$$A(BC)=(AB)C$$

Şekil-4.2'de toplamanın birleşme özelliği geçitlere uygulanmıştır. Şekil-4.3'de ise çarpmanın birleşme özelliği geçitlere uygulanmıştır.



Şekil-4.2 Toplamada Birleşme özelliği.



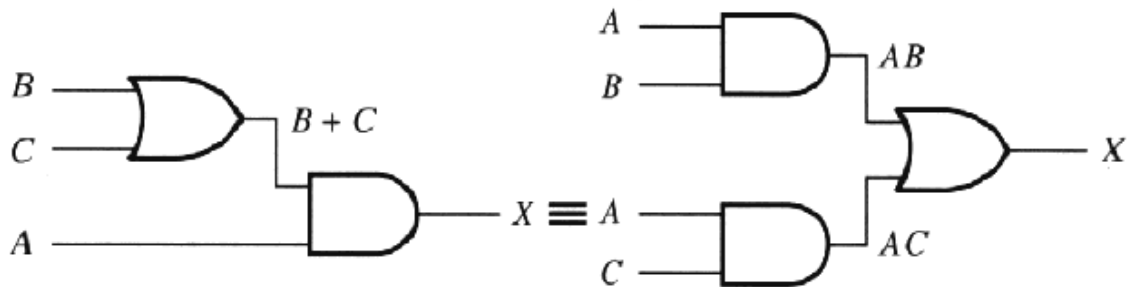
Şekil-4.3 Çarpmada birleşme özelliği

Dağılma özelliği

Üç değişkenli yazılmış hali aşağıdadır.

$$A(B+C)=AB+AC$$

Bu kanun birçok değişkeni VEYAladıktan sonra tek bir değişkenle VElemek ile bir değişkeni tüm diğer değişkenlerle ayrı ayrı VEledikten sonra VEYAlamak eşdeğer işlemlerdir. Şekil-4.4'te dağılma özelliğinin geçitlere uygulanması gösterilmiştir.



Şekil-4.4 Dağılma özelliğinin uygulaması.

BOOLEAN KURALLARI

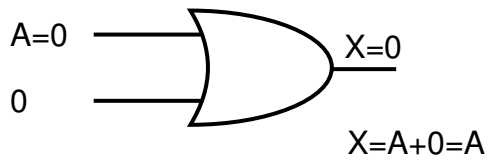
Boolean kuralları boolean eşitliklerini sadeleştirmek için kullanılır. 12 adet olan kuralların özeti Tablo-4.1'de gösterilmiştir.

1.	$A+0=A$
2.	$A+1=1$
3.	$A\bullet 0=0$
4.	$A\bullet 1=A$
5.	$A+A=A$
6.	$A\neq A=1$
7.	$A\bullet A=A$
8.	$A\bullet \bar{A}=0$
9.	$\bar{\bar{A}}=A$
10.	$A+AB=A$
11.	$A+\bar{A}B=A+B$
12.	$(A+B)(A+C)=A+BC$

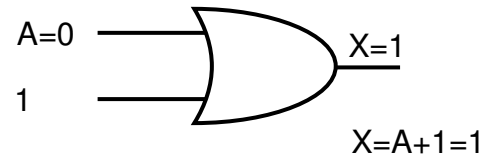
Tablo-4.1 Boolean kuralları.

Şekil-4.5'de 1-8 numaralı boolean kurallarının geçitlere uygulanması gösterilmiştir.

KURAL 1 $A+0=A$

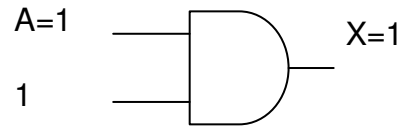
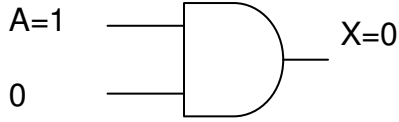
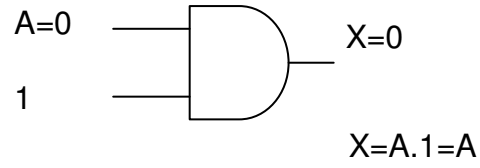
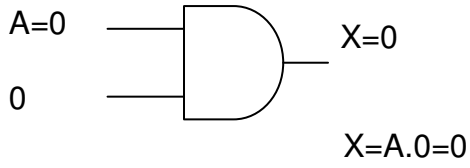
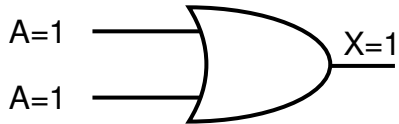
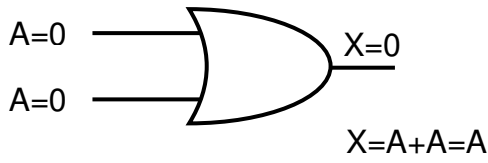
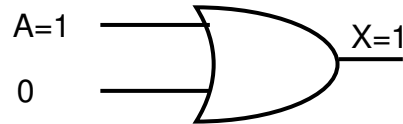
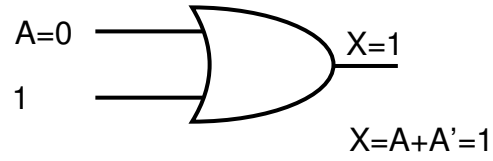
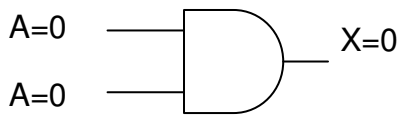
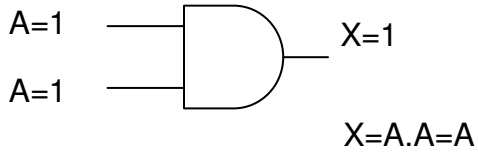
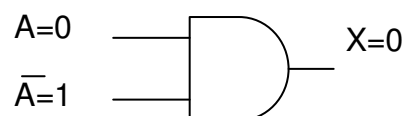
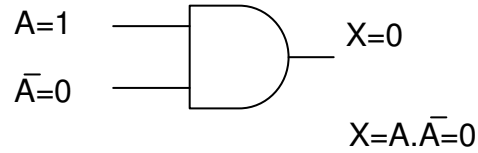


KURAL 2 $A+1=1$



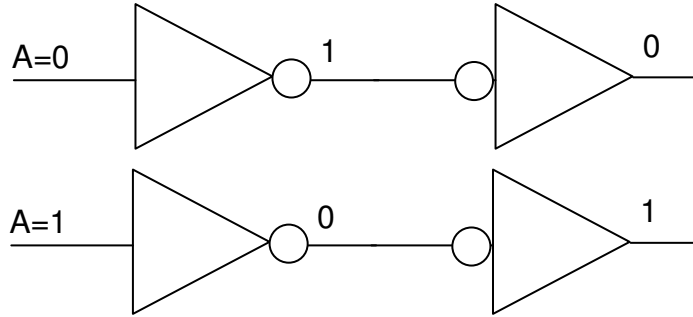
KURAL 3 $A\bullet 0=0$

KURAL 4 $A\bullet 1=A$

**KURAL 5** $A+A=A$ **KURAL 6** $A+A'=1$ **KURAL 7** $A.A=A$ **KURAL 8** $A.A'=0$ 

Şekil-4.5 1-8 nolu kuralların geçitlere uygulanması.

Şekil-4.6'da 9. kuralın geçide uygulanması gösterilmiştir. Bu kurala göre değişkenin deęilinin deęili kendisidir. Bir deęişken iki kez DEĞİL geçidi geçidinden geçirildiğinde kendisi elde edilir. Bu işlem size çok gereksiz gibi gelebilir fakat yaygın şekilde kullanılır. Özellikle dijital işareti güçlendirmek için kullanılır. Dijital işaret gerilim olarak yükseltilmez, sadece akım olarak yükseltilir.



Şekil-4.6 KURAL 9'un geçitler ile gösterimi

KURAL 10: $A + AB = A$ Bu kural dağılma özelliği kullanılarak açıklanabilir.

$$A + AB = A(1 + B) \quad \text{Dağılma özelliği.}$$

$$= A \cdot 1 \quad \text{Kural 2}$$

$$= A \quad \text{Kural 4}$$

Tablo ile doğrulayalım.

A	B	AB	A+AB
0	0	0	0
0	1	0	0
1	0	0	1
1	1	1	1

Eşit

KURAL 11:

Diğer kuralları kullanarak doğruluğunu gösterelim;

$$A + \overline{AB} = A + B$$

$$A + \overline{AB} = (A + AB) + \overline{AB} \quad \text{Kural 10}$$

$$= (AA + AB) + \overline{AB} \quad \text{Kural 7,}$$

$$= AA + AB + A\overline{A} + \overline{AB} \quad \text{Kural 8}$$

$$= (A + \overline{A})(A + B) \quad \text{Paranteze alın}$$

$$= 1 \cdot (A + B) \quad \text{Kural 6}$$

$$= A + B \quad \text{Kural 4}$$

DE MORGAN TEOREMİ

Şöyle açıklanabilir;

$$\overline{XY} = \overline{X} + \overline{Y} \quad (1)$$

$$\overline{X + Y} = \overline{X} \cdot \overline{Y} \quad (2)$$

1 nolu eşitliği özetleyecek olursak; çarpımın değil değışkenlerin deęillerinin toplamına eşittir. 2 nolu eşitliği özetlersek; toplamın değil değışkenlerin deęillerinin çarpımına eşittir.

De Morgan Teoremi ikiden fazla değışkenler için de geçerlidir.

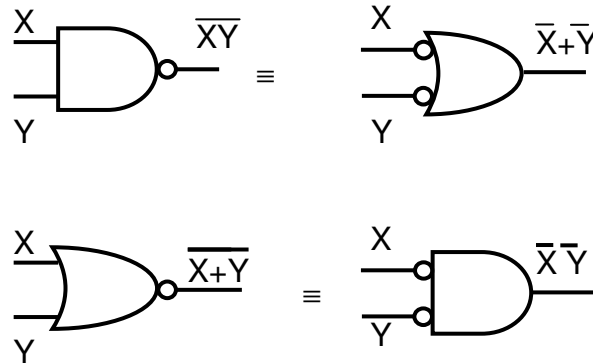
$$\overline{XYZ} = \overline{X} + \overline{Y} + \overline{Z}$$

$$\overline{WXYZ} = \overline{W} + \overline{X} + \overline{Y} + \overline{Z}$$

$$\overline{X + Y + Z} = \overline{X} \cdot \overline{Y} \cdot \overline{Z}$$

$$\overline{\overline{W + Y + X + Z}} = \overline{W} \cdot \overline{Y} \cdot \overline{X} \cdot \overline{Z}$$

De Morgan teoremlerinin geçitlere uygulanışı şekil-4.7'de gösterilmiştir.



Şekil-4.7 De Morgan teoremlerinin geçitler üzerinde uygulanması.

ÇARPIMLARIN TOPLAMI

Boolean aritmetikinde çarpımı VE işleminin yaptığını daha önce belirtmiştik. İki veya daha fazla değışkenin çarpımı VE geçidi ile gerçekleştirilebilir. İki değışkenin çarpımı AB, üç değışkenin çarpımı ABC ve dört değışkenin çarpımı ABCD şeklinde açıklanabilir. Toplama işlemi ise VEYA geçitleri ile gerçekleştirilir. Çarpımların toplamı şeklindeki eşitlikler ise önce VE geçidi kullanılarak çarpma yapılır ve sonra birden fazla çarpım VEYA geçidi kullanılarak toplanır. Örnek olarak aşağıdaki eşitlikleri verebiliriz;

$$AB + BCD$$

$$ABC + DEC$$

$$\overline{ABC} + \overline{DEFG} + AEG$$

$$ABC + \overline{ABC} + ABC$$

ÇT ifadelerde değişkenlerin üzerindeki değil çizgileri birleşik çizilemez. Örneğin aşağıdaki ifadeler birbirine eşit değildir.

$$\overline{A} \overline{B} \overline{C} \neq \overline{ABC}$$

Tüm eşitlikler ÇT olarak ifade edilebilir. Boolean kanunları kullanılarak ÇT olmayan eşitlikler ÇT haline dönüştürülebilirler.

ÖRNEK:

$$A(AB + CD) = AB + ACD$$

TOPLAMLARIN ÇARPIMI (TÇ)

Toplamların çarpımı şeklindeki ifadeler birden fazla VEYA işleminin sonucunu VE işlemine tabi tutulmuş halidir.

Örnek:

$$(A + B)(B + C + D)$$

$$(A + B + C)(D + E + F)$$

$$(A + \overline{B} + C)(D + C + \overline{F} + G)(A + \overline{F} + \overline{G})$$

TÇ ifadelerin bazılarında çarpımlardan biri veya birkaçı tek değişken olabilir.

$$A(B + C + D)(E + F + G)$$

Boolean Kanunlarının Kullanılarak Mantık Fonksiyonlarının Basitleştirilmesi:

Boolean kanunları genellikle karmaşık mantık fonksiyonlarının basitleştirilerek daha az sayıda geçitle veya daha az girişli geçit kullanarak gerçeklemek için kullanılır. Bu işlem için boolean kanunlarının iyi bilinmesi gerekir.

ÖRNEK:

$AB + A(B + C) + B(B + C)$ fonksiyonunu boolean kanunlarını kullanarak en basit hale indirgeyin.

ÇÖZÜM:

Adım 1. Dağılıma kanunu 2. ve 3. terimlere uygulanırsa;

$$AB + AB + AC + BB + BC$$

Adım 2. 7 nolu kanun uygulanırsa ($BB=B$)

$$AB + AB + AC + B + BC$$

Adım 3. 3 nolu kanun uygulanırsa ($AB+AB=AB$)

$$AB + AC + B + BC$$

Adım 4. B çarpan parantezine alınır.

$$AB + AC + B(1 + C)$$

Adım 5. 2 nolu kanun uygulanırsa.

$$AB + AC + B \cdot 1$$

Adım 6. 4 nolu kanun uygulanırsa.

$$AB + AC + B$$

Adım 7. Birinci ve üçüncü terim B ortak parantezine alınır.

$$AC + B(A + 1)$$

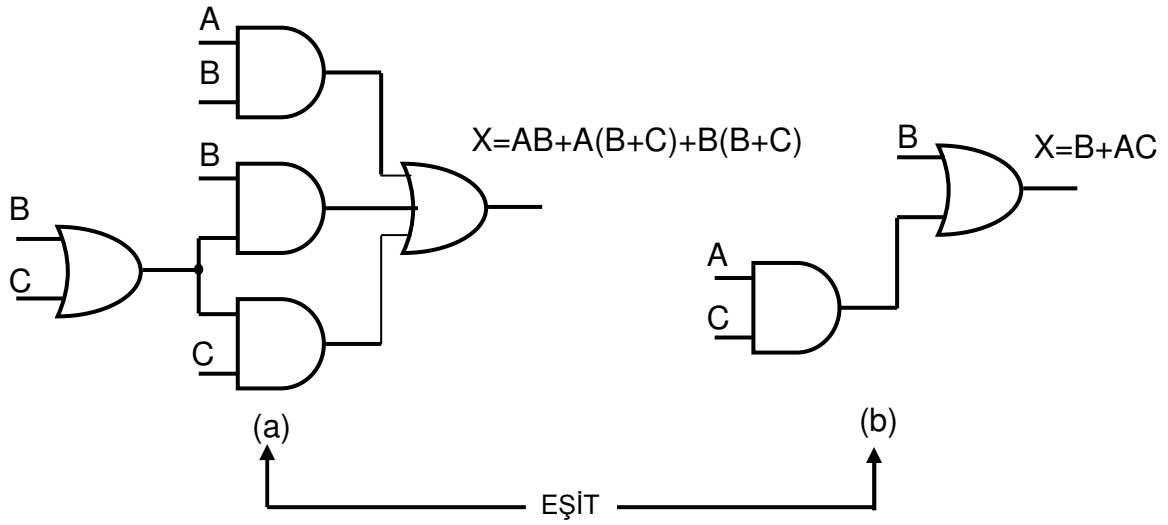
Adım 8. 2 nolu kanun uygulanırsa.

$$AC + B \cdot 1$$

Adım 9. 4 nolu kanun uygulanırsa en sade hali elde edilir.

$$AC + B$$

Karmaşık ve sade hallerinin devrelerini birlikte çizersek aradaki fark daha iyi anlaşılabilir. Şekil-4.8'de sade ve karmaşık halleri ayrı ayrı iki devre olarak çizilmiştir. Karmaşık olanda 5 geçit sade olanda ise 2 geçit kullanılmıştır.



Şekil-4.8 Sade ve karmaşık haliyle fonksiyonun devresi.

BİRLEŞİK MANTIK DEVRELERİNİN TASARIMI

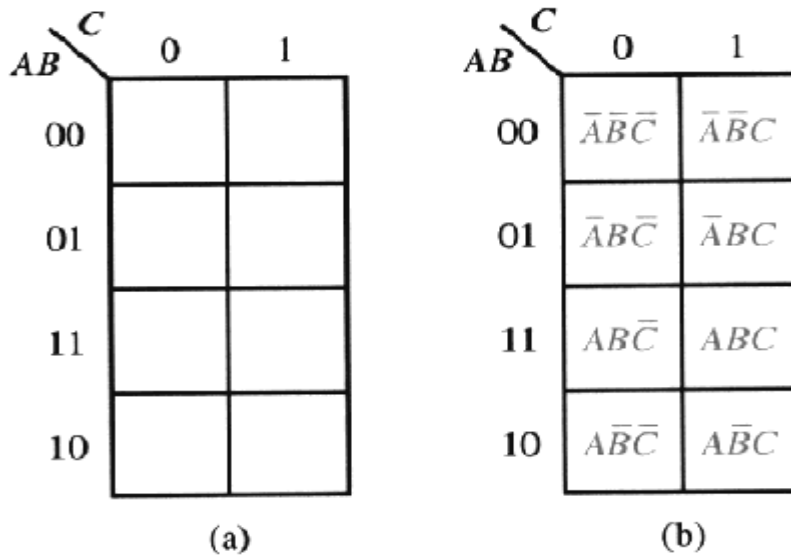
Birleşik mantık devresinin çıkışı giriş değişkenleri tarafından belirlenir. Bu tür devrelerin tasarımında öncelikle doğruluk tablosu elde edilir. Doğruluk tablosundan elde edilen ÇT ifade Karnaugh haritasına yerleştirilerek en sade ifade yazılır. Sadeleştirmenin amacı devreyi daha az sayıda geçit kullanarak gerçeklemektir. Karnaugh haritası 4 ve daha az değişken olduğunda kullanışlıdır. Daha fazla değişkene sahip olduğunda Boolean kanunları kullanılarak ifadeler sadeleştirilebilir.

3-DEĞİŞKENLİ KARNAUGH HARİTASININ KULLANIMI

3-değişkenli Karnaugh haritası Şekil-4.9'da gösterilmiştir. (a)'da değişkenler sıralanmış ve ikilik karşılıkları yazılmıştır. Değişkenlere isim olarak A, B, C kullanılmıştır ki bu isimler isteğe göre değiştirilebilir. (b)'de ise her hücrenin adresi yazılmıştır. Bir devrenin doğruluk tablosu elde edildikten sonra kolay bir şekilde devrenin değerleri bu adreslerden yola çıkarak Karnaugh haritasına yerleştirilebilir.

4-DEĞİŞKENLİ KARNAUGH HARİTASI

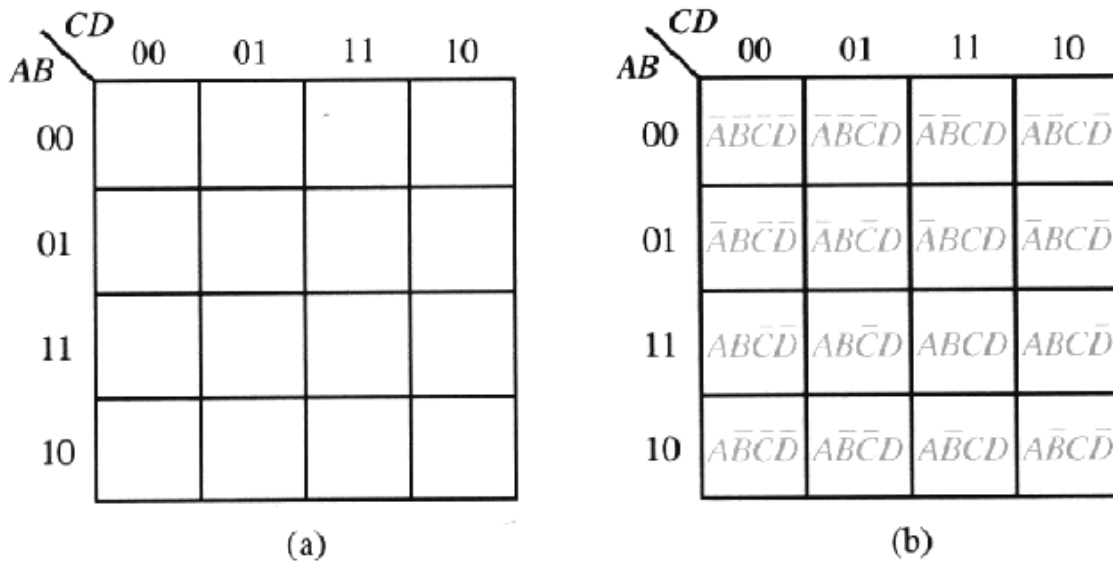
4-değişkenli Karnaugh haritası Şekil-4.10'da gösterilmiştir. (a)'da değişkenler sıralanmış ve ikilik karşılıkları yazılmıştır. Değişkenlere isim olarak A, B, C, D kullanılmıştır bu isimler isteğe göre değiştirilebilir. (b)'de ise her hücrenin adresi yazılmıştır. Bir devrenin doğruluk tablosu elde edildikten sonra kolay bir şekilde devrenin değerleri bu adreslerden yola çıkarak Karnaugh haritasına yerleştirilebilir.



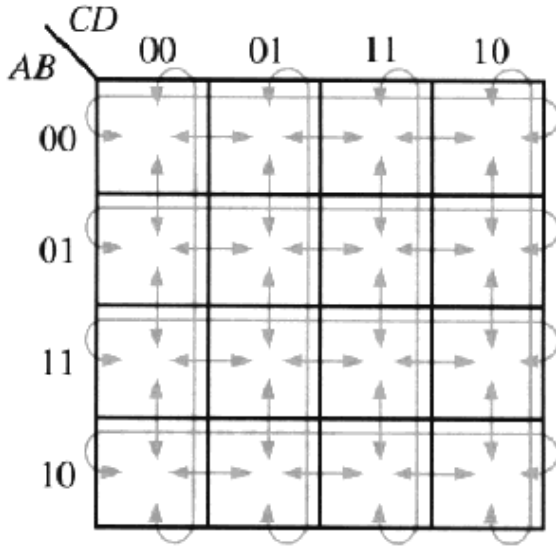
Şekil-4.9 3-değişkenli Karnaugh haritası ve adresleri.

Hücre komşulukları

Karnaugh haritasında hücrelerin komşu olabilmeleri için sadece bir değişkeni farklı olmalıdır. Yerleşim planı tablodaki sıra ile DEĞİL geçidi bu kurala uyacak şekilde düzenlenmiştir. Fiziki olarak her hücre diğerine komşu olarak görünse bile çapraz yerleşmiş hücreler kurala uymadıkları için komşu sayılmazlar. Şekil-4.11'de 4-değişkenli Karnaugh haritasında komşuluklar gösterilmiştir.



Şekil-4.10 4-değişkenli Karnaugh haritası ve hücrelerin adresleri.



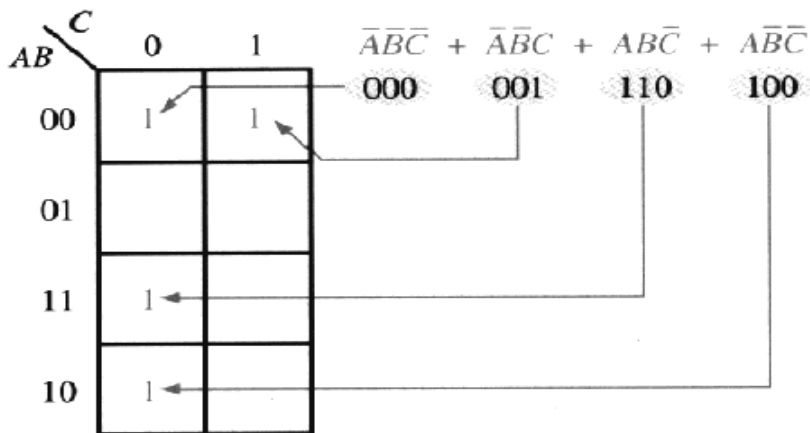
Şekil-4.11 Hücre komşulukları.

Standart Çarpımların Toplamı İfadelerinin Yerleştirilmesi

Standart formdaki ÇT ifadelerde her bir terim bir hücreye yerleştirilir. İfadeye yer alan terimlerin yerine 1, yer almayanların bulunması gereken adrese 0 yerleştirilir. Yerleştirme bittikten sonra haritadaki 1 sayısı ile standart formdaki ÇT ifadedeki çarpım terimlerin sayısı eşit olmalıdır. Şekil-4.12'de buna 3-değişkenli haritada örnek verilmiştir.

ÖRNEK 1: $A'B'C + A'B'C' + ABC' + AB'C'$ ifadesini haritaya yerleştirin.

ÇÖZÜM: Öncelikle ikilik karşılıklarını yazalım.



Şekil-4.12 Standart formdaki bir ÇT ifadenin karnaugh haritasına yerleştirilmesi.

ÖRNEK 2: $A'B'C + A'BC' + ABC + ABC'$ ifadesini haritaya yerleştirin.

ÇÖZÜM: Öncelikle ikilik karşılıklarını yazalım.

$$A'B'C + A'BC' + ABC + ABC'$$

$$001 \quad 010 \quad 111 \quad 110$$

		<i>C</i>		
		0	1	
<i>AB</i>	00		1 ← $\bar{A}\bar{B}C$	
	01	1 ← $\bar{A}B\bar{C}$		
	11	1 ← ABC	1 ← ABC	
	10			$AB\bar{C}$

ÖRNEK 3: $\bar{A}\bar{B}\bar{C}\bar{D} + A\bar{B}\bar{C}\bar{D} + \bar{A}\bar{B}C\bar{D} + A\bar{B}C\bar{D} + \bar{A}B\bar{C}D + ABCD + A\bar{B}C\bar{D}$

İfadesini Karnaugh haritasına yerleştirin.

ÇÖZÜM:

		<i>CD</i>				
		00	01	11	10	
<i>AB</i>	00		1 ← $\bar{A}\bar{B}\bar{C}\bar{D}$	1 ← $\bar{A}\bar{B}CD$		
	01	1 ← $\bar{A}B\bar{C}\bar{D}$				
	11	1 ← $AB\bar{C}\bar{D}$	1 ← $AB\bar{C}D$	1 ← $ABCD$		
	10				1 ← $A\bar{B}C\bar{D}$	

Standart olmayan ÇT ifadelerinin haritaya yerleştirilmesi.

Standart formda olmayan ÇT ifadeleri öncelikle standart hale getirilir. Boolean kuralları uygulanarak ifadeler genişletilir. Standart olmayan terimde bulunması gereken değişken bu terime eklenir. Ekleme yapılırken standart olmayan terim kayıp terimin kendisi ile değilinin toplamı ile çarpılır. Böylece artık iki terim elde edilir. Bir anlamda sadeleştirmenin tersi yapılır. $ABC + A'C$ ifadesinde standart olmayan terim $A'C$ terimidir, eksik olan değişken ise B değişkenidir. B 'nin kendisi ve B 'nin değili ile bu terim ayrı ayrı çarpılır. $ABC + A'CB + A'CB'$ ifadesi elde edilir. Bu genişletme pratik şekilde harita üzerinde de yapılabilir.

ÖRNEK 1: $A' + AB' + ABC'$ ifadesini haritaya yerleştirin.

ÇÖZÜM: İfade standart olmadığı için standart hale getirelim.

$A' + AB' + ABC'$
 000 100 110
 001 101
 010
 011

	C	
AB	1	1
	1	1
	1	
	1	1

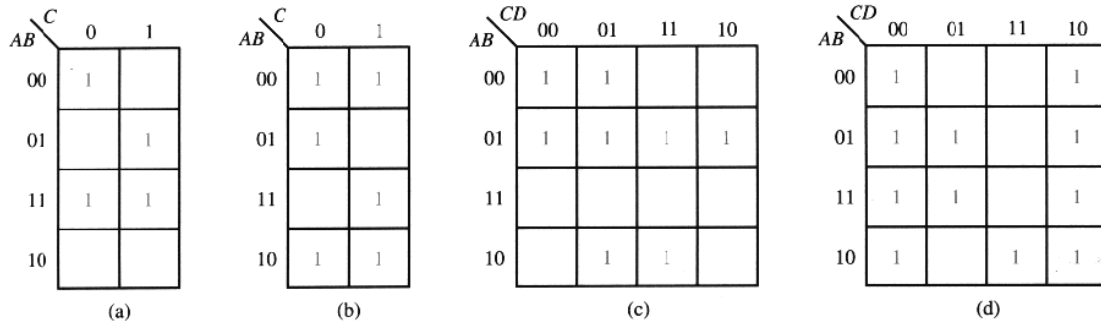
ÇT İFADELERİNİN KARNAUGH HARİTASI İLE SADELEŞTİRİLMESİ

Amaç ÇT ifadeleri en az terime ve en az değişkenli hale getirmektir. Bir önceki kısımda ÇT ifade haritaya yerleştirildi, bu kısımda haritada terimler birleştirilerek tekrar ÇT ifade haline getirilecek. Birleştirmenin yapılabilmesi için öncelikle komşu 1'ler belirlenmeli, komşu 1'ler ikişer dörder veya sekizerli gruplar halinde birleştirilmelidir. Gruplar belirlenirken aşağıdaki kurallara uyulmalıdır.

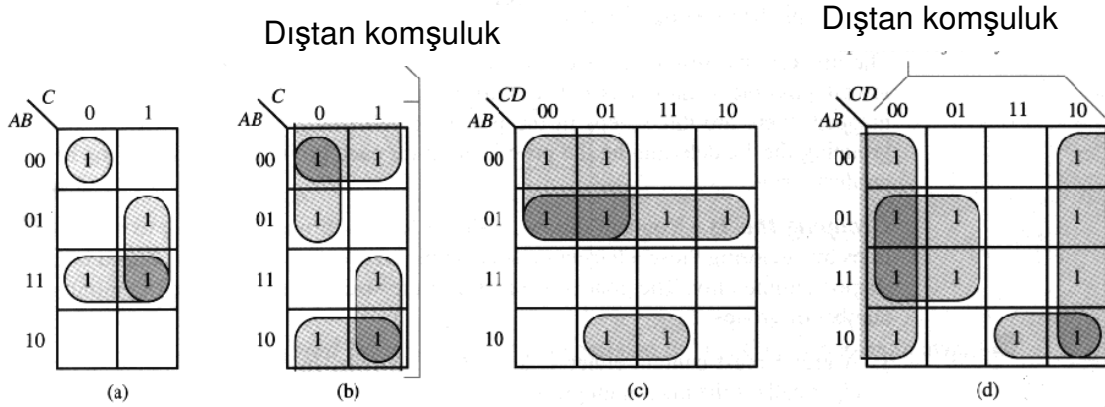
- Grup 1,2,4,8,16 hücrelerden (içinde 1 olan hücre) oluşabilir.
- Gruptaki her hücre bir veya daha fazla hücre ile komşu olmalı. Gruptaki tüm 1'lerin birbirlerine komşu olması beklenmez.
- Koşul 1'deki kurala uymak koşulu ile grup maksimum sayıda 1'den oluşmalıdır.
- Karnaugh haritasındaki tüm birler bir grup tarafından içermelidir. İçerilmiyor ise tek başına grup yapılmalıdır.

ÖRNEK:

Bu kurallara uyarak Şekil-4.13'de verilen Karnaugh haritalarındaki grupları belirleyin. Gruplar belirlendikten sonra en az sayıda değişken ile yazılması gerekir. Gruplar doğru belirlendiğinde en az sayıda terim elde edilir. Çünkü her grup bir terimi temsil eder. Grubun genişliğini ise terimdeki değişken sayısını belirler. 4-değişkenli Karnaugh haritasında grup bir adet 1'den oluşuyor ise bu grup 4 değişkenle, 2 adet 1'den oluşuyorsa 3 değişkenle, 4 adet 1'den oluşuyorsa 2 değişkenle, 8 adet 1'den oluşuyorsa 1 değişkenle tanımlanabilir. Eğer Karnaugh haritası tamamı bir grup ise ifadenin değeri 1'dir. 3-değişkenli Karnaugh haritasında ise grup 1 adet 1'den oluşuyor ise bu grup 3 değişkenle, 2 adet 1'den oluşuyorsa 2 değişkenle, 4 adet 1'den oluşuyorsa 1 değişkenle tanımlanabilir. 8 adet 1'den oluşan grup varsa bu ÇT ifadenin değeri 1'e eşittir.



F1 Aşağıda gösterilenden farklı bir şekilde gruplar oluşturulabilir fakat ifadedeki terim sayısı veya değişken sayısında bir azalma olmaz.



Şekil-4.13

ÖRNEK: Şekil-4.14'te verilen Karnaugh haritalarının en sade ÇT ifadelerini yazın.

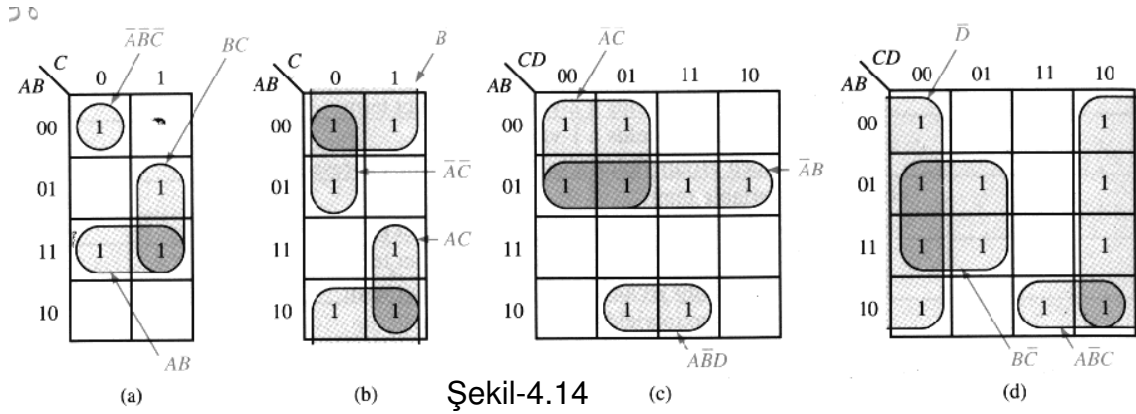
ÇÖZÜM: Grupların terimleri Karnaugh haritası üzerine yazılır ve sonra bu terimler toplanarak ÇT ifade elde edilir.

(a). $AB+BC+A' B' C'$

(b). $B'+A'C'+AC$

(c). $A' B+A' C'+AB' D$

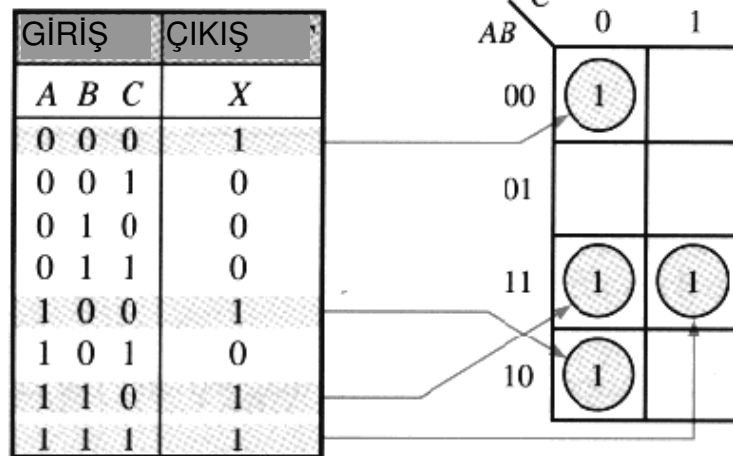
(d). $D'+AB' C+BC'$



DOĞRULUK TABLOSUNDAN KARNAUGH HARİTASINA GEÇİŞ

Doğruluk tablosu bir devrenin değişkenlerinin tüm olasılıklarını ve bu olasılıklara göre devrenin çıkışını belirler. Doğruluk tablosundan çıkışın 1 olduğu satırlar ÇT ifadenin terimlerini oluşturur. Tablodan yazılan ÇT ifade en geniş ifadedir. Tablonun her satırının Karnaugh haritasında bir karşılığı vardır. Daha önce belirlediğimiz adreslere göre tablonun değeri Karnaugh haritasına geçirilebilir. Karnaugh haritasında sadeleştirilerek sade halde yazılabilir. Tasarım yapılırken önce problem tanımlanır daha sonra değişkenler belirlenir. Değişken sayısı oranında doğruluk tablosu yapılır ve tüm giriş olasılıkları bu tabloda yer alır. Çıkışın 1 olması gereken satırlar belirlenir ve 1 yazılır. Buradan Karnaugh haritasına geçirilir ve en sade ÇT ifade elde edilir. En son adımda mantık geçitleri kullanılarak önce devrenin şekli çizilir ve devre kurularak çalıştırılır.

$$X = \bar{A}\bar{B}\bar{C} + \bar{A}B\bar{C} + A\bar{B}\bar{C} + ABC$$

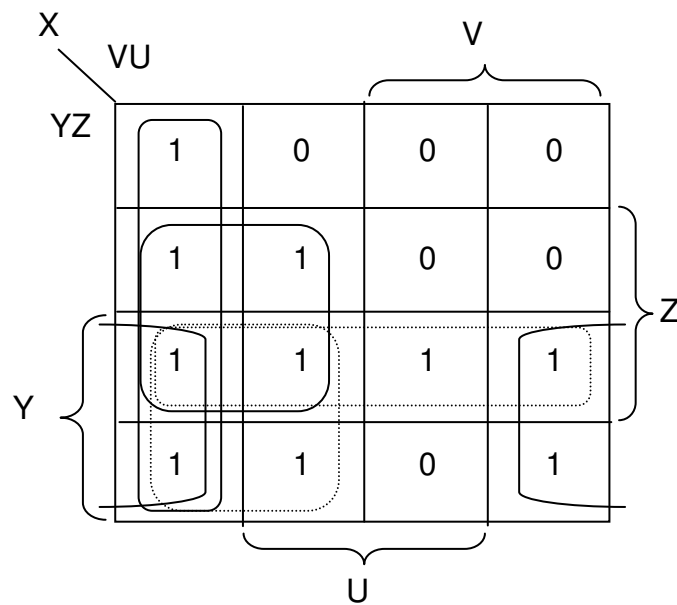


Şekil-4.15

Şekil-4.15'te doğruluk tablosundan Karnaugh haritasına geçişe örnek verilmiştir.

A		B		X
Y	Z	V	U	
0	0	0	0	1
0	0	0	1	0
0	0	1	0	0
0	0	1	1	0
0	1	0	0	1
0	1	0	1	1
0	1	1	0	0
0	1	1	1	0
1	0	0	0	1
1	0	0	1	1
1	0	1	0	1
1	0	1	1	0
1	1	0	0	1
1	1	0	1	1
1	1	1	0	1
1	1	1	1	1

Tablo-4.2



$$X = Y\bar{V} + Z\bar{V} + YZ + \bar{U} \bar{V} + \bar{U}Y$$

BÖLÜM 5

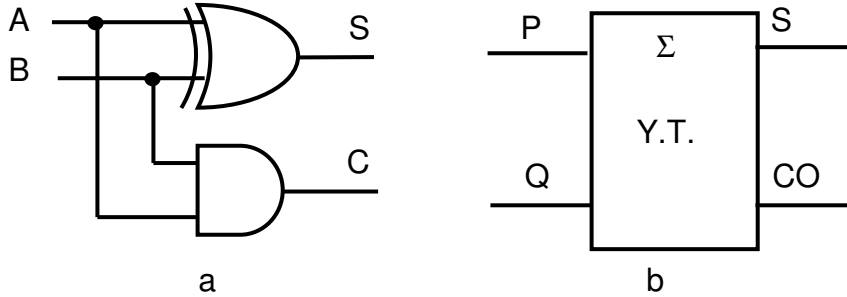
TOPLAYICI VE KARŞILAŞTIRICI

TOPLAYICILAR

Toplayıcı devreler ikilik sayıları toplar ve sonucu yine ikilik olarak çıkışa yazarlar. Elde girişi olmayan toplayıcı devreye yarım toplayıcı, olan devrelere tam toplayıcı adı verilir. Yarım toplayıcı devreyi geçitler kullanarak tasarlayalım. Devrenin yapması gereken işlevi doğruluk tablosunda oluşturalım.

A	B	TOPLAM (S)	ELDE (C)
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Tablo-5.1 Yarım toplayıcının doğruluk tablosu.



Şekil-5.1 a. Yarım toplayıcının devresi b. Mantık sembolü.

Toplam çıkışının fonksiyonunu yazacak olursak ifadesi elde edilir. Elde çıkışınıninki ise $C=AB$ 'dir. Yarım toplayıcıyı bir EXOR ve bir VE geçidi ile oluşturulabilir. Şekil-5.1'de yarım toplayıcı devresi gösterilmiştir.

Yarım toplayıcı devresi sadece iki adet bir bitlik ikilik sayının toplanmasında kullanılabilir. İki bitlik sayının toplanması için devrenin elde girişi olan bir devre tasarlanması gerekir. Elde girişi olan bir toplayıcı devresine tam toplayıcı devresi adı verilir. Tam toplayıcı devresinin doğruluk tablosu Tablo-5.2'de ve devresi Şekil-5.2'de verilmiştir. Tablodan elde çıkışı ve toplam çıkışının eşitliklerini yazalım;

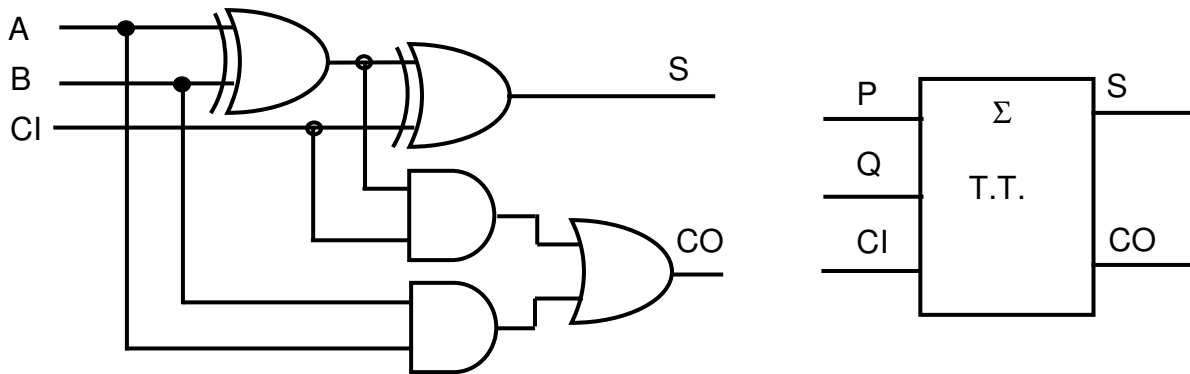
$$\text{Toplam çıkışının fonksiyonu; } S = A \oplus B \oplus CI$$

$$\text{Elde çıkışının fonksiyonu;}$$

Şekilden de anlaşılacağı gibi iki adet yarım toplayıcıdan tam toplayıcı elde edilebilir.

A	B	CI	S	CO
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Tablo-5.2 Tam toplayıcı doğruluk tablosu

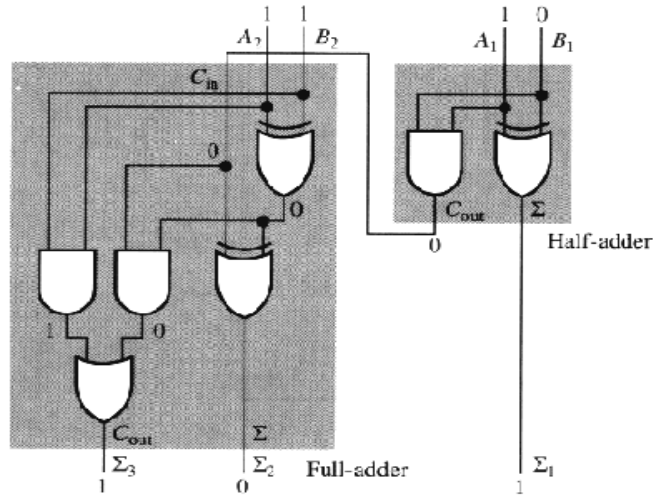


Şekil-5.2 Tam toplayıcının devresi ve mantık simgesi.

PARALEL TOPLAYICILAR

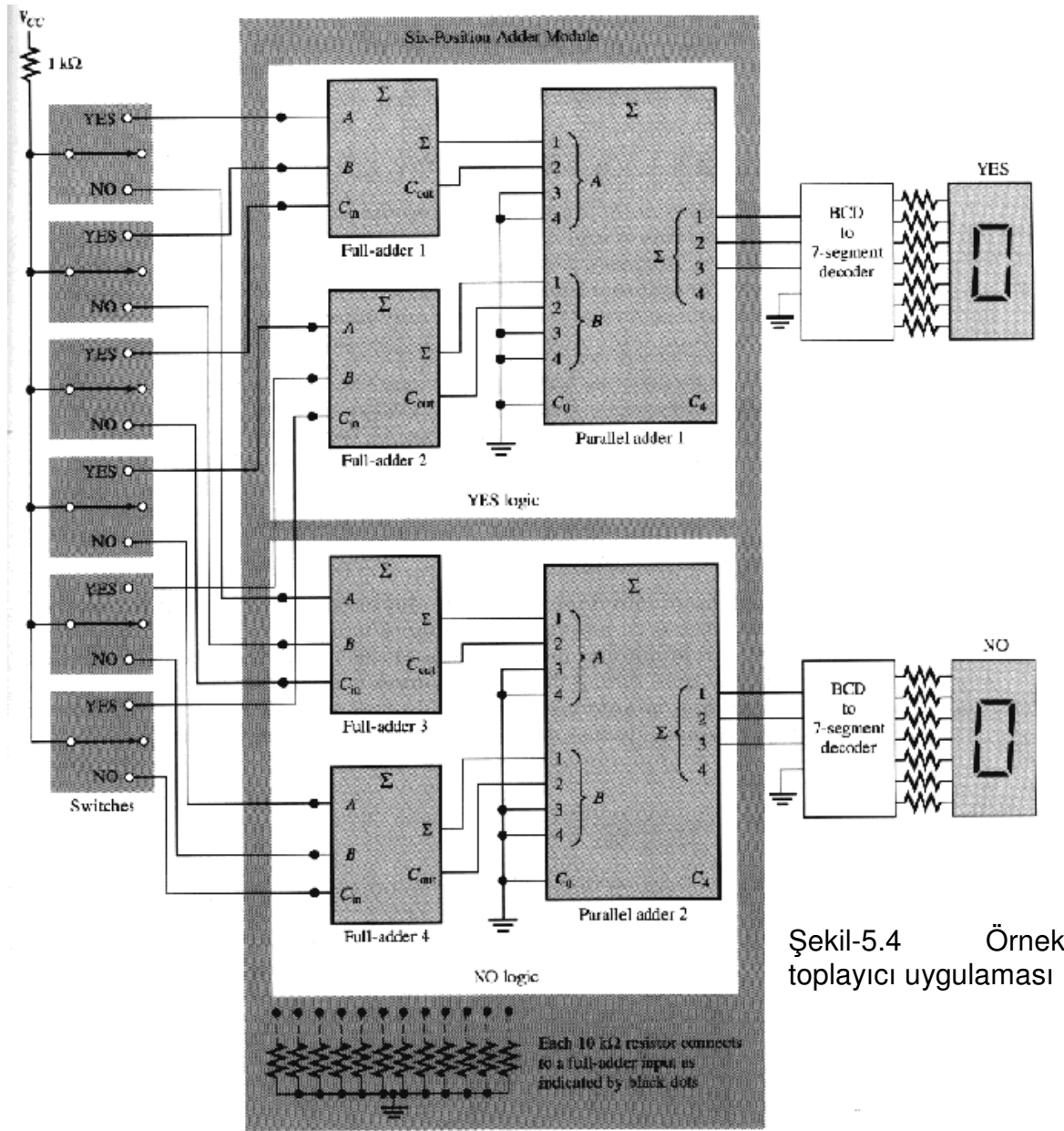
Tam toplayıcı bir bitlik iki sayıyı toplamak için kullanılabilir. Birden fazla bitlik ikilik sayıların toplanmasında tam toplayıcılar paralel bağlanarak kullanılır. Paralel toplayıcılarda en düşük değerli bit önce toplanır ve en son yüksek değerli bit toplanır. Düşük değerli bitlerin toplanmasından oluşan elde bir üst değerdeki sayıların toplanmasında kullanılır. En yüksek değerli bitlerin toplamından oluşan elde toplamın eldesidir. Toplam sonucunda sayının bit sayısı artmıştır. En yüksek değerli bitten oluşan eldeye toplam adı verilir.

Toplayıcılar SSI (Small-Scale Integration) tümdevreler kullanılarak elde edilebileceği gibi MSI (Medium-Scale Integration) tümdevreler kullanılarak da elde edilebilir. MSI tümdevrelerde kullanılan bağlantının azalmasının yanı sıra fiyatları da SSI tümdevrelere oranla daha düşüktür. Bu konunun deneyinde MSI ailesinden 7483 4 bit tam look-ahead-carry toplayıcıyı kullanarak iki adet 4 bitlik sayıyı toplayacaksınız. 74283 ise diğerinin yeni sürümüdür, besleme bacakları standart hale getirilmiştir. Onun dışında işlevsel olarak farkları yoktur.



Şekil-5.3 Paralel toplayıcı.

Şekil-5.4'te toplayıcı kullanılarak oy sayım makinesi şeması verilmiştir. Devrenin evet ve hayır sayan ayrı iki göstergesi vardır. Oy verme anahtarları üç konumludur; evet ve hayır konumlarında +5 volt uygulanır. Evet ve hayır oyları farklı iki devre tarafından toplanır.



Şekil-5.4 Örnek toplayıcı uygulaması

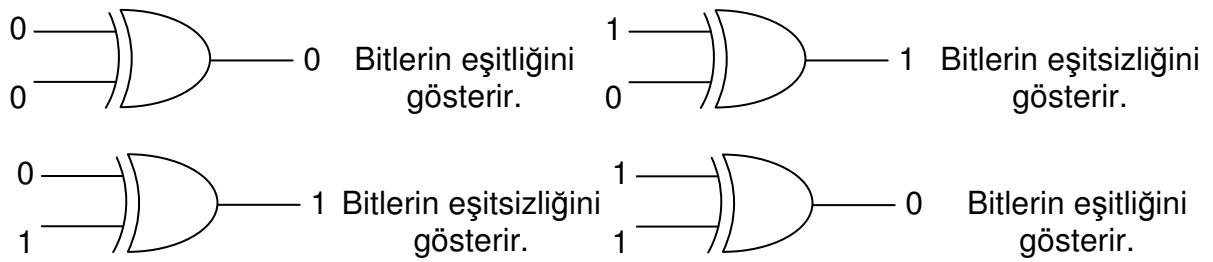
KARŞILAŞTIRICILAR

Karşılaştırıcıların temel işlevi iki niceliğin büyüklüğünü karşılaştırarak bu niceliklerin arasındaki ilişkiyi belirlemektir. En basit şekliyle karşılaştırıcı iki sayının eşitliğini denetler.

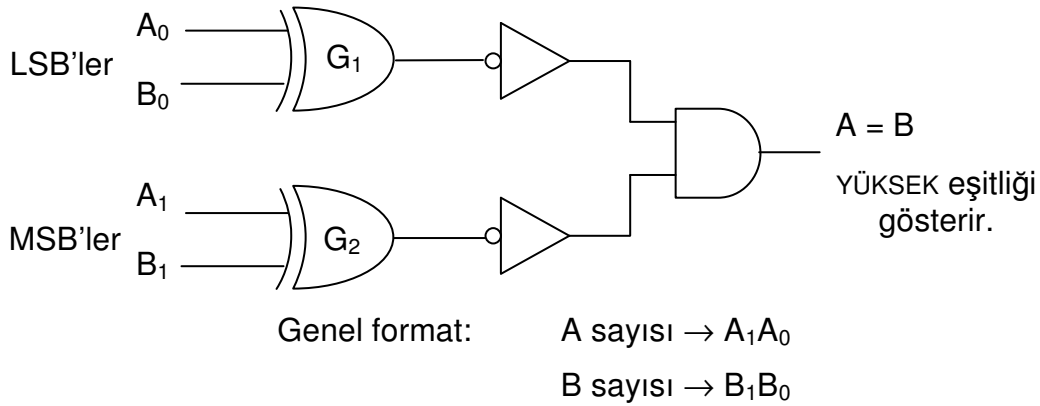
Zıt geçidi (EXOR) temel bir karşılaştırıcı olarak kullanılabilir çünkü bu geçidin çıkışı iki giriş biti eşit değilse "1", girişler eşitse "0"dır. Şekil-5.5'te EXOR geçidinin iki bitlik karşılaştırıcı olarak kullanılması gösterilmektedir.

İki bitlik ikilik sayıları karşılaştırmak için ek bir EXOR geçidine gerek vardır. İki sayının en az anlamlı iki biti (LSB) G_1 geçidiyle ve en anlamlı iki biti (MSB) G_2 geçidiyle karşılaştırılır (şekil-5.6). İki sayı eşitse, karşılıklı bitleri eşittir ve EXOR geçitlerinin çıkışları "0"dır. Karşılıklı bitleri eşit değilse, EXOR geçidinin çıkışı

"1"dir. İki sayının eşit yada eşit olmadığını belirten *tek* bir çıkış elde edebilmek için, Şekil-5.6'da gösterildiği gibi, iki evirici ve bir VE geçidi kullanılır. Her bir EXOR geçidinin çıkışı değillenir ve VE geçidinin girişine uygulanır. Her bir EXOR geçidinin giriş bitleri eşit olduğunda, sayıların karşılıklı bitleri de eşittir ve VE geçidinin her iki girişi "1" olduğu için çıkışında da "1" görülür. Sayılar eşit değilse, karşılıklı bit setlerinden biri yada her ikisi eşit değildir ve VE geçidinin en az bir girişine "0" geldiği için çıkışında da "0" görülür. Sonuç olarak, VE geçidinin çıkışı "1" ise sayıların eşitliği, "0" ise eşitsizliği belirtilir. Örnek-5.1'de her iki durumda da çalışması anlatılmaktadır. EXOR geçidi ile eviricilerin yerini EXNOR (eş) geçidi almıştır.



Şekil-5.5 Temel karşılaştırıcının çalışması.



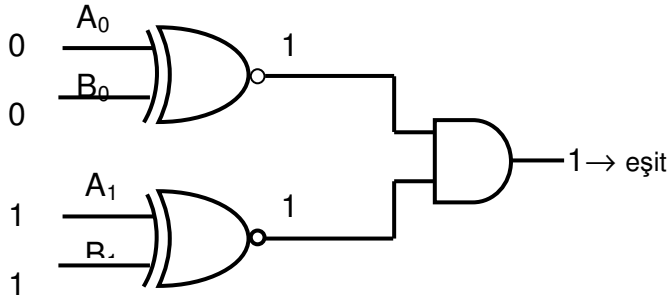
Şekil-5.6 İki 2-bitlik ikilik sayının karşılaştırmasını yapan devrenin mantık çizgesi.

Örnek-5.1 Her bir ikilik sayı setini Şekil-5.7'deki karşılaştırıcının girişlerine uygulayın ve devrenin her noktasının mantık seviyelerini belirleyerek çıkışı bulun.

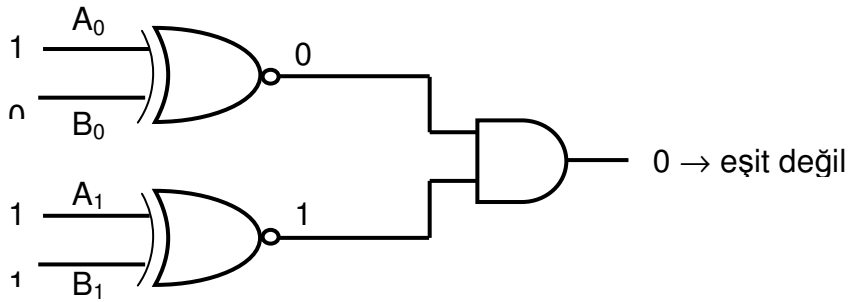
(a) 10 ve 10 (b) 11 ve 10

Çözüm:

Şekil-5.7(a)'da gösterildiği gibi, 10 ve 10 girişleri için çıkış "1"dir.



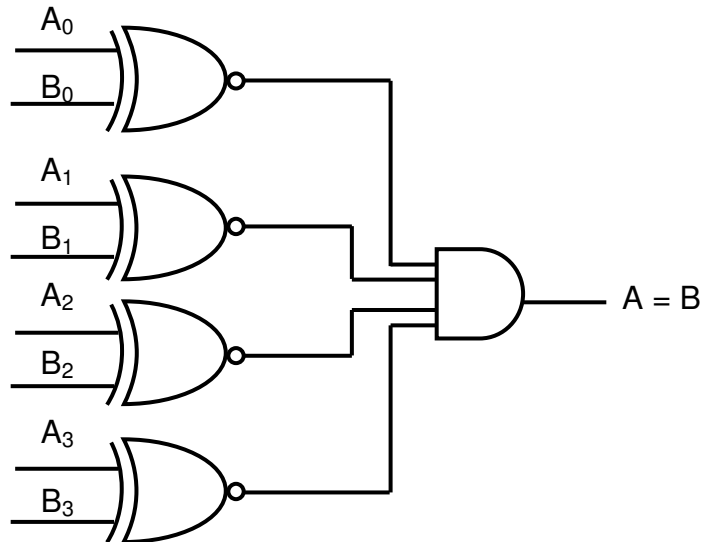
Şekil-5.7(a)



Şekil-5.7(b)

Şekil-5.7(b)'de gösterildiği gibi, 11 ve 10 girişleri için çıkış "0"dır.

İki adet 4-bitlik sayı için Şekil-5.8'de gösterildiği gibi, karşılaştırıcı devresi herhangi bir bit sayısına genişletilebilir. Bu devreye göre, iki sayı eşitse karşılıklı bitleri de eşittir ve VE geçidinin çıkışı "1" olur.



Şekil-5.8 İki 4-bitlik sayının, $A_3A_2A_1A_0$ ve $B_3B_2B_1B_0$, karşılaştırılmasını gösteren mantık devresi

MSI KARŞILAŞTIRICILAR

Bazı tümdevre karşılaştırıcıların karşılaştırılan iki sayıdan hangisinin daha büyük olduğunu belirten çıkışları da vardır. Şekil-5.9'daki dört-bitlik karşılaştırıcı mantık simgesinde görüldüğü gibi, A sayısı B sayısından büyük olduğunda ($A>B$) ve A sayısı B sayısından küçük olduğunda ($A<B$) belirten çıkışlar da eklenmiştir.

Bu iki çıkış işlevinin devreye eklenmiş hali Şekil-5.10'da gösterilmiştir. $A>B$ ve $A<B$ işlevlerini yerine getiren mantık devresinin çalışmasını anlayabilmek için iki adet ikilik sayıyı inceleyerek sayıların eşitsizliğini belirleyen durumu açıklayalım.

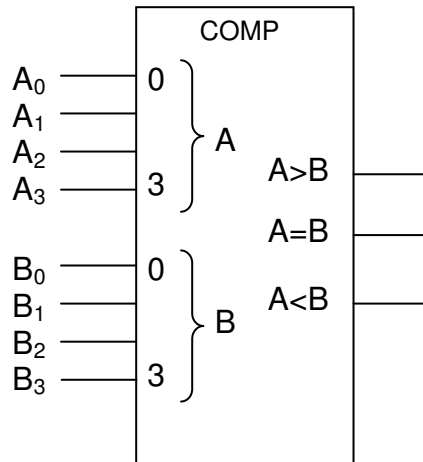
Genel biçimi A sayısı için $A_3A_2A_1A_0$ ve B sayısı için $B_3B_2B_1B_0$ olan iki 4-bitlik sayıyı kullanalım. A ve B sayılarının eşitsizliğini belirlemek için öncelikle her sayıda en önemli bitleri incelemeliyiz. Aşağıdaki koşullar olasıdır:

$A_3 = 1$ ve $B_3 = 0$ ise, A sayısı B sayısından büyüktür.

$A_3 = 0$ ve $B_3 = 1$ ise, A sayısı B sayısından küçüktür.

$A_3 = B_3$ ise, eşitsizlik için bir sonraki bitleri incelemeliyiz.

Bu üç inceleme sayılardaki her bit konumu için geçerlidir. Genel yöntem bir bit konumundaki eşitsizliği en anlamlı bitlerden başlayarak sınamaktır. Bir eşitsizlik bulunduğu, iki sayı arasındaki ilişki belirlenmiştir ve diğer daha az değerli bitlerin durumları *gözardı edilmelidir* çünkü zıt bir eşitsizlik belirtimi ortaya çıkabilir; en değerli sıradaki eşitsizlik ifadesi *önceliklidir*. Bunu açıklamak için A sayısı 0111 ve B sayısı da 1000 olsun. A_3 ve B_3 bitlerinin karşılaştırılması sonucu A sayısının B sayısından küçük olduğu ortaya çıkar, çünkü $A_3 = 0$ ve $B_3 = 1$ 'dir. Ancak A_2 ve B_2 bitleri karşılaştırıldığında $A_2 = 1$ ve $B_2 = 0$ olduğundan $A>B$ ifadesiyle karşılaşıyoruz. Aynı koşul diğer daha az değerli bitler için de geçerlidir. Bu durumda öncelik A_3 ve B_3 bitlerine verilmelidir çünkü uygun eşitsizlik koşulunu bu bitler sağlar.



Şekil-5.9 Eşitsizlik gösterimine sahip 4-bitlik büyüklük karşılaştırıcısının mantık simgesi.

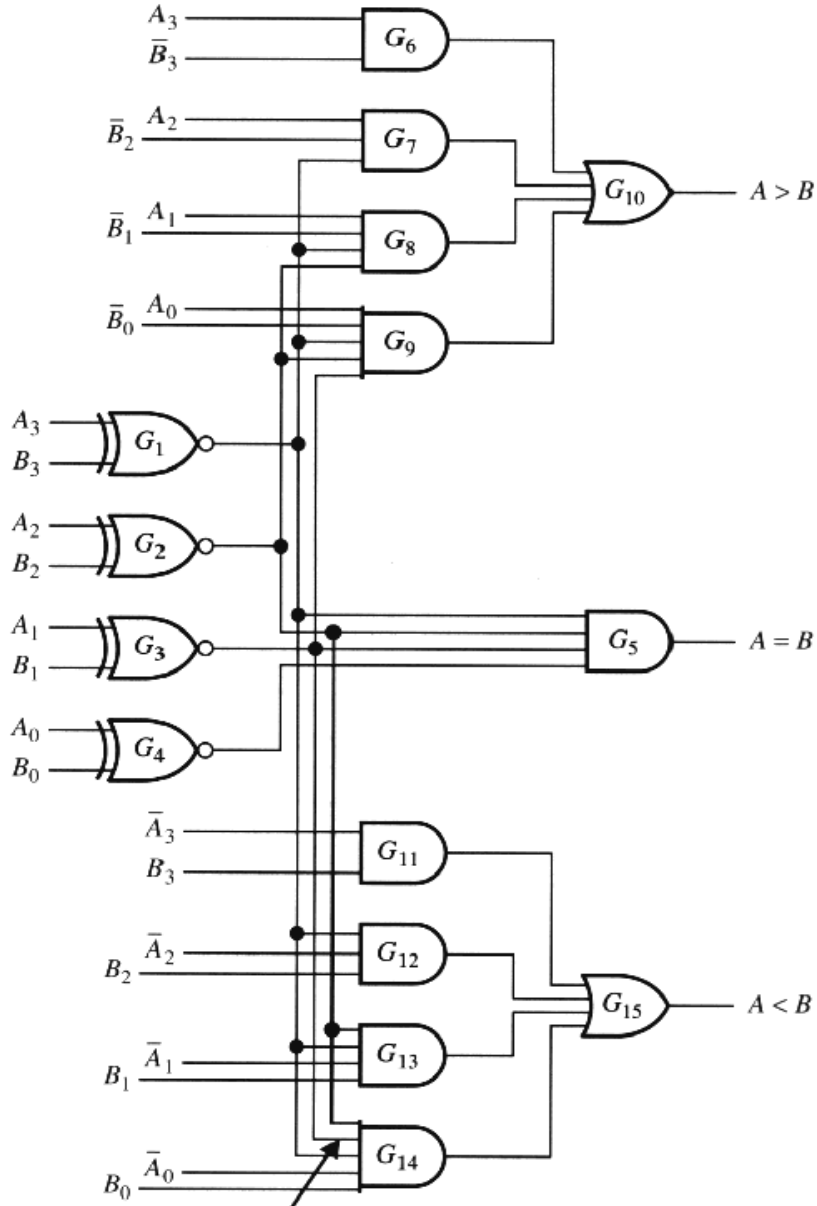
Şekil-5.10 iki 4-bitlik sayının karşılaştırılması ve $A > B$, $A < B$ veya $A = B$ çıkışlarının üretilmesi yöntemini göstermektedir. $A > B$ koşulu G_6 ile G_{10} arası geçitlerle sağlanmaktadır. G_6 geçidi $A_3=1$ ve $B_3=0$ 'ı denetlemektedir ve çıkış fonksiyonu A_3B_3 'tür. G_7 geçidi $A_2=1$ ve $B_2=0$ 'ı denetlemektedir ve çıkış fonksiyonu A_2B_2 'dir. G_8 $A_1=1$ ve $B_1=0$ 'ı denetlemektedir (A_1B_1). G_9 $A_0=1$ ve $B_0=0$ 'ı denetlemektedir (A_0B_0). Bu koşulların hepsi A sayısının B sayısından daha büyük olduğunu gösterir. Tüm bu geçitlerin çıkışları $A > B$ çıkışını elde etmek için G_{10} geçidiyle VEYA'lanır.

G_1 geçidinin çıkışı G_7 , G_8 , G_9 geçitlerinin girişlerine bağlanmıştır. Bu bağlantıyla, A_3 ve B_3 bitlerinde uygun eşitsizlik oluştuğunda ($A_3 < B_3$), diğer daha az değerli bitlerin incelenmesi engellenir. Aynı amaçla G_2 geçidi G_8 ve G_9 'a, G_3 geçidi de G_9 'a bağlanmıştır.

G_{11} ile G_{15} arası geçitler $A < B$ koşulunu sınamaktadır. Her VE geçidi A sayısında 0'ın ve B sayısında 1'in olup olmadığını denetler. VE geçidi çıkışları G_{15} geçidi ile VEYA'lanarak $A < B$ çıkışı elde edilir. Uygun eşitsizlik bulunduğu daha az değerli bitlerin incelenmesinin engellenmesi yukarıda anlatıldığı gibidir. A sayısındaki dört bit B'deki bitlere eşitse, her bir EXNOR geçidinin çıkışı 1'dir. Bu da $A = B$ çıkışındaki G_5 geçidinin çıkışının "1" olmasını sağlar.

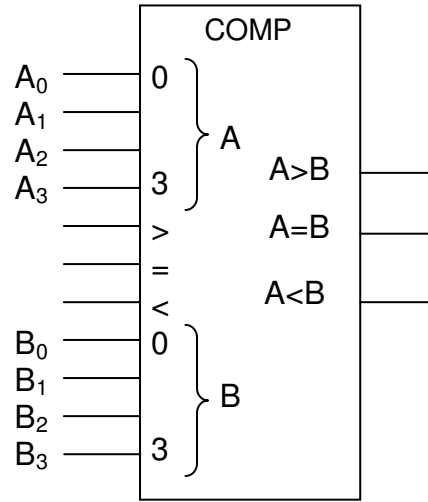
7485 Dört-Bitlik Büyüklük Karşılaştırıcı

7485 MSI teknolojisi ile üretilen büyüklük karşılaştırıcısıdır. Mantık simgesi Şekil-5.11'dedir. (Bu tümdevre hakkındaki teknik bilgiyi EK 1'de veri yaprakları kısmından bulabilirsiniz.)



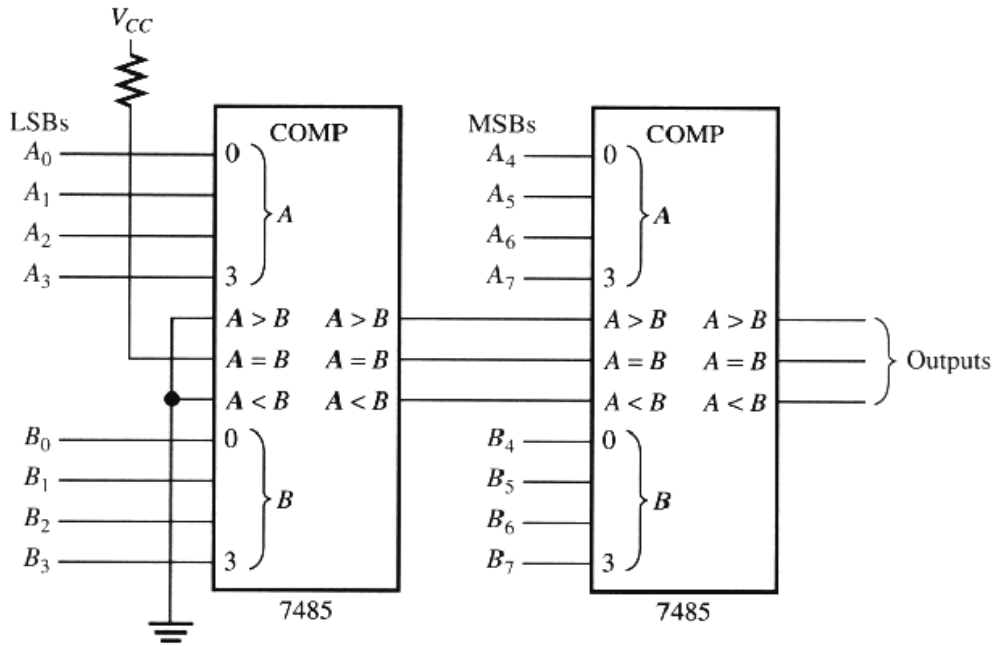
Şekil-5.10 4 bitlik büyüklük karşılaştırıcı

Daha önce bir büyüklük karşılaştırıcısında olduğunu belirttiğimiz giriş ve çıkışlara ek olarak üç adet *kaskat bağlama girişi* de bulunmaktadır. ($<$, $=$, $>$). Bu girişler dört bittten büyük sayıların karşılaştırılabilmesi için birkaç karşılaştırıcının kaskat bağlanabilmesine olanak sağlar. Karşılaştırıcıyı genişletebilmek için $A < B$, $A = B$ ve $A > B$ çıkışları bir sonraki daha değerli karşılaştırıcının karşılık gelen girişlerine bağlanır. En az değerli karşılaştırıcının $=$ girişi YÜKSEK, $<$ ve $>$ girişleri DÜŞÜK seviyeye bağlanmalıdır.



Şekil-5.11 7485 4-bitlik büyüklük karşılaştırıcı tümdevresinin mantık simgesi

Bu tümdevre kullanılarak 4 bitten daha fazla uzunluktaki ikilik sayılar karşılaştırılabilir. Şekil-5.12 8 bitlik iki sayının karşılaştırılmasına örnek verilmiştir.



Şekil-5.12 8 bitlik karşılaştırıcı.

BÖLÜM 6

KODÇÖZÜCÜLER VE KODLAYICILAR

GİRİŞ

Kodçözücü giriş bitlerinin belirli kombinasyonunun girişe gelip gelmediğini denetler, geldiğinde çıkışı etkin hale getirir. Girişine gelen veriyi bir başka forma dönüştürür. Günlük yaşamda onluk sayı sistemi kullanırız, fakat bilgisayar ikilik sayılar ile işlem yaptığı için bilgisayar kullanmak istediğimizde onludan ikiliye kodlayıcıya gereksinim duyarız. Kodlayıcının kodladığı bilgi tekrar günlük yaşamın parçası olabilmesi için kodçözücü kullanılarak kodu eski haline getirilmelidir.

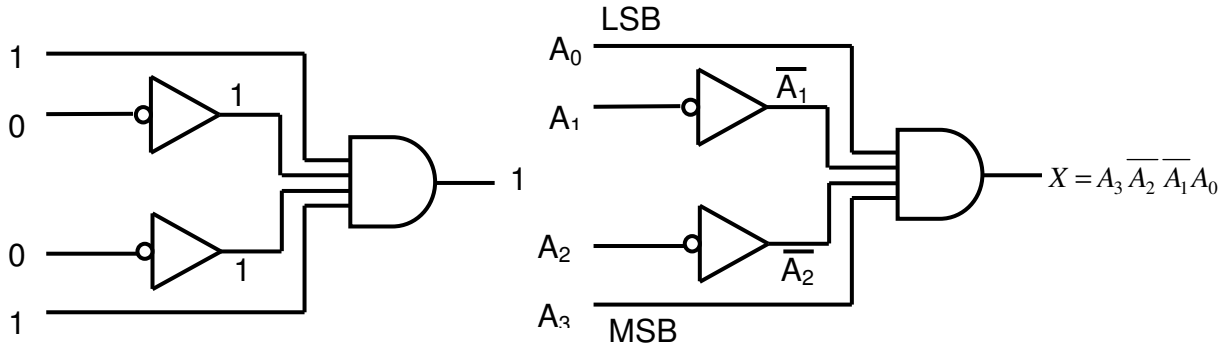
KOD ÇÖZÜCÜLER (DECODERS)

Kodçözücünün temel işlevi girişlerinde belirli bir bit birleşiminin (kod) olup olmadığını algılamak ve varsa bunu belirli bir çıkış seviyesiyle göstermektir. Genel olarak, kodçözücülerin n girişi varsa bir ile 2^n adet çıkışı olabilir.

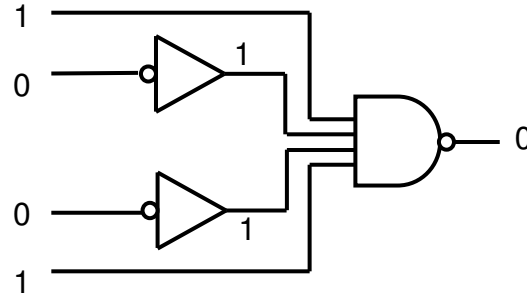
TEMEL İKİLİK KODÇÖZÜCÜ

Bir dijital devrenin girişlerine ikilik 1001 sayısı gelip gelmediğini belirlemek için bir VE geçidi temel kodçözücü elemanı olarak kullanılabilir çünkü bu geçit tüm girişleri YÜKSEK olduğunda YÜKSEK seviye üretir. Dolayısıyla, 1001 ikilik sayısı oluştuğunda VE geçidinin bütün girişlerine YÜKSEK seviye gelebilmesi için ortadaki iki biti (0'lar) deęillemek gerekir Şekil-6.1(a)'daki kodçözücü için mantık fonksiyonu Şekil-6.1(b)'de gösterildiği gibidir. Çıkış fonksiyonu $A_0 = 1$, $A_1 = 0$, $A_2 = 0$ ve $A_3 = 1$ girişlere uygulanmadığı sürece "0"dır. A_0 en az değerli bit ve A_3 en değerli bittir.

VE geçidinin yerine VED geçidi kullanıldığında (Şekil-6.2), uygun ikilik kodun varlığını DÜŞÜK seviye ile belirtir.



Şekil-6.1 1000 için etkin-YÜKSEK seviyeli kodçözücü devre.



Şekil-6.2 1001 için etkin-DÜŞÜK seviyeli kodçözücü devre.

Örnek-6.1

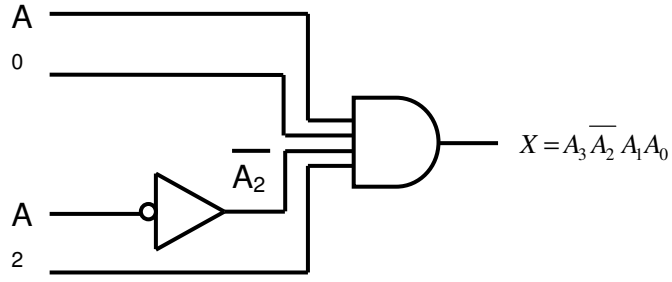
1011 ikilik sayısının kodunun çözülmesi için gerekli etkin-YÜKSEK çıkışlı mantık devresini çizin.

Çözüm:

İkilik sayıdaki sıfır olarak görünen değişkenin değillemesiyle fonksiyon bulunmuş olur:

$$X = A_3 \overline{A_2} A_1 A_0$$

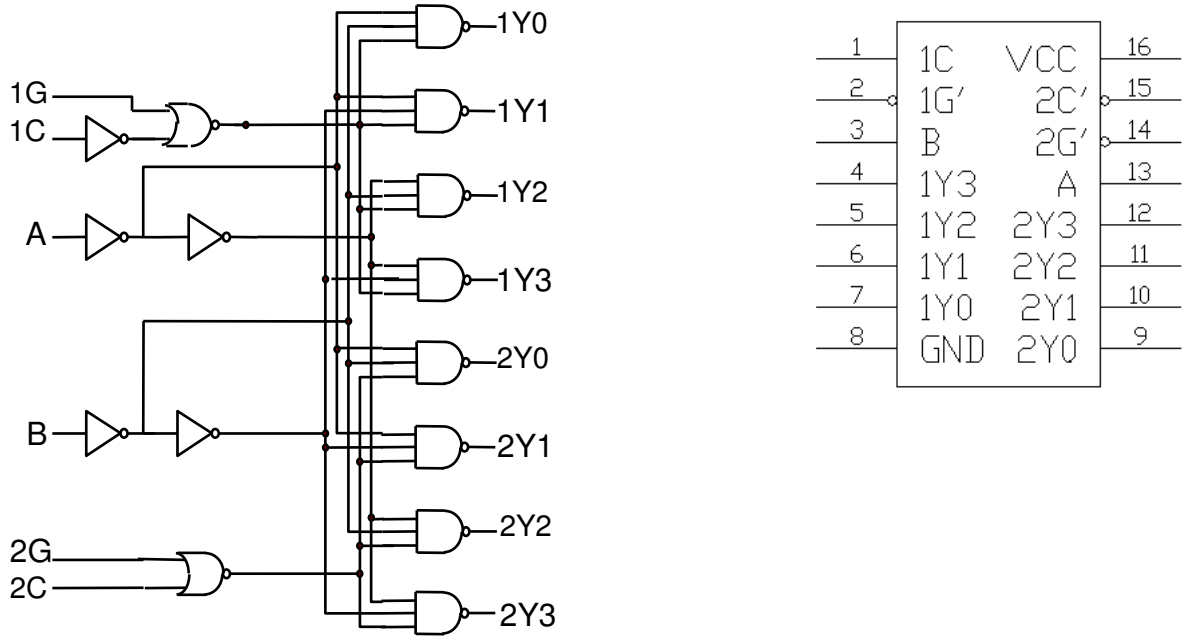
Bu fonksiyon değillenmemiş değişkenler olan A_0 , A_1 , A_3 'ü VE geçidinin girişine doğrudan bağlayarak ve A_2 değişkenini değilleyerek VE geçidinin girişine uygulayarak gerçekleştirir (Şekil-6.3).



Şekil-6.3 1011 için etkin-YÜKSEK seviyeli kodçözücü devre.

74156 VE 74138 3-E-8 KODÇÖZÜCÜLER

Tümdevre şeklinde üretilmiş kod çözücülerde (MSI) olası tüm çıkışlar tümdevre üzerine yerleştirilir, kullanıcılar istediği çıkışları kullanırlar. 3 bit ikilik kodçözücü 74LS156 tümdevresinin çıkış sayısı $2^3 = 8$ 'dir çıkışı vardır. Bu tümdevrenin doğruluk tablosu Tablo-6.1'de verilmiştir. Tümdevrenin veri yaprağı ekte verilmiştir. Açık kollektör olduğu için çıkış uçlarına yükseğe çekme dirençleri bağlanmalıdır. Aksi halde çalışmaz.



Şekil-6.4 74156 3 bit kodçözücü tümdevresinin mantık sembolü ve iç bağlantı şeması.

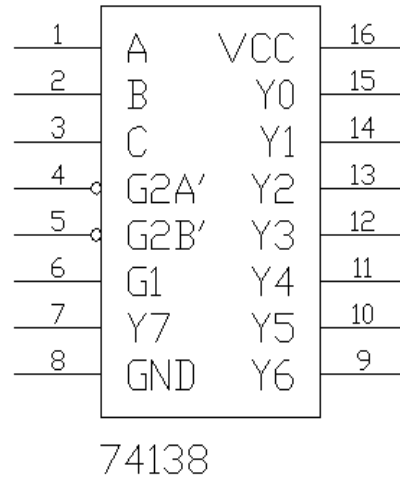
74LS156 tümdevresinde A, B girişleri tek, 3. giriş 1C ve 2C olarak iki parçaya bölünmüştür. 1C ve 2C girişleri birleştirilip kullanıldığında 3'e 8 kodçözücü olarak kullanılırken ayrı kullanıldığında iki adet 2'ye 4 kodçözücü olarak çalıştırılabilir.

Bacak bağlantısı ve iç yapısı Şekil-6.4'te gösterilmiştir. 1G ve 2G girişleri birinci ve ikinci 2'ye 4 kodçözümlerinin izin verme girişleridir. 3'e 8 kodçözümlerinin izin verme girişleri olarak kullanıldığında bu iki giriş birleştirilerek kullanılır.

74LS138 tümdevresi 74LS156 tümdevresi gibi 3'e 8 kodçözümlüdür. Farkı ise 3 adet kontrol girişine sahip olması ve açık kollektör olmaması ve direnç bağlantısına ihtiyaç duymamasıdır. Kontrol girişlerinden iki tanesi düşükte etkin olurken diğeri YÜKSEKte etkin olur. Çıktılar etkin düşüğe göre çalışır. Şekil-6.5'te bu tümdevrenin mantık sembolü verilmiştir. Tablo-6.2'de ise aynı tümdevrenin doğruluk tablosu verilmiştir.

GİRİŞLER				ÇIKIŞLAR							
SEÇME			İZİN	0	1	2	3	4	5	6	7
C	B	A	G	2Y0	2Y1	2Y2	2Y3	1Y0	1Y1	1Y2	1Y3
X	X	X	1	1	1	1	1	1	1	1	1
0	0	0	0	0	1	1	1	1	1	1	1
0	0	1	0	1	0	1	1	1	1	1	1
0	1	0	0	1	1	0	1	1	1	1	1
0	1	1	0	1	1	1	0	1	1	1	1
1	0	0	0	1	1	1	1	0	1	1	1
1	0	1	0	1	1	1	1	1	0	1	1
1	1	0	0	1	1	1	1	1	1	0	1
1	1	1	0	1	1	1	1	1	1	1	0

Tablo-6.1 3 bit ikilik kodçözümlü 74LS156 tümdevresinin doğruluk tablosu.



Şekil-6.5 74138 3 bit kodçözücü tümdevresinin mantık sembolü.

GİRİŞLER					ÇIKIŞLAR							
SEÇME			İZİN		0	1	2	3	4	5	6	7
C	B	A	G ₁	G ₂	Y0	Y1	Y2	Y3	Y4	Y5	Y6	Y7
X	X	X	X	1	1	1	1	1	1	1	1	1
X	X	X	0	X	1	1	1	1	1	1	1	1
0	0	0	1	0	0	1	1	1	1	1	1	1
0	0	1	1	0	1	0	1	1	1	1	1	1
0	1	0	1	0	1	1	0	1	1	1	1	1
0	1	1	1	0	1	1	1	0	1	1	1	1
1	0	0	1	0	1	1	1	1	0	1	1	1
1	0	1	1	0	1	1	1	1	1	0	1	1
1	1	0	1	0	1	1	1	1	1	1	0	1
1	1	1	1	0	1	1	1	1	1	1	1	0

Tablo-6.2 3 bit ikilik kodçözücü 74LS138 tümdevresinin doğruluk tablosu.

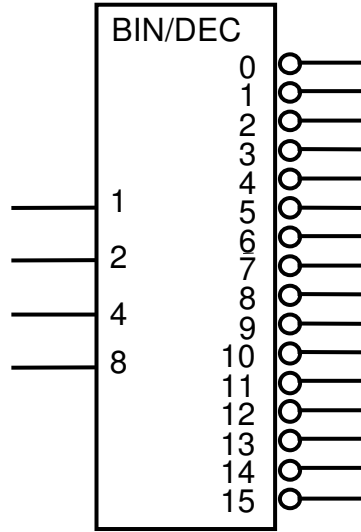
DÖRT BİTLİK İKİLİK KODÇÖZÜCÜ

Dört bit için tüm olası kombinasyonların kodunun çözülebilmesi için on altı kod çözücü geçide gerek vardır ($2^4 = 16$). Bu tür bir kod çözücüye *4-hattan-16-hatta* kod çözücü denir çünkü dört girişi ve on altı çıkışı vardır. On altı ikilik kod sözcüğü ve karşılık gelen mantık fonksiyonları Tablo-6.3'te verilmiştir.

Girişler				Mantık	Çıkışlar															
A ₃	A ₂	A ₁	A ₀	Fonksiyonu	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
0	0	0	0	$\overline{A_3} \overline{A_2} \overline{A_1} \overline{A_0}$	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	$\overline{A_3} \overline{A_2} \overline{A_1} A_0$	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	0	$\overline{A_3} \overline{A_2} A_1 \overline{A_0}$	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	1	1	$\overline{A_3} \overline{A_2} A_1 A_0$	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1
0	1	0	0	$\overline{A_3} A_2 \overline{A_1} \overline{A_0}$	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1
0	1	0	1	$\overline{A_3} A_2 \overline{A_1} A_0$	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1
0	1	1	0	$\overline{A_3} A_2 A_1 \overline{A_0}$	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
0	1	1	1	$\overline{A_3} A_2 A_1 A_0$	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
1	0	0	0	$A_3 \overline{A_2} \overline{A_1} \overline{A_0}$	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
1	0	0	1	$A_3 \overline{A_2} \overline{A_1} A_0$	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
1	0	1	0	$A_3 \overline{A_2} A_1 \overline{A_0}$	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
1	0	1	1	$A_3 \overline{A_2} A_1 A_0$	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
1	1	0	0	$A_3 A_2 \overline{A_1} \overline{A_0}$	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
1	1	0	1	$A_3 A_2 \overline{A_1} A_0$	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1
1	1	1	0	$A_3 A_2 A_1 \overline{A_0}$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
1	1	1	1	$A_3 A_2 A_1 A_0$	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Tablo-6.3 4-hattan-16-hatta kod çözücü için mantık fonksiyonu ve doğruluk tablosu.

Her sayı için etkin-DÜŞÜK çıkış gerekirse, kodçözücü devrenin VED geçitleriyle ve eviricilerle tasarlanması gerekir. Öncelikle her değişkenin kendisi ve değıline gerek olduğundan değışkenlerin değılleri bir defa üretilir ve gerekli olan tüm geçitler için kullanılır. Onaltı ikilik kodun kodunun çözülebilmesi için onaltı VED geçidi gereklidir. Kodçözücünün devresi bu şekilde çizilebilir. Ancak bu şekildeki bir devrenin çizimi karmaşık olacağı için 4-e-16 kodçözücü için Şekil-6.6'daki mantık simgesi kullanılır. BIN/DEC etiketi ikilik (binary) girişin karşılık gelen onluk (decimal) çıkışı etkin yaptığını belirtir. Giriş etiketleri 1, 2, 4 ve 8 giriş bitlerinin ikilik ağırlıklarını gösterir.



Şekil-6.6 İkilden onluya kod çözücü.

74154 4-E-16 KODÇÖZÜCÜ

74154 TTL MSI bir kodçözücüdür. 74154 4 bit kodçözücü tümdevresinin mantık sembolü Şekil-6.7'de, doğruluk tablosu ise, Tablo-6.4'tedir. Tümdevre, 4 adet giriş, 16 adet çıkış ve iki adet içeriden VE'lenmiş etkin düşükte çalışan kontrol (izin) girişine (G_1 ve G_2) sahiptir. Çıkışlar etkin düşükte çalışır. Kontrol girişlerinin her ikisine birden sıfır gelmediği durumda girişlerin durumu ne olursa olsun çıkışların tamamı "1" seviyesinde kalacaktır. Çıkışlar açık kollektör bağlantı olmadığından çıkışlara direnç bağlantısı gerekmez.

1	0	VCC	24
2	1	A	23
3	2	B	22
4	3	C	21
5	4	D	20
6	5	G_2'	19
7	6	G_1'	18
8	7	15	17
9	8	14	16
10	9	13	15
11	10	12	14
12	GND	11	13

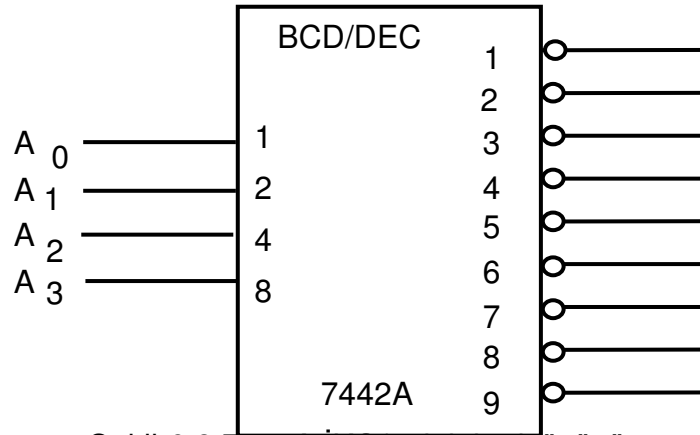
Şekil-6.7 74154 4-e-16 kodçözücü (a) bacak bağlantısı, (b) mantık simgesi.

GİRİŞLER						ÇIKIŞLAR																
G ₁	G ₂	A	B	C	D	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
1	1	X	X	X	X	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	1	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	0	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	0	0	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	0	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1	1
0	0	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1	1
0	0	1	0	0	0	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1	1
0	0	1	0	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1	1
0	0	1	0	1	0	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1	1
0	0	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1	1
0	0	1	1	0	0	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1	1
0	0	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1	1
0	0	1	1	1	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1	1
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0	1
0	0	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	0

Tablo-6.4 74154 tümdevresinin doğruluk tablosu.

İKO/ONLUK KODÇÖZÜCÜ

İKO/onluk kodçözücü her İKO kod sözcüğünü (8421 kodu) olası on onluk basamak gösterimlerinden birine çevirir. *4-hattan-10-hatta* kodçözücü olarak da adlandırılır. Bu kodçözücünün iç yapısı 4-e-16 kodçözücüyle temelde aynıdır, ancak farkı 16 yerine 10 adet kodçözücü geçit kullanılmasıdır, çünkü İKO kod 0 ile 9 arasında on adet onluk basamakla gösterilmektedir. On İKO kod sözcüğünün listesi ve karşılık gelen fonksiyonları Tablo-6.5'te verilmiştir. Bu fonksiyonların her biri etkin-DÜŞÜK çıkış üretmek için VED geçitleriyle gerçekleştirilir. Etkin-YÜKSEK çıkış üretmek gerekirse, VE geçitleri kullanılır.



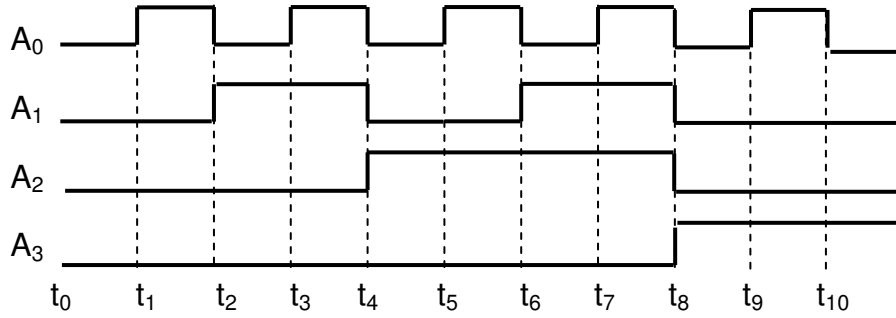
Şekil-6.8 7442A İKO/onluk kodçözücü.

Onluk Basamak	İKO Kod				Mantık Fonksiyonu
	A ₃	A ₂	A ₁	A ₀	
0	0	0	0	0	$\overline{A_3} \overline{A_2} \overline{A_1} \overline{A_0}$
1	0	0	0	1	$\overline{A_3} \overline{A_2} \overline{A_1} A_0$
2	0	0	1	0	$\overline{A_3} \overline{A_2} A_1 \overline{A_0}$
3	0	0	1	1	$\overline{A_3} \overline{A_2} A_1 A_0$
4	0	1	0	0	$\overline{A_3} A_2 \overline{A_1} \overline{A_0}$
5	0	1	0	1	$\overline{A_3} A_2 \overline{A_1} A_0$
6	0	1	1	0	$\overline{A_3} A_2 A_1 \overline{A_0}$
7	0	1	1	1	$\overline{A_3} A_2 A_1 A_0$
8	1	0	0	0	$A_3 \overline{A_2} \overline{A_1} \overline{A_0}$
9	1	0	0	1	$A_3 \overline{A_2} \overline{A_1} A_0$

Tablo-6.5 İKO Kodçözücü fonksiyonları.

Örnek 6.2:

7442A İKO/onluk kodçözücü tümdevresidir. Mantık simgesi Şekil-6.8'de verilmiştir. Şekil-6.9'daki dalga şekilleri 7442A'nın girişlerine uygulandığında çıkış dalga şeklini çizin.



Şekil-6.9

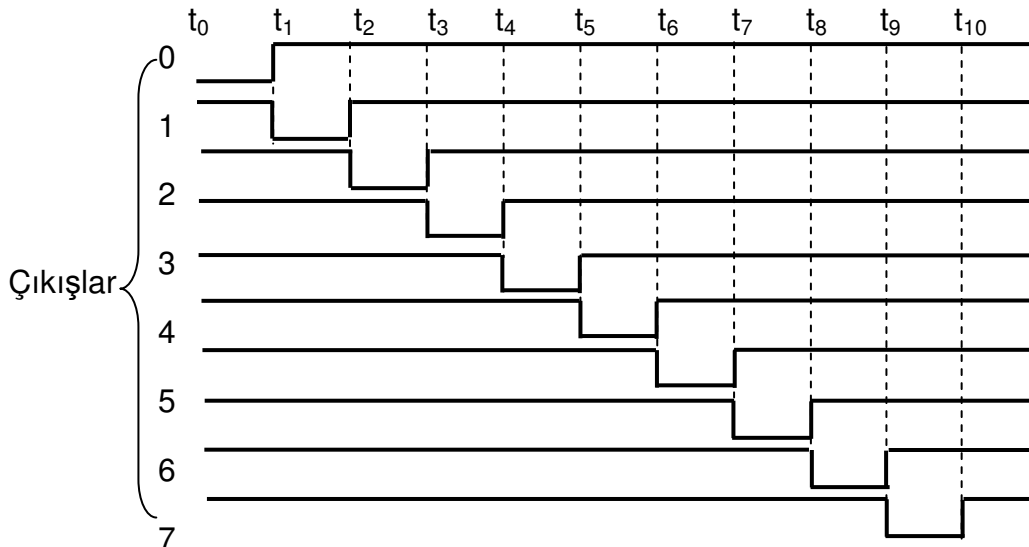
Çözüm:

Çıkış dalga şekilleri Şekil-6.10'da gösterildiği gibidir. Girişler artan sırayla İKO kodlarını ifade ettiğinden çıkışlarda da aynı sırayla karşılık gelen onluk çıkışlar etkin olmaktadır.

Kod çözümler pratikte mikrobilgisayar sistemlerinde farklı tipteki bellek tümdevreleri kullanıldığında bu tümdevrelerin adres bilgisinin doğru bir şekilde ulaştırılması için kullanılır.

KODLAYICILAR (ENKODERS)

Kodçözücünün tersi işlem yapar. Onluk sayıları İKO'ya veya sekizli sayıları ikiliye, onaltılık sayıları ikiliye dönüştürür.



Şekil-6.10

ONLUDAN İKO'YA KODLAYICI

Bu kodlayıcının 10 adet girişi ve 4 adet çıkışı vardır. Tablo-6.6'da onluk sayıların İKO karşılıkları verilmiştir. Bu devrenin 4 adet çıkışı olacaktır, bu çıkışları başlangıçta diğerlerinden bağımsız düşünerek mantık devresini tasarlayabiliriz. A_3 çıkışının 1 olabilmesi için 8 veya 9 sayılarından biri 1 olmalıdır. Buna göre A_3 'ün ifadesi şöyle yazılabilir.

$$A_3=8+9$$

A_2 biti ise 4,5,6,7 sayılarının herhangi birisi girişte etkin olduğu zaman 1 olur. Diğer durumlarda sıfır olur. VEYA işlemi ile tanımlayabiliriz.

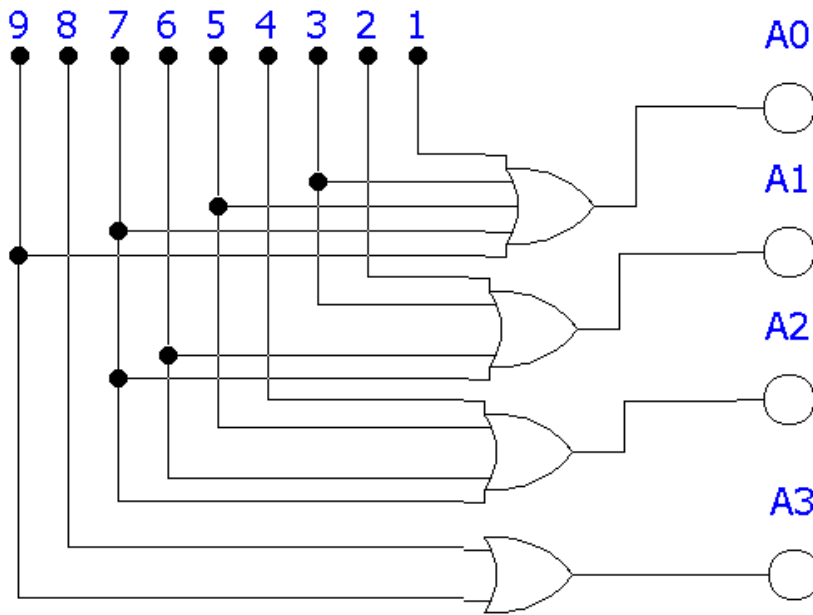
$$A_2=4+5+6+7$$

Aynı yöntemle $A_1=2+3+6+7$ ve $A_0=1+3+5+7+9$ olarak tanımlayabiliriz.

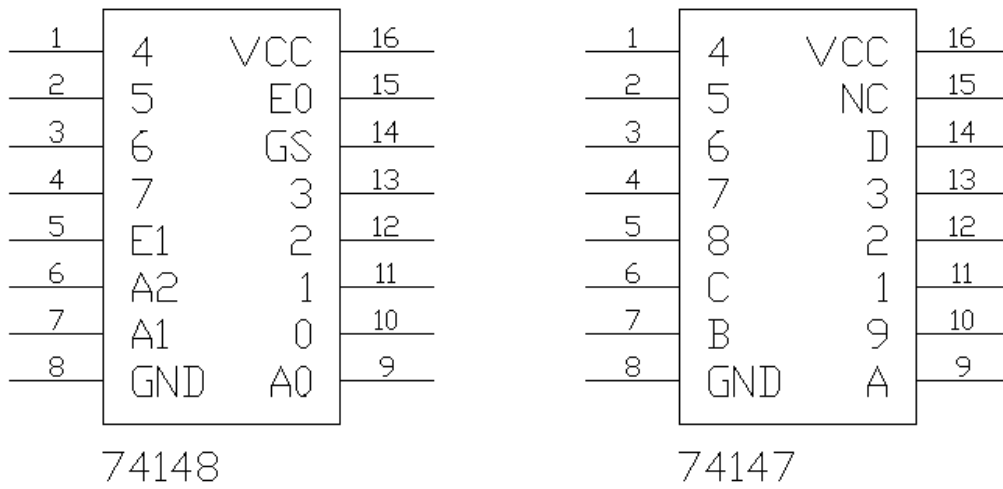
Şekil-6.11'de kodlayıcı mantık geçitleri ile gösterilmiştir. Kodlayıcıların girişi genellikle kboarda bağlıdır, kullanıcının kboardun birden fazla tuşuna aynı anda basması durumunda çıkışların oluşmasında kargaşa oluşacaktır. Tuşlara öncelik verilmelidir, öncelik genellikle en büyük değerli girişe verilir. Onluk-İKO kodlayıcıda 9'a öncelik verildiğinde devre karmaşık hale gelecektir. Şekil-6.12'de 8 girişli ve yüksek değerli giriş öncelikli 3 bit ikilik çıkışlı 74148'in ve 10 girişli İKO çıkışlı YÜKSEK değerli giriş öncelik veren 74147'nin bacak bağlantıları verilmiştir. Girişler ve çıkışlar DÜŞÜK seviyede etkin olurlar.

ONLU SAYI	İKO KOD			
	A_3	A_2	A_1	A_0
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0
9	1	0	0	1

Tablo-6.6



ŞEKİL-6.11 Onluk-İKO kodlayıcı.



Şekil-6.12

BÖLÜM 7

MULTİPLEXER DEMULTİPLEXER

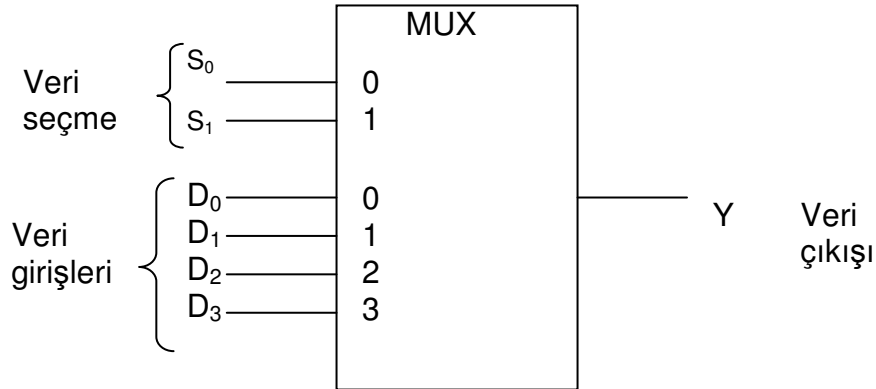
GİRİŞ

Multiplexer veya veri seçiciler her zaman aralığında girişlerindeki verinin birini tek bir çıkışa bağlar. Demultiplexer ise bu işlemin tersini gerçekleştirir. Aynı tümdevre (dekoder) kodçözücü olarak kullanılabilir. Multiplexer kısaca MUX, demultiplexer ise DEMUX olarak adlandırılır. Multiplexer ve demultiplexer tümdevreleri kullanılarak birleşik mantık devreleri tasarlanabilir. Eşlik biti üretici/denetleyici iletişim sistemlerinde oluşan hataları denetlemek için üretilir. Sadece bir bitin değerinin değişmesini algılar. İletilecek olan veri bitleri içerisindeki "1"lerin sayısının tek ya da çift olmasına göre hata denetimi yapılır. Eşlik biti üretici/denetleyici çift (EVEN) veya tek (ODD) eşliğe göre çıkışı etkin yapar. Tümdevreler üzerinde hem tek eşlik biti için hem de çift eşlik biti için çıkış bulunur.

MULTİPLEXER (VERİ SEÇİCİLER)

Multiplexer (MUX) sayısal veriyi birkaç kaynaktan alıp iletim için ortak tek bir yola aktaran bir aygıttır. Temel bir veri seçicide birkaç veri giriş hattı ve tek bir çıkış hattı vardır. Hangi girişin çıkışa aktarılacağını belirleyen birkaç veri seçme hattı da bulunmaktadır.

Dört girişli bir veri seçicinin (multiplexer) mantık simgesi Şekil-7.1'de verilmiştir. Burada iki tane veri seçme hattı vardır ki bu iki hatla dört veri hattından hangisinin çıkışa aktarılacağı seçilebilir. Şekil-7.1'e baktığımızda, veri seçme girişlerindeki iki bitlik ikilik kod seçilen veri girişindeki verinin veri çıkışına aktarılmasını sağlayacaktır. İkilik 0 ($S_1 = 0$ ve $S_0 = 0$) veri seçme hatlarına uygulanırsa, D_0 girişindeki veri çıkışa aktarılır. İkilik 1 ($S_1 = 0$ ve $S_0 = 1$) veri seçme hatlarına uygulanırsa, D_1 girişindeki veri çıkışa aktarılır. İkilik 2 ($S_1 = 1$ ve $S_0 = 0$) veri seçme hatlarına uygulanırsa, D_2 girişindeki veri çıkışa aktarılır. İkilik 3 ($S_1 = 1$ ve $S_0 = 1$) veri seçme hatlarına uygulanırsa, D_3 girişindeki veri çıkışa aktarılır. Bu çalışmanın özeti Tablo-7.1'dedir.



Şekil-7.1 4-giriş 1-çıkış hatlı (4×1) veri seçici/ multiplexer için mantık simgesi.

Veri Seçme Girişleri		
S ₁	S ₀	Seçilen veri
0	0	D ₀
0	1	D ₁
1	0	D ₂
1	1	D ₃

Tablo-7.1 Dört girişli veri seçici için veri seçimi.

Veri seçicinin bu işlemi yapmasını sağlayan mantık devresini inceleyelim. Veri çıkışı *seçilen* veri girişinin durumuna eşittir. O zaman, veri çıkışını veri girişleri ve veri seçme girişleri cinsinden yazabilmeliyiz. Bunu şu şekilde yapabiliriz:

Yalnızca S₁ = 0 ve S₀ = 0 olduğunda, Y veri çıkışı D₀ veri girişine eşittir:

$$Y = D_0 \overline{S_1} \overline{S_0}$$

Yalnızca S₁ = 0 ve S₀ = 1 olduğunda, veri çıkışı D₁ veri girişine eşittir:

$$Y = D_1 \overline{S_1} S_0$$

Yalnızca S₁ = 1 ve S₀ = 0 olduğunda, veri çıkışı D₂ veri girişine eşittir:

$$Y = D_2 S_1 \overline{S_0}$$

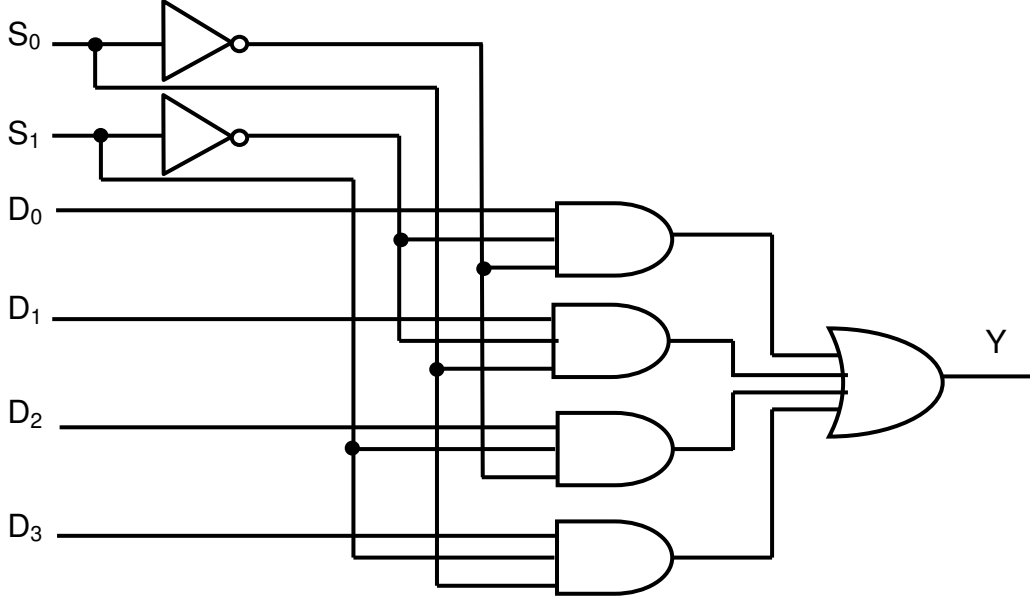
Yalnızca S₁ = 1 ve S₀ = 1 olduğunda, veri çıkışı D₃ veri girişine eşittir:

$$Y = D_3 S_1 S_0$$

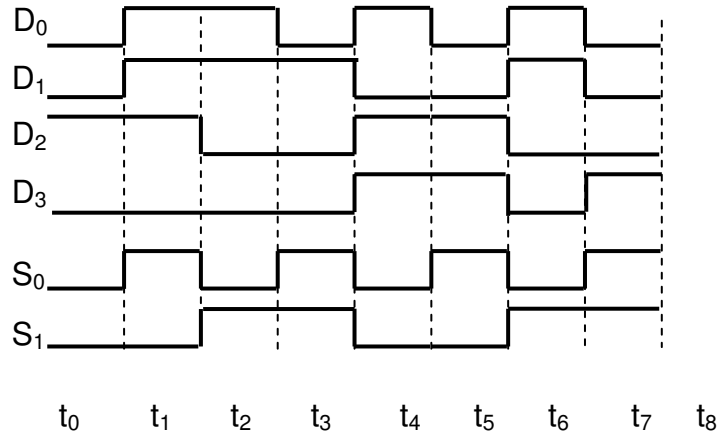
Bu terimler VEYAlandığında veri çıkışının toplam ifadesi elde edilir:

$$Y = D_0 \overline{S_1} \overline{S_0} + D_1 \overline{S_1} S_0 + D_2 S_1 \overline{S_0} + D_3 S_1 S_0$$

Bu eşitliğin mantık geçitleriyle gerçekleştirilmesi için 4 adet 3 girişli VE geçidi, 1 adet 4 girişli VEYA geçidi ve S_1 ve S_0 'ın deęillerini almak için iki adet evirici gerekmektedir (Şekil-7.2). Herhangi bir giriş hattından veri seçilebildiğinden bu devreye *veri seçici* de denilmektedir.



Şekil-7.2 Dört girişli veri seçicinin mantık devresi.



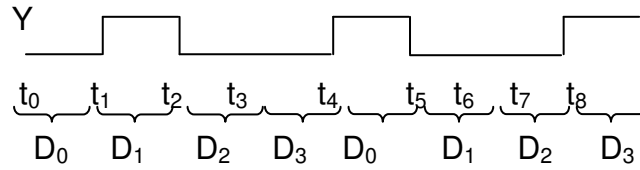
Şekil-7.3

Örnek 7.1

Şekil-7.3'teki veri seçici dalga şekilleri Şekil-7.2'deki veri seçicinin girişlerine uygulanmıştır. Girişlere göre çıkış dalga şeklini çizin.

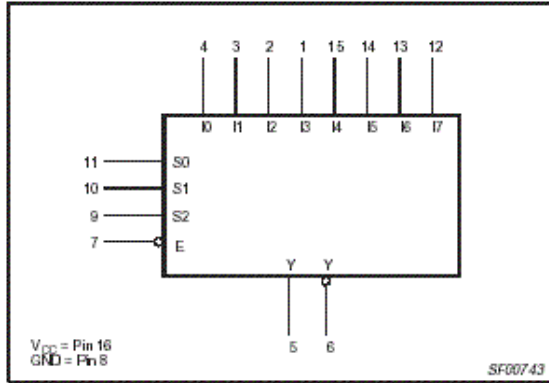
Çözüm:

Her aralıkta veri seçme hatlarının ikilik durumları hangi verinin seçileceğine karar verir. Veri seçme girişlerinin 00, 01, 10, 11, 00, 01, 10, 11,... şeklinde yinelenen kodlar şeklinde devam ettiğine dikkat edin. Çıkış dalga şekli Şekil-7.4'te görülmektedir.

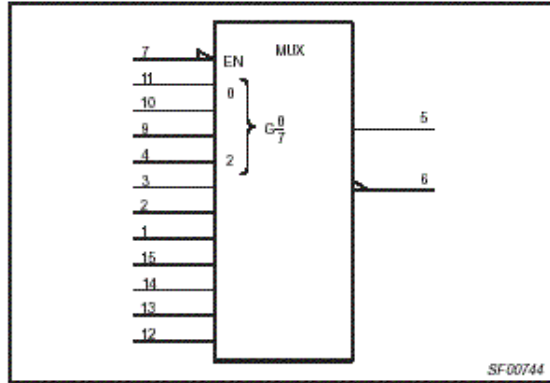


Şekil-7.4

LOGIC SYMBOL



IEC/IEEE SYMBOL

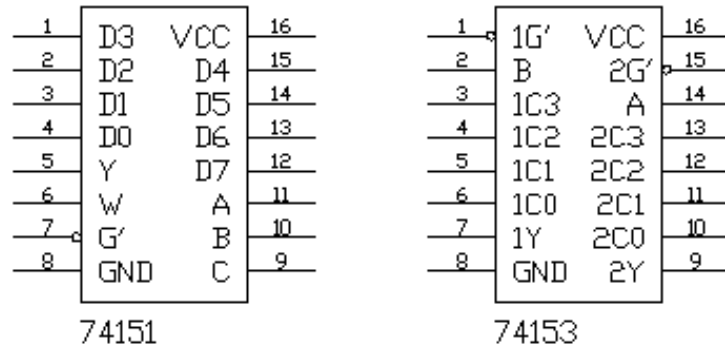


Şekil-7.5 74151 multiplexer tümdevresinin mantık gösterimi.

Şekil-7.5'te deneyde kullanılacak olan 74151 multiplexer tümdevresinin mantık gösterimi verilmiştir. 8 veri girişi ve 3 adet veri seçimi yapan denetim girişi vardır. Veri çıkışının kendisi (Y) ve deęili (Y':W) tümdevre üzerinde bulunur, ayrıca izin verme girişi G' (EN: enable) vardır. Şekil-7.6'da 8-1 veri seçici içeren 74151 ve 4-1 çift veri seçici içeren 74153 tümdevrelerinin bacak bağlantıları verilmiştir. Burada D veya I olarak adlandırılan girişler veri girişleridir. A, B, C girişleri veri seçme hatlarıdır.

VERİ SEÇİCİ / MULTİPLEXER UYGULAMALARI

Mantık fonksiyon üretici Veri seçici/multiplexer'ın bir uygulama alanı toplamların çarpımı şeklinde birleşimsel mantık fonksiyonlarının üretilmesidir. Bu şekilde kullanıldığında, aygıt ayırık geçitlerin yerini alabilir, kullanılacak tümdevrelerin sayısını azaltabilir ve tasarımdaki deęişikliklerin daha kolay yapılabilmesine olanak sağlar.



Şekil-7.6 Multiplexer tümdevreleri.

74151A sekiz-girişli veri seçici/multiplexer değişkenler veri seçme girişlerine bağlanır ve her bir veri girişi doğruluk tablosundaki o fonksiyon için gerekli mantık seviyesine kurulursa, üç değişkenli herhangi bir mantık ifadesini belirtmek için kullanılabilir. Örneğin, fonksiyon değişken birleşimi $\overline{A_2} A_1 \overline{A_0}$ olduğunda “1” ise, 2 girişi (010 ile seçilen) YÜKSEK seviyeye bağlanır. Veri seçme hatlarına bu özel değişken birleşimi uygulandığında, bu ‘YÜKSEK seviye’ çıkışa aktarılır.

Örnek 7.2

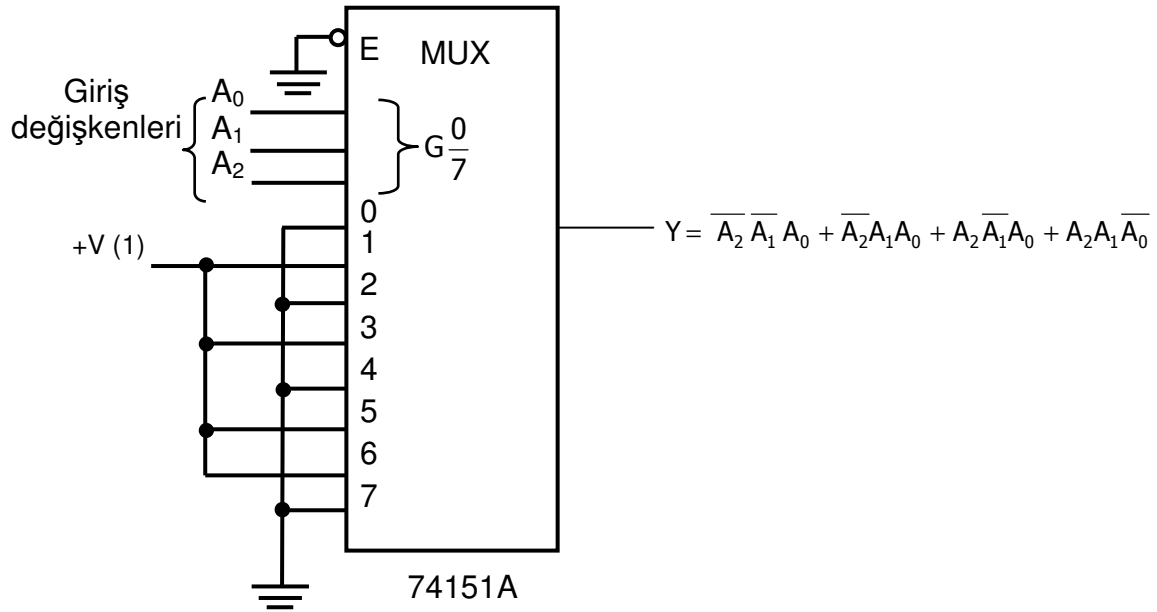
Tablo-7.2’de tanımlanan mantık fonksiyonunu 74151A multiplexer ile gerçekleştirin. Bu yöntemi ayrıntılı mantık geçitleriyle yapılan tasarım ile karşılaştırın.

Çözüm:

Doğruluk tablosundan Y’nin 001, 011, 101 ve 110 giriş değişken birleşimleri için “1” olduğunu görüyoruz. Diğer bütün birleşimler için Y “0”dır. Bu fonksiyonun veri seçici ile tasarlanabilmesi için, yukarıda belirtilen her bir birleşim tarafından seçilen veri girişinin YÜKSEK seviyeye bağlanması gerekir. Şekil-7.7’de gösterildiği gibi diğer bütün veri girişleri DÜŞÜK seviyeye bağlanmalıdır. Bu fonksiyonun mantık geçitleriyle tasarlanması için dört tane 3-girişli VE geçidi, bir tane 4-girişli VEYA geçidi ve üç eviriciye gerek vardır.

Girişler			Çıkış
A ₂	A ₁	A ₀	Y
0	0	0	0
0	0	1	1
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0

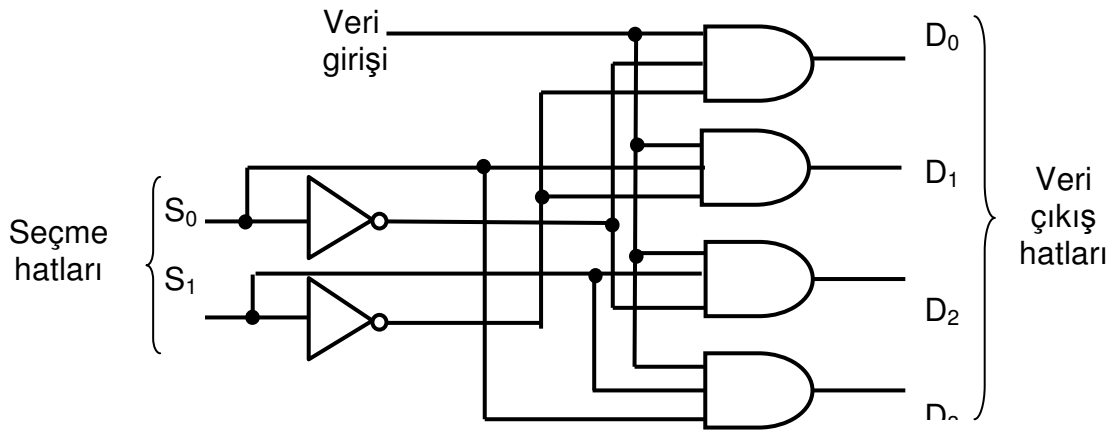
Tablo-7.2



Şekil-7.7 Üç-değişkenli fonksiyon üretici olarak veri seçici.

DEMÜLTİPLEXER

Demultiplexer (DMUX) temel olarak multiplexer'ın yaptığı işin tersini yapar. Bir hattan veriyi alır ve belirli sayıdaki çıkış hatlarından birine aktarır. Bu nedenle, bu ayyığta veri çoklayıcı yada veri dağıtıcı denilir.

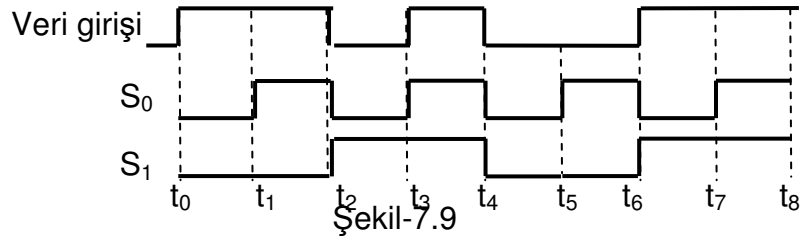


Şekil-7.8 1-hattan-4-hatta demultiplexer

Şekil-7.8'de 1-hattan-4-hatta veri dağıtıcı devresi görölmektedir. Veri-giriş hattı tüm VE geçitlerinin girişlerine gitmektedir. İki veri-seçme hattı her seferinde yalnız bir geçide izin vermektedir ve veri-giriş hattındaki veri seçilen geçitten ilgili veri-çıkış hattına aktarılır.

Örnek 7.3

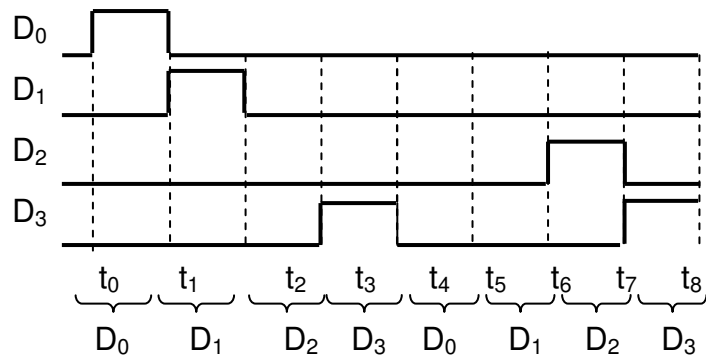
Seri veri-giriş dalga şekli ve veri-seçme girişleri Şekil-7.9'da verilmiştir. Şekil-7.8'deki demultiplexer için veri-çıkış dalga şeklini çizin



Şekil-7.9

Çözüm:

Seçme hatları ikilik sayma düzeninde gitmektedir, dolayısıyla birbirini izleyen her bir giriş Şekil-7-10'daki gibi sırayla D_0 , D_1 , D_2 , D_3 'e yönlendirilir.

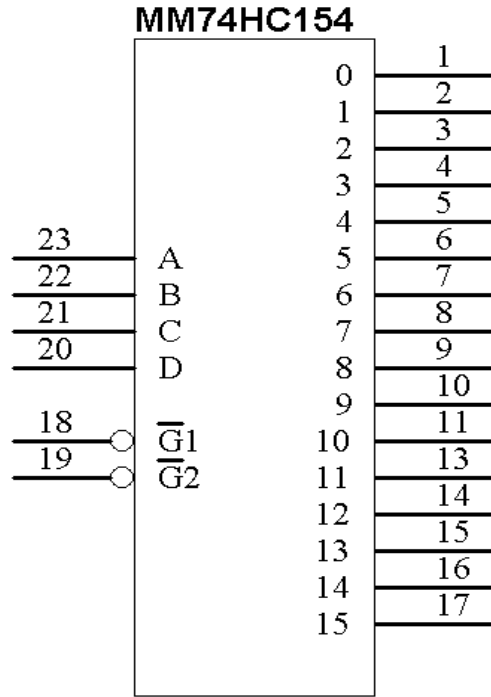


Şekil-7.10

74154 'ÜN DEMULTİPLEXER OLARAK KULLANILMASI

74154'ü 4-hattan-16-hatta kodçözücü olarak kullanmıştık. Bu aygıt ve diğer kodçözücüler demultiplexer uygulamalarında da kullanılabilir. Demultiplexer olarak kullanıldığında bu tümdevrenin mantık simgesi Şekil-7-11'de gösterilmektedir.

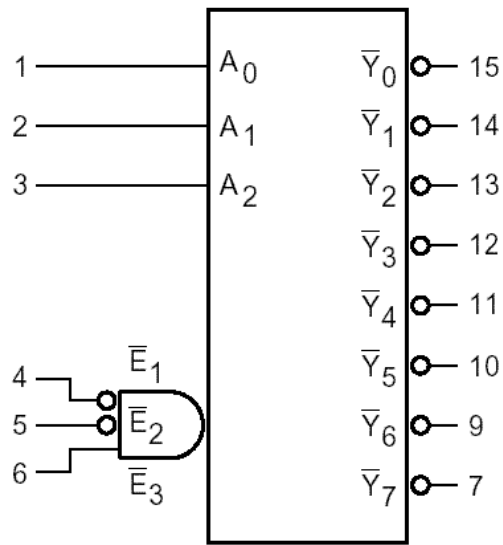
Demultiplexer uygulamalarında giriş hatları veri seçme hatları olarak kullanılır. İzin girişlerinden biri veri-giriş hattı olarak kullanılırken diğer izin girişi de DÜŞÜK tutularak negatif-VE geçidine izin verilir.



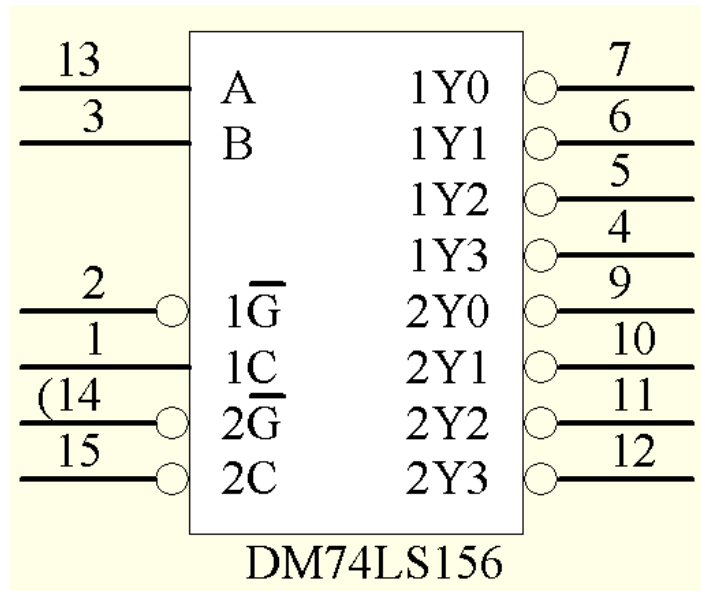
Şekil-7.11 74154'ün demultiplexer olarak kullanılması.

74156 VE 74138 'İN DEMULTİPLEXER OLARAK KULLANILMASI

Demultiplexer olarak 74156 ve 74138 tüm devresi kullanılmaktadır. Kodçözücü olarak da kullanılan bu tüm devrelerin mantık gösterimleri Şekil-7.12 ve 13'tedir. 74156 tümdevresi dual 4×1 demultiplexer tümdevresidir. 1C girişi birinci veri girişi, 2C ikinci veri girişi olarak, 1G girişi birinci izin girişi 2G girişi ikinci izin girişi olarak kullanılır. Denetim girişleri A_0 ve A_1 her iki demultiplexer için ortaktır. 74138 tümdevresi 1×8 demultiplexer olarak kullanılabilir, daha önce bu tümdevre kodçözücü olarak kullanılmıştı. Demultiplexer olarak kullanmak için E_3 girişi veri girişi olarak kullanılır, E_1 , E_2 girişleri izin girişleri olarak kullanılır. A, B, C girişleri denetim için kullanılır. Çıktılar etkin DÜŞÜKte çalışır.



Şekil-7.13 74138 demultiplexer tümdevresinin mantık gösterimi.



Şekil-7.12 74156 demultiplexer tümdevresinin mantık gösterimi

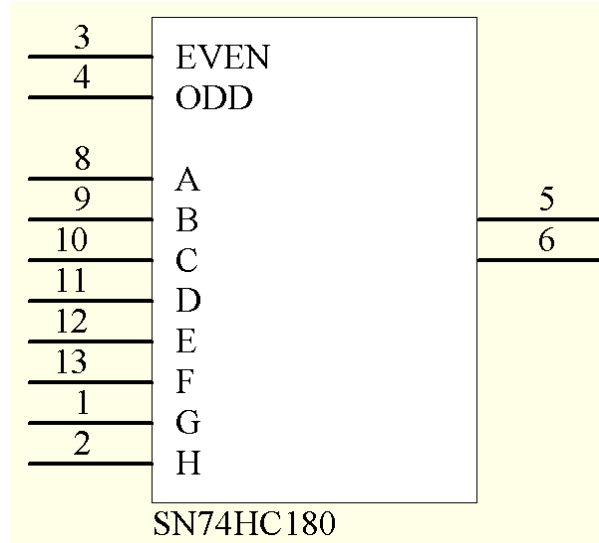
EŞLİK BİTİ ÜRETECİ

Eşlik biti üretici/denetleyicisi olarak 74180 tümdevresi kullanılacaktır. Bu tümdevrenin mantık gösterimi Şekil-7.14'te, doğruluk tablosu da Tablo-7.3'te verilmiştir. Girişteki tek ve çift girişleri daha önceki üretcin çıkışından gelen eşlik biti içindir. Bu tümdevreler veri iletişim sistemlerinde kullanılır.

Girişler			Çıkışlar	
Girişteki 1'lerin sayısı	Tek	Çift	Σ Tek	Σ Çift
Çift	1	0	1	0
Tek	1	0	0	1
Çift	0	1	0	1
Tek	0	1	1	0
X	1	1	0	0
X	0	0	1	1

Tablo-7.3 74180 tümdevresinin doğruluk tablosu

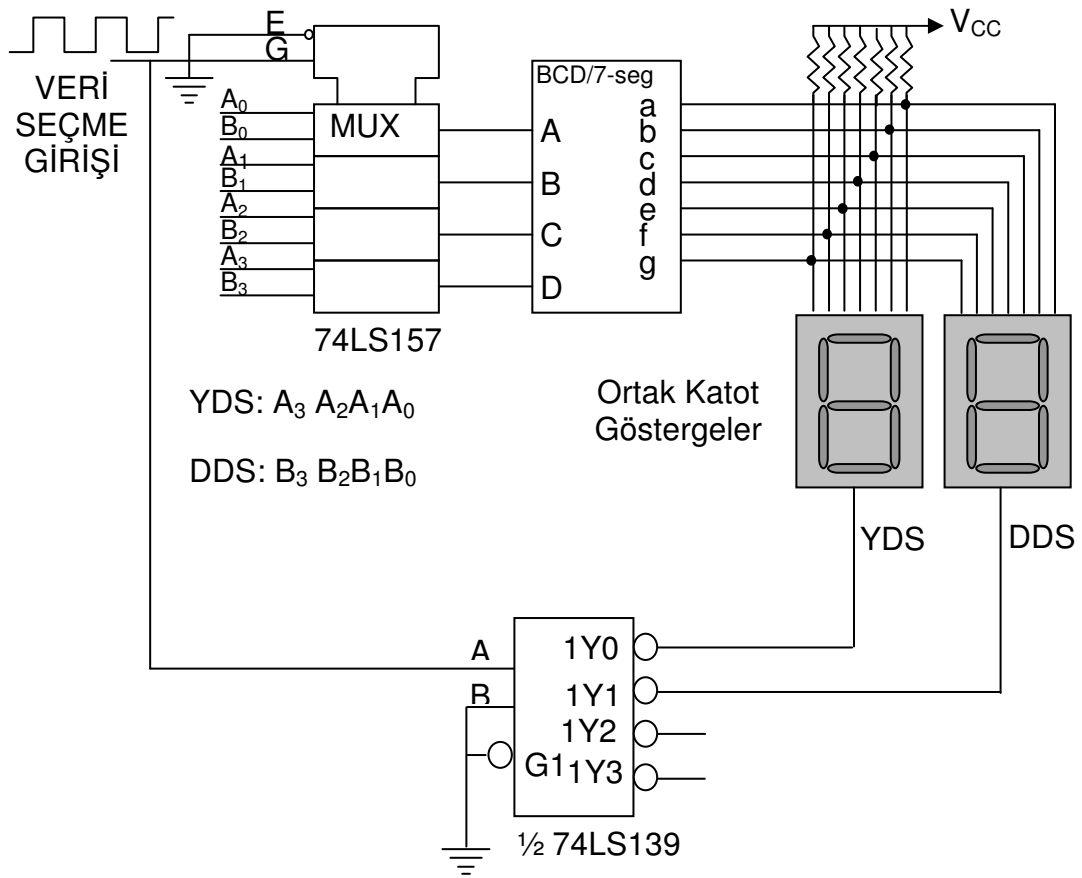
Eğer girişteki tek ve çift girişleri kullanılmıyorsa kullanılan tek veya çift eşlik bitine bağlı olarak uygun bağlantı yapılmalıdır. İletişim sistemlerinde verici kısmında bu girişlerin her ikisi de kullanılmaz, alıcı kısmında ise seçilen eşlik bitine bağlı olarak tek veya çift girişlerden biri kullanılır diğerine tabloya göre uygun bağlantı yapılır.



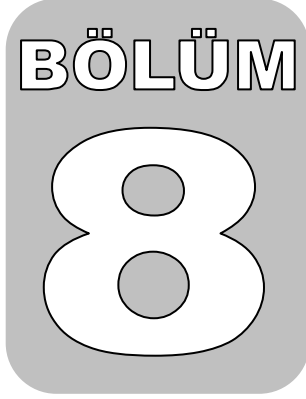
Şekil-7.14

MULTİPLEXER UYGULAMALARI

Şekil-7.15'de multiplexer ve dekoder uygulaması gösterilmiştir. Kodçözücü/sürücü tümdevreleri genellikle bu şekilde kullanılır. Bu devrede iki gösterge zaman paylaşımı olarak aynı Kodçözücü/sürücü tümdevre kullanılarak sürülmektedir. İki veya daha fazla gösterge bu yöntemle sürülebilir. 74139'un yarısı kullanılmıştır, sayıların katotlarını sürmek için kullanılır. Multiplexer ise gelen A verisini YDS'da B verisini DDS'da gösterir.



Şekil-7.5 Multiplexer uygulama devresi.



TUTUCULAR VE FLİP- FLOPLAR

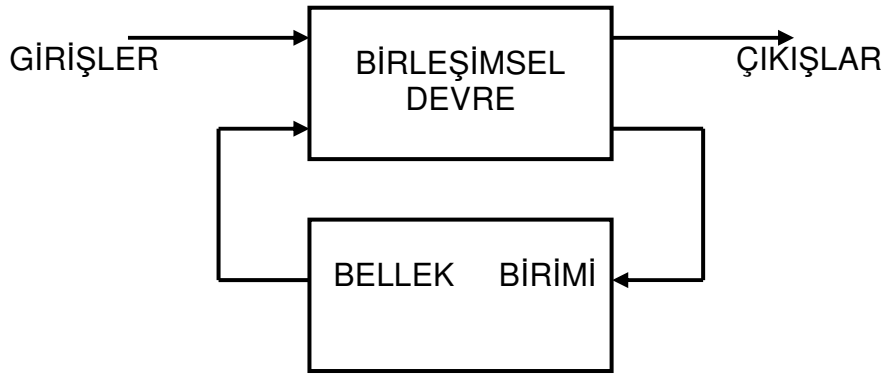
GİRİŞ

Bu bölümde *çift kararlı*, *tek kararlı* ve *kararsız* mantık aygıtları incelenecektir. Kararsızlar ile kararlılar arasındaki ayırım, durum değiştirme biçimleridir. Flip-Flop; sayaçlar, yazaçlar ve diğer ardıl denetim mantığı aygıtları için temel yapı ögesidir. Genelde *tek-atımlı* olarak ta adlandırılan aygıtlar, *tutucu* ve *Flip-Flop* olarak iki ana öbeğe ayrılır. Kararsız aygıtlarda SET (KUR) ve RESET (SİL) olarak adlandırılan iki ayrı karar durumu vardır ve bu durumlar sürekli olduğundan, kararsız aygıtlar saklama amaçlı kullanılabilirler. Flip-Flop ile tutucu tek kararlıdır, yalnızca bir tane kararlı durumu vardır ve bu nedenle tetiklendiğinde (yada etkinleştirildiğinde), süresi denetlenebilen tek bir vurum üretir. Kararsızda ise kararlı bir durum yoktur ve genellikle kendiliğinden dalga biçimi üreten bir titreşken olarak kullanılır. Sayısal sistemlerde bu tür titreşkenler saat adı verilen zamanlayıcı devrelerin yapımında kullanılırlar.

SIRALI ARDIL MANTIK

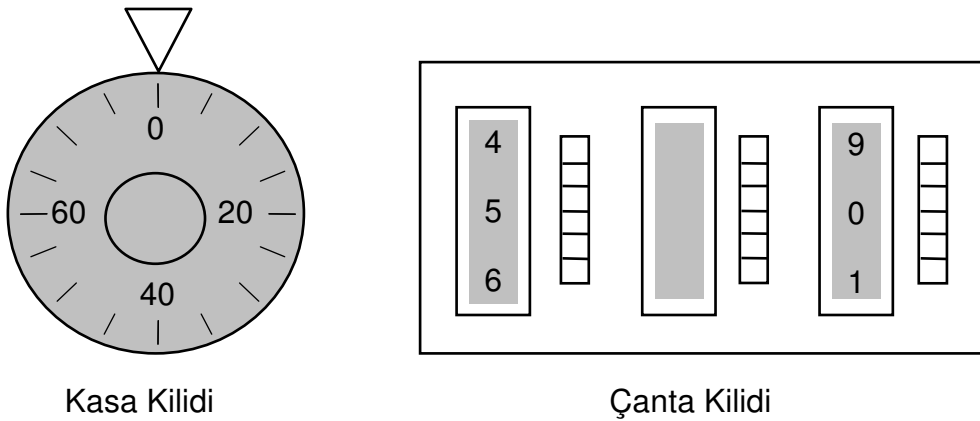
Birleşimsel mantıkta devrelerin çıkışları, o anki girişlere bağlıdır ve girişlerin değişmesi durumunda hemen etkilenirler. Tüm mantık geçitleri, toplayıcılar, sayaçlar, kodlayıcılar, kodçözücüler, yol (veri) seçiciler ve yol çoklayıcılar birleşimsel mantık devreleridir. Her sayısal dizgede birleşimsel mantık devreleri vardır ve uygulamadaki bu dizgelerin çoğunda bellek birimleri de kullanılmaktadır. İşte bu sistemleri yapılandıran mantık sıralı ardıl mantıktır. Ardıl mantıkta kullanılan temel yapı şekil-8.1'de görülmektedir.

Ardıl mantık ile birleşimsel mantığın en önemli ayırımı ardıl mantıkta dizge (sistem) tepkisinin o anki girişlerle birlikte, dizgenin önceki çıkışlarına da bağlı olmasıdır. Buna örnek olarak kasa ve çanta kilitlerini verelim. Çanta kilidinin şifresi 5-7-0 ise, kilidi açmak için 5-7-0 sayılarının kodlanması yeterlidir. Sayıların hangi sıra ile yazıldığı yada göstergenin bir önceki konumunun hiç bir etkisi olmaz.



Şekil-8.1 Sıralı ardıl mantık devrelerin temel yapısı.

Kasa kilidinin şifresi ise sözgelimi 34 sağa, 22 sola, 48 sola olsun. Kilidin açılabilmesi için bu sayıların doğru sıra ile (yani önce 34, sonra 22 ve en son 48) ve verilen yönlere doğru girilmesi gerekir. Buna göre kasa kilidi sıralı ardıl, çanta kilidi de birleşimsel mantık ile çalışır diyebiliriz.

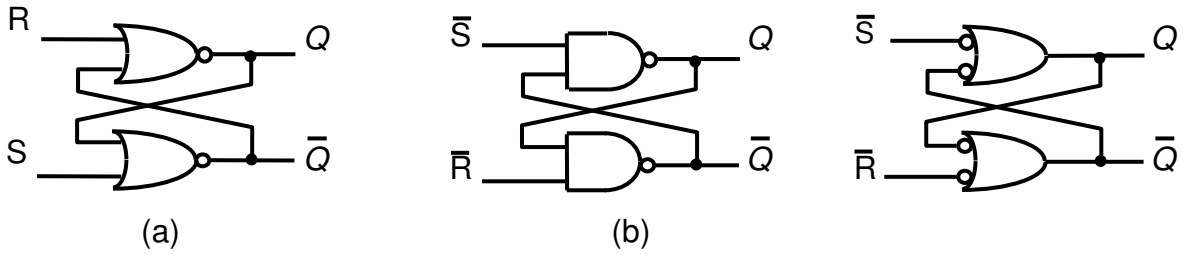


Şekil-8.2 Birleşimsel ve ardıl mantık için örnekler. Kasa kilidi ve çanta kilidi.

TUTUCU (LATCH) DEVRELERİ

Tutucu, Flip-Floptan ayrı olarak değerlendirilen çift kararlı bir veri saklama aygıtıdır. Şekil-1.3'te çapraz bağlı mantık negatif veya geçitleriyle oluşturulmuş S - R tutucu devresinin V_{ed} ve veya D geçitli devrelere nasıl dönüştüğü de gösterilmiştir. Her devrede görülen çapraz bağlantı en basit bellek yapısıdır ve bütün titreşkenlerde mutlaka bulunur. Flip-Floplarla tutucular arasındaki benzerlik, tutucunun da Flip-Flop gibi iki ayrı kararlı durumda bulunabilmesidir. Bununla birlikte tutuculara çıkışın kararlı kalması girişteki verilerin sürekliliğine bağlıdır. Tutucunun, her zaman birbirinin tersi olan iki çıkışı vardır. Q ve Q' . En temel

tutucu, $S - R$ türüdür. Burada S set (kur) R reset (sil) anlamındadır. Şekil-8.4'te $S - R$ tutucunun ETKİN-1 ve ETKİN-0 mantık için devreleri verilmiştir. veyaD geçitleriyle oluşturulmuş devrede olası durumlar şekil-8.5'te sırayla gösterilmiştir. $S - R$ tutucuda dikkat edilmesi gereken nokta, girişlerin ikisinin de aynı anda kesinlikle etkin olmaması gerektiğidir. Bu durum çıkışları aynı duruma gelmeye zorlar ve titreşmeye yol açar. Yasak giriş durumu sona erdiğinde tutucunun alacağı durum da önceden bilinemez. Şekil-8.5'te gösterilen olası durumlar tablo-8.1'de etkin-1 mantık için düzenlenmiştir.

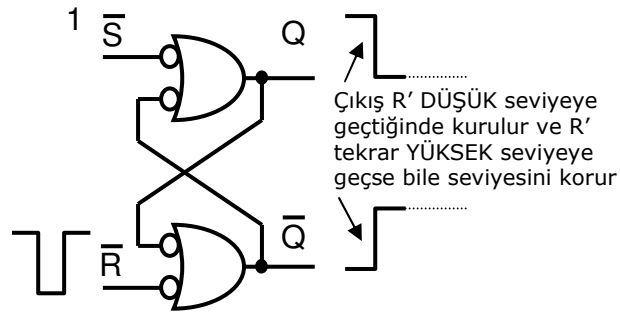


Şekil-8.3 a) VEYAD b) VED geçitleri ile kurulmuş $S - R$ tutucular. c) VED geçitlerinin TERS-VEYA olarak gösterimi.

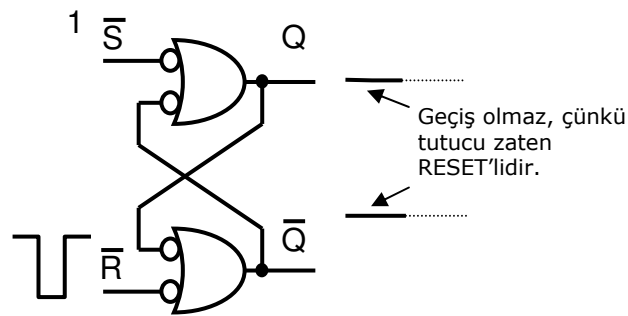
S	R	Q	Q	YORUM
0	0	D.Y	D.Y	Tutucu önceki durumunda kalır.
0	1	0	1	Tutucu sıfırlanır.
1	0	1	0	Tutucu kurulur.
1	1	1	1	YASAK DURUM

Tablo-8.1 ETKİN-1 girişli $S - R$ tutucu için doğruluk tablosu.

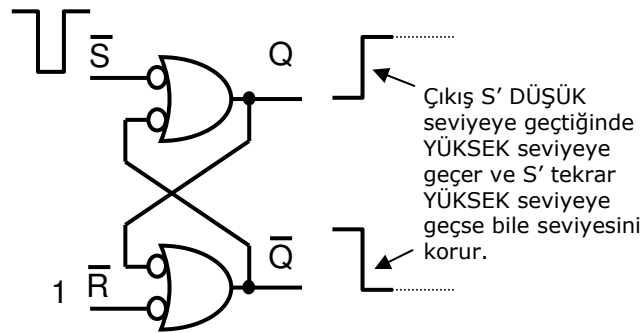
Tutucunun çalışması negatif veya değil geçidi temel alınarak tüm olası girişler için Şekil-8.4'te açıklanmıştır. $S - R$ tutucunun üç çalışma kipi ve bir de yasak çalışma durumu vardır. Birinci çalışma kipine KURma adı verilir. Eğer tutucunun Q çıkışı önceden DÜŞÜK seviyede ise S' girişine geçici de olsa bir DÜŞÜK seviye uygulanırsa (etkin yapılırsa) çıkış DÜŞÜK seviyeden YÜKSEK seviyeye geçiş yapar. Giriş tekrar YÜKSEK seviyeye geçse bile çıkış kurulu durumunu devam ettirir.



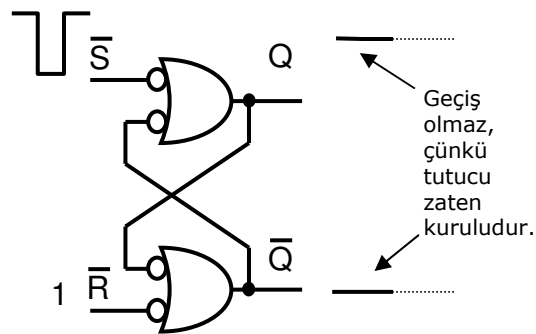
Tutucu KUR'ulu başlamıştır.



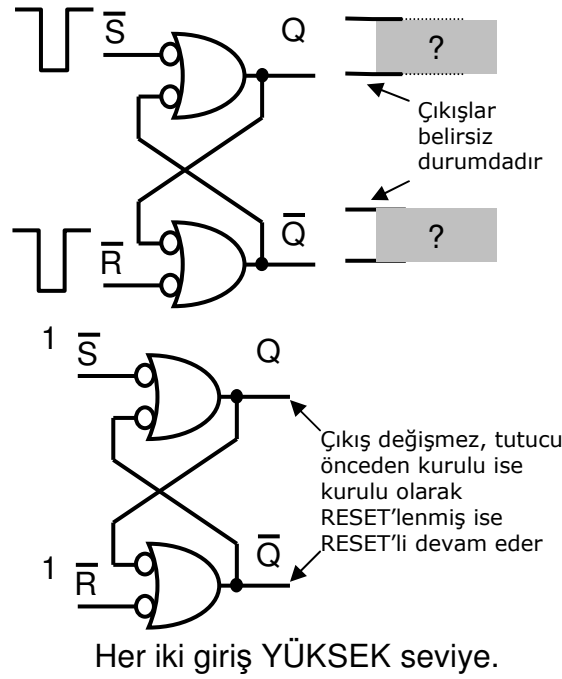
Tutucu durumunu korur.



Tutucu RESET konumunda başlamıştır.



Tutucu durumunu korur.

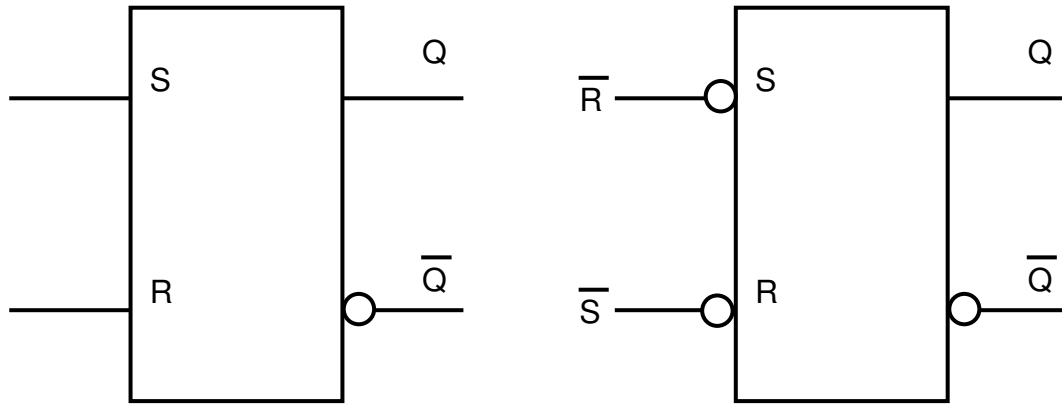


Şekil-8.4 S-R tutucunun RESET, KUR, DEĞİŞME YOK ve YASAK durumları

Q çıkışı zaten önceden kurulu ise S' girişi etkin olduğunda çıkış durum değiştirmez, kurulu olarak devam eder. Tutucunun çıkışı önceden kurulu ve S' girişi YÜKSEK seviyede iken R' girişine geçicilde olsa DÜŞÜK seviye bir vuru uygularsak çıkış RESETlenir. Bu işleme tutucunun silinmesi veya RESETlenmesi adı verilir. Eğer önceden RESETlenmiş tutucunun R' girişine yine bir DÜŞÜK seviye bir vuru uygulanırsa çıkış durum değiştirmez.

Tutucunun her iki girişine YÜKSEK seviye uygulanırsa çıkış durum değiştirmez. Bu duruma değişme yok veya bekleme durumu denir. Çıkışlar girişlerden herhangi biri etkin olana dek eski halini korur. Çıkış önceden ne olursa olsun her iki giriş aynı anda DÜŞÜK seviye birer vuru uygulanırsa çıkışın durumu tahmin edilemez. Tutucu bu turumda kararsız olarak çalışır. Kararsız çalışma durumu kullanılmaz, yasak durum adı verilir.

S-R tutucunun mantık simgesi Şekil-8.5'te DÜŞÜK seviye etkin girişli (a) ve YÜKSEK seviye etkin girişli (b) olarak gösterilmiştir.



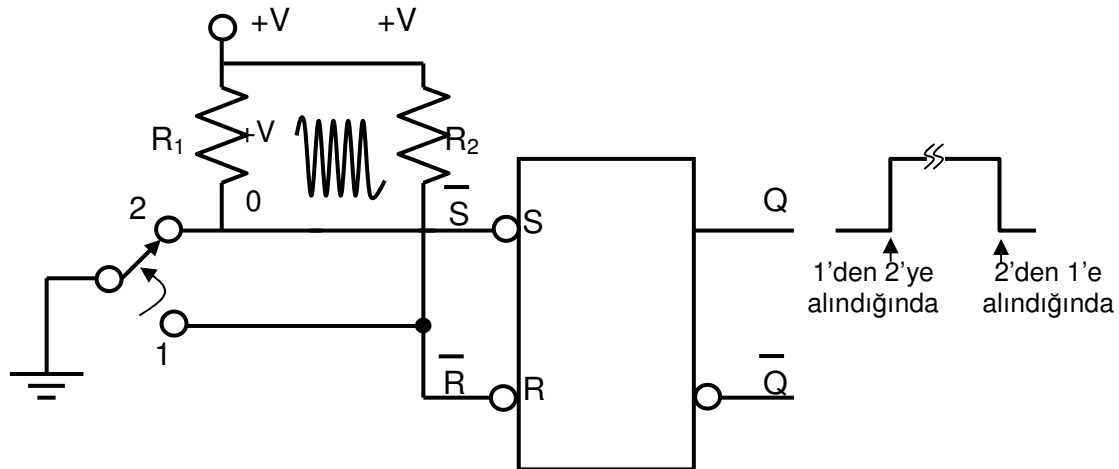
(a) YÜKSEK seviye etkin girişli S-R tutucu

(a) DÜŞÜK seviye etkin girişli S-R tutucu

Şekil-8.5 S-R tutucunun mantık simgesi.

S-R tutucunun uygulama alanlarından birisi olarak, kontak sıçraması giderme devresini verebiliriz. Mekanik anahtarlarda gözlenen ve kontağın hemen yerine oturmayıp kısa süreli titreşmesi nedeniyle oluşan kontak sıçraması (bouncing), anahtarın çıkışına S - R tutucu bağlanarak önlenir. Bu tür bir uygulama Şekil-8.6'da görülmektedir.

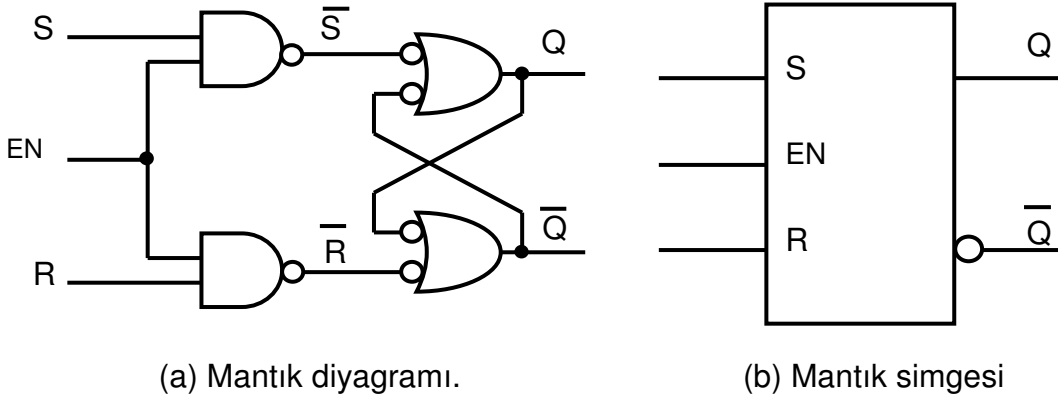
Anahtar 1 konumundan 2 konumuna alındığında S' girişi DÜŞÜK seviyeye gelir. Anahtar kontağı mekanik olduğu için çarparak geriye sıçrar bu anda S' girişi tekrar YÜKSEK seviyeye çıkar. Bu çarpma ve sıçrama süresi azalarak yüzlerce defa tekrar edebilir. Bu arada tutucunun R' girişine direnç üzerinden YÜKSEK seviye uygulanmaktadır. S' girişinde oluşan gürültü işareti şekilde gösterilmiştir. Bu gürültünün elektronik devreleri etkilemesini engellemek için tutucu kullanılır. S' girişinin ilk DÜŞÜK seviyeye düşmesiyle tutucu çıkışı kurulur. İkinci veya daha sonraki DÜŞÜK seviyeye düşüşlerde ise zaten çıkış kurulu olduğu için eski halini devam ettirecektir.



Şekil-8.6 S-R tutucunun ile kontak titreşiminin yok edilmesi.

GEÇİTLİ S-R TUTUCU

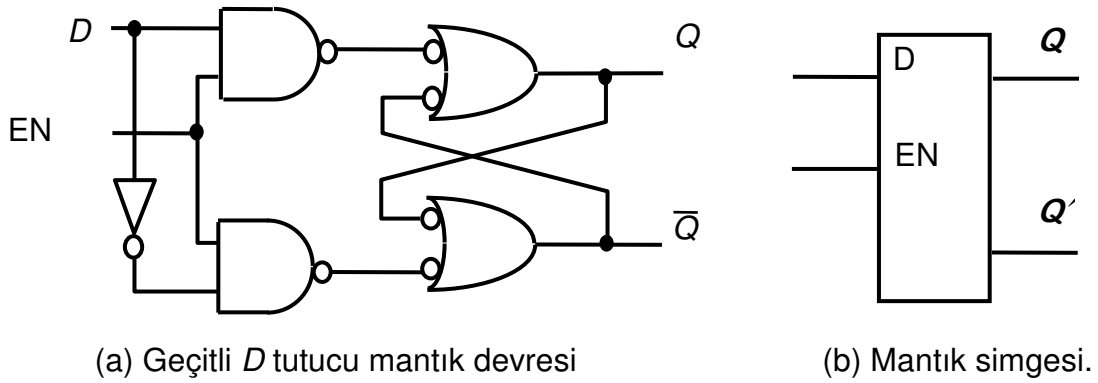
Tutucunun girişindeki veri değişimlerine sürekli olarak duyarlı olması hız gerektiren sistemlerde sorunlar yaratır. Bu nedenle tutucuya bir onay (İZİN-ENABLE) girişi eklenerek gerekli veri dizilerinin etkili olması sağlanır. Bu yeni bağlantıda S ve R girişleri yalnızca izin girişi '1' iken çıkışın durumu üzerinde etkili olabilirler. Şekil-1.7'de İZİN girişli S-R tutucunun mantık simgesi ve mantık diyagramı gösterilmiştir.



Şekil-8.7 Geçitli S-R tutucunun mantık simgesi.

GEÇİTLİ D TUTUCU

Sayısal dizgelerde veri saklamak yada bekletmek amacı ile en çok kullanılan tutucu, D (data - veri) türüdür. Bu tutucu, S-R tutucunun bir tümleyici ile yeniden düzenlenmiş biçimindedir. Tutulacak veri S girişine doğrudan ve R girişine de tümlenerek uygulanır. İzin girişi yüksek olduğunda D girişindeki bilgi ('1' yada '0') Q çıkışına taşınır ve tutulur.



Şekil-8.8 Geçitli D tutucu.

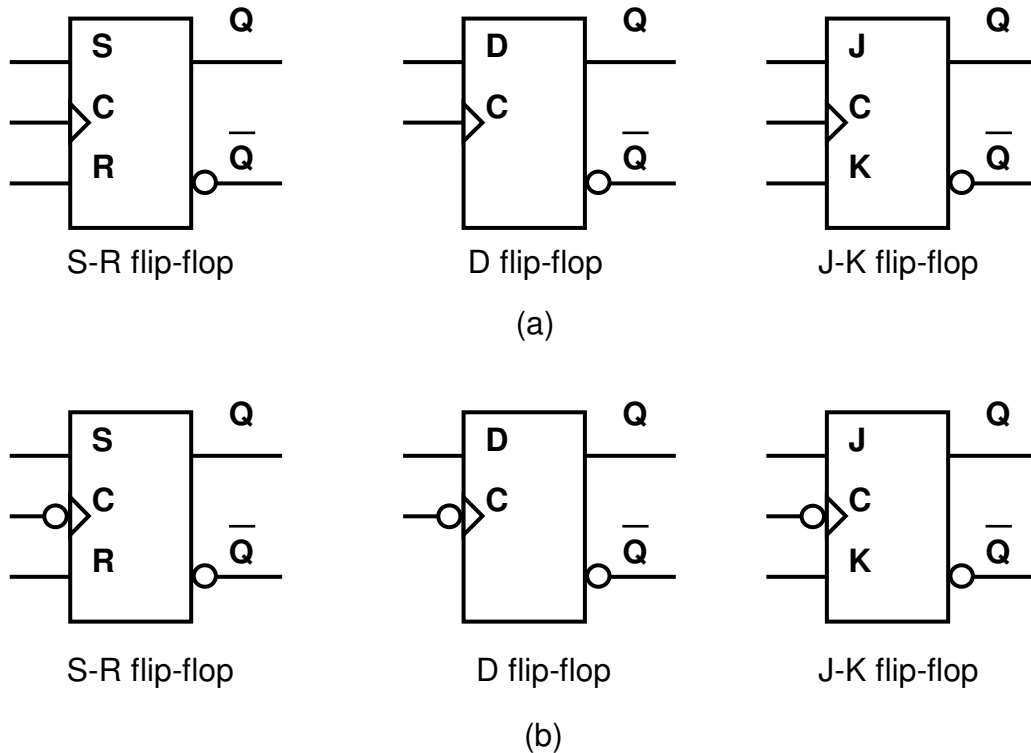
D	EN	Q	Q'	YORUM
0	1	0	1	Sıfırlanır
1	1	1	0	Kurulur
X	0	$Q_{(t-1)}$	$Q'_{(t-1)}$	Değişme yok

Tablo-8.2 D tutucunun doğruluk tablosu

74LS75, içinde dört ayrı **D** türü tutucu bulunan bir tümdevredir. (Bakınız ekteki veri yapraklarına) Bu tümdevrede tutucuların izin girişleri ikişer ikişer ortaklanmış ve **C** olarak adlandırılmıştır. İşte tam bu nokta tutucu ile Flip-Flop arasındaki geçiş noktasıdır.

KENAR TETİKLİ FLİP-FLOP'LAR

Flip-Floplar, çift kararlı ve eşzamanlı aygıtlardır. Eşzamanlı terimi burada, çıkışın yalnızca *saat - clock* denilen tetikleme işaretinin belirli bir noktasında durum değiştirdiğini anlatmaktadır. Özetle, çıkış değişimleri saatle eşzamanlı olarak oluşur.

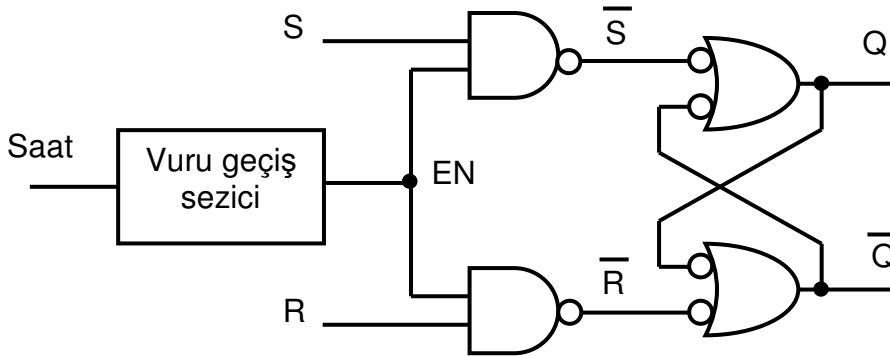


Şekil-8.9 (a) Yükselen, (b) düşen kenar tetiklemeli flip-flop'ların mantık simgeleri

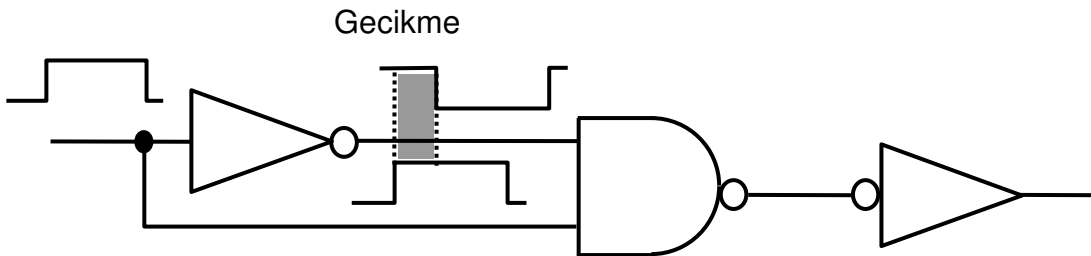
Geçitli tutucuda kullanılan izin işaretinin etkisi, işaret düşük yada yüksek iken süreklidir. Oysa sayısal dizgelerde veriler, yüzlerce Mbit/s hızlarda işlenebilmelidir ancak tutucunun vuru süresince değişimlere duyarlı kalması, hız için çok kısıtlayıcı bir etkidir. Buna çözüm olarak yalnızca saat işaretinin durum değiştirmesi sırasında girişler etkin kılınarak hız artışı sağlanır. Böylece elde edilen tutucu türevine Flip-Flop denir ve elektronik aygıtlarda en fazla kullanılan basit geçici bellek türüdür. Kenar tetikli Flip-Floplarda tetikleme, artı (yükselen) yada eksi (düşen) kenarda yapılabilir. Girişler yalnızca bu geçişler sırasında çıkışlar üzerinde etkili olur. Şekil-8.9'da yaygın olarak kullanılan Flip-Flop simgeleri verilmiştir. Saat girişindeki küçük üçgen Flip-Flopun kenar tetikli olduğunu gösterir. Düşen kenar tetikleme ise yine saat girişindeki yuvarlak ile belirtilmiştir.

S-R Flip-Flop

Şekil-8.10'da basit bir yükselen kenar tetikli **S-R** flip-flop devresi verilmiştir. **S-R** tutucudan ayrı olarak burada bir de geçiş sezici vardır. Bu devre en basit olarak bir DEĞİL geçidinin gecikmesinden yararlanılarak gerçekleştirilebilir. Bu birkaç nano saniyelik gecikmeden üretilen kısa süreli vuru, sürücü devresine uygulanarak saat işaretinin yükselmesi sırasında çok kısa süreli bir İZİN işareti olarak kullanılır.



(a) Basitleştirilmiş pozitif kenar tetikli Mantık S-R Flip-Flop diyagramı.

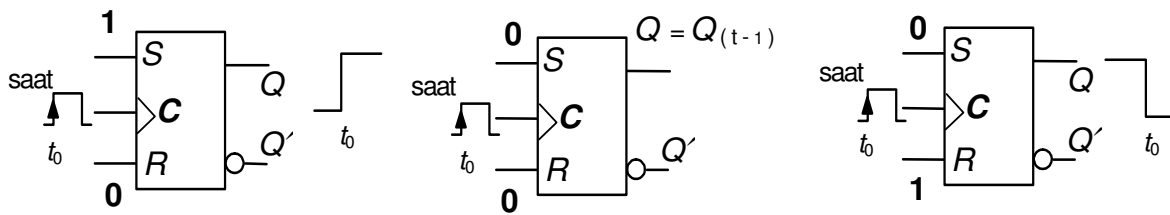


b) Vuru geçiş sezici devresi (Yükselen kenar için).

Şekil-8.10 Kenar tetikleme ve S-R Flip-Flopun yapısı.

S	R	C	Q	Q'	YORUM
0	0	X	$Q_{(t-1)}$	$Q'_{(t-1)}$	Değişme yok
0	1	↑	0	1	Sıfırlanır
1	0	↑	1	0	Kurulur
1	1	↑	?	?	YASAK

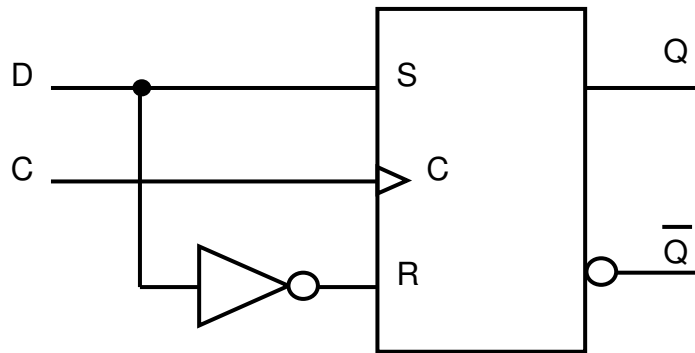
Tablo-8.3 Yükselen kenar tetikli S –R tutucu için doğruluk cetveli.



Şekil-8.11 S-R Flip-Flopun çalışma durumları.

D FLİP-FLOP

D flip-flop tek bir veri biti (1 yada 0) saklanacağında kullanılır. **S-R** flip-flopa bir DEĞİL geçidi eklenerek **D** flip-flop elde edilir. **D** ucuna '1' yada '0' uygulandıktan sonra gelen ilk tetikleyici geçişinde bu veri çıkışa ulaşır ve orada tutulur.



Şekil-8.12 D türü flip-flop için S-R flip-flopta yapılan değişiklik.

<i>D</i>	<i>C</i>	<i>Q</i>	<i>Q'</i>	YORUM
1	↑	1	0	Kurulur (1 yüklenir)
0	↑	0	1	Temizlenir (0 yüklenir)

↑ = saatin düşük'ten yüksek'e geçişi

Tablo-8.4 Yükselen kenar tetikli D flip-flop için durum cetveli.

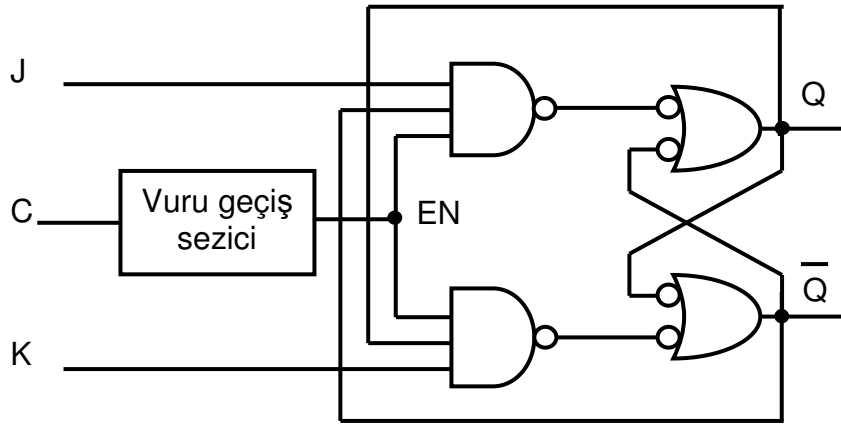
J-K FLİP-FLOP

J-K flip-flop piyasada belki de en çok kullanılan flip-floptur. **J-K** Flip-Flopun çalışması kur, sil ve değişme yok durumlarında **S-R** flip-flop ile aynıdır. Aralarındaki ayırım **J-K** flip-floplarda yasak durum bulunmamasıdır. Her iki girişin de '1' olduğu durum tanımlıdır. Şekil-8.13'te yükselen kenar tetikli **J-K** Flip-Flopun iç mantığı gösterilmiştir. Burada **S-R** flip-floplardan değişik olarak **Q** çıkışı **K** girişindeki geçide, **Q'** çıkışı da **J** girişindeki geçide bağlanmıştır.

<i>J</i>	<i>K</i>	<i>C</i>	<i>Q</i>	<i>Q'</i>	YORUM
0	0	↑	$Q_{(t-1)}$	$Q'_{(t-1)}$	Değişme yok
0	1	↑	0	1	Sıfırlanır
1	0	↑	1	0	Kurulur
1	1	↑	$Q'_{(t-1)}$	$Q_{(t-1)}$	Tümleyen

Tablo-1.5 Yükselen kenar tetikli J-K Flip-Flopun doğruluk tablosu.

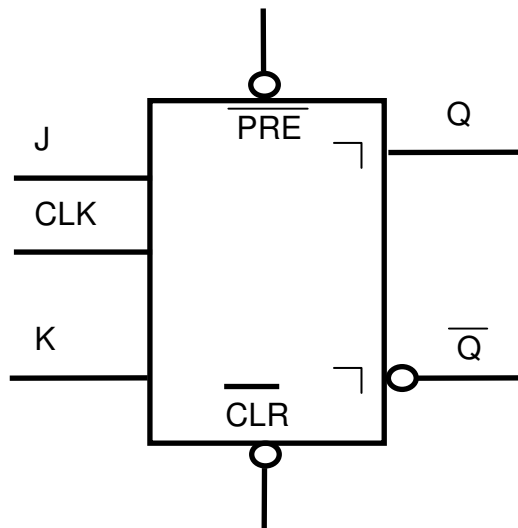
J-K Flip-Flop D Flip-Flop gibi kenar tetikli olabileceği gibi vuru tetikli de olabilir. Vuru tetikli J-K Flip-Flopa master-slave flip-flop adı da verilir. Yükselen kenarda veri master Flip-Flopun çıkışına, düşen kenarda ise veri slave Flip-Flopun çıkışına aktarılır. Böylece vuru tamamlandığında girişteki veri çıkışa aktarılmış olur. Vuru tetikli olarak adlandırılmasının sebebi de girişteki verinin çıkışa aktarılması için vurunun tamamının gerekli olmasıdır.



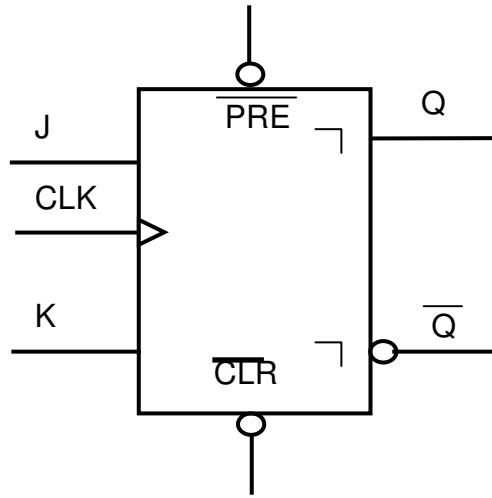
Şekil-8.13 Yükselen kenar tetikli J-K flip-flopun mantık diyagramı.

J	K	CLK	Q	Q'
0	0	Pozitif Vuru	Q_0	Q'_0
0	1	Pozitif Vuru	0	1
1	0	Pozitif Vuru	1	0
1	1	Pozitif Vuru	Q'_0	Q_0

Tablo-8.6 Master-Slave J-K Flip-Flopun doğruluk tablosu.



Şekil-8.14 Master slave Flip-Flopun mantık simgesi.



J	K	CLK	Q	Q ₀
0	0	Pozitif Vuru	Q ₀	Q' ₀
0	1	Pozitif Vuru	0	1
1	0	Pozitif Vuru	1	0
1	1	Pozitif Vuru	Q' ₀	Q ₀

Tablo-8.7 Veri kilitlemeli J-K Flip-Flopun doğruluk tablosu

Bu Flip-Flopların sakıncası ise veri master çıkışına aktarıldıktan sonra giriş değiştiğinde master çıkışı değişir ve çıkışa değişen veri aktarılır. Özetle veri çıkışa aktarılan kadar (vuru tamamlana kadar) giriş sabit kalmalıdır. Bu sakıncayı ortadan kaldıran Flip-Floplara ise veri kilitlemeli (data lockout) Flip-Flop denir. Bu Flip-Floplarda girişteki veri vuru tamamlanmadan değişse bile çıkış bu durumdan etkilenmez. Her iki Flip-Flopun doğruluk tabloları ve mantık sembolleri şekil-8.14 ve 15'te gösterilmiştir.

T FLİP-FLOP

J-K Flip-Flopun **J** ve **K** girişleri birbirine bağlanarak elde edilen flip-flop türüdür. **J = K = 0** iken saat işareti uygulansa da çıkışlar durum değiştirmez. **J = K = 1** olduğunda ise her tetikleme de çıkışlar bir önceki durumlarının tümleyene dönüşürler. Flip-flop adını bu özelliğinden (toggle) alır. T flip-flopun doğruluk tablosu tablo-1.8'de mantık simgesi ise şekil-8.18'de gösterilmiştir.

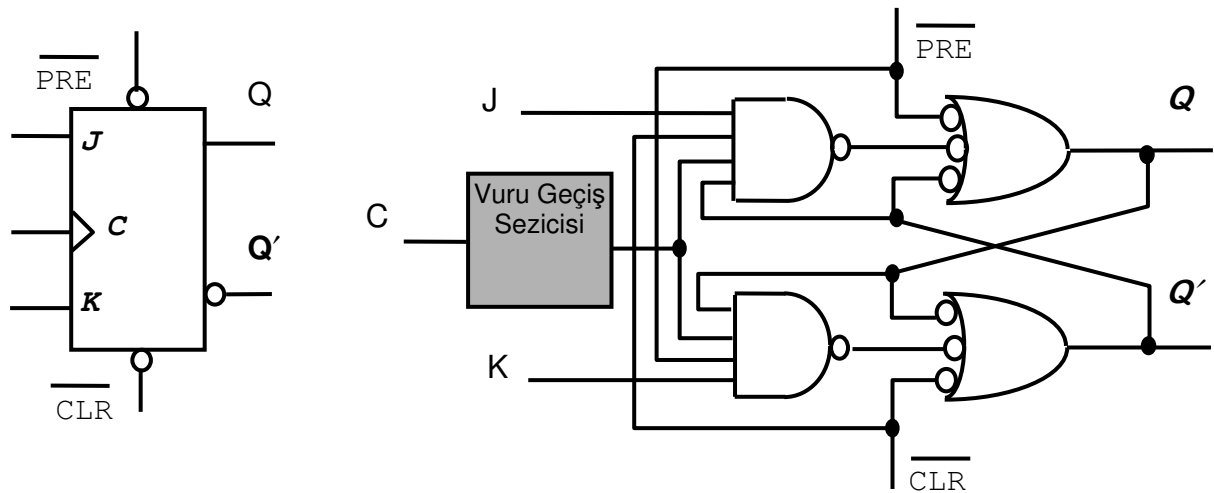
T	C	Q	Q'	Yorum
0	↑	$Q_{(t-1)}$	$Q'_{(t-1)}$	Değişme Yok
1	↑	$Q'_{(t-1)}$	$Q_{(t-1)}$	Tümleyen

Tablo-8.6 Yükselen kenar tetikli t flip-flopun doğruluk tablosu.

ASENKRON GİRİŞLER

Flip-Flopun D , $J-K$, $S-R$ ve T girişlerindeki veriler, Flip-Flopun çıkışına yalnızca saat işaretinin tetikleyen kenarında aktarıldıklarından, başka deyişle bu veriler saatle eşzamanlı (senkron) işlendiklerinden bu girişlere eşzamanlı girişler denir. Çoğu Flip-Flop tümdevresinde ayrıca eşzamansız (asenkron) girişler de bulunur. Bu uçlar öndeğerle (preset -PRE) ve temizle (clear-CLR), bazı üreticiler tarafından ise doğrudan kur S_D ve doğrudan sil R_D olarak adlandırılırlar. Bu girişler kullanılarak Flip-Flop istenilen anda kurulabilir yada sıfırlanabilir. Genellikle ETKİN-0 olarak eklenen bu girişler eşzamanlı çalışmada yüksek mantıkta tutulmalıdır.

74LS74A birbirinin aynısı iki ayrı D flip-flop içeren TTL bir tümdevredir. Flip-Flopun yükselen kenar tetiklidir ve asenkron girişleri ETKİN-0'dır. 74LS76A içinde de iki $J-K$ flip-flop vardır. Bu flip-flopun düşen kenar tetiklidir ve eşzamansız denetim girişleri de vardır. Her iki flip-flopun mantık diyagramları ve simgelerini EK-A'daki veri yapılarından bakabilirsiniz.



Şekil-8.17 asenkron girişli J-K flip-flopun mantık simgesi ve diyagramı.

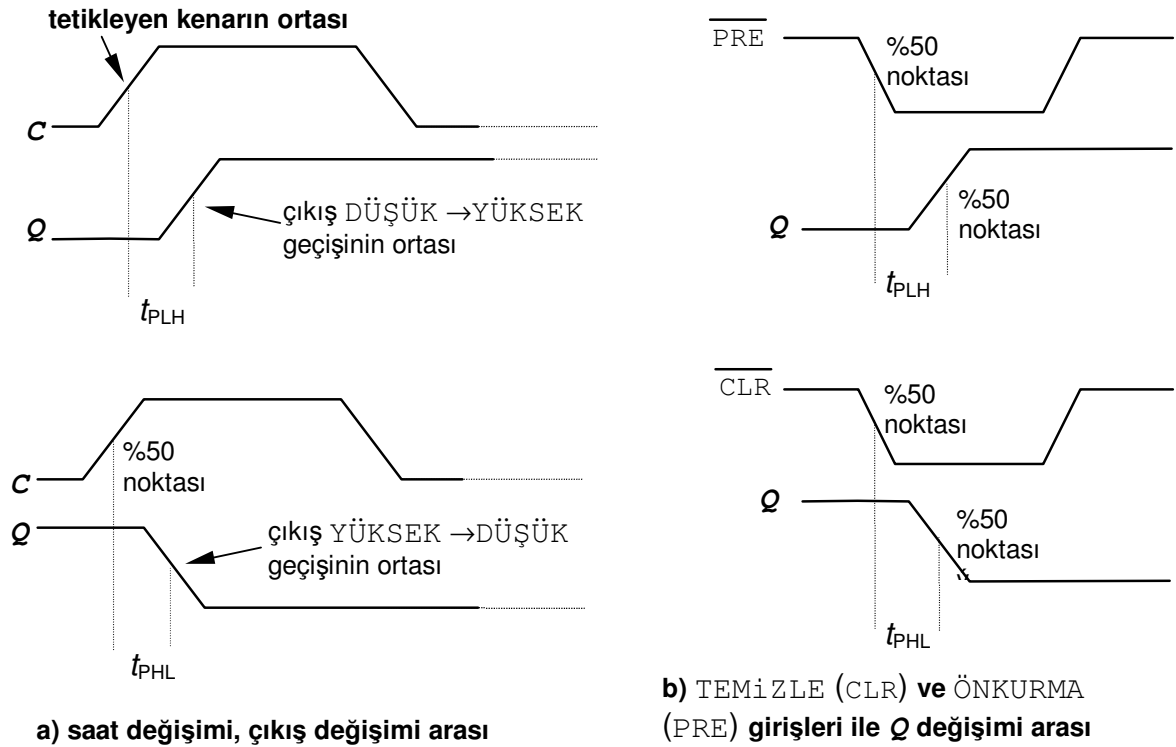
FLİP-FLOPLARIN BAŞARIM ÖZELLİKLERİ

Flip-flopların başarım, sınırlama ve doğru çalışma için gereksinimleri, bazı değişkenler ile belirtilir. Bu teknik özellikler, tüm mantık ailelerindeki (*LS, S, AS, TTL, CMOS*) flip-floplar için geçerlidir.

YAYILMA GECİKMESİ SÜRESİ

Yayılma gecikmesi süresi giriş işaretinin uygulanmasından sonra çıkışın oluşabilmesi için gerekli zaman aralığıdır. Bir ikilinin çalışmasında önemli olan yayılma gecikmesi tanımları aşağıdaki gibidir:

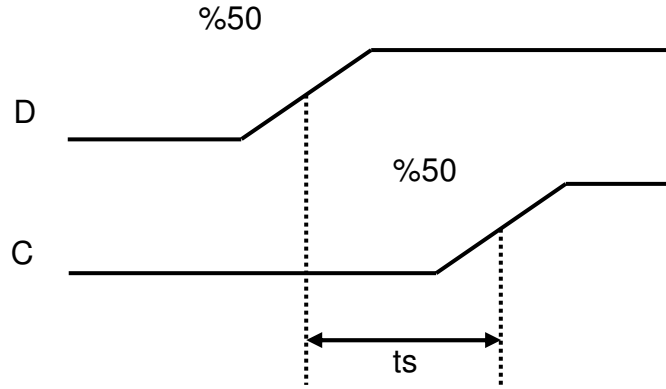
1. Saat darbesinin tetikleme kenarından çıkışın *DÜŞÜK'ten-YÜKSEK'e* geçişine kadar ölçülen t_{PLH} yayılma gecikmesi.
2. Saat darbesinin tetikleme kenarından çıkışın *YÜKSEK'ten-DÜŞÜK'e* geçişine kadar ölçülen t_{PHL} yayılma gecikmesi.
3. *Preset (önkurma)* girişinden çıkışın *DÜŞÜK'ten-YÜKSEK'e* geçişine kadar ölçülen t_{PLH} yayılma gecikmesi. Bu gecikme şekilde bir aktif-DÜŞÜK *preset* için gösterilmiştir.
4. *Clear (Sil)* girişinden çıkışın *YÜKSEK'ten-DÜŞÜK'e* geçişine kadar ölçülen t_{PHL} yayılma gecikmesi. Bu gecikme şekilde bir aktif-DÜŞÜK *sil* için gösterilmiştir.



Şekil-8.18 Yayılma gecikmeleri.

KURULMA (SET-UP) SÜRESİ

Saat darbesinin tetikleme kenarından önce, seviyelerin ikiliye güvenilir olarak ulaşması için J;K veya S;R veya D girişlerindeki mantık seviyelerinin sabit kalması için gerekli minimum zaman aralığına *kurulma süresi* (t_s) denir.

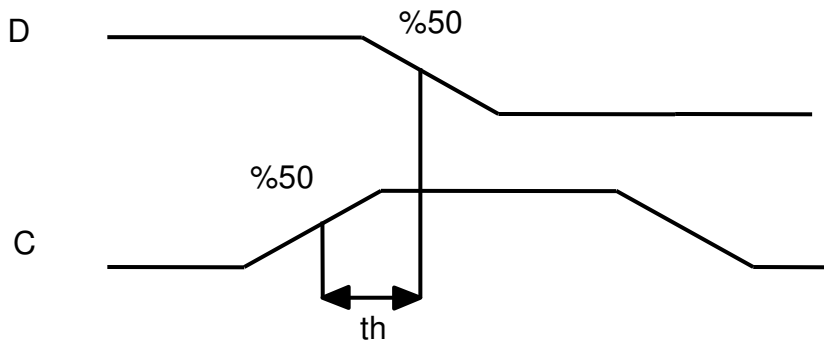


Şekil-8.19 Kurulma süresi.

Saat darbesinin tetikleme kenarından önce veri girişlerinin gerekli mantık seviyesinde kalmalarını sağlamak için gerekli süre.

TUTMA (HOLD) SÜRESİ:

Seviyelerin ikiliye güvenilir olarak ulaşması için saat darbesinin tetikleme kenarından sonra girişlerin mantık seviyelerini koruması için gerekli minimum zaman aralığına *tutma süresi* (t_h) denir.



Şekil-8.20 Kurulma süresi.

MAKSİMUM SAAT FREKANSI

Maksimum saat frekansı (f_{max}) bir ikilinin güvenilir olarak tetiklenebileceği maksimum hızı ifade eder. Maksimum değer üzerindeki saat frekanslarında, ikili yeteri kadar hızlı yanıt (tepki) verememekte ve çalışması bozulmaktadır.

VURU GENİŞLİKLERİ

Güvenilir çalışma için minimum vuru genişlikleri (t_w) saat, önkurma (preset) ve silme (clear) girişleri için üretici tarafından tanımlanmaktadır. Tipik olarak saat minimum yüksek süresi ve minimum düşük süresi ile tanımlanır.



Şekil-8.8

GÜÇ TÜKETİMİ

Herhangi bir sayısal devrenin güç tüketimi aygıtın tüm güç tüketimidir. +5V'luk d.c. kaynakla beslenen ve 50mA akım çeken bir ikilinin güç tüketimi,

$$P = V_{CC} \times I_{CC} = 5V \times 50mA = 250mW$$

DC kaynağın kapasitesi söz konusu olduğunda pek çok uygulamalarda güç tüketimi önem taşımaktadır. Örnek olarak 10 ikiliden oluşan bir sayısal sistemde her ikili 250mW güç tüketir toplam güç gereksinimi

$$P_{TOP} = 10 \times 250mW = 2.5W$$

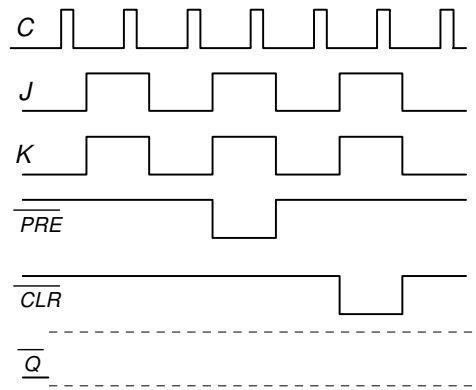
Bu bize d.c. kaynak için gerekli çıkış kapasitesini verir. İkili +5V d.c. ile beslenmekte ise, kaynağın sağlaması gereken akım miktarı:

$$I = \frac{250W}{5V} = 0.5A \text{ olur.}$$

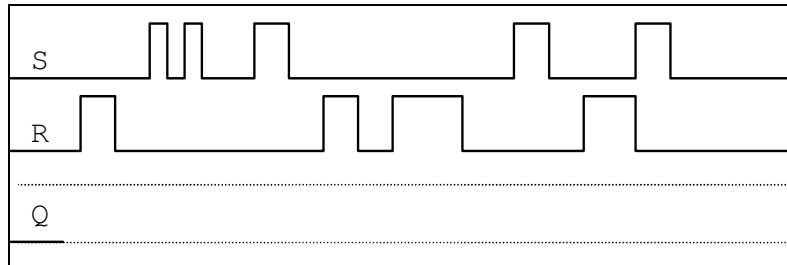
En az 0.5A'lık akım verebilen +5V'luk bir d.c. kaynak gerekmektedir.

SORULAR

1. Birleşimsel ve sıralı ardıl mantık arasındaki farkı açıklayın.
2. Flip-Flop ve tutucu arasındaki fark nedir, açıklayın.
3. Flip-Flop uygulamalarında, yayılma gecikmesi, değiştirme süresi, tutma süresi, en yüksek çalışma sıklığı kavramlarını açıklayın.
4. İşaret genişliği ve güç tüketimi değerlerinin önemlerini nedir? Açıklayın.
5. Yandaki J-K ikilinin girişlerine uygulanan sinyallere göre Q ve Q' çıkışında oluşan sinyalleri çizin.



6. Etkin-1 giriřli bir S-R tutucunun giriřlerine yanda grlen dalga biçimleri uygulanmıřtır. Çıkıř dalga biçimini verilen bořluęa çizin.



BÖLÜM 9

555 ZAMANLAYICI VE ÜÇ DURUMLU TAMPON

GİRİŞ

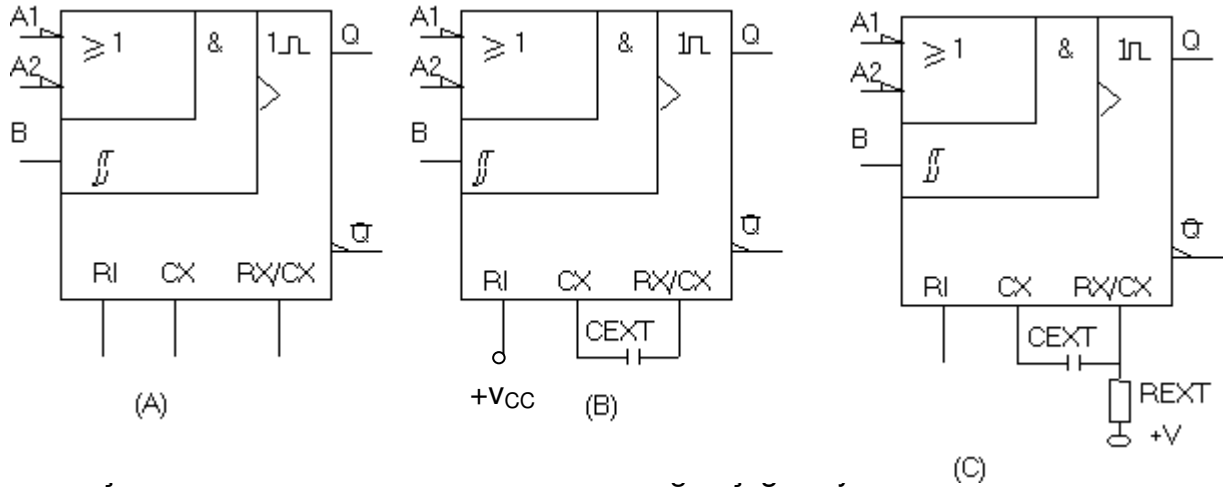
Bu bölümde tek atımlılar, zamanlayıcılar ve üç durumlu geçitler gibi sayısal sistemlerde yardımcı görev üstlenen elemanlar incelenecektir.

TEK ATIMLI VURU ÜRETEÇLERİ

Tek atımlılar tek kararlı çalışan bir multivibratördür (titreşim üretici). Eleman bir kararlı duruma sahiptir. Girişine bir uyarma işareti geldiğinde kararsız duruma geçerek önceden belirlenmiş süre bu kararsız durumda kalır ve tekrar kararlı durumuna geri döner. Yeri uyarma gelmesini bekler. Gelmediği sürece kararlı durumda kalır. Şekil-9.1'de 74121 yeniden tetiklenemez tek atımlının sembol ve bağlantıları gösterilmiştir.

- (A) Dış eleman yok ($t_w=30ns$)
- (B) Dahili R ve C_{EXT} ($R=2K$) ($T_w=0.7R \cdot C_{EXT}$)
- (C) R_{EXT} ve C_{EXT} ($t_w=0.7R_{EXT} \cdot C_{EXT}$)

Çıkış vuru genişliği dış bileşenlerin (elemanların) seçimi ile değiştirilebilir. 74121 tümdevresinde dış zamanlama elemanları kullanılmadığında minimum 30ns'lik bir vuru genişliği elde edilmektedir. Vuru genişliği dış elemanlar ile 40ns ile 28s arasında bir değere ayarlanabilmektedir.



ÖRNEK 9.1:

Bir uygulamada yaklaşık 1s'lik darbe genişliğine sahip bir tek atımlı kullanılacaktır. 74121'i kullanarak bağlantıları ve bileşen değerlerini gösteriniz.

ÇÖZÜM:

$R_{EXT}=10M\Omega$ seçelim ve gerekli kondansatörün kapasite değerini hesaplayalım.

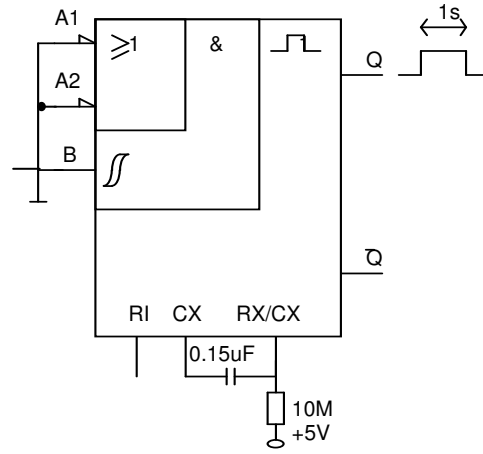
$$t_w = 0.7 \cdot R_{EXT} \cdot C_{EXT}$$

$$C_{EXT} = \frac{t_w}{0.7 \cdot R_{EXT}}$$

$$C_{EXT} = \frac{1s}{(0.7)(10 \cdot 10^6)} = 0.143 \times 10^{-6} F$$

$$C_{EXT} = 0.143 \mu F$$

Standart $0.15\mu F$ 'lık kondansatör kullanabiliriz. Uygun bağlantılar aşağıdaki gibidir.



ÖRNEK 9.2:

74122'ye bağlandığında $1\mu s$ 'lik bir darbe genişliği üretecek R_{EXT} ve C_{EXT} değerlerini bulun.

ÇÖZÜM:

74122 için $K=0.32$ olduğundan:

$$t_w = K \cdot R_{EXT} \cdot C_{EXT} \left(1 + \frac{0.7}{R_{EXT}}\right) \quad C_{EXT} = 1.5nf$$

$$t_w = K \cdot R_{EXT} C_{EXT} + 0.7 \left(\frac{K \cdot R_{EXT} \cdot C_{EXT}}{R_{EXT}}\right)$$

$$t_w = K \cdot R_{EXT} C_{EXT} + 0.7 \cdot K \cdot C_{EXT}$$

$$R_{EXT} = \frac{t_w - 0.7 \cdot K \cdot C_{EXT}}{K \cdot C_{EXT}} = \frac{t_w}{K \cdot C_{EXT}} - 0.7$$

$$R_{EXT} = \frac{1 \times 10^{-6} s}{(0.32)(1500 \cdot 10^{-12} F)} - 0.7 = 2.1K$$

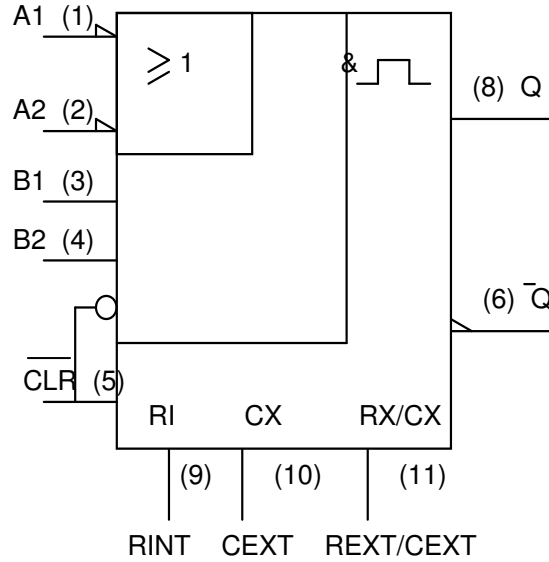
KARARSIZ MULTIVBRATÖRLER VE ZAMANLAYICILAR

İki durum arasında gidip gelen hiçbir zaman kararlı olarak bir duruma geçmeyen multivibratör çeşididir. Tetikleme ve zamanlama gerektiren sistemlerde osilatör olarak kullanılır.

\int simgesi Schmitt tetikleyici girişini belirtir. Bu çeşit giriş histerezis üreten bir özel eşik devresine sahiptir. Giriş gerilimi kritik giriş seviyesi etrafında dolaşırken durumlar arasında hatalı anahtarlamayı önleyen bir özelliktir. Giriş çok yavaş değişirken bile güvenilir tetikleme yapılmasına olanak sağlar.

Clear (temizle) girişine sahip bir yeniden tetiklenebilir tek atımlı tümdevredir. Harici R ve C girişlerine sahip, A1, A2, B1, B2 geçitli tetikleme girişleridir.

$$t_w = K \cdot R_{EXT} \cdot C_{EXT} \left(1 + \frac{0.7}{R_{EXT}}\right)$$



Şekil-9.2 74122'nin ANSI / IEEE standardı mantık sembolü

K; üretici firma tarafından veri kataloglarında verilen ve her bir tek atımlıya özgü bir sabittir. 74122 için $K=0.32$

ÖRNEK 9.3:

74122'ye bağlandığında $1\mu s$ 'lik bir darbe genişliği üretecek R_{EXT} ve C_{EXT} değerlerini bulun.

ÇÖZÜM:

74122 için $K=0.32$ olduğundan:

$$t_w = K \cdot R_{EXT} \cdot C_{EXT} \left(1 + \frac{0.7}{R_{EXT}}\right) \quad C_{EXT} = 1500 pF$$

$$t_w = K \cdot R_{EXT} \cdot C_{EXT} + 0.7 \left(\frac{K \cdot R_{EXT} \cdot C_{EXT}}{R_{EXT}}\right)$$

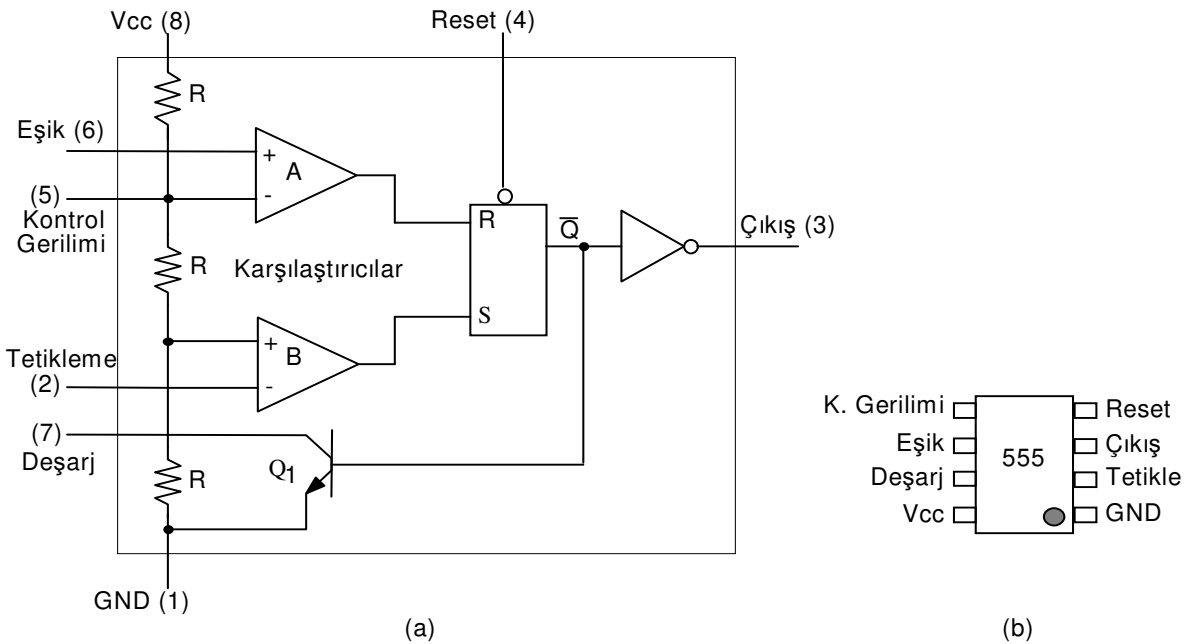
$$t_w = K \cdot R_{EXT} \cdot C_{EXT} + 0.7 K \cdot C_{EXT}$$

$$R_{EXT} = \frac{t_w - 0.7 \cdot K \cdot C_{EXT}}{K \cdot C_{EXT}} = \frac{t_w}{KC_{EXT}} - 0.7$$

$$R_{EXT} = \frac{1 \times 10^{-6} s}{(0.32) \cdot (1500 \times 10^{-12} F)} - 0.7 = 2.1 K\Omega$$

555 ZAMANLAYICI

555 zamanlayıcısı astable (kararsız) ve monostable (tek kararlı) olarak çalışabilen bir tümdevredir. İç yapısı (a) ve bacak bağlantısı (b) Şekil-9.3'de gösterilmiştir. V_{CC} gerilimi 4.5 V ile 18 V arası olabilir.



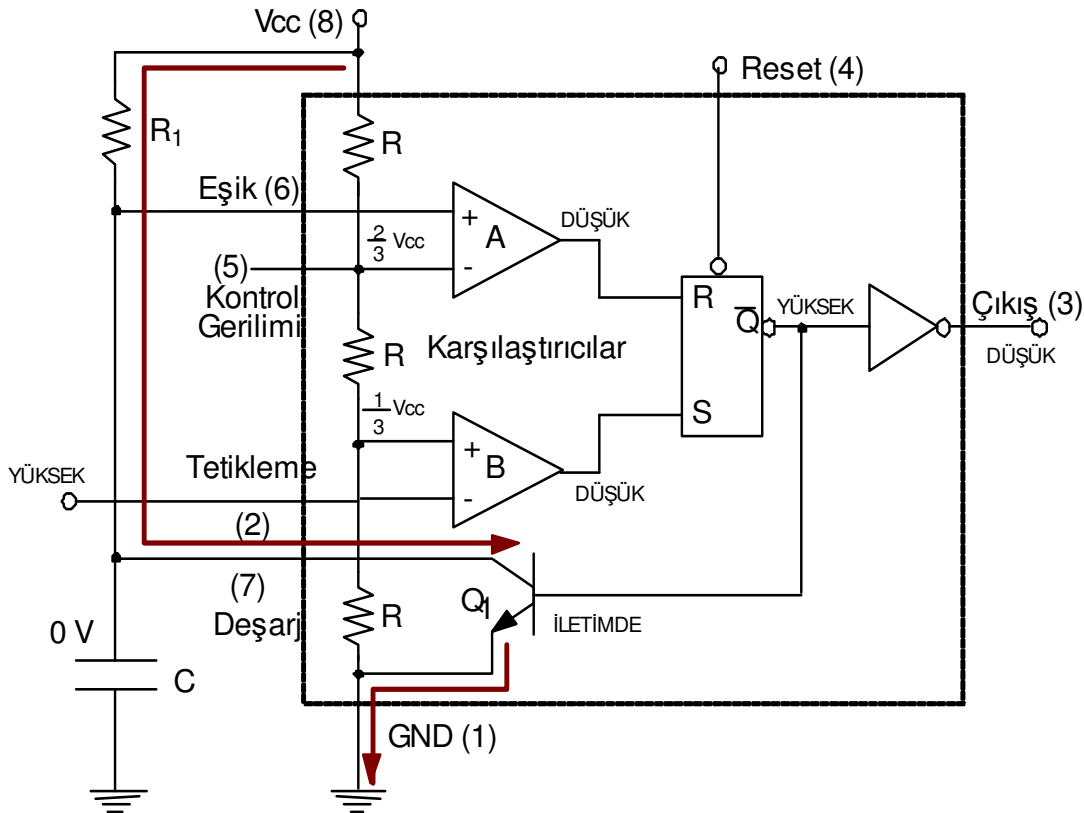
Şekil-9.3 (a). 555'in iç yapısı, (b) bacak bağlantısı.

Karşılaştırıcı B tetikleme girişi (2 nolu bacak) ile $(1/3)V_{CC}$ değerini karşılaştırır. Eğer karşılaştırıcının eksi (-) girişi $(1/3)V_{CC}$ 'den küçük ise, karşılaştırıcı çıkışı YÜKSEK olur ve ikiliyi (f-f) kurarak (SET) 3 nolu bacadaki Q çıkışını YÜKSEK seviyeye geçirir; deşarj transistörünü kesime götürür. Karşılaştırıcı A eksi girişindeki $(2/3)V_{CC}$ ile artı (+) girişine uygulanan eşik gerilimini karşılaştırır. Eşik gerilimi $(2/3)V_{CC}$ 'den büyük olana kadar karşılaştırıcı çıkışı DÜŞÜKTÜR.

Eşik gerilimi $(2/3)V_{CC}$ 'yi aştığında karşılaştırıcı A'nın çıkışını DÜŞÜK'ten-YÜKSEK'e geçirir ve ikilinin çıkışı temizlenir (RESET). Bu durum çıkışı tekrar DÜŞÜK seviyeye geçirerek deşarj transistörünü de iletme götürür. Harici TEMİZLE (RESET) girişi tutucuyu eşik devresinden bağımsız olarak temizlemek için kullanılır. 5 nolu bacak $(2/3)V_{CC}$ yerine başka bir değerde karşılaştırma yapılacak ise kullanılır. Genellikle bu bacak kullanılmaz ve eşik ve tetikleme girişlerinden gelebilecek gürültüyü engellemek için $0.01\mu F$ 'lık kondansatör ile ortak uca bağlanır.

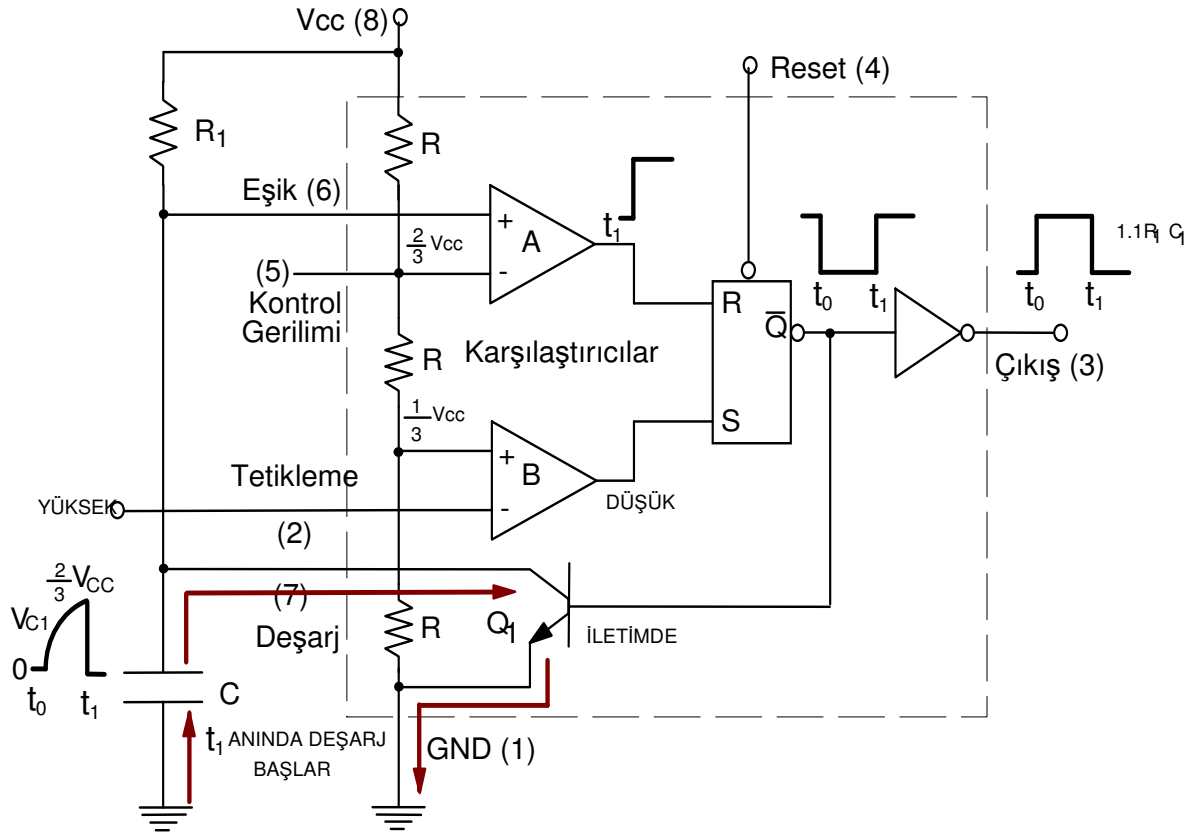
555'İN TEK ATIMLI MODDA ÇALIŞTIRILMASI

ŞEKİL-9.4'te 555'in tek kararlı olarak kullanımını gösteren devre görülmektedir, tetikleme girişine herhangi bir işaret gelmediğinde (yüksek seviyesini koruduğunda) karşılaştırıcı A ve B'nin çıkışları her ikisi aynı anda düşük seviyededir. Dolayısıyla FF'nin \bar{Q} çıkışı yüksek seviyede olacak ve transistörü iletime götürecektir. Bu anda çıkış \bar{Q} 'nun değillenmiş hali olduğu için sıfır seviyesinde olacaktır. V_{CC} kaynağından R_1 direnci ve Q_1 transistörü üzerinden ŞEKİL-9.3'de gösterilen yönde ortak uca bir I akımı akacaktır.



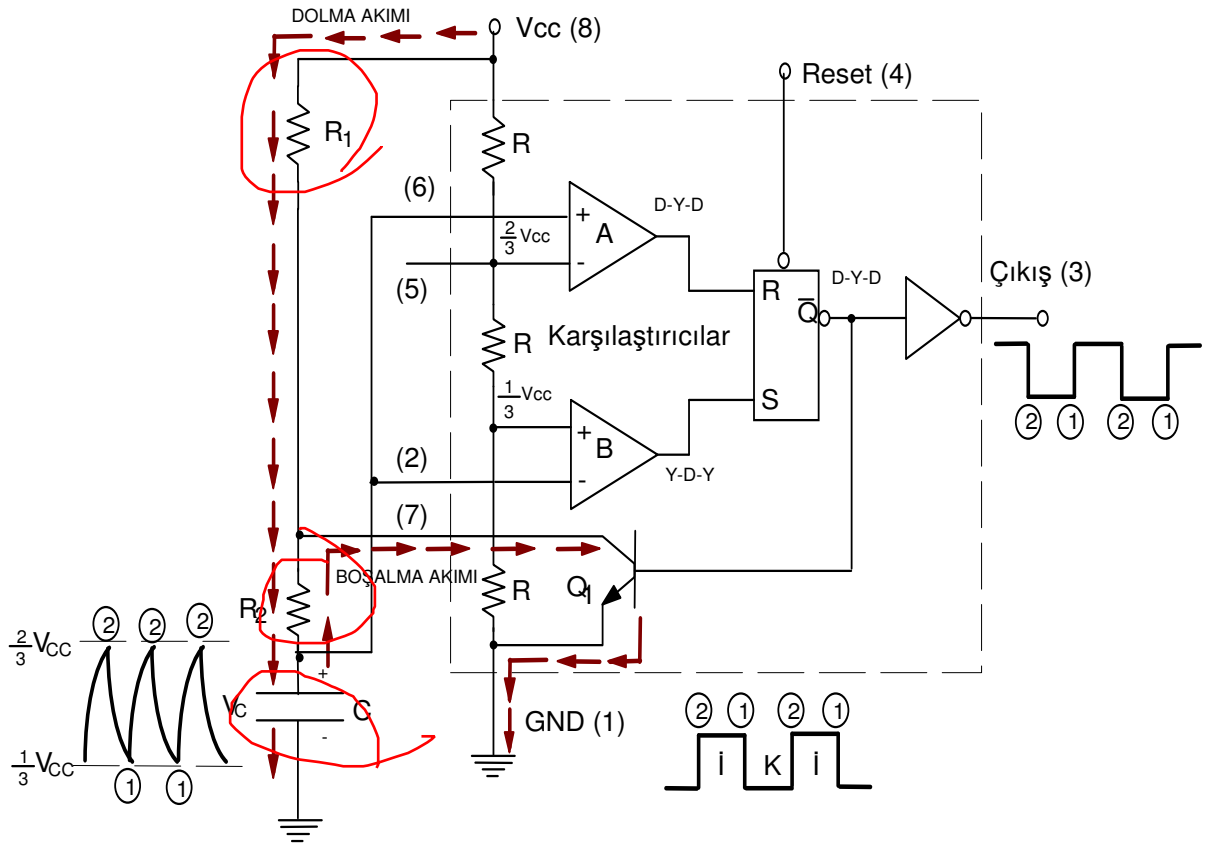
ŞEKİL-9.4 Tetikleme olmadığında akımın yolu ve çıkışların durumu.

Şekil-9.4'te olduğu gibi tetikleme girişi belirli bir süre için sıfır seviyesine çekilirse karşılaştırıcı B'nin çıkışı o süre kadar yüksek seviyeye çıkar ve FF'u kurar. Kurulan FF'un \bar{Q} çıkışı yüksek seviyeden düşük seviyeye düşecektir. Düşüğe geçtiği andan, t_0 anı, itibaren Q_1 transistörü kesime geçer ve V_{CC} kaynağından ortak uca doğru akan I akımı kondansatör üzerinden devresini tamamlamak zorunda kalacaktır, bu arada kondansatör dolmaya başlayacaktır. t_0 anında \bar{Q} çıkışı yüksekten düşüğe geçerken devre çıkışı düşüktür. Düşüğe geçince çıkış yüksek olur ve Q_1 transistörünü iletime götürür. Transistör iletime geçtiğinde kondansatör transistör üzerinden boşalmaya başlar ve ŞEKİL-9.4'te görüldüğü gibi bir I akımı transistör üzerinden ortak uca akar. t_1 anında \bar{Q} çıkışı düşüğe çıkarken devre çıkışı düşüğe geçecektir ve bu durumunu bir dahaki tetikleme gelene kadar korur.



ŞEKİL-9.6 Kondansatör $(2/3)V_{CC}$ değerine dolduktan sonraki (t_1 anında) akımın yolu ve çıkışların durumu.

Q çıkışı düşükten yükseğe geçecek ve transistörü iletime götürecektir. Kondansatör uçlarında toplanan enerji R_2 direnci ve transistör üzerinden ortak uca doğru bir boşalma akımı akıtacaktır. Bu durumda devre çıkışı yüksekte düşük seviyesine kondansatör uçlarındaki gerilimin $(1/3)V_{CC}$ 'ye düşüncüye kadar devam edecektir. V_C $(1/3)V_{CC}$ 'ye ulaştığında karşılaştırıcı A'nın artı girişine $(1/3)V_{CC}$, eksi girişine $(2/3)V_{CC}$ uygulandığından ve çıkışı yüksekte düşüğe geçecek, karşılaştırıcı B'nin eksi girişine sabit $(1/3)V_{CC}$ uygulandığından artı girişine gelen gerilim $1/3V_{CC}$ 'nin altına düşer düşmez ((1) nolu durum) çıkışı düşükten yükseğe geçer ve FF'ü kurar. Q çıkışı yüksekte düşüğe geçecek, transistörü kesime götürecektir. Bu durum V_C 'nin değeri $2/3V_{CC}$ değerini aşınca kadar devam eder. Bu değere ulaştığında tekrar FF silinir ve transistör iletime geçer ve kondansatör tekrar boşalmaya başlar. Bu olaylar sürekli bir şekilde olduğundan çıkıştan sürekli bir kare dalga elde edilir. Kare dalganın düşük seviyesi kondansatörün $2/3V_{CC}$ 'den $1/3V_{CC}$ değerine kadar boşalması sırasında geçen süre uzunluğundadır. Yüksek seviyesi ise kondansatörün $1/3V_{CC}$ 'den $2/3V_{CC}$ değerine kadar dolması sırasında geçen süre uzunluğundadır. Yüksek seviye ile düşük seviye eşit değildir. Eşit olabilmesi için R_1 direncinin mümkün olduğu kadar küçük seçilmesi gerekir.



ŞEKİL-9.7 555'in kare dalga üretici olarak kullanılması.

Yaklaşık olarak üreticinin frekansı;

$$f = \frac{1}{0.7(R_1 + 2R_2)C}$$

formülü ile hesaplanabilir. Kondansatör dolduğu sırada zamanlayıcı çıkışı yüksek seviyede olduğuna göre yüksek seviyede kaldığı süre (t_H);

$$t_H \cong 0.7(R_1 + R_2)C$$

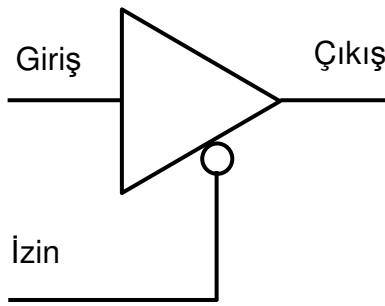
Kondansatör boşalma sırasında zamanlayıcı çıkışı düşük seviyede olduğuna göre düşük seviyede kaldığı süre (t_L);

$$t_L \cong 0.7R_2C$$

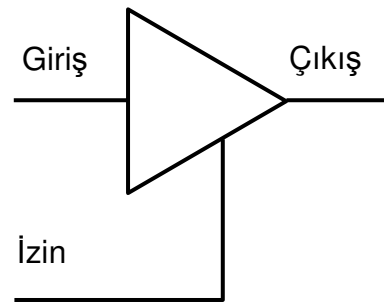
Kare dalga üreticilerinde genellikle t_H ve t_L sürelerinin eşit olması istenir. Eşit yapmak için R_1 mümkün olduğu kadar küçük seçilmelidir. Bu süreleri eşit yapabilmek için R_2 direnci uçlarına dolma akımını üzerinden geçmesine izin verecek şekilde diyot bağlanır ve her iki direncin değeri eşit seçilir.

ÜÇ-DURUMLU TAMPONLAR

Üç durumlu mantıkta üç düzey vardır: HI ("1"), LO ("0") ve HE (açık devre-yüksek empedans). Üç-durumlu tampon İZİNlenince girişi çıkışına ulaşır. İZİN yokken çıkış ucu açık devredir. Üç-durumlu tampon-ÜÇ DURUMLU TUTUCU bilgisayarlar da iki-yönlü (bi-directional) ve çoğullanmış (multiplexed) yollarda çokça kullanılır. İki-yönlü yol, aynı iletim hattından ayrı zamanlarda ayrı yönler e veri iletebilmek için kullanılır. Çoğullanmış yol, birkaç bilgisayar aygıtının aynı yolu ortaklaşa kullanabilmesi (bir anda yalnız biri) için kullanılır. Üç-durumlu tamponun (Three State Buffer) birkaç çeşidi vardır. Tüm çeşitlerinin mantık simgeleri Şekil-9.8 ve 9.9'da gösterilmiştir. Doğruluk tabloları Tablo-9.1, 9.2, 9.3 ve 9.4'te verilmiştir.

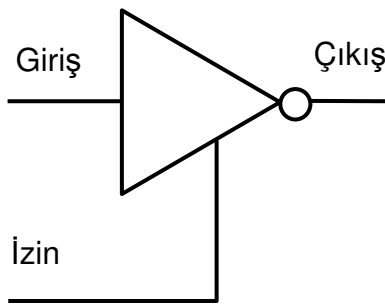


(a) DÜŞÜK seviye izinli.

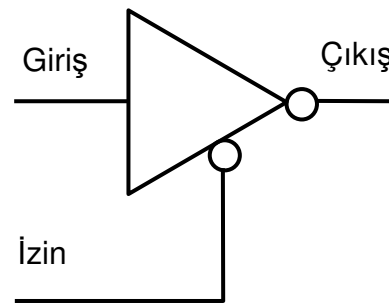


(a) YÜKSEK seviye izinli.

Şekil-9.8 Terslemeyen çıkışlı DÜŞÜK seviye ve YÜKSEK seviye izinli üç durumlu tamponların mantık simgeleri.



(a) YÜKSEK seviye izinli.



(b) DÜŞÜK seviye izinli.

Şekil-9.9 Tersleyen çıkışlı DÜŞÜK seviye ve YÜKSEK seviye izinli üç durumlu tamponların mantık simgeleri.

İZİN	GİRİŞ	ÇIKIŞ	YORUM
1	1	0	Girişin tersi çıkışa aktarılır.
1	0	1	Girişin tersi çıkışa aktarılır.
0	1	Y.E.	Çıkış açık devredir.
0	0	Y.E.	Çıkış açık devredir.

Tablo-9.1 İZİN girişi YÜKSEK seviyede etkin olan girişi tersleyen üç-durumlu tamponun doğruluk tablosu.

İZİN	GİRİŞ	ÇIKIŞ	YORUM
0	1	0	Girişin tersi çıkışa aktarılır.
0	0	1	Girişin tersi çıkışa aktarılır.
1	1	Y.E.	Çıkış açık devredir.
1	0	Y.E.	Çıkış açık devredir.

Tablo-9.2 İZİN girişi DÜŞÜK seviyede etkin olan girişi tersleyen üç-durumlu tamponun doğruluk tablosu.

İZİN	GİRİŞ	ÇIKIŞ	YORUM
1	1	1	Giriş aynen çıkışa aktarılır.
1	0	0	Giriş aynen çıkışa aktarılır.
0	1	Y.E.	Çıkış açık devredir.
0	0	Y.E.	Çıkış açık devredir.

Tablo-9.3 İZİN girişi YÜKSEK seviyede etkin olan girişi terslemeyen üç-durumlu tamponun doğruluk tablosu.

Verilen doğruluk tablolarından da görüldüğü gibi, İZİN girişi etkin kılınıncı girişteki bilgi (boncuk varsa tümlenerek) çıkışa ulaşmaktadır. ÜÇ DURUMLU TUTUCULAR bilgisayar yollarında yaygın olarak kullanılmaktadırlar. Yol (bus), sayısal bilgilerin üzerinden iletildiği iletim hattıdır. Bilgisayar içinde veriler ve adresleri birimler arasında gerek duyuldukça alınıp, gönderilir. Bu iletim 8, 16 yada 32 bitlik paralel hatlar üzerinden yapılır ve aynı kablolar bilginin hem gönderilmesi hem de alınması için kullanılır.

İZİN	GİRİŞ	ÇIKIŞ	YORUM
0	1	1	Giriş aynen çıkışa aktarılır.
0	0	0	Giriş aynen çıkışa aktarılır.
1	1	Y.E.	Çıkış açık devredir.
1	0	Y.E.	Çıkış açık devredir.

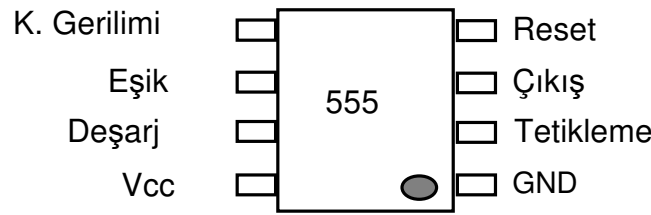
Tablo-9-4 İZİN girişi DÜŞÜK seviyede etkin olan girişi terslemeyen üç-durumlu tamponun doğruluk tablosu.

İki türlü yol vardır: İki-yönlü ve çoğullanmış. İki-yönlü yollar verinin her iki yöne doğru akmasına izin verir. İki-yönlü bilgisayar yollarına en iyi örnek veri yoludur (data bus). Çevre aygıtlar (peripheral devices) bu yola veri yazarak MİBe (Merkezi İşlem Birimi-CPU-Central Processing Unit) veri gönderirler. MİB bu verileri gerektiği gibi işleyip gereksinen çevre aygıtlara yine aynı veri yolu üzerinden gönderir. MİBe veri gönderen çevre aygıtlar, RAM, ROM, klavye, floppy disk driver, hard disk driver, tape driver, modem, scanner, digitizer, mouse, lite pen, ve trackball olarak; işlemcinin veri yolladığı çevre aygıtlar da, tape, floppy & hard disk drivers, RAM, modem, lite pen, printer, plotter ve video card olarak sıralanabilir.

Çoğullanmış bir yolda veri yalnızca bir yönde akabilir ama birkaç ayrı birim aynı yola (tabii ki ayrı zamanlarda) veri yazabilirler. Çoğullanmış yol için örnek olarak adres yolunu verebiliriz. MİB bu yol üzerinden bütün adresli çevre aygıtlarına (bellekler) yazılacak ve/veya okunacak verilerin adreslerini iletir. Yolun çoğullanmasının temel amacı, MİBin adres çıkış iskelesinin elektriksel olarak yüklenmesini önlemektir. Zaten adres bilgilerinin tüm çevre birimlere ayrı ayrı gitmesi de istenmektedir, çünkü her birinin adres yapılandırması değişik olabilir.

SORULAR

1. ~~Kararsız durumda (osilatör olarak)~~ çalışan 555 zamanlayıcısına bağlanan direnç ve kondansatör değerlerine göre frekansını hesaplayın.
 - a. $R_1 = 1k\Omega$, $R_2 = 2k\Omega$, $C = 0.01 \mu F$
 - b. $R_1 = 2k\Omega$, $R_2 = 4k\Omega$, $C = 0.001 \mu F$
2. 555 tüm devresini kullanarak t_H/ t_L oranı 0.8 , $f=500kHz$ olan kararsız zamanlayıcı devresini 10nF, 1nF, 100nF'lık kondansatörlerden uygun olanı kullanarak tasarlayın. (Devresi çizilerek eleman değerleri devre üzerinde gösterilecek.)



BÖLÜM 10

SAYICILAR (COUNTERS)

Sayıcılar sayısal elektroniğin temel devreleridir. Sayıcılar istenilen aralıkta her saat darbesinde ileri veya geri doğru sayma yaparlar. Sayıcılar flip-flop kullanılarak yapılır, kullanılan flip-flop sayısı sayıcının sayma aralığını belirler. Örneğin 4 flip-flop kullanılırsa sayılacak durum sayısı 2^4 tür ve sayma aralığı $0000_2 - 1111_2$ 'dir. Sayıcılar iki ana gruba ayrılır, eş zamansız anlamında asenkron ve eş zamanlı anlamında senkron. Asenkron sayıcılar aynı zamanda ripple (dalgacık) sayıcı olarak da adlandırılırlar.

ASENKRON SAYICILAR

Asenkronun kelime anlamı eş zamanda olmayan demektir. Sayıcılara bu adın verilmesinin sebebi ise, sayıcıyı oluşturan Flip-Flopların durum değiştirme anlarının birbirleri ile aynı olmayışdır. Bu sayıcıları oluşturan Flip-Flopların saat girişleri aynı işaret ile tetiklenmez.

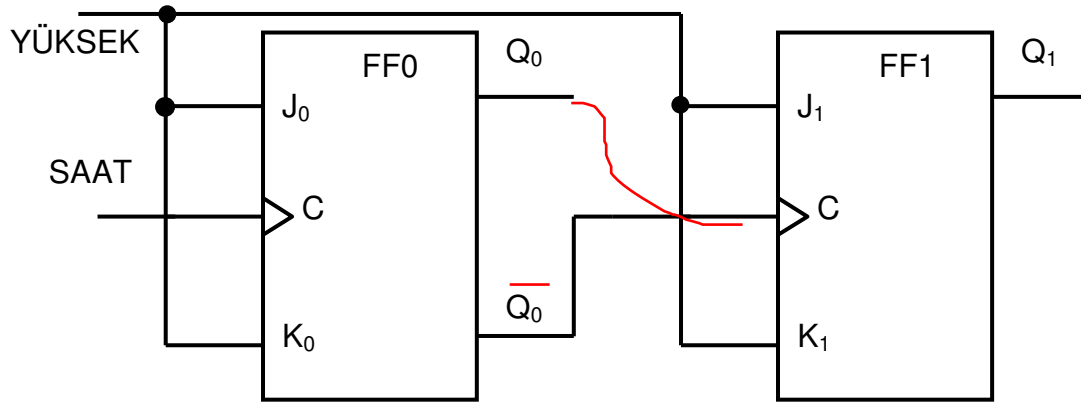
2 BİT ASENKRON SAYICI

Şekil-10.1'de 2 bit asenkron sayıcının bağlantısı verilmiştir. Saat işareti, bu işaret sayma hızını belirler, sadece FF0'ın saat girişine bağlanmıştır. FF1'nin saat girişine ise FF0'ın Q' çıkışı bağlanmıştır. Saat işaretinin yükselen kenarında FF0 tetiklenirken FF1 ise FF0'ın Q' çıkışında oluşan yükselen kenar ile tetiklenerek durum değiştirir. FF0 tetiklendikten belirli bir süre sonra ve sadece yükselen kenarlarda, o da ancak iki saat işaretinde bir gerçekleşir, girişe gelen saat işareti yayılma gecikmesi kadar geç çıkışa ulaştığından dolayı aynı anda tetikleme hiçbir zaman gerçekleşmez.

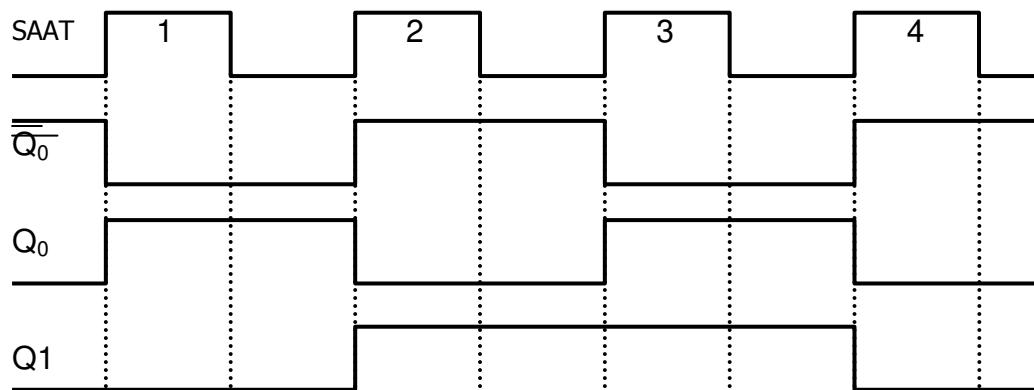
Şekil-10.2'de sayıcının zamanlama diyagramı verilmiştir. Birinci saat vurusunda FF0 yükselen kenar ile tetiklenmiştir. Q çıkışı düşükten yükseğe geçerken Q' çıkışında bir düşen kenar oluşmuştur. FF1 Q' yükselen kenarı ile tetiklendiğinden durum değiştirmemiştir. İkinci saat vurusunda FF0 durum değiştirmiştir, dolayısıyla Q' çıkışında bir yükselen kenar olduğundan FF1 tetiklenmiş ve durum değiştirmiştir. Üçüncü saat vurusunda FF0 yine durum değiştirir fakat Q' çıkışında düşen kenar oluştuğu için FF1'de durum değişikliği olmaz.

SAAT VURUSU	Q ₁	Q ₀
Başlangıç	0	0
1	0	1
2	1	0
3	1	1
4	0	0

Tablo-10.1 İki bit asenkron sayıcının durum değişimi.



Şekil-10.1 İki bit asenkron sayıcı.



Şekil-10.2 İki bit asenkron sayıcının zamanlama diyagramı.

Dördüncü ve son saat vurusunda ise FF0 durum değiştirir, Q' çıkışında oluşan yükselen kenar FF1'in durum değiştirmesine neden olur. Sayıcının saat vuruşlarına göre durum değiştirmesi Tablo-10.1'de verilmiştir. Sayıcı iki bit olduğu için çıkışların

alabileceği durum sayısı dördtür, beşinci çevrimde başlangıç değerine geri dönecektir. Q_1 YÜKSEK değerlikli bit Q_0 ise DÜŞÜK değerlikli biti gösterir.

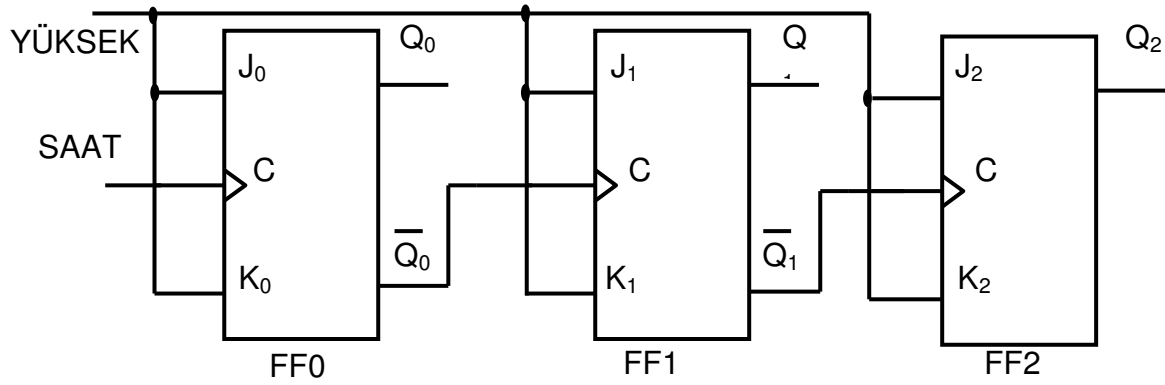
3 BİT ASENKRON SAYICI

3 bit asenkron sayıcı bağlantısı şekil-10.3(a)'da gösterilmiştir. 2 bit sayıcı ile aynı şekilde çalışır. Farklı olarak 8 çıkış durumu vardır, bir çevrimi 8 makine vurusunda tamamlayacaktır. Şekil-10.3(b)'de gösterilmiştir. 3 bit sayıcıda 8 çıkış durumu oluşacaktır, bu durumlar saat saykılına göre Tablo-10.2'de gösterilmiştir.

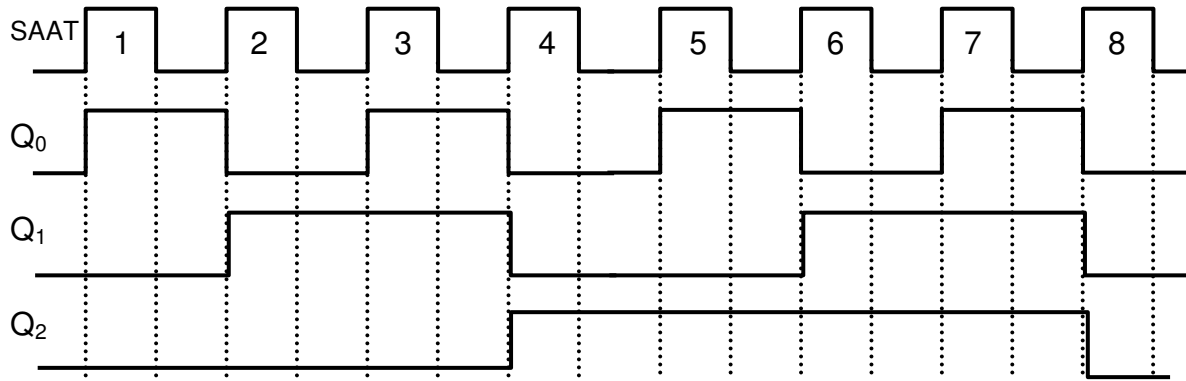
SAAT VURUSU	Q_2	Q_1	Q_0
Başlangıç	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1
8	0	0	0

Tablo-10.2 3 bit asenkron sayıcının durum değişimi.

Daha önce bu sayıcıların diğer bir adının ripple sayıcı olduğunu söylemiştik. Bunun nedeni şekil-10.4'te gösterilmiştir. Birinci saat vurusunda sadece Q_0 çıkışında saat işaretine göre bir gecikme olacaktır. Diğer çıkışlarda ise değişim olmadığı için gecikme olmayacaktır. İkinci saat vurusunda Q_0 çıkışında bir gecikme olacak, aynı zamanda Q_0 ' ile Q_1 arasında bir gecikme oluşacaktır. Dördüncü saat vurusunda Q_2 çıkışı da Q_1 'e göre gecikecektir. Sonuç olarak çıkışlar aynı anda oluşmayacak her çıkış bir DÜŞÜK değerli çıkışın durum değişimini bekleyecektir. Sonucun bu şekilde dalgalar halinde gerçek değerini almasından dolayı bu sayıcılara dalgacık anlamına gelen ripple adı verilmiştir. Bu gecikmeler hızlı çalışan sayısal dizgelerde sorun olabilir.

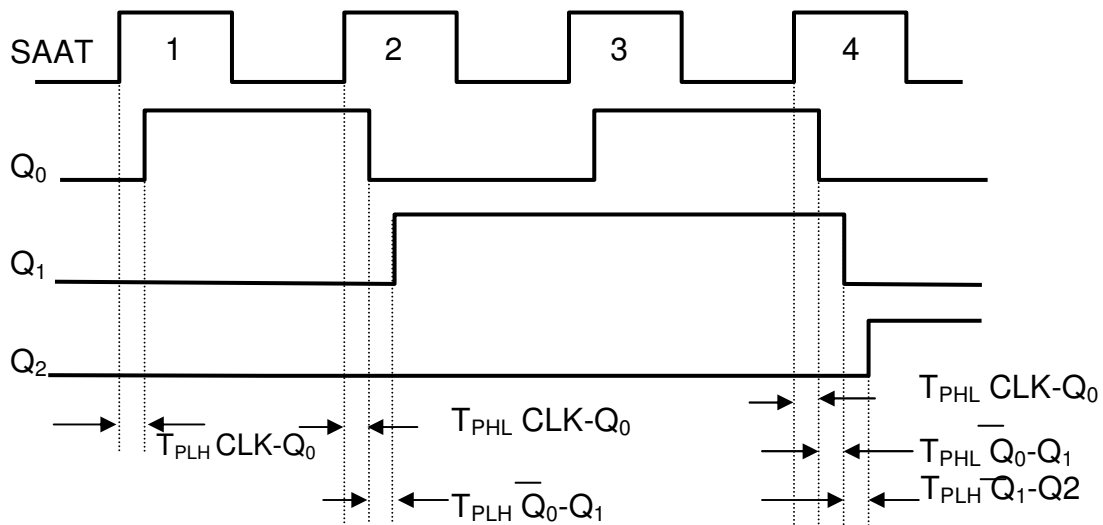


(a)



(b)

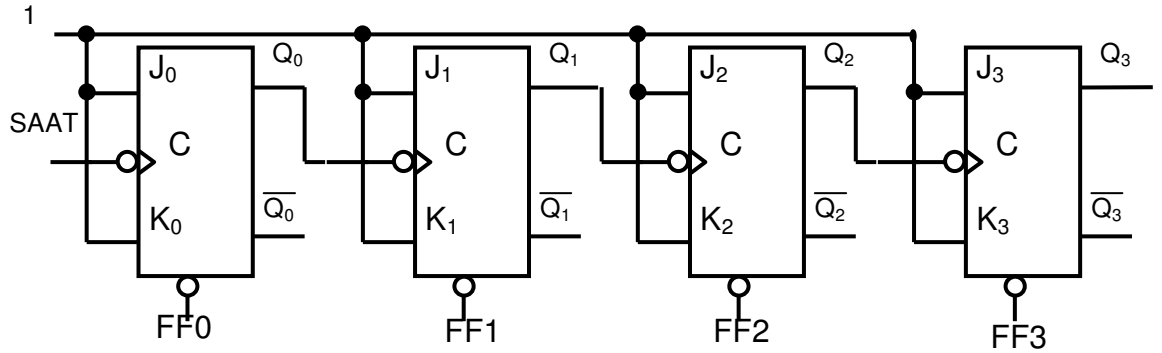
Şekil-10.3 3 bit asenkron sayıcı bağlantısı ve zamanlana diyagramı.



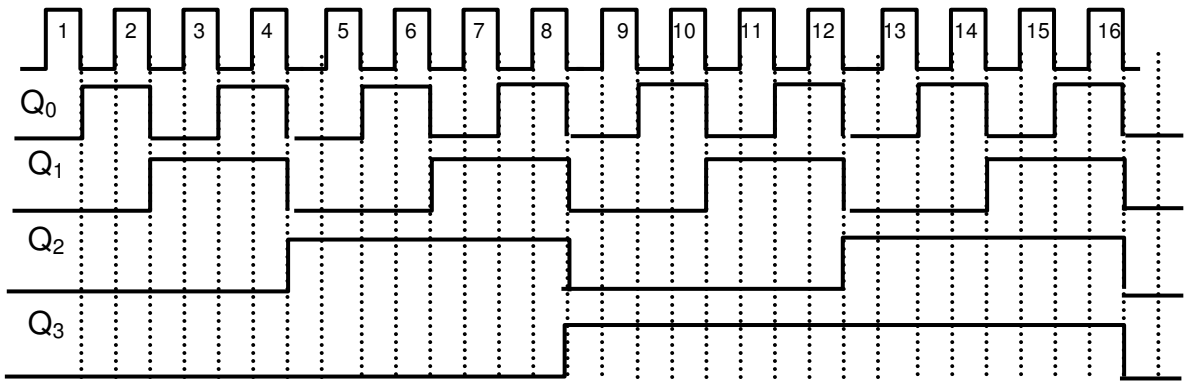
Şekil-10.4 3 bit asenkron sayıcının yayılma gecikmesi.

ÖRNEK 1

ŞEKİL-10.5'de 4 bit asenkron sayıcı bağlantısı verilmiştir. Her flip-flop düşen kenar tetiklidir ve yayılma gecikmesi süresi 10ns'dir. Her flip-flopun Q çıkışını saat vurusuna göre değişimini gösteren zamanlama diyagramını çizin. Toplam gecikmeyi hesaplayın.

Çözüm:

(a)



(b)

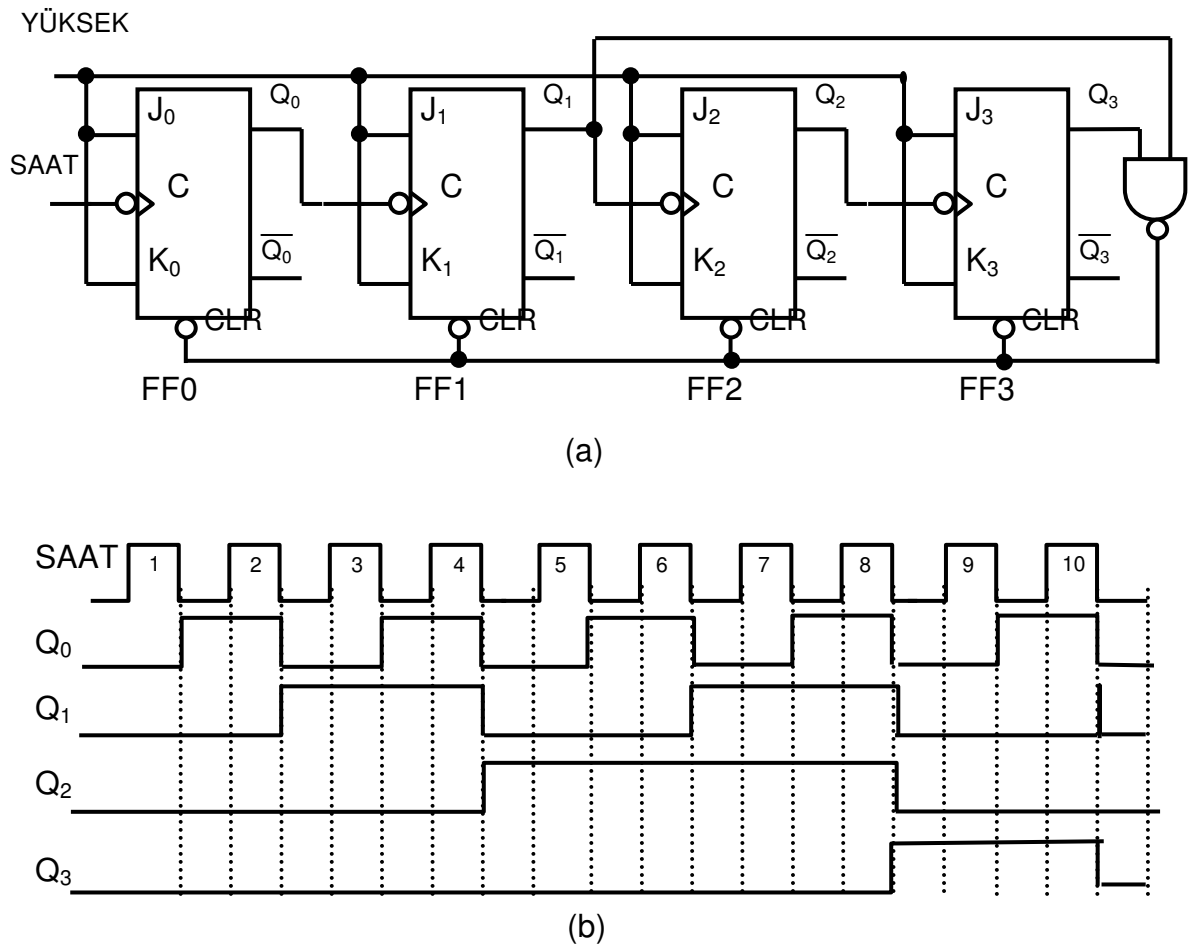
Şekil-10.5 4 bit asenkron sayıcı (a) bağlantısı, (b) zamanlama diyagramı.

$$T_p = 4 \times 10 \text{ ns} = 40 \text{ ns}$$

ASENKRON ONLUK SAYICI

İkilik sayıcılar olası tüm durumları verilen saat vurularına göre çıkışlarında oluştururlar. Bu tür sayıcılarda maksimum durum sayısı 2^n formülü ile hesaplanabilir, burada n flip-flop adedidir. Maksimum durumundan daha az sayan sayıcı tasarlanabilir. Günlük yaşamda kullanılan sayı sistemi olan onluk saymayı yaptırmak için mod 10 sayan sayıcı tasarlanabilir. Genel mantığı, bu çıkış durumu

algılandığında flip-flopların asenkron girişlerine etkin seviye uygulanarak çıkışlar ya sıfırlanır veya birleşir. Sıralı 10 kesen asenkron sayıcılara **onluk sayıcı** (decade counter) adı verilir. Onluk sayıcı çıkışlarında 1010 durumunu algılayınca flip-flopun CLR girişlerine bir sıfır seviye uygulanarak sayma işlemi başlangıç durumuna getirilir. 10'da kesme yapabilmek için Q_1 ve Q_3 çıkışlarının her ikisinin birden bir olduğu durum bir VED geçidi ile gözlenir. Bu durum geldiğinde VED geçidi çıkışı "0" olur ve bu sıfır Flip-Flopların CLR girişlerine uygulanarak tüm Q çıkışları sıfırlanır. Sayma işlemi bir sonraki saat vurusunda tekrar başlar. Çıkışlarda 1010 (on) durumu çok kısa süreli oluşur. Q_1 çıkışı bu durumda sıfırdan bir seviyesine çıktığından bu çıkışta çok kısa süreli iğne dalgacık oluşur. Şekil-10.6'da onluk asenkron sayıcının bağlantısı ve zamanlama diyagramı verilmiştir.



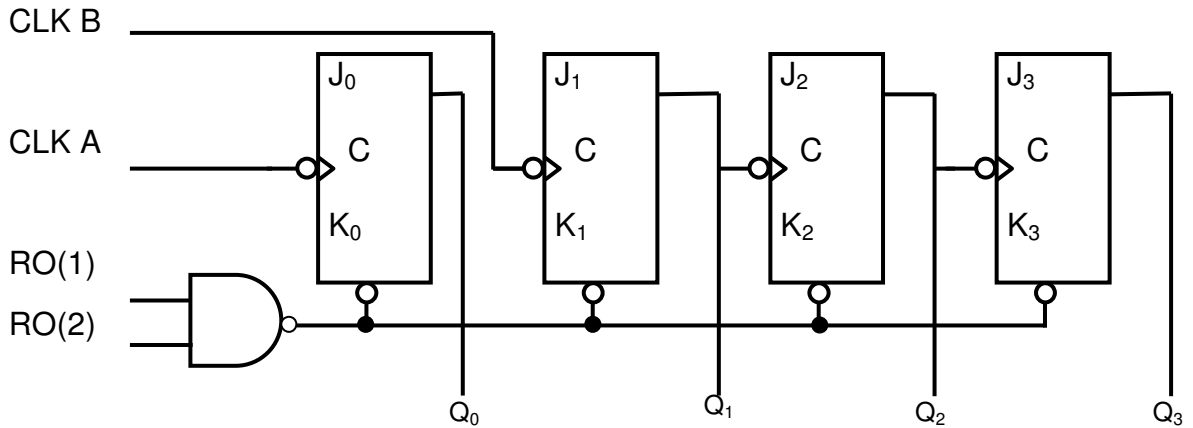
Şekil-10.6 4 bit asenkron sayıcı (a) bağlantısı, (b) zamanlama diyagramı.

İKİLİK ASENKRON SAYICI TÜMDEVRESİ

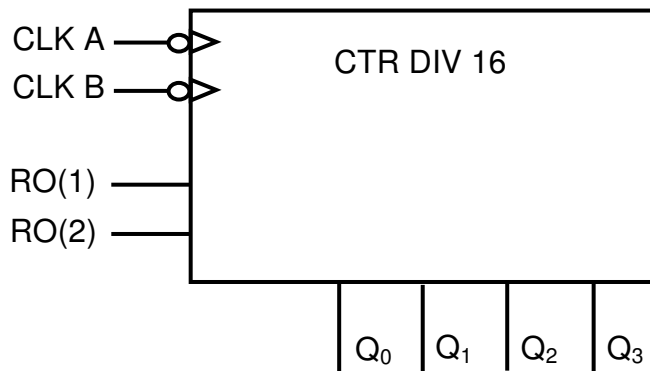
Genel formülünü yazacak olursak n adet flip-flop kullanıldığında saydığı durum sayısı 2^n saydığı en yüksek sayı ise 2^{n-1} 'dir. Şekil-10.1'deki devrede J-K flip-floplar kullanılarak gerçekleştirilmiş 4 bit sayıcı devresi gösterilmiştir. 4 flip-flop

kullanıldığına göre saydığı durum sayısı $2^4=16$ 'dır, sayma aralığı 0000_2-1111_2 'dir. Asenkron sayıcılarda sadece ilk flip-flop girişine saat darbesi dışarıdan uygulanır, diğerlerinin saat girişlerine bir önceki flip-flopların çıkışları uygulanır. Tüm J-K girişleri YÜKSEK seviyede tutulur. Şekil-10.7'de ikilik yukarı doğru sayan asenkron sayıcının şekli görülmektedir. İkilik sayıcılar flip-floplar kullanılarak yapılabileceği gibi bu işlemi yapan tümdevreler de vardır. Dışarıdan yapılan birkaç bağlantı ile istenilen nitelikte sayıcı bu tümdevreler kullanılarak elde edilebilir.

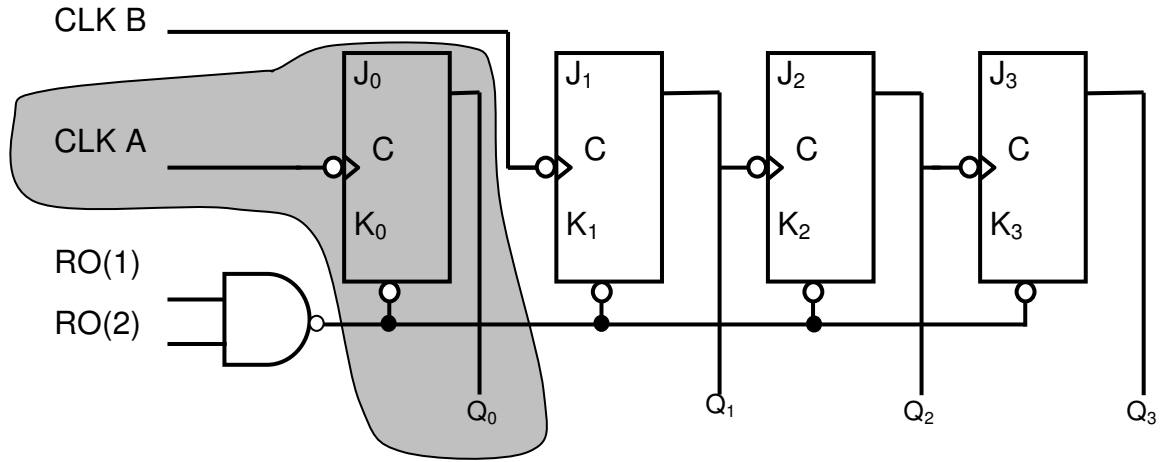
ŞEKİL-10.8'de deney setinde kullanılan 7493 tümdevresinin bacak bağlantısı verilmiştir. Bu tümdevre içerisinde dört adet J-K flip-flop vardır, bu flip-flopların üç tanesi asenkron sayıcı olarak içeriden bağlantıları yapılmıştır. Diğer flip-flop ise tek başına kullanılabilir şekilde diğerlerinden bağımsızdır. 7493 tümdevresinin iç bağlantısı ŞEKİL-10.9'da gösterilmiştir. Şekilden de anlaşılacağı gibi bu tümdevre 3 bit veya 4 bit ikilik asenkron sayıcı olarak kullanılabilir. 3 bitlik sayıcı olarak kullanıldığında CLKB girişi saat girişi olarak kullanılacaktır. 4 bitlik sayıcı olarak kullanılırken saat girişi olarak CLKA kullanılacak ve CLKB girişi QA çıkışına bağlanacaktır.



Şekil-10.7 4-bit yukarı doğru sayan asenkron sayıcı.

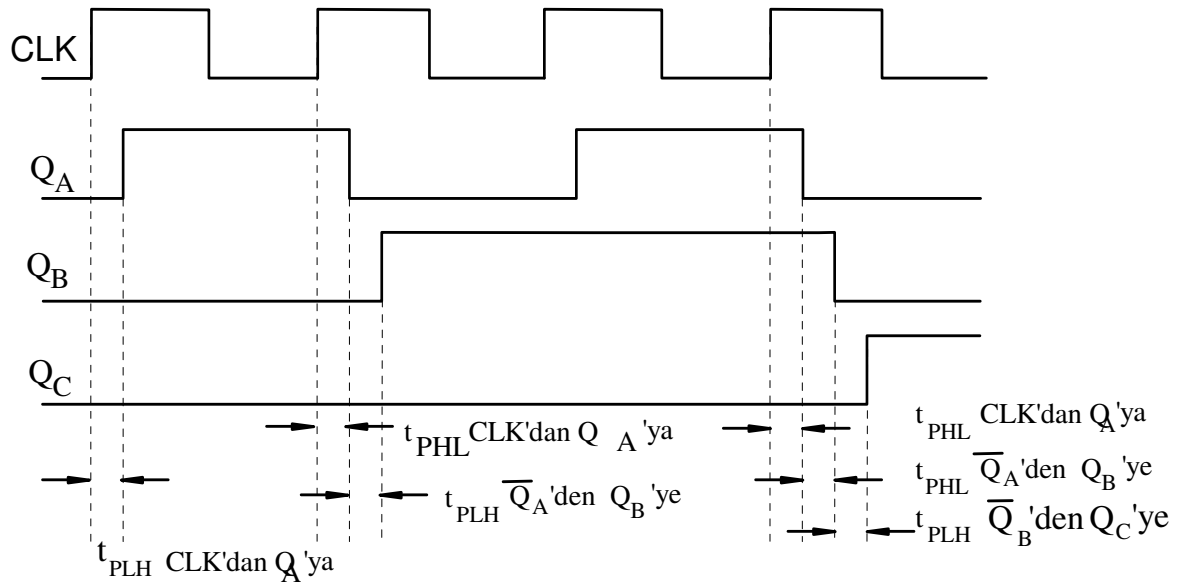


Şekil-10.8 7493 asenkron sayıcının mantık simgesi.



Şekil-10.9 7493'ün 3 bit sayıcı bağlantısı.

Asenkron sayıcıların tasarımı kolaydır ve özel durumlar istenmediğinde dışarıdan başka eleman bağlamak gerekmez. Asenkron sayıcılarda girişten uygulanan saat işareti önce FF0'a sonra FF1'e ve sonra FF2'ye ulaşır. Girişten verilen saat işaretinin çıkıştan elde edilmesi için 4 saat işaretine gerek vardır. Giriş işaretinin dalgalar halinde çıkışa iletilmesinden dolayı bu tür sayıcılara *Ripple sayıcılar* adı da verilir. Bir öncekinin çıkışı bir sonrakinin saat girişi olduğu için giriş ile çıkış arasında propagasyon (yayılma) gecikmeleri toplanarak en son çıkışa aktarılacaktır. FF1 saat vuruşunun tetikleme kenarı ile durum değiştirirken FF3 tetikleme kenarında hemen durum değiştirmeyecek belirli bir gecikme sonunda durum değiştirecektir. Şekil-10.9'da 7493 tümdevresinin 3 bit asenkron sayıcı olarak bağlantısı verilmiş ve Şekil-10.10'da ise bu sayıcıya ait çıkış dalga şekilleri gösterilmiştir. (Gecikmeler olayın anlaşılabilmesi için biraz fazla gösterilmiştir.)

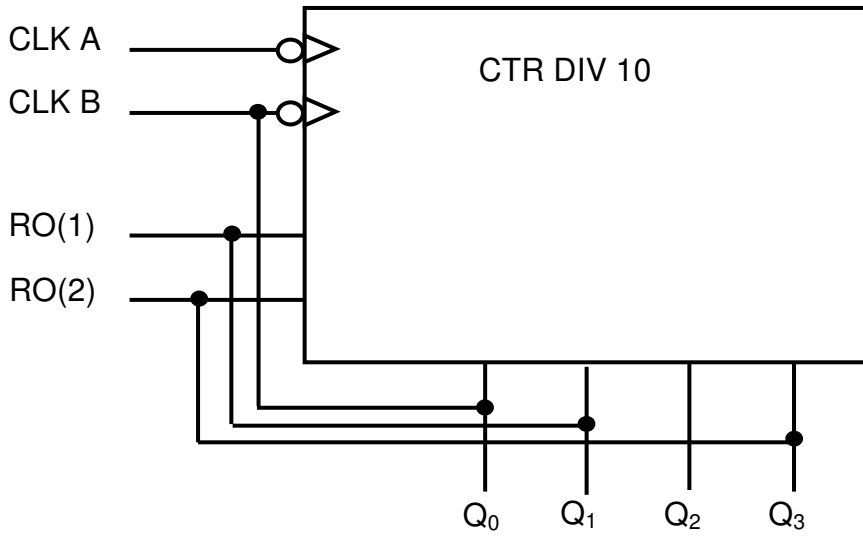


Şekil-10.10 3 bit asenkron sayıcı devresi ve çıkış dalga şekilleri.

7493'ün RO(1), RO(2) girişleri kullanılarak dışarıdan herhangi bir geçit bağlantısına gerek duymadan istenilen modda sayıcı elde edilebilir. Fakat 3 adetten veya daha fazla YÜKSEK seviye çıkışını denetlemek isterseniz bu modu gerçekleyemezsiniz. Örneğin 11, 13, 14 gibi.

SENKRON SAYICILAR

Senkron'un kelime anlamı aynı anda gerçekleşen veya eş zamanlı demektir. Sayıcılarda ise tüm Flip-Floplar aynı saat vurusu ile tetiklenir ve çıkışları durumlarını aynı anda değiştiriyorlarsa senkron sayıcı adı verilir.



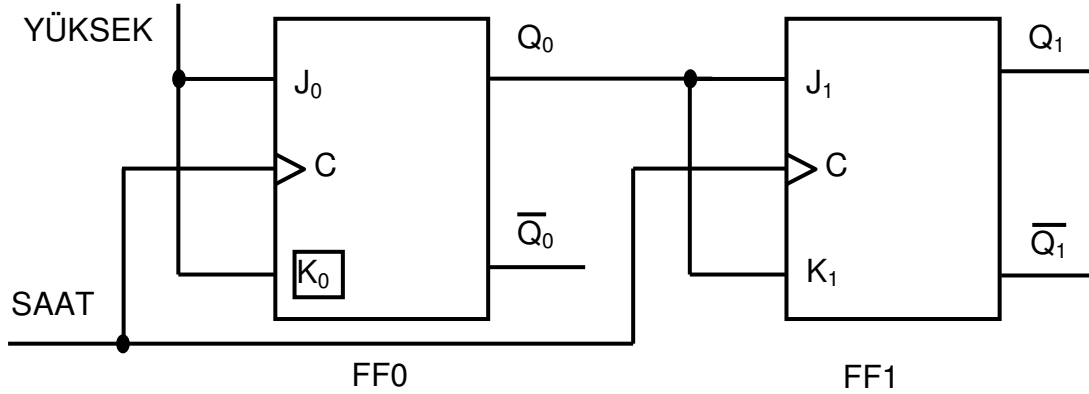
Şekil-10.11 7493'ün onluk sayıcı olarak kullanılması.

2 BİT SENKRON SAYICI

Şekil-10.12'de 2 bit senkron sayıcının Flip-Floplarla bağlantısı verilmiş-tir. Asenkron sayıcıdan farkı saat girişlerinin ortak kullanılmasıdır. Ayrıca Q_0 çıkışı J_1, K_1 girişlerine uygulanmıştır.

Başlangıç olarak her iki Flip-Flopun çıkışının sıfır durumunda olduğunu varsayalım. İlk saat vurusunda FF0 tümleyen modda çalışacak ve Q_0 YÜKSEK seviyeye gelecektir. FF1'de çıkışlar eski durumunu koruyacaktır. Çünkü saat vurusunun yükselen kenarı geldiğinde daha J_1 ve K_1 girişlerine DÜŞÜK seviye uygulanıyor olacaktır. Flip-Flopların yayılma gecikmesinden dolayı çıkışlarını saat vurusu geldikten belli bir süre sonra yenilediklerini hatırlayın. Saat vurusu 1'in sonunda $Q_0=1$ ve $Q_1=0$ olacaktır, bu da ikilik 1 sayısını gösterir.

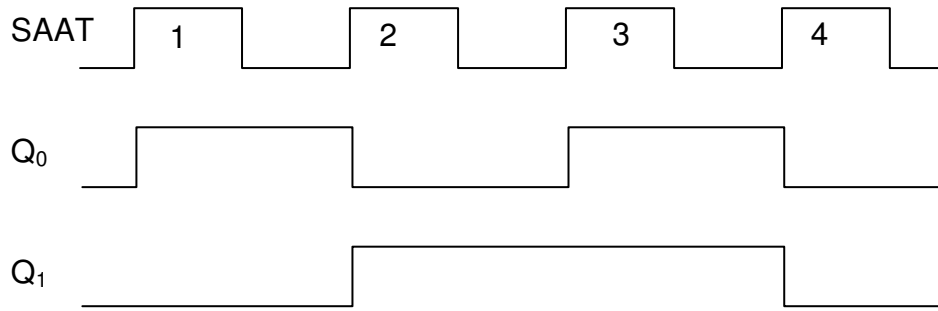
Saat vurusu 2 Flip-Floplara uygulandığında FF0 tümleyen modda çalışacak ve Q_0 düşük seviyeye gelecektir. FF1'de ise J_1 ve K_1 girişleri YÜKSEK seviye olduğundan tümleyen modda çalışacak ve Q_1 YÜKSEK seviyeye gelecektir. Saat vurusu 2'nin sonunda $Q_0=0$ ve $Q_1=1$ olacaktır, bu da ikilik 2 sayısını gösterir.



Şekil-10.12'de 2 bit senkron sayıcı

Saat vurusu 3 Flip-Floplara uygulandığında FF0 tümleyen modda çalışacak ve Q_0 YÜKSEK seviyeye gelecektir. FF1'de ise J_1 ve K_1 girişleri DÜŞÜK seviye olduğundan çıkış eski durumunu korur ve Q_1 YÜKSEK seviyeye kalır. Saat vurusu 2'nin sonunda $Q_0=1$ ve $Q_1=1$ olacaktır, bu da ikilik 3 sayısını gösterir.

Saat vurusu 4 Flip-Floplara uygulandığında FF0 tümleyen modda çalışacak ve Q_0 düşük seviyeye gelecektir. FF1'de ise J_1 ve K_1 girişleri YÜKSEK seviye olduğundan tümleyen modda çalışacak ve Q_1 DÜŞÜK seviyeye gelecektir. Saat vurusu 2'nin sonunda $Q_0=0$ ve $Q_1=0$ olacaktır, bu da ikilik 0 sayısını gösterir. 4 saat vurusu sonunda sayıcı başlangıç konumuna geri dönecektir. Şekil-10.13'de zamanlama diyagramı verilmiştir.



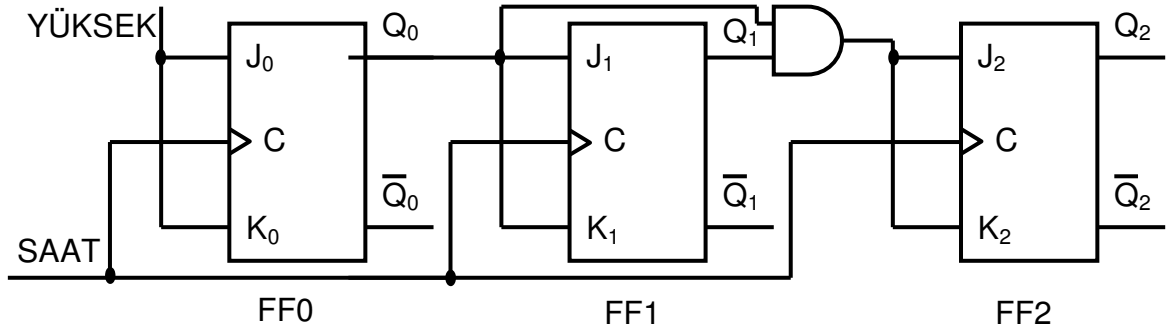
Şekil-10.13 2 bit senkron sayıcı zamanlama diyagramı

Senkron sayıcılarda yayılma gecikmesi asenkron sayıcılarda olduğu gibi YÜKSEK değerlikli bitlere doğru gidildiğinde artmayacaktır. Her çıkışın yayılma gecikmesi sabit olacaktır.

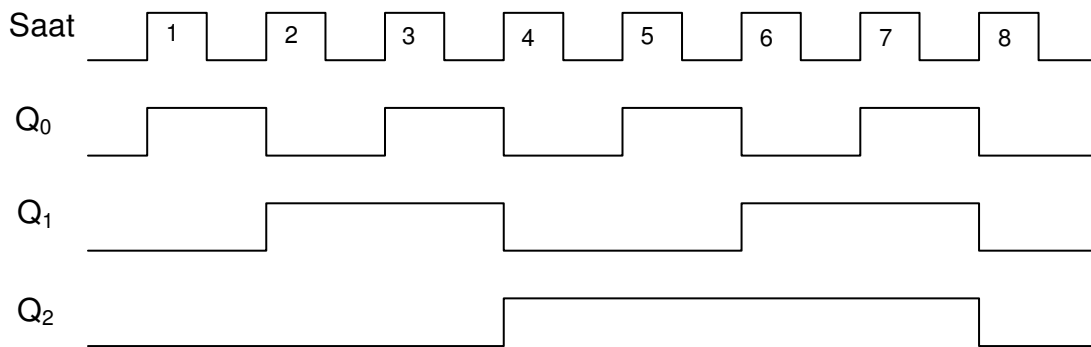
3 BİT SENKRON SAYICI

3 bit senkron sayıcı bağlantısı Şekil-10.14'te zamanlama diyagramı ise Şekil-10.15'te verilmiştir. Çalışmasını anlamak için her saat vurusuna göre çıkışların değişimini incelemek gerekir. 3 bit sayıcının çıkışlarının her saat vurusuna göre

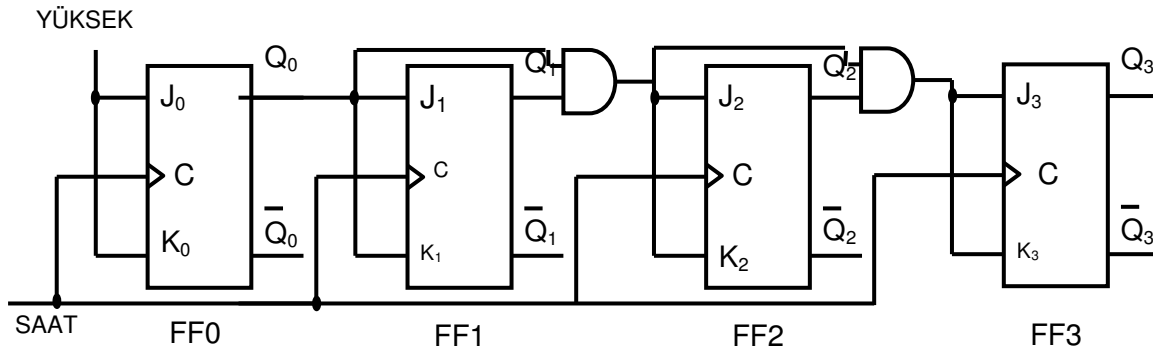
değişimi Tablo-10.3'te verilmiştir. Öncelikle Q_0 'a bakalım, bu çıkış başlangıç durumunda başlayarak her saat vurusunda durum değiştiriyor. Bu çıkışı elde edebilmek için FF0'ın tümleyen kipte tutulması gerekir. Bunun için de J_0 ve K_0 sürekli YÜKSEK seviyede tutulmalıdır. Q_1 çıkışı ise Q_0 'ın YÜKSEK olmasını takip eden saat vurularında durum değiştirir. Durum değiştirme 2, 4,6,8, nolu saat vurularında gerçekleşir. Bu değişimi sağlamak için J_1 ve K_1 girişlerine Q_0 bağlanır. $Q_0=1$ ve saat vurusunun yükselen kenarı geldiğinde FF1 tümleyen kipte çalışır. Q_2 'ye bakacak olursak bu çıkış sadece Q_0 ve Q_1 çıkışlarının her ikisi birden YÜKSEK olduğunda durum değiştirmiştir. J_2 ve K_2 girişlerine bu iki çıkışın VE'lenmiş hali uygulanmalıdır. Ne zamanki Q_0 ve Q_1 çıkışlarının her ikisi birden YÜKSEK olursa VE geçidi çıkışı 1 olur ve bu 1 J_2 ve K_2 girişlerine uygulanarak FF2'nin tümleyen kipte çalışmasını sağlar. Çıkışlardan herhangi biri sıfır olduğunda J_2 ve K_2 girişlerine sıfır uygulandığından FF2 değişmez kipinde çalışacak ve Q_2 durum değiştirmeyecektir.



Şekil-10.14 3 bit senkron sayıcı



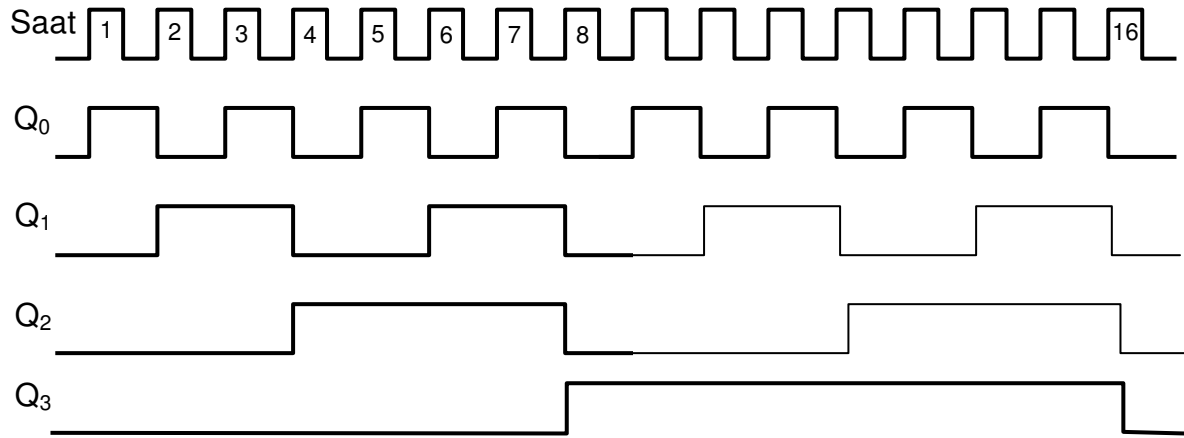
Şekil-10.15 3 bit senkron sayıcı zamanlama diyagramı.



Şekil-10.16 4 bit senkron sayıcı

4 BİT SENKRON SAYICI

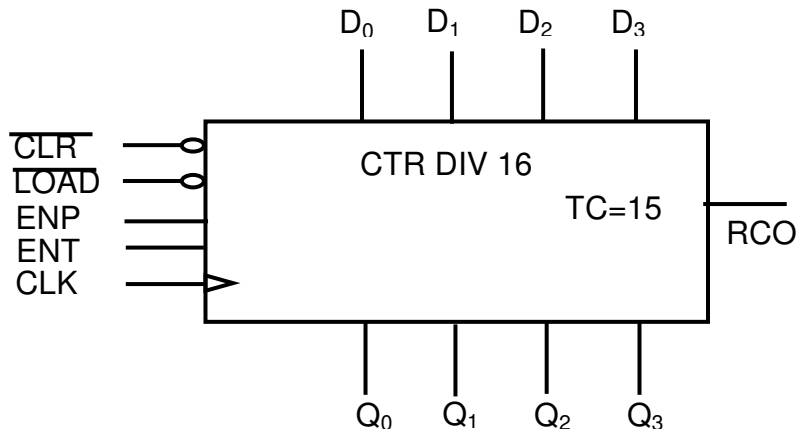
Şekil-10.16'da 4 bit senkron sayıcının bağlantısı, şekil-10.17'de zamanlama diyagramı gösterilmiştir. İlk üç Flip-Flop da herhangi bir değişik bağlantı yoktur, 3 bit senkron sayıcı da olduğu gibi yapılmıştır. Dördüncü Flip-Flopta durum değişimi diğer üç Flip-Flopun çıkışları YÜKSEK olduğunda gerçekleşir. Q_0 Q_1 çıkışları VE'lenmiştir. Bu geçidin çıkışını Q_2 ile VE'lediğimizde J_3 ve K_3 için gerekli olan YÜKSEK seviye elde edilir.



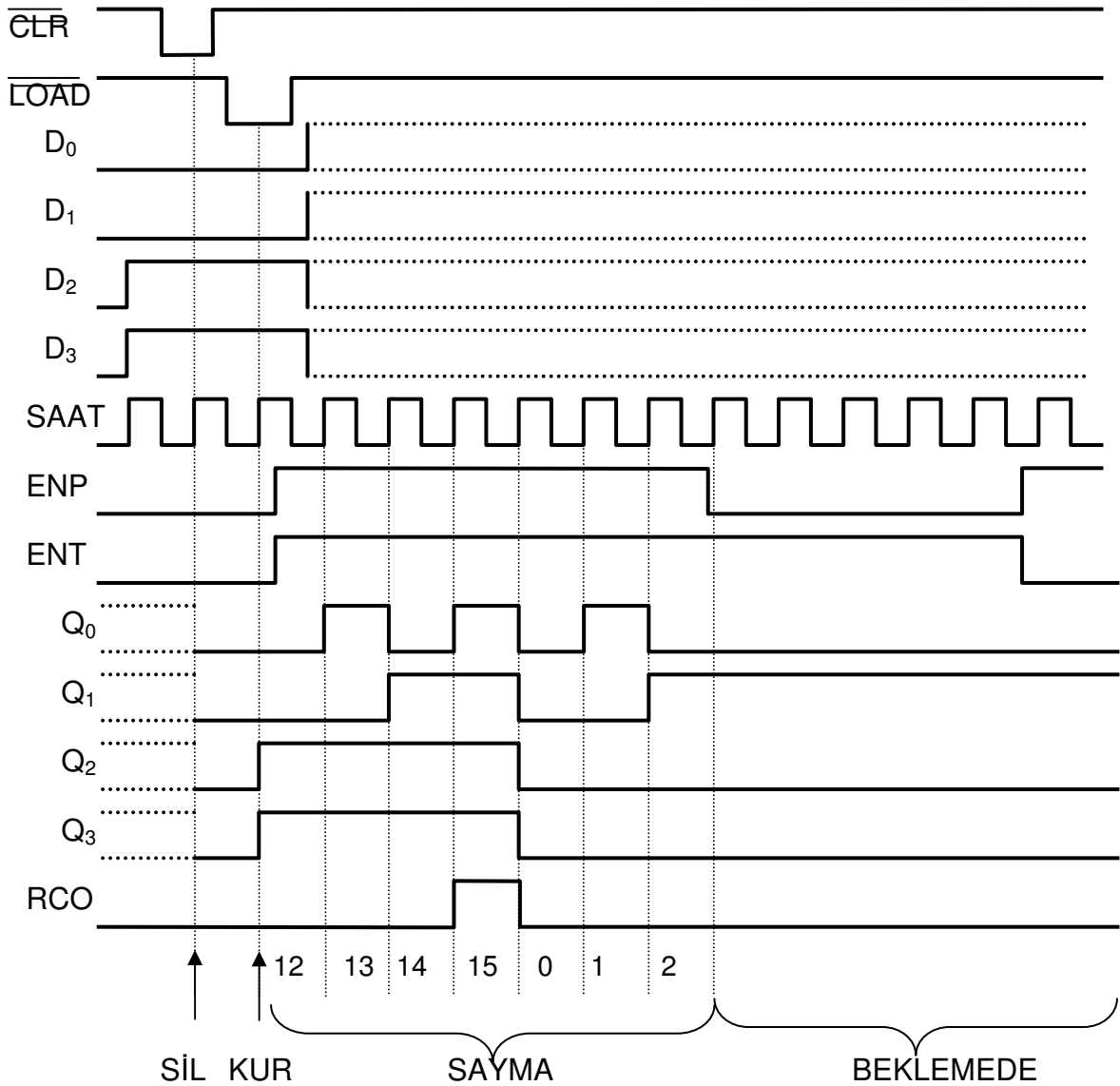
Şekil-10.17 4 bit senkron sayıcı zamanlama diyagramı.

74LS163A 4 BİT İKİLİK SENKRON SAYICI

74LS163A ikilik senkron sayıcı olarak üretilmiş bir tümdevredir. Bu tümdevre daha önce açıklanan senkron sayıcı özelliklerine ek olarak bir çok özelliğe sahiptir. Şekil-10.18'de mantık sembolü gösterilmiştir. Bunlardan bir tanesi sayıcı başlangıç değeri olarak sayıcının sayma aralığında herhangi bir sayıya ayarlanabilir. Bu işlem için paralel veri girişlerine istenilen sayı, LOAD girişine de bir DÜŞÜK seviye uygulanmalıdır. Bu işlem sonrası bir sonraki saat vurusunda girilen bu değer sayıcının çıkış durumu olacaktır.



Şekil-10.18 74LS163A'nın mantık gösterimi.

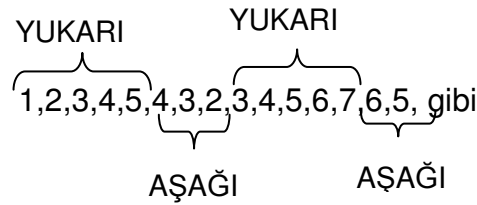


Şekil-10.19 74LS163A'nın çalışmasına bir örnek.

Aktif DÜŞÜK CLR girişi tüm Flip-Flopları sıfırlar. İki adet İZİN girişi, ENP ve ENT sayıcının normal sırada sayabilmesi için YÜKSEK'te tutulmalıdır. Kas kat bağlama sırasında bir üst basamağa izin vermek için bu İZİN girişleri kullanılacaktır. Ripple clock output (RCO) çıkışı ise sayma değeri en büyük duruma ulaştığında bir YÜKSEK seviyeli vuru üretir. Bu sayıcı için en büyük sayma durumu $(1111)_2$, yani onluk 15'tir.

YUKARI/AŞAĞI SENKRON SAYICILAR

Aşağı/yukarı sayıcılar belirlenen şekilde ileri ve geri doğru sayabilirler. Bazı kaynaklarda çift yönlü (bidirectional) sayıcı adı da verilir. Aşağı/yukarı sayıcıdan beklenen sayma durumunun her hangi birinde geri doğru sayabilmesidir. Aşağıda sayıcıdan beklenen davranış onlu sayılar ile örneklenmiştir.



Tablo-10.4'te 3 bitlik bit senkron aşağı/yukarı sayıcının tablosu verilmiştir. Oklar sayıcının bulunduğu durumdan verilen yöne göre hangi yöne doğru sayma yapacağını gösterir. Q_0 çıkışı incelendiğinde her iki yönde de FF0'ın tümleyen kipte çalışması gerektiği sonucu çıkmaktadır.

$$J_0=K_0=1$$

Yukarı doğru saymada Q_1 durumunu $Q_0=1$ durumunu takip eden saat vurularında değiştirir. Aşağı doğru saymada ise Q_1 durumunu $Q_0=0$ durumunu takip eden saat vurularında değiştirir. FF1'in J_1 ve K_1 girişleri aşağıdaki eşitliğe göre YÜKSEK olmalıdır.

Saat Vurusu	Yukarı	Q_2	Q_1	Q_0
0	→	0	0	0
1	→	0	0	1
2	→	0	1	0
3	→	0	1	1
4	→	1	0	0
5	→	1	0	1
6	→	1	1	0
7	→	1	1	1

Tablo-10.4

$$J_1 = K_1 = Q_0 \bar{Q}_3$$

Yukarı doğru saymada Q_2 durumunu $Q_0=Q_1=1$ durumunu takip eden saat vurularında değiştirir. Aşağı doğru saymada ise Q_2 durumunu $Q_0=Q_1=0$ durumunu takip eden saat vurularında değiştirir. FF1'in J_1 ve K_1 girişleri aşağıdaki eşitliğe göre YÜKSEK olmalıdır.

$$J_2 = K_2 = Q_0 Q_1$$

Bu eşitliklerden yola çıkarak 3 bit senkron yukarı/aşağı sayıcı devresi Şekil-10.20'de gösterilmiştir. Yukarı/aşağı girişi DÜŞÜK olduğunda aşağı, YÜKSEK olduğunda yukarı sayma yapacaktır.

$$J_3 = K_3 = Q_0 Q_1 Q_2 + Q_0 Q_3$$

Örnek:

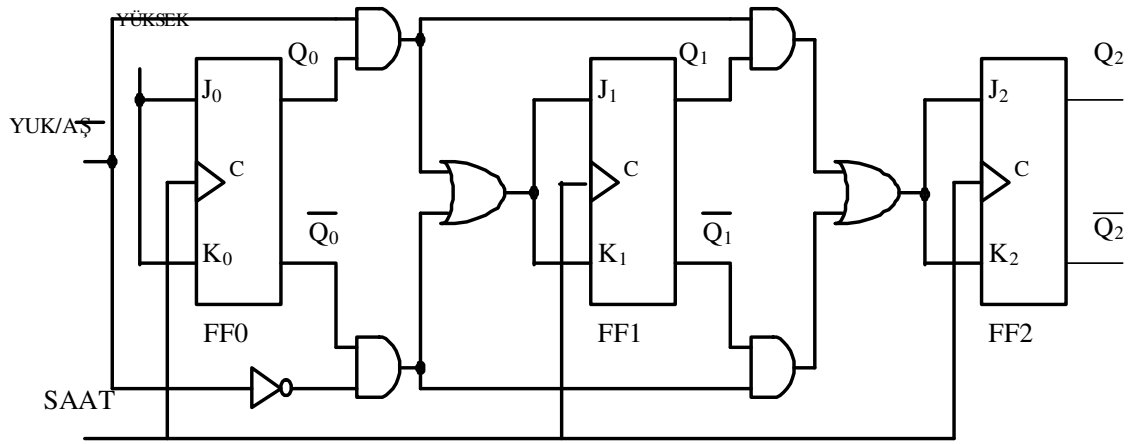
Şekil-10.21'de yukarı/aşağı ve saat işaretleri verilen 4 bit senkron yukarı/aşağı sayıcının çıkış dalga şekillerini ve durum değişim tablosu çizin. Sayıcı 0000 durumundan saymaya başlar ve Flip-Floplar yükselen kenar tetiklemelidir.

ÇÖZÜM:

Saat Vurusu	Q_3	Q_2	Q_1	Q_0
Başlangıçta	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	0	1	1
6	0	0	1	0
7	0	0	0	1
8	0	0	0	0
9	1	1	1	1
10	0	0	0	0
11	0	0	0	1

Tablo-10.5 Yukarı/aşağı sayıcının sayma adımları

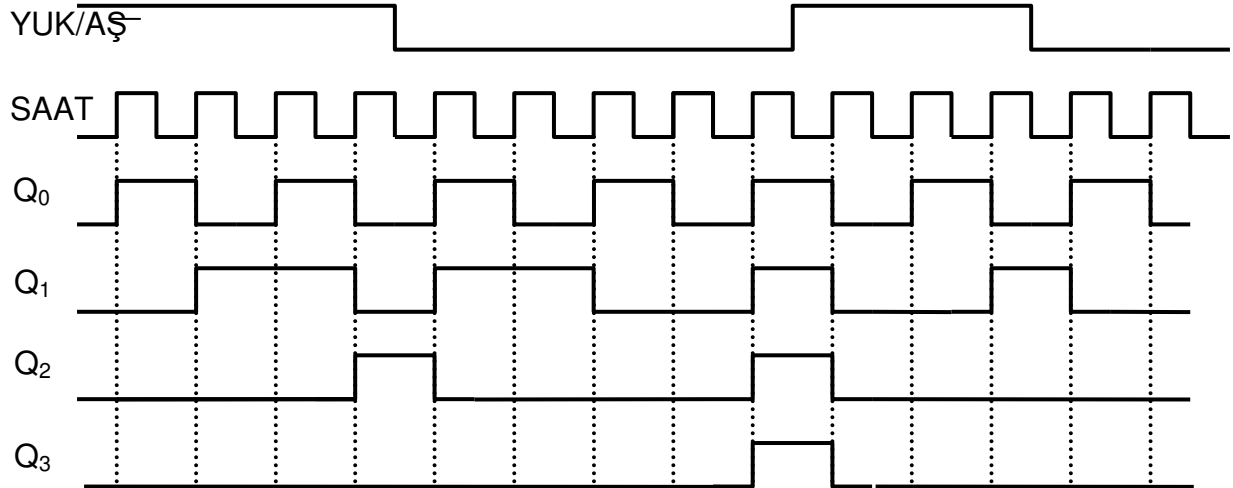
4 bit senkron yukarı/aşağı sayıcının çıkış dalga şekilleri Şekil-10.21'de çizilmiştir. Sayıcının durum değiştirme tablosu Tablo-10.5'te gösterilmiştir.



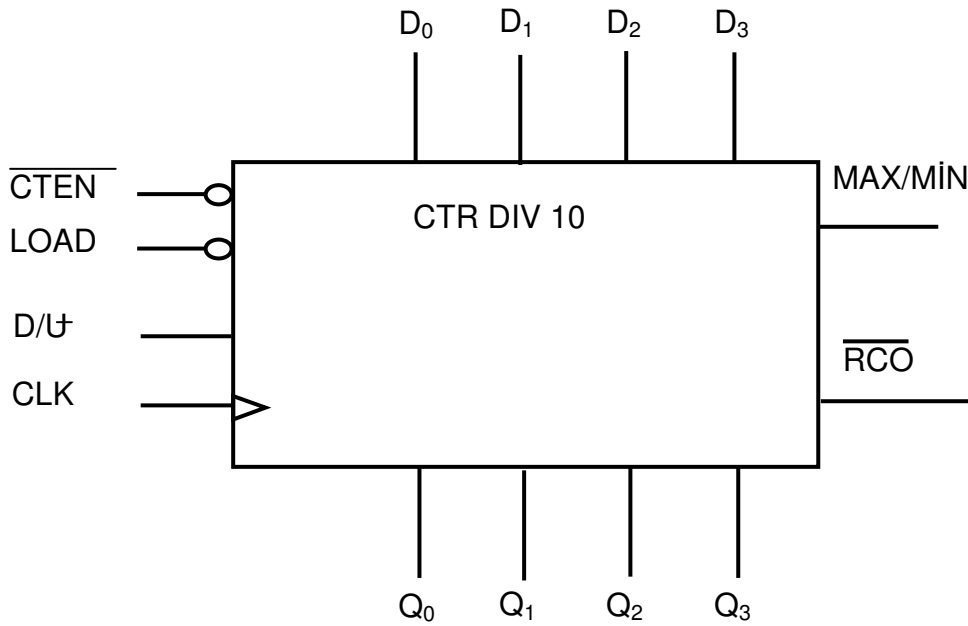
Şekil-10.20 3 bit senkron aşağı/yukarı sayıcı bağlantısı.

74190 YUKARI/AŞAĞI İKO (BCD) SAYICI

Bizler onluk sayı düzenini kullanmaya alışkın olduğumuzdan ikilik sayı düzenini kullanan sayısal devreler bu onluk sayıları ikilik düzene dönüştürüp işlemleri yaptıktan sonra yeniden onluk düzene çevirmek zorundadır. Onluk sayıları ikilik sayılara çevirmek için sayısal devrelerin en çok kullandığı kodlardan biri İkili Kodlanmış Onluk (İKO) [Binary Coded Decimal (BCD)] 'dir. Bir İKO sayıcı her saat darbesi ile İKO düzende sayma yapan bir devredir. Çıktısı İKO biçimindedir ve başka bir devre aracılığıyla sayısal formda kodlanması gerekir.



Şekil-10.21 4 Bit senkron yukarı/aşağı sayıcı.



Şekil-10.23 74LS190'ın mantık gösterimi.

74190 tümdevresi bir senkron yukarı/aşağı İKO (BCD) sayıcıdır.

A, B, C, D girişler: Sayıcıya LOAD ile bu girişlerdeki İKO sayı yüklenebilir ve sayma bu değerden başlar.

QA, QB, QC, QD çıkışlar: Sayma değerleri bu çıkışlardan alınır.

CTEN (Counter Enable): Saymaya İZİN verme girişi, aktif-düşük seviyeli

RCO (Ripple Clock Output): Son sayma değerine ulaşıldığında DÜŞÜK seviyeye geçen bu çıkış CTEN girişi ve MAX/MIN çıkışı ile birlikte kaskat bağlantılar için kullanılmaktadır. Aktif-DÜŞÜK seviyelidir.

MAX/MIN: Yukarı sayma kipinde 9 (1001), aşağı sayma kipinde 0 (0000) sayma değerine ulaşıldığında bir YÜKSEK seviyeli vuru üreten çıkış.

D/U : Yüksek seviyeli iken aşağı, düşük seviyeli iken yukarı sayma yaptıran denetim girişi.

CLK : Saat girişi

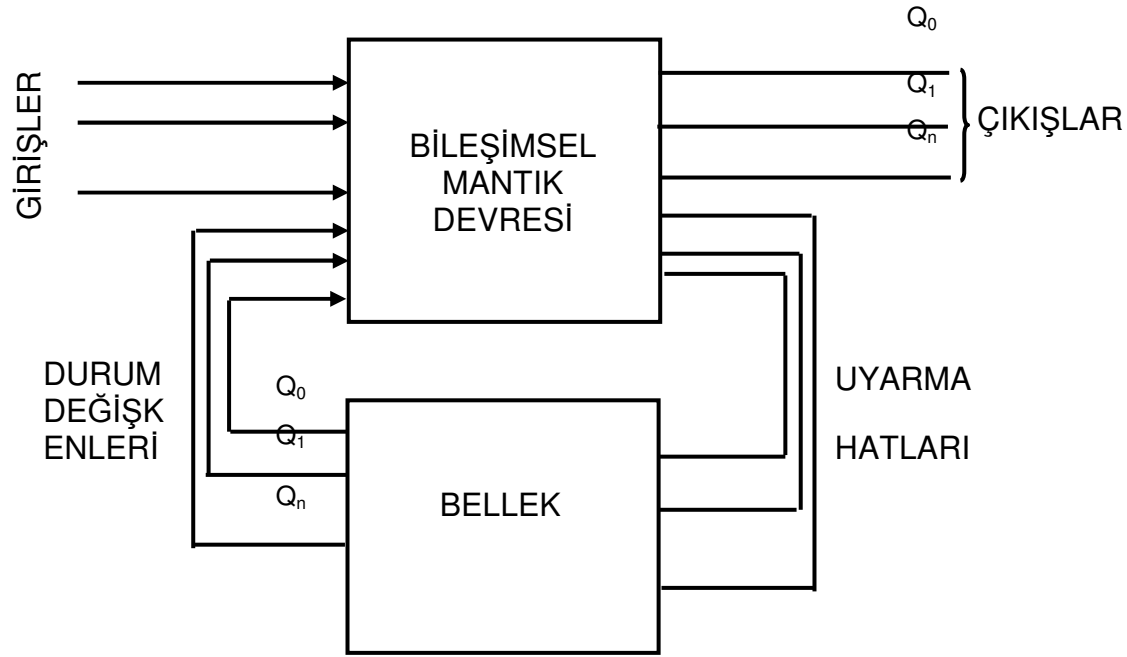
LOAD (YÜKLE): Düşük seviyeli olduğunda A-D arası girişlerdeki bilgiyi sayıcıya yükler.

NOT: 74190 tümdevresinin zamanlama grafiği için veri yapraklarına bakınız.

SENKRON SAYICI TASARIMI

SIRALI ARDIL (SEQUENTIAL) DEVRE MODELİ

Sıralı ardıl devre birleşimsel mantık devresi ve bellek kısmından oluşur. Sıralı ardıl devrenin saat girişi bellek kısmına uygulanır. Bir sonraki durumda gerekli sonuçlar bellek kısmında saklanır. Saklanan bilgiler ile girişe uygulanan işaretler birleşimsel mantık devresinde uygun mantık işlemine sokulur ve sonraki durum elde edilir. Saklanan duruma *şimdiki durum* mantık işlemi sonrası oluşan duruma ise *gelecek durum* adı verilir. Şekil-10.24'te ardıl devrenin blok bağlantısı verilmiştir.

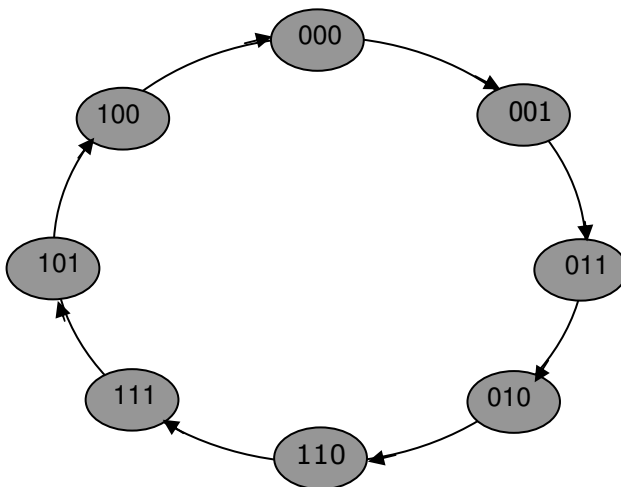


Şekil-10.24 Sıralı ardıl devre modeli.

Bu model tüm ardıl devrelerde geçerlidir. Sayıcılar ise ardıl devrelerin özel bir uygulamasıdır. Durumlarını saat işaretine göre değiştirirler. Şimdi senkron sayıcı tasarılmanın adımlarını öğrenelim.

Q ₃	Q ₂	Q ₁	Q ₀	
0	0	0	0	
0	0	0	1	
0	0	1	0	Yukarı
0	0	1	1	
0	1	0	0	
0	0	1	1	
0	0	1	0	
0	0	0	1	Aşağı
0	0	0	0	
1	1	1	1	
0	0	0	0	
0	0	0	1	Yukarı
0	0	1	0	
0	0	0	1	Aşağı
0	0	0	0	

Tablo-10.6

ADIM 1: DURUM DİYAGRAMININ ÇİZİMİ

Şekil-10.25 durum diyagramı.

Durum diyagramında sayıcının her saat vurusu geldiğinde ilerleyeceği durumlar

sıra ile gösterilir. Şekil-10.25'te gram kodunda sayan sayıcının durum diyagramı gösterilmiştir. Bu devrenin saat vurusundan başka girişi, Flip-Flop çıkışlarından başka çıkışı yoktur.

ADIM 2: GELECEK DURUM TABLOSUNUN ELDE EDİLMESİ

Gelecek durum tablosu durum diyagramından elde edilir. Başlangıç değeri şimdiki değer olarak tablonun birinci satırına yazılır. Saat vurusu uygulandığında oluşacak durum gelecek durum olarak ikinci satıra yazılır. İkinci saat öncesi bu durum şimdiki durumdur, gelecek durum ise ikinci saat vurusu geldiğinde oluşacak durum olarak üçüncü satıra yazılır. Diğer satırlarda aynı mantığa göre düzenlenir.

ŞİMDİKİ DURUM			GELECEK DURUM		
Q ₂	Q ₁	Q ₀	Q ₂	Q ₁	Q ₀
0	0	0	0	0	1
0	0	1	0	1	1
0	1	1	0	1	0
0	1	0	1	1	0
1	1	0	1	1	1
1	1	1	1	0	1
1	0	1	1	0	0
1	0	0	0	0	0

Tablo-10.7 3 bit gram sayıcının gelecek durum tablosu

ADIM 3: GEÇİŞ TABLOSU

Tablo-10.8'de J-K Flip-Flopun geçiş tablosu verilmiştir. Devrelerin çalışması incelenirken giriş değerlerine göre çıkışı belirlenir. Sayıcı tasarımında çıkış bellidir. Bu çıkışın elde edilebilmesi ve sayma işleminin istenilen sırada olması için giriş verilmesi gerekli değerler belirsizdir. Flip-Flop geçiş tabloları kullanılarak giriş değerleri bulunur. Tabloda Q_N şimdiki çıkış durumunu, Q_{N+1} ise gelecek çıkış durumunu gösterir. J ve K Flip-Flopun girişleri, X fark etmez durumları için kullanılmıştır. Tablo-10.7 ile Tablo-10.8 birleştirirsek her Flip-Flop için giriş J, K değerlerini belirleyebiliriz. Tablo-10.9 bu değerler verilmiştir.

Q_N	Q_{N+1}	J	K
0	0	0	X
0	1	1	X
1	0	X	1
1	1	X	0

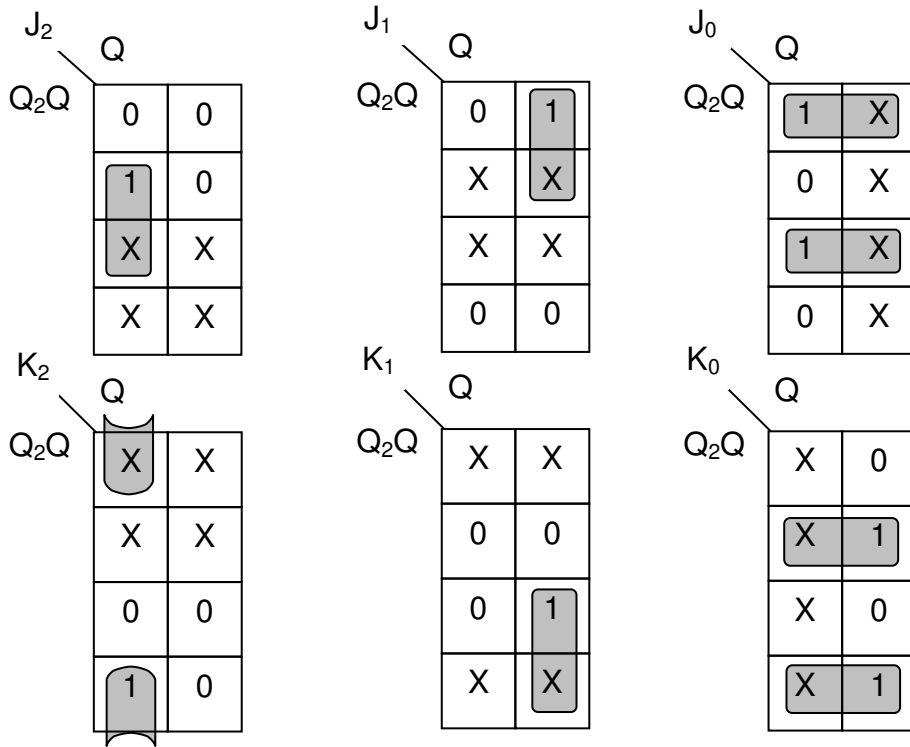
Tablo-10.8 J-K Flip-Flop geçiş tablosu

ŞİMDİKİ DURUM			GELECEK DURUM			FF2		FF1		FF0	
Q_2	Q_1	Q_0	Q_2	Q_1	Q_0	J_2	K_2	J_1	K_1	J_0	K_0
0	0	0	0	0	1	0	X	0	X	1	X
0	0	1	0	1	1	0	X	1	X	X	0
0	1	1	0	1	0	0	X	X	0	X	1
0	1	0	1	1	0	1	X	X	0	0	X
1	1	0	1	1	1	X	0	X	0	1	X
1	1	1	1	0	1	X	0	X	1	X	0
1	0	1	1	0	0	X	0	0	X	X	1
1	0	0	0	0	0	X	1	0	X	0	X

Tablo-10.9 Flip-Flop girişlerinin belirlenmesi

ADIM 4: KARNAUGH HARİTALARI

Her Flip-Flop J ve K girişlerinin en sade fonksiyonlarını Karnaugh haritası ile elde edebiliriz. Her giriş Karnaugh haritasına yerleştirildikten sonra en büyük gruplar seçilir ve en sade fonksiyonu yazılır. Şekil-10.26'da tüm Karnaugh haritaları ve gruplamalar gösterilmiştir.



Şekil-10.26 J, K girişleri için Karnaugh haritaları

ADIM 5: EŞİTLİKLERİN ELDE EDİLMESİ

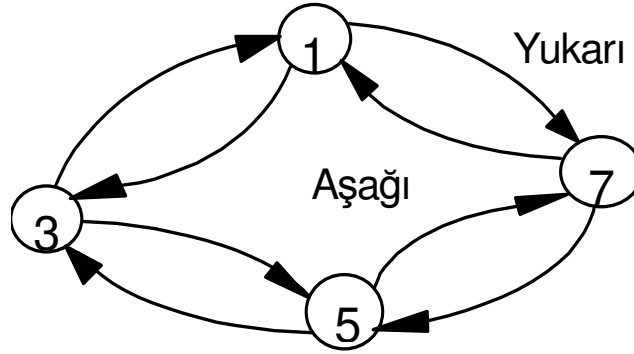
Elde edilen eşitlikler yanda verilmiştir. J_0 ve K_0 girişleri EXOR geçitleri ile yapılırsa daha az geçit kullanarak çizilebilir. Diğer giriş eşitlikleri tek geçitle elde edilebilir.

$$J_0 = Q_1Q_2 + \bar{Q}_1\bar{Q}_2 = \overline{Q_1 \oplus Q_2}, \quad K_0 = Q_1\bar{Q}_2 + \bar{Q}_1Q_2 = Q_1 \oplus Q_2$$

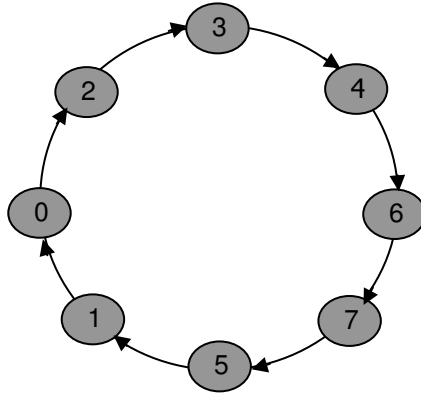
$$J_1 = \bar{Q}_2Q_0, \quad K_1 = Q_0Q_2$$

$$J_2 = Q_1\bar{Q}_0, \quad K_2 = \bar{Q}_1\bar{Q}_0$$

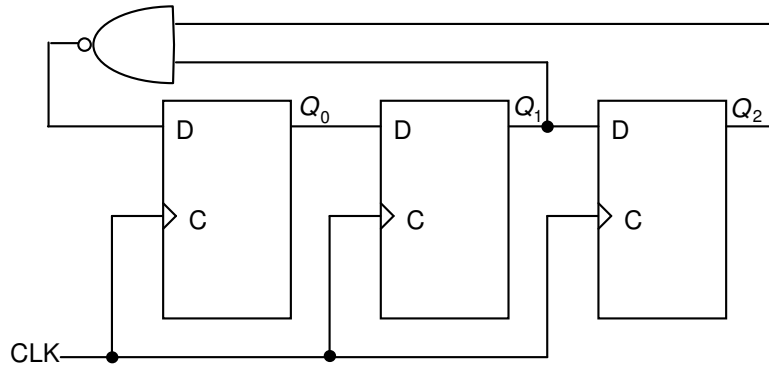
3. Aşağıda durum diyagramı verilen senkron sayıcıyı tasarlayın.



4. Durum diyagramı aşağıda verilmiş senkron sayıcıyı tasarlayın, devresini çizin.

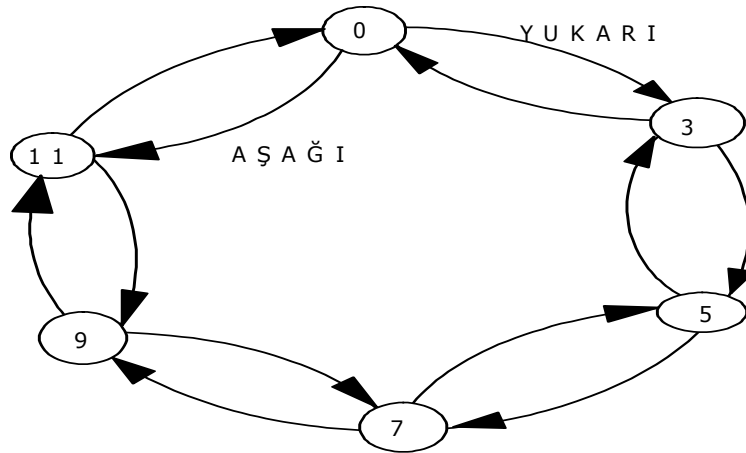


8. Aşağıdaki şekilde sayıcının sayma sırasını belirleyiniz.

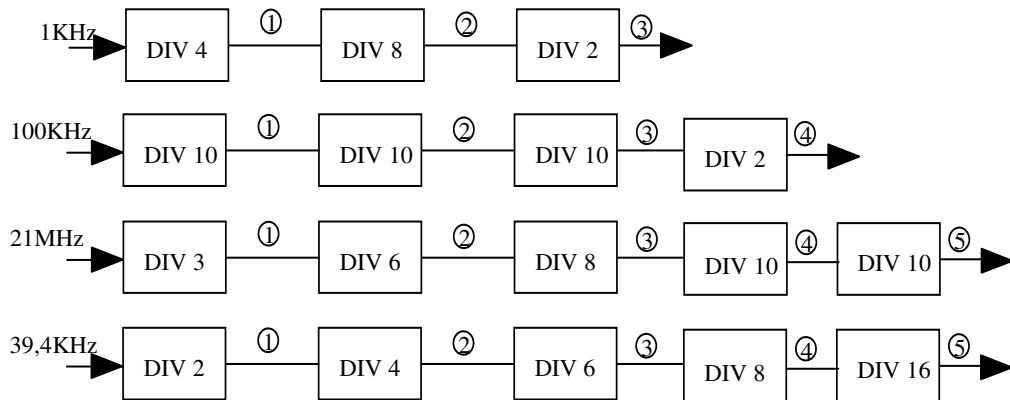


9. J-K flip-floplar kullanarak aşağıdaki sıraya göre sayan sayıcı devreyi tasarlayın.
00, 10, 01, 11, 00,.....
10. J-K flip-floplar kullanarak aşağıdaki sıraya göre sayan ikili sayıcı devreyi tasarlayın.
0, 9, 1, 8, 2, 7, 3, 6, 4, 5, 0

11. Aşağıdaki durum diyagramına göre sayan sayıcı devreyi tasarlayın.



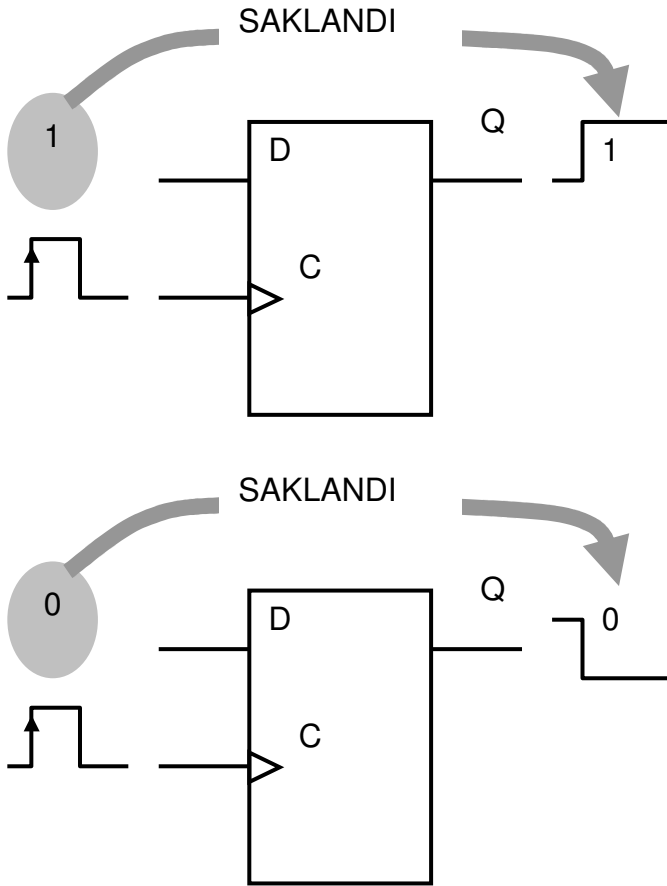
12. Şekilde gösterilen kaskat sayıcıların yuvarlak içinde numara ile belirtilen noktalarındaki frekans değerlerini yazınız.



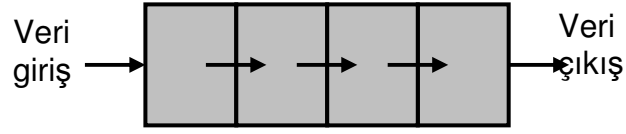
BÖLÜM 11

KAYAR YAZAÇLAR

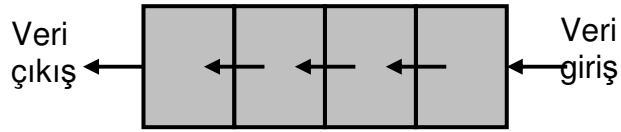
Yazaçlar flip-floplardan oluşur ve sayısal veriyi saklamak ve/veya işlemek için kullanılmaktadır. Her flip-flop bir bitlik veri saklama kapasitesine sahip olduğundan bir yazaçtaki flip-flop sayısı saklanacak veya işlenecek bit sayısı kadardır. Tümdevre yazaçların bit sayısı 4 bitten 4000 bite kadar değişmektedir.



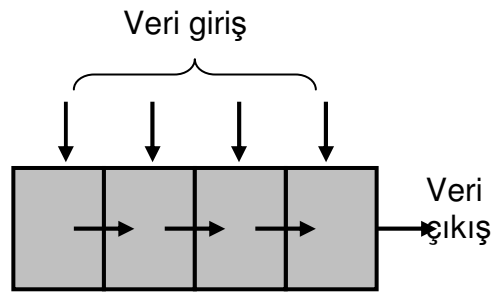
Şekil-11.1 "0" ve "1" bilgisinin flip-flopta saklanması.



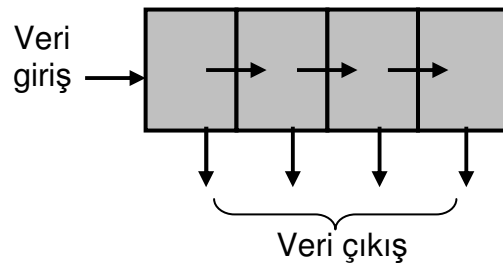
(a) Seri giriş-sağa ötele-seri çıkış.



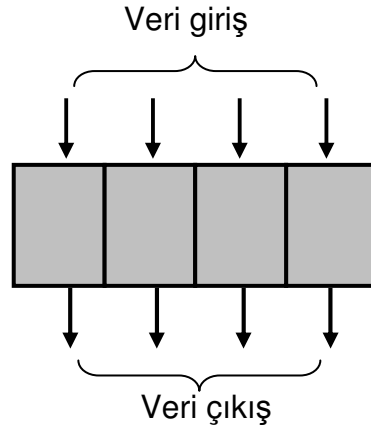
(b) Seri giriş-sola ötele-seri çıkış.



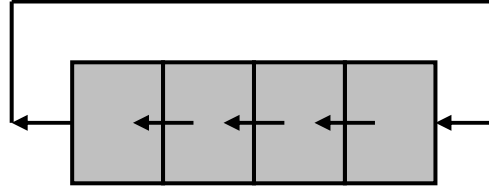
(c) Paralel giriş-sağa ötele-seri çıkış.



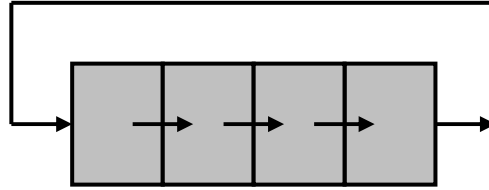
(d) Seri giriş-sağa ötele- Paralel çıkış.



(e) Paralel giriş- Paralel çıkış.



(g) Sola döndür.



(f) Sağa döndür.

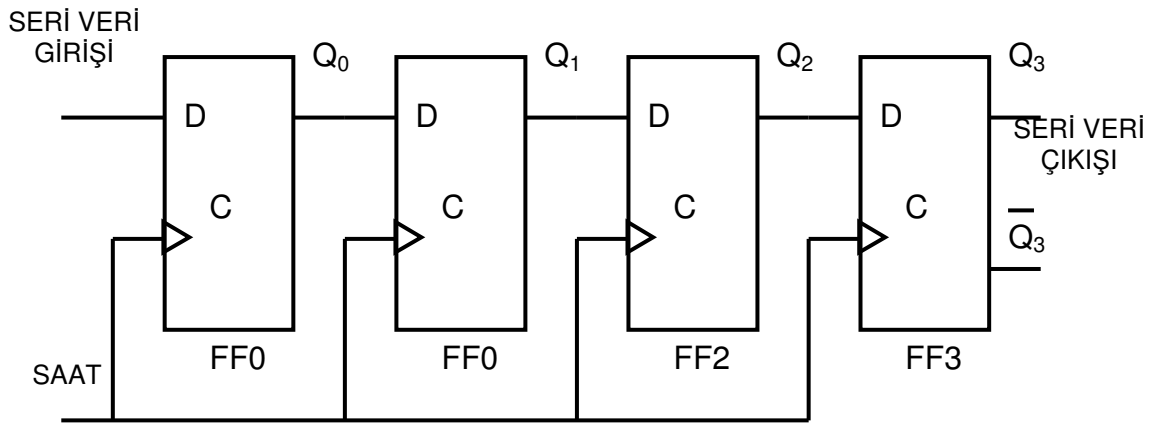
Şekil-11.2 Temel öteleme ve döndürme işlemleri

Yazaçlar iki ana grupta incelenirler. Birinci grup saklama yapan yazaçlar, ikinci grup ise öteleme yapan yazaçlardır. Bir saklama yazacı gerek duyulana kadar sayısal veriyi saklar. Ötelemeli yazaç ise, veriyi bir flip-floptan bir sonrakine öteleyerek işler. Devrenin yapısına bağlı olarak, veri sağa, sola veya hem sağa hem de sola saat darbesiyle, eşzamanlı olarak ötelenir. 4 bit yazaçta öteleme çeşitleri Şekil-11.2'de gösterilmiştir. Bir saklama yazacına veri aynı anda girilir (paralel giriş) ve çıkıştan aynı şekilde alınır (paralel çıkış). Ötelemeli yazaçta veri tek bir hattan her seferinde bir bit girilir veya çıkıştan aynı şekilde alınır (seri giriş veya seri çıkış) veya hepsi bir defada girilir veya çıkıştan alınır (paralel giriş veya

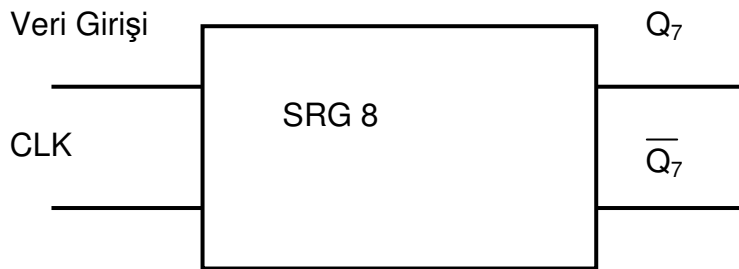
paralel çıkış). Dört giriş/çıkış düzenlemesi şöyledir: Seri giriş/seri çıkış (SISO), seri giriş/paralel çıkış (SIPO), paralel giriş/ paralel çıkış (PIPO), paralel giriş/seri çıkış (PISO). Yazaçlar bir bilgisayarda pek çok işlemi yerine getirmek için kullanılırlar. Ötelemeli yazaçlar veriyi bir flip-floptan diğerine ötelerken çarpma ve bölme işlemlerini yerine getirmektedir. Saklama yazaçları ise komutları ve veriyi bilgisayarın bir başka bölümüne iletilene kadar saklarlar. Ötelemeli yazacın saklama yazacından farkı her flip-flopun veri girişinin önceki flip-flopun çıkışından alınması için bağlı olmasıdır. Bu da verinin flip-floplar içinde kaydırılmasını sağlar. Bilgisayar işlemlerinde bilginin seri formdan paralele veya paralel formdan seriye dönüştürülmesi önem taşımaktadır.

SERİ GİRİŞLİ-SERİ ÇIKIŞLI KAYAR YAZAÇLAR

Şekil-11.3'te seri girişli seri çıkış yazaç 4 adet d flip-flop ile elde edilmiştir. FF0'ın D girişinden girilen veri 4 saat vurusu sonunda FF3'ün Q çıkışından aynen Q' çıkışından terslenerek alınır. Bu tür devreler veriyi belirli bir süre bekletmek amacıyla kullanılır. Şekil-11.4'te 8 bit SISO kayar yazacın mantık simgesi gösterilmiştir.



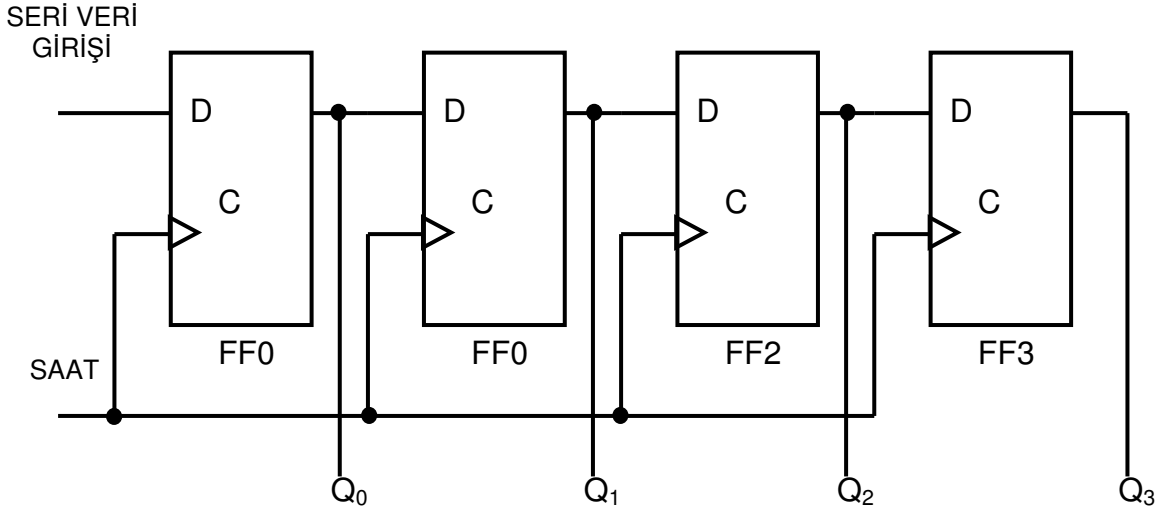
Şekil-11.3 Seri giriş seri çıkış kayar yazaç. (SISO)



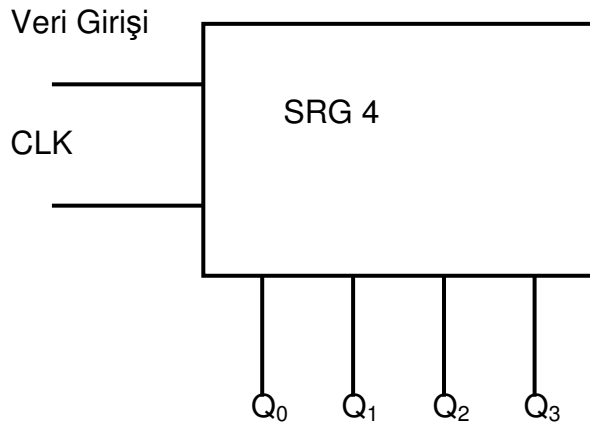
Şekil-11.4 8 bit SISO'nun mantık simgesi.

SERİ GİRİŞLİ PARALEL ÇIKIŞLI KAYAR YAZAÇLAR

Veri biti seri olarak birinci flip-flop'un D girişinden girer birinci saat vurusunda FF0'in Q_0 , ikincide FF1'in Q_1 , üçüncüde FF2'nin Q_2 ve dördüncüde FF3'nin Q_3 çıkışına ulaşır. Veriler düz sırada alınmak isteniyorsa ters sırada kayar yazaca girilmelidir. Şekil-11.5'te D flip-floplarla elde edilmiş 4 bit SIPO yazacın mantık diyagramı verilmiştir. Şekil-11.6'da ise 4 bit SIPO kayar yazacın mantık simgesi verilmiştir.



Şekil-11.5 Seri giriş paralel çıkış kayar yazaç. (SIPO)



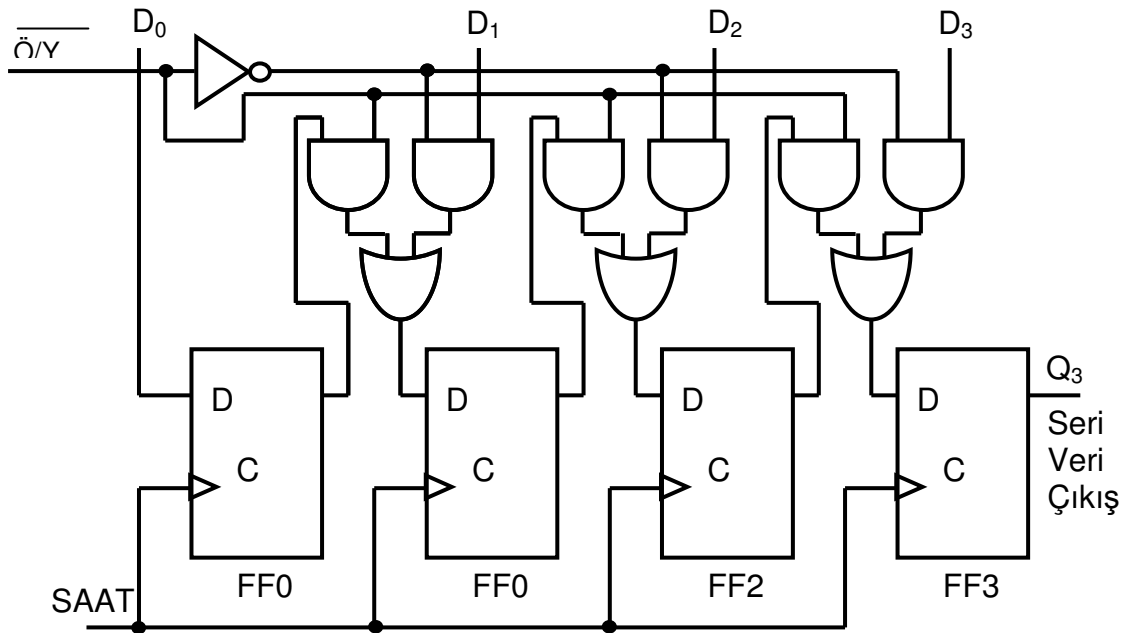
Şekil-11.6 4 bit SIPO'nun mantık simgesi.

Bu tür yazaçlar seri formda saklanmış verileri veya seri formda gelen verileri paralel hale getirmek için kullanılır. Dijital sistemlerde kaliteli veri ancak bit sayısı artırılarak elde edilebilir. Fakat bit sayısı fazla olan sistemlerde veri iletimi çok fazla taşıyıcı hat gerektirdiğinden sistemi özellikle baskı devre kartını karmaşık hale

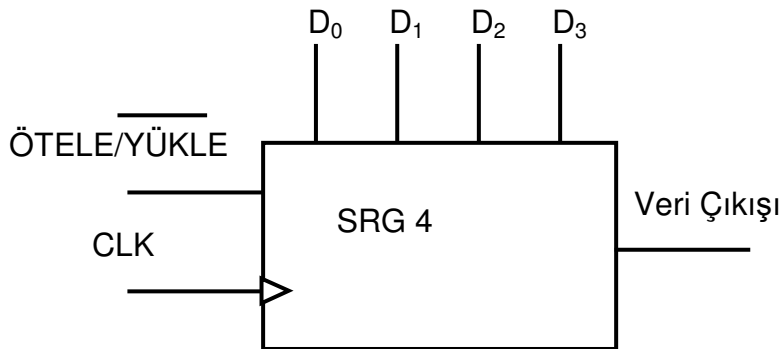
getirir. Bunu engellemek için iki birim arasında veri iletimi hız engeli yoksa seri olarak yapılarak bu karmaşanın önüne geçilebilir. Seriyeye dönüştürülen veriler SIPO kullanılarak tekrar paralele dönüştürülür. Çünkü bir çok sistem veriyi paralel olarak kullanırlar.

PARALEL GİRİŞLİ SERİ ÇIKIŞLI KAYAR YAZAÇLAR

Bir önceki kısımda SIPO yazaçların seri veriyi paralel kullanan sistemler için paralele dönüştürdüğünü sebepleri ile birlikte açıklamıştık. Paralel girişli seri çıkışlı yazaçlarda SIPO'nun tersini yapar. Birden fazla bitten oluşan veri kelimelerinin seri olarak tek hat üzerinden başka birime iletimi sağlamak amacıyla paralel girişli seri çıkışlı (PISO) kayar yazaçlar kullanılır. Genellikle seri veri iletim sistemini verici kısmında PISO alıcı kısmında ise SIPO yer alır.



Şekil-11.7 Paralel giriş seri çıkış kayar yazaç. (PISO).

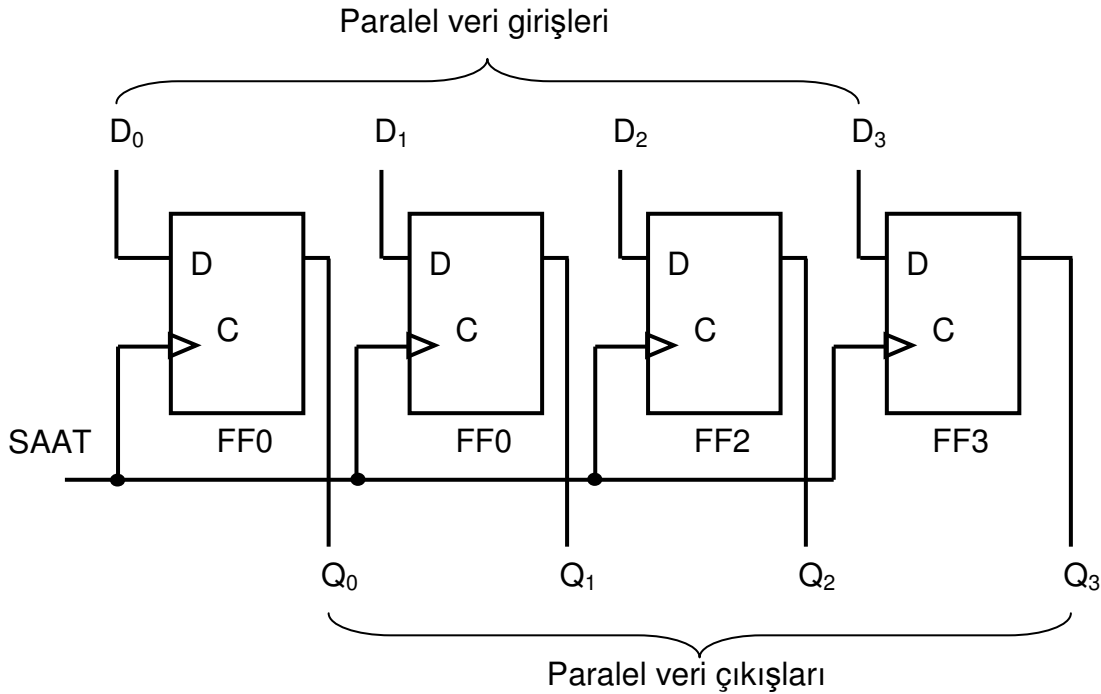


Şekil-11.8 4 bit PISO'nun mantık simgesi.

Şekil-11.7'de PISO kayar yazacın mantık simgesi gösterilmiştir. Şekil-11.8'de ise PISO kayar yazacın mantık diyagramı verilmiştir. Bu yazacın paralel girişlerindeki veri ÖTELE/YÜKLE girişi "00 yapılarak ilgili D flip flop girişlerine yüklenir. Bu giriş "1" yapıldığında D girişlerindeki veriler çıkışlarına ötelenir. Paralel girişteki veri ile ötelenen verinin karışmasını engellemek için geçitler kullanılmıştır.

Paralel Girişli Paralel Çıkışlı Kayar Yazaçlar

Paralel girişli paralel çıkışlı kayar yazaç kısa süreli veri saklama amaçlı kullanılan yazaç türüdür. Aslında bu tip yazaçlara kayar yazaç adı verilmez sadece yazaç olarak adlandırılır. Şekil-11.9'da 4 bitlik paralel girişli paralel çıkışlı (PIPO) kayar yazacın mantık diyagramı, Şekil-11.10'da ise 74195 kayar yazacının mantık simgesi gösterilmiştir.

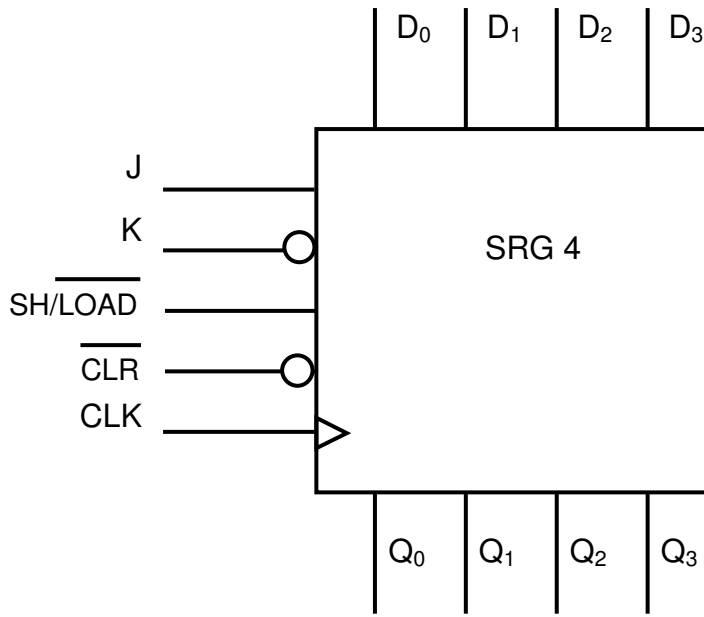


Şekil-11.9 Paralel giriş Paralel çıkış kayar yazaç. (PIPO)

74195 tümdevresinin iki adet seri veri giriş hattı, 4 adet paralel girişi ve 4 adet paralel çıkışı vardır. Bu giriş ve çıkışlar kullanılarak bu tüm devre ile seri giriş seri çıkış, seri giriş paralel çıkış, paralel giriş paralel çıkış, paralel giriş seri çıkışlı kayar yazaç elde etmek mümkündür. Bu tüm devre sadece sağa doğru öteleme yapabilir.

74194 SH/ $\overline{\text{LOAD}}$ girişine DÜŞÜK seviye uygulandığında paralel girişlerde yer alan veri bitleri saat vurusuyla senkron olarak ilgili flip-flop'un çıkışına aktarılır. Bu giriş eğer YÜKSEK seviye yapılırsa yüklenmiş veriler bir bit sağa ötelenir.

CLR girişi asenkron giriş olup, DÜŞÜK seviye uygulandığında paralel çıkışları temizler. J ve K girişleri seri veri girişi olarak kullanılır. Q₃ çıkışı seri veri çıkışı olarak kullanılabilir.



Şekil-11.10 74195 4 bit PISO kayar yazacın mantık simgesi.

ÜNİVERSAL KAYAR YAZAÇ

Sağa sola ötelemenin her ikisini yapabilen kayar yazaçlara universal kayar yazaç adı verilir. 74194 yazacı bu tür bir yazaçtır. Bu yazaçta veri giriş çıkış hatlarının yanı sıra iki adet veri akışını veya kayar yazacın çalışmasını denetleyen iki adet giriş hattı vardır. Tablo-11.1'de bu girişlerin işlevleri verilmiştir.

Yazaçlara veri paralel olarak seri olarak da sağ ve soldan olmak üzere üç şekilde yapılabilir. Yükleme çeşidi S0 ve S1 girişleri ile belirlenebilir, Tablo-11.1'de yükleme çeşidine göre anahtarların alması gereken durumlar verilmiştir.

S1	S0	İşlem	Açıklama
0	0	İşlem yapma	Sıfırlama haricinde tüm işlemler engellenir.
0	1	Sağa kaydır	Bilgi girişi sol taraftaki bilgi girişinden yapılır.
1	0	Sola kaydır	Bilgi girişi sağ taraftaki bilgi girişinden yapılır.
1	1	Paralel yükleme	Bilginin tamamı aynı anda paralel olarak yüklenir.

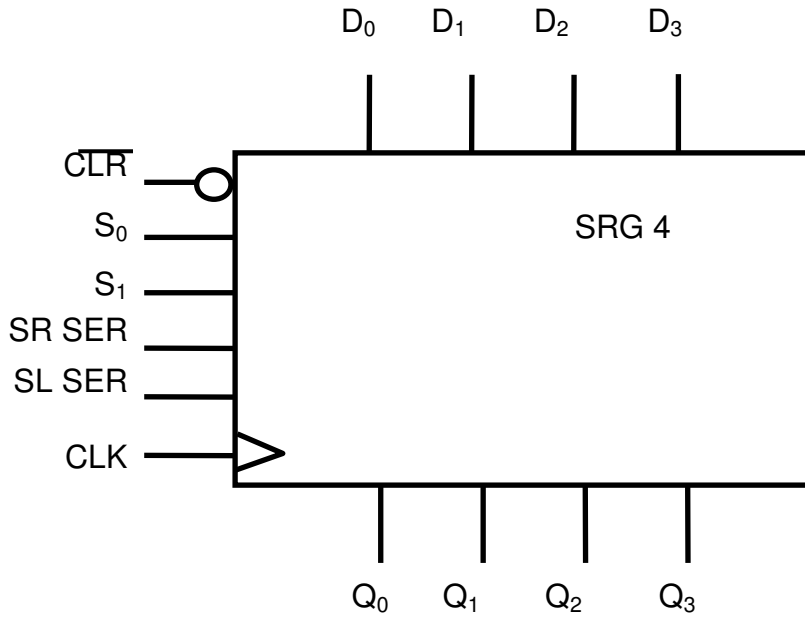
Tablo-11.1 Yazaca veri yüklenmesi.

Girişlerin her ikisi 0 olduğunda yazaç temizleme dışındaki tüm işlemleri yapmaz. Bu durum yazacı başlangıç durumuna getirmek için kullanılır. S1 girişi 0, S0 girişi 1 olduğu durumda sol seri girişte bulunan veri alınır ve diğer Flip-Flop çıkışlarındaki veriler bir sağa ötelenir. En sağdaki Flip-Flop çıkışındaki veri eğer başka devreye bağlı değil ise kaybolur. Bu çalışma modunda seri girişe 0

yazıldığında her ötelemede yazaçta yüklü sayı 2'nin üstlerine bölünür. Bir ötelemede yapıldığında 2^1 'e, iki öteleme yapıldığında 2^2 'ye, üç öteleme yapıldığında 2^3 'e böler... v.b.

Tam bölme olmadığında kalanın saklanabilmesi için en sağdan çıkan veriyi saklayacak yine seri girişli bir yazaca gereksinim vardır. Üçüncü çalışma modunda, S1 girişi 1 ve S0 girişi 0 olduğu durum, en sağda seri girişteki bilgi alınır ve bir sola ötelenir. Eğer seri girişe 0 verisi girilirse her öteleme sonucunda yazaçta yüklü olan sayı 2 ile çarpılır. Çarpma sonucu 4 bitten büyük olursa en değerlikli bittin (en soldaki bit) başka bir yazaca kaydedilmesi gerekir. Dördüncü çalışma modunda, her iki giriş 1 olduğunda, paralel yükleme için kullanılır. Bu çalışma modu bölünecek çarpılacak sayının yüklemesi için kullanılır.

Ötelemeli yazaçların diğer bir kullanım alanı da seri veri iletişimidir. Verici ile alıcı arasındaki mesafenin uzak olduğu sayısal sistemlerde ikili tabandaki verilerin iletimi için kablo maliyetini düşürmek için seri iletişim kullanılır. Paralel veri iletiminde iletilecek verinin genişliği kadar (kaç bitten oluşuyorsa) hat kullanılması gerekir. Oysa seri iletişimde kaç bitlik olursa olsun tüm veri tek hattan vericiden alıcıya ulaştırılabilir. Seri iletişim sistemin hızını düşürebilir fakat kablo maliyetini düşürdüğü için tercih edilmektedir ve yaygın olarak kullanılmaktadır.



Şekil-11.11 74194A Üniversal 4 bit kayar yazaç.

Alıcı ile vericinin senkronize çalışabilmesi için seri iletişim yapan sistemlerde saat işaretinin de vericiden alıcıya iletilmesi gerekir. Paralel yükleme modunda gönderilecek olan sayı yazaca bir saat vurusu ile yüklenir ve daha sonra en yüksek değerli bit birinci sırada gönderilecek ise sola doğru kaydırma yapılır, eğer en düşük değerli bit birinci sırada gönderilecekse sağa kaydırma yapılır. Alıcı kısmı da verici kısmına göre değiştirilmelidir. Dört bitlik veri iletimi için dört bitlik

yazaç kullanılırsa yükleme vurusu ile birlikte birinci veri alıcıya gönderilir. Eğer bu durum sakınca oluşturuyor ise çıkışa bir D tipi tutucu yerleştirilmelidir. Propagasyon gecikmesine önlem olarak alıcı saat vurusu vericinininkinden bir kaç on nano saniye geciktirilirse devrenin doğru çalışması sağlanmış olur. Bu geciktirme devresi uygulaması D tipi tutucuların deneyinde kullanılmıştı. Alıcı kısmında çıkış paralel çıkışlardan alınarak seriye dönüşen veri tekrar paralele dönüştürülmüş olur. Seri veri iletiminde kayar yazaçların öteleme özelliğinin dışında paralel verinin seriye, seri verinin de paralele dönüştürme özellikleri kullanıldı.

Bunların dışında kayar yazaçlar sayısal sistemlerde zaman geciktirme elemanı olarak ve sayıcı olarak kullanılırlar. Zaman geçirme elemanı olarak kayar yazacın seri girişine uygulanan veri çıkışından alınabilmesi için kayar yazaç kaç bitlik ise o kadar öteleme sonrası çıkıştan alınabilir. Örneğin saat frekansı 1kHz olan bir 8 bitlik kayar yazacın girişine verilen veri çıkışından ancak 8 ms sonra alınabilir. 8 ms'ye veri kayar yazaç içerisinde geciktirilmiştir. Sayısal sistemlerde ortak saatle senkronize çalışan devrelerde hızlı çalışan birim ile yavaş çalışan birimin sonuçlarının aynı anda değerlendirilmesi gerektiğinde hızlı çalışan sistemin sonucu bu yöntemle bekletilebilir. Diğer uygulama alanı ise Johnson ve ring (halka) tipi sayıcılardır.

SORULAR

1. 74195 dört bitlik paralel erişimli ötelemeli yazacın veri giriş ve çıkışlarını tanımlayarak J, K, \overline{CLR} ve $\overline{SH/LD}$ girişlerinin bilgi yazılması için hangi mantık düzeylerinde olması gerektiğini yazın. Veri girişine 1100 bilgisi uygulandıktan sonra veri çıkışlarının ne olduğunu yazın. $\overline{SH/LD}$ YÜKSEK, giriş 0000 yapıldığında bir saat darbesi uyguladıktan sonra veri çıkışlarındaki sayıyı yazın.
2. 4 bitlik paralel giriş/seri çıkışlı bir ötelemeli yazacın devresini çizin. (Paralel veri yüklemeyi düşük seviyeli bir kontrol sinyali ile yapın.) $D_0...D_3 = 1011$ verisini paralel yükleyerek bu veriyi seri olarak çıkıştan elde eden dalga şeklini çizin.
3. 74194 dört bitlik çift yönlü ötelemeli yazacın mantık sembolü sağda ve özellikleri aşağıda verilmiştir. $(1100)_2$ verisinin paralel girişlerden yüklenmesi için S_0 , S_1 ve \overline{CLR} girişlerinin mantık düzeylerini yazın. Aynı verinin sağa ötelenmesi için kontrol girişlerinin hangi seviyede olması gerekir? Sayı yüklendikten sonra iki kere sağa, sonra üç kere sola öteleme yaptıktan sonra $Q_3Q_2Q_1Q_0$ çıkışındaki veriyi yazın.

BÖLÜM

12

BELLEKLER VE PROGRAMLANABİLİR ELEMANLAR

Veri saklama aygıtı olarak kullanılan kayar yazaçlar, her ikilide bir bit saklayan küçük ölçekli belleklerdir. Ancak bilgisayarda ve diğer mikroişlemci tabanlı sistemlerde, büyük miktarlarda ikilik verinin geçici yada yarı geçici olarak saklanması gerekir. Bu denli büyük miktarda verinin saklanmasında yazaçların kullanılması hacim ve maliyet olarak olası değildir. Sistemlerde programların ve programların ürettiği, işlediği ve gereksindiği verilerin saklanmasında *bellekler* (memories) kullanılır. Bellekler, yarıiletken, manyetik ve optik bellekler olarak üç ana öbeğe ayrılmıştır.

YARI İLETKEN BELLEKLER

En küçük ikilik veri birimi bittir ve bellekler veriyi genellikle, *bayt* (çoklu) denilen 8-bitlik öbekler yada bunun katları olarak saklarlar. Bir bayt, *nibble* denilen 4-bitlik iki parçaya ayrılabilir. Sözcük (word) denilen ve tam bir bilgiyi oluşturan birim, bir yada daha çok bayt içerir. Sözcük 8-bitten daha kısa olsa da en az bir bayt ile işlenir.

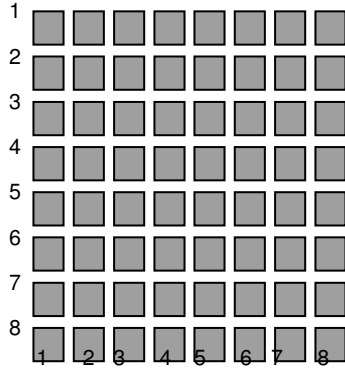
TEMEL YARIİLETKEN BELLEK DİZİMİ

Bellekteki saklama elemanlarına ***hücre*** (cell) denir ve hücrelerin her biri, 1 yada 0 olmak üzere bir bitlik veriyi saklar. Bellekler bu hücrelerden oluşturulan dizimlerden (array) oluşurlar. Bellek dizimindeki her hücre, bir satır ve bir sütun numarası belirtilerek tanımlanabilir. Şekil-12.1'de 64 hücreli bir dizimin elde edilmesi için kullanılacak üç yöntem verilmiştir. Bu dizimlerin tümü 64-bitlik bellek olarak adlandırılabilir gibi, 8×8lik dizime 8-bayt bellek, 16×4lük dizime 16-nibble bellek, 64×1'lik dizime de 64-bit bellek denilebilir. **Bellekler, saklayabilecekleri sözcük sayısı çarpı sözcük uzunluğu olarak tanımlanırlar.** Sözcümleri bir $16k \times 4$ bellek, her biri 4-bit uzunlukta 16384 sözcük saklayabilir.

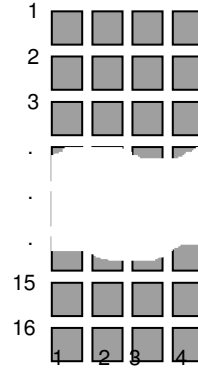
BELLEK ADRES VE KAPASİTESİ

Veri biriminin, bellek dizimi içindeki yerine adres denir. Adres Şekil-12.2 (a)'da bitin satır ve sütun numarası ile Şekil-12.2 (b)'de ise baytın satır numarası ile belirlenmiştir. Buna göre adreslemenin, verilerin bellek içinde nasıl düzenlendiğine bağlı olduğunu söyleyebiliriz. Bellek kapasitesi de saklanabilecek toplam bit sayısı ile ifade edilir. Buna göre Şekil-12.2'deki her iki düzenlemenin kapasiteleri birbirine

eşit ve 64-bittir.



(a) 8x8 dizim

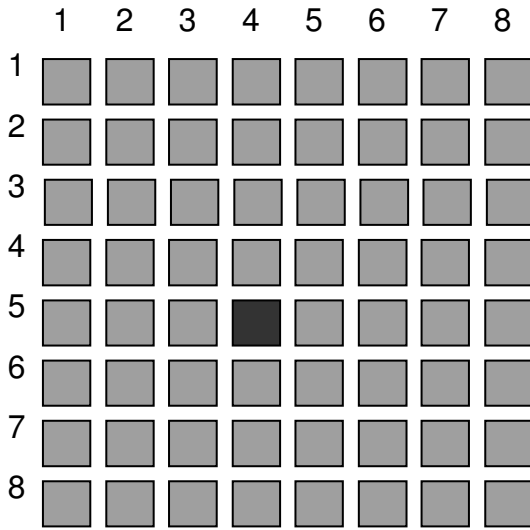


(b) 16x4 dizim

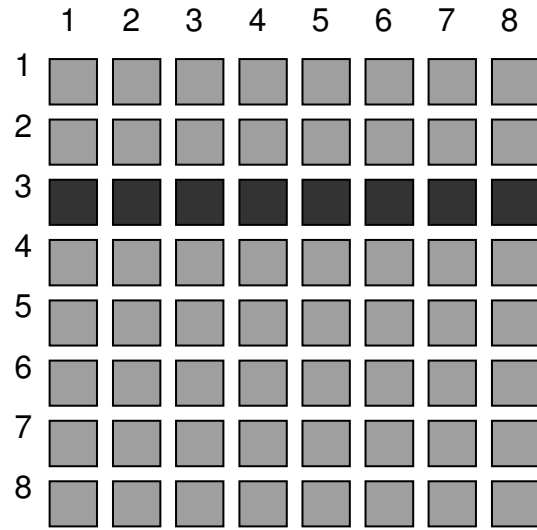


(c) 64x1 dizim

Şekil-12.1 Üç farklı şekilde hücrelerin dizimi



(a) Bitin adresi satır 5, sütun 3



(b) Baytın adresi satır 3

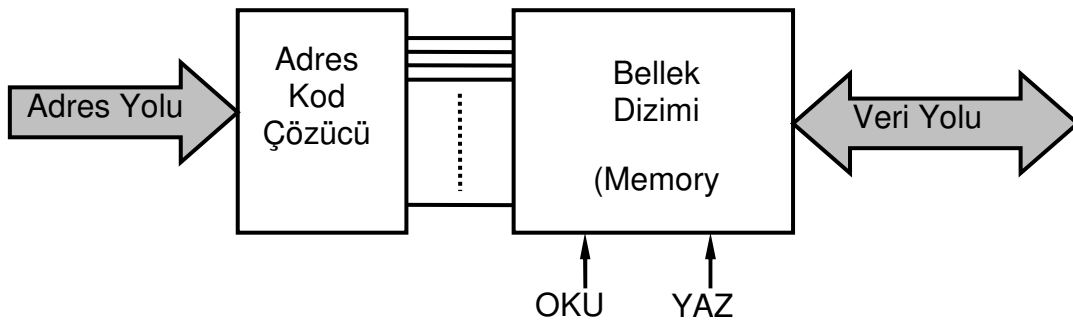
Şekil-12.2 Belleğin adreslenmesi.

TEMEL BELLEK İŞLEMLERİ

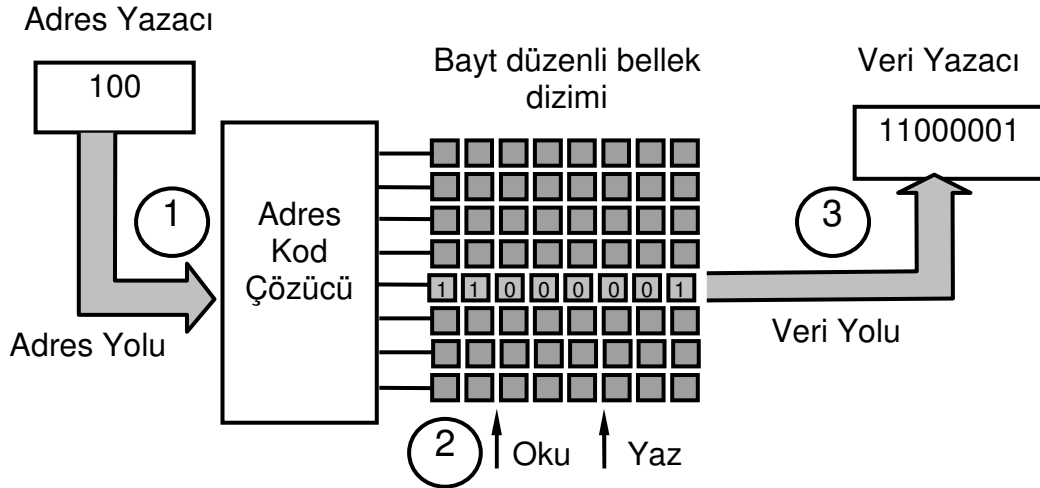
Verinin ve/veya bilginin bellekte belirli bir adrese koyulmasına **yazma** (write), bellekte belirli bir adresten alınmasına da **okuma** (read) işlemi denir. Okuma ve yazma işlemlerinin bir parçası olan adresleme işlemi, işlem yapılacak adresi seçer. Veri belleğe yazılırken ve bellekten okunurken veri yolu (data bus) olarak adlandırılan iki yönlü (bidirectional) bir hat dizisinden geçer.

Okuma işlemi Temel okuma işlemi Şekil-12.4'te verilmiştir. Bu işlemde de önce yazaçtaki adres bilgisi adres yoluyla adres kod çözücüsüne ulaştırılır. Kod çözücü okuma yapılacak bellek satırını seçer ve uygulanan **oku** komutuyla seçilen adresteki verinin bir kopyası, veri yoluna koyulup geçici olarak veri yazacına yazılır. Okuma işlemi sonunda, okunan veri bellekten silinmez.

Yazma işlemi Temel yazma işlemi Şekil-12.5'te verilmiştir. Bir baytı belleğe yazmak için, adres yazacında bulunan adres kodu adres yoluna (address bus) koyulur ve adres kod çözücüsüne iletilir. Kod çözücü kodu alınca belirttiği bellek yerini bularak seçer. Sonra belleğe bir **yaz** komutu verilerek, veri yazacında hazır tutulan veri, veri yolu ile belleğe gönderilir ve bellek yerindeki eski verinin yerine yenisi yazılır. Belleğe yeni veri yazılınca eskisi yok olur.

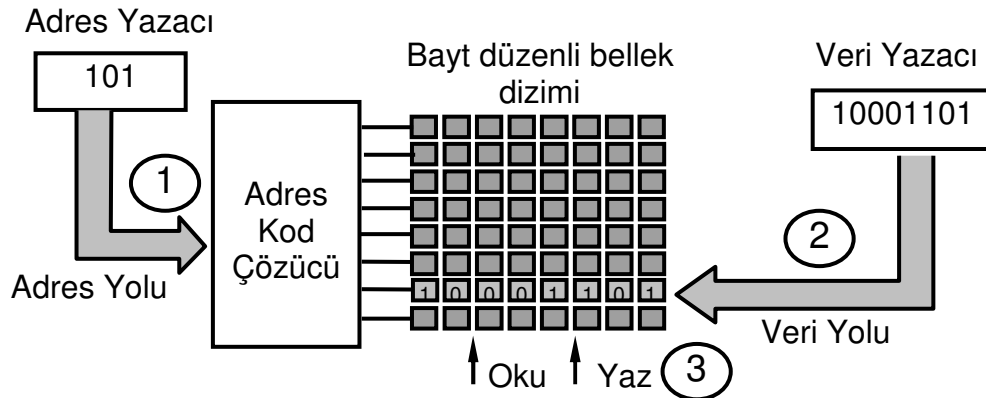


Şekil-12.3 Bellek blok diyagramı.



1. Adres kodu adres yoluna verilerek adres 4 seçilir.
2. Oku komutu belleğe uygulanır.
3. 4 nolu bellek satırın içeriği veri yolu kullanılarak veri yazacına alınır.

Şekil-12.4 Okuma işlemi.



1. Adres kodu adres yoluna verilerek adres6 seçilir.
2. Veri baytı veri yoluna yazılır.
3. Yaz komutu ile önceki veri silinerek yeni veri adres6 ya yazılır.

Şekil-12.5 Yazma işlemi.

RAM'LAR VE ROM'LAR

Yarıiletken belleklerin iki ana gurubu RAM ve ROM belleklerdir. RAM (random-access memory/doğrudan erişimli bellek), okuma ve yazma işlemleri için bütün adreslerine eşit sürede ve istenilen sırada erişilebilen bir bellek türüdür. Bütün RAMlerde hem okuma hem de yazma işlemleri yapılabilir. RAM, uygulanan gerilim kesilince yazılı verileri yitirdiği için volatile (uçucu) bellek olarak da adlandırılır.

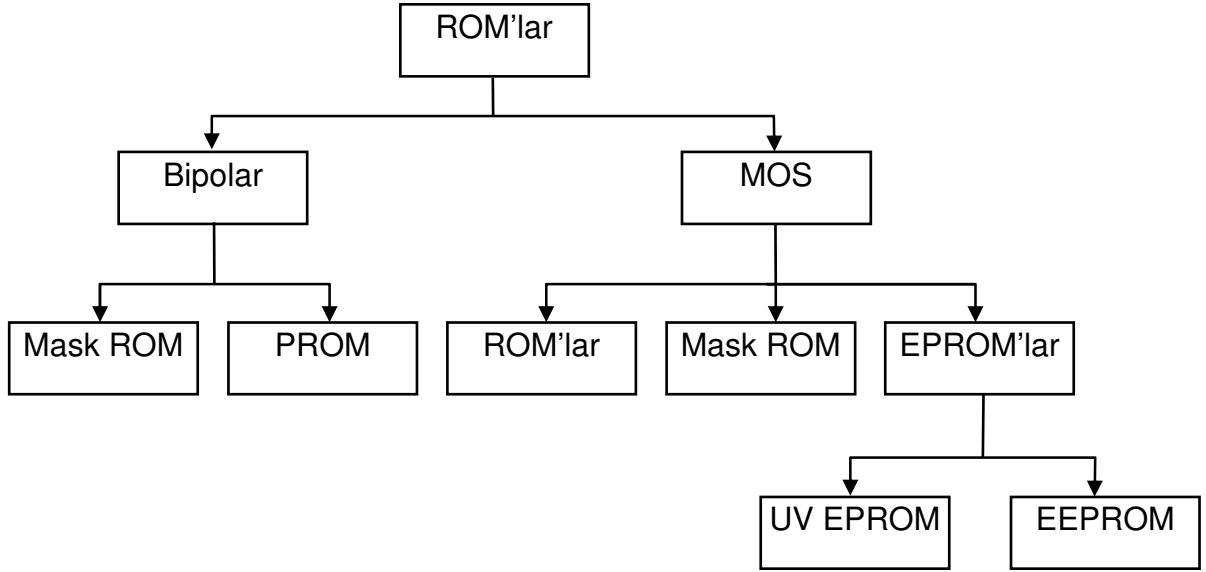
ROM (read-only memory/yalnızca okunan bellek), verinin kalıcı yada yarı kalıcı olarak saklandığı bir bellektir. ROMdan veri okunur ama RAMde olduğu gibi veri yazma işlemi yapılamaz. ROM da RAM gibi doğrudan erişimlidir ama bu terim geleneksel olarak yalnızca oku/yaz bellekleri için kullanılmaktadır. ROM bellek, gerilim kesildiğinde de verileri sakladığı için nonvolatile (uçucu olmayan) bellek olarak adlandırılır.

SALT OKU BELLEKLER (ROM'LAR)

Daha önce söz edildiği gibi, bir ROM bellekten okunabilen fakat ya hiç değiştirilmeyen yada özel bir donanım olmaksızın değiştirilemeyen kalıcı veya yarı kalıcı depolanmış veri içerir. Bir ROM tablolar, dönüştürmeler, veya sistemin başlangıca getirilmesi ve çalışması için programlanmış buyruklar gibi sistem uygulamalarında tekrar tekrar kullanılan veriyi depolamak için kullanılır.

ROM AİLESİ

Yarıiletken ROMlar iki kutuplu (bipolar TTL gibi) yada metal-oksit yarıiletken (MOS) teknolojileriyle üretilirler. Mask ROM, verinin üretim sırasında kalıcı olarak yüklendiği ROM türüdür. PROM yada programlanabilir ROMa veri, kullanıcı tarafından özel bir donanımla elektriksel olarak yüklenebilir.



Şekil-12.6 Yarıiletken ROM ailesi.

Mask ROM ve PROM her iki teknoloji ile de üretilirler. EPROM yada silinebilir (erasable) PROM ise yalnızca MOS teknolojisi ile üretilebilir. UV EPROM, kullanıcı tarafından elektriksel olarak programlanabilen ve bir kaç dakikalık UV ışık uygulamasıyla yeniden silinebilen bir bellek aygıtıdır. Elektriksel olarak silinebilen PROM (EEPROM yada EAPROM) birkaç milisaniye içinde silinebilir. **Mask ROM** kısaca ROM olarak adlandırılır. Yaygın olarak kullanılan standart işlevleri; en çok kullanılan dönüşümler gibi, veya kullanıcı tarafından tanımlanan işlevleri sağlamak için üretim aşamasında kalıcı olarak programlanır. Bellek bir kere programlandıktan sonra değiştirilemez. IC ROM'ların çoğu bir "1" veya bir "0" ı göstermek için bir SIRA/SÜTUN eklemindeki bir transistör bağlantısının varlığı veya yokluğundan yararlanır. Bir ROM ya bipolar ya da MOS olabilir.

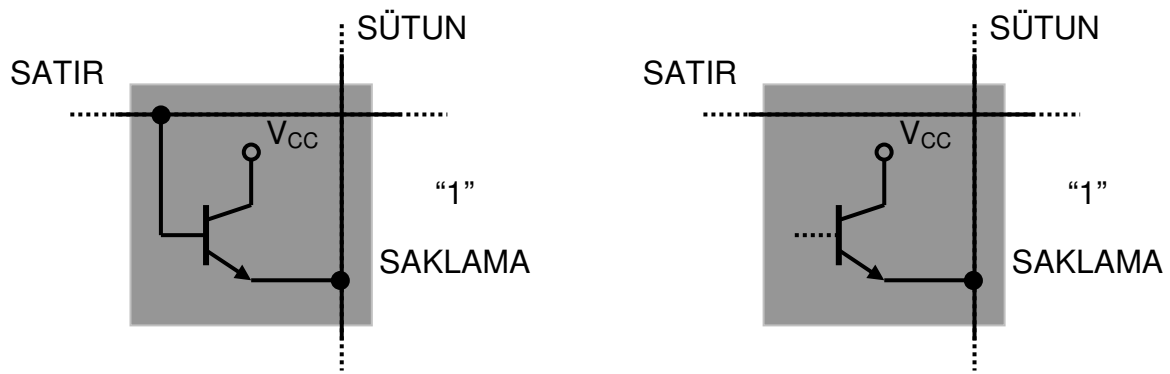
Şekil-12.7(a)'da bipolar ROM hücreler görülmektedir. Bir SIRA hattından transistörün bazına olan bağlantının varlığı o yerde bir "1" i temsil eder, çünkü SATIR hattı YÜKSEK olarak alındığında, o SATIR hattına bir baz bağlantısı olan tüm transistörler ilettime geçer ve YÜKSEK (1) seviyesini ilgili SÜTUN hatlarına bağlar. Baz bağlantılarının olmadığı SATIR /SÜTUN eklemlerinde, SATIR adreslendiğinde SÜTUN hatları DÜŞÜK (0) seviyede kalır. Şekil-12.7(b)'de MOS ROM hücreleri gösterilmektedir. MOSFET'lerle yapılmasının dışında temelde bipolar hücrelerle aynıdır. Bir ekleme bir geçit (gate) bağlantısının varlığı veya yokluğu kalıcı olarak bir "1"i veya "0"ı depolar.

ROM'UN YAPISI

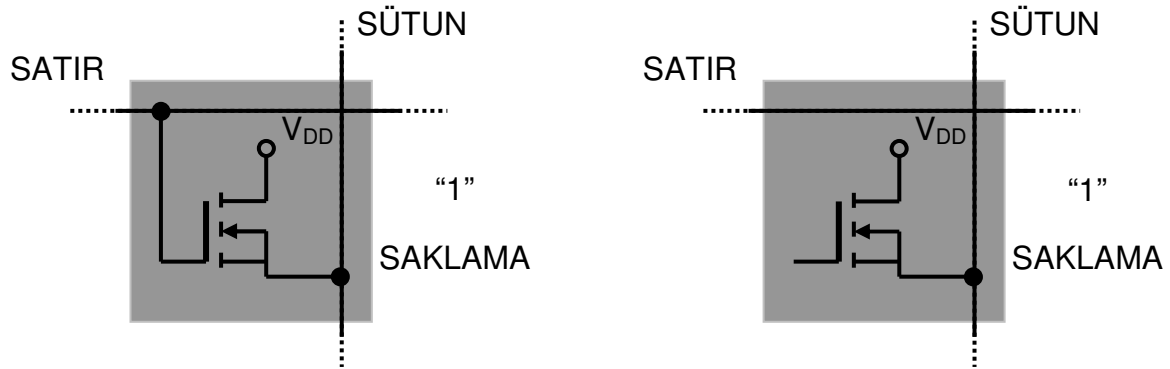
Kavramı açıklamak için, Şekil-12.8'te küçük, basitleştirilmiş bir ROM dizimi gösterilmektedir. Açık renkli kareler depolanmış "0"ları temsil etmekte, koyu renk kareler depolanmış "1"leri göstermektedir.

Temel okuma işlemi şöyledir: adres girişine, ikilik bir adres kodu uygulandığında, karşılık gelen SATIR hattı YÜKSEK seviyeye geçer. Bu YÜKSEK seviye bir "1" in depolandığı her bir hücredeki transistörler aracılığıyla SÜTUN hatlarına bağlanır. Bir "0" in depolandığı her bir hücrede, SÜTUN hatları dirençler yüzünden DÜŞÜK seviyede kalır. SÜTUN hatları veri çıkışını oluşturur. Seçilen SATIR'da depolanmış sekiz adet veri biti çıkış hatlarında görünür.

Gördüğünüz gibi, bu örnek ROM her biri 8 veri biti depolayan 16 adres için düzenlenmiştir. Bu yüzden, bu bir 16×8 'lik ROM' dur ve toplam kapasitesi 128 bittir.



(a) Bipolar hücreler.



(a) Bipolar hücreler.

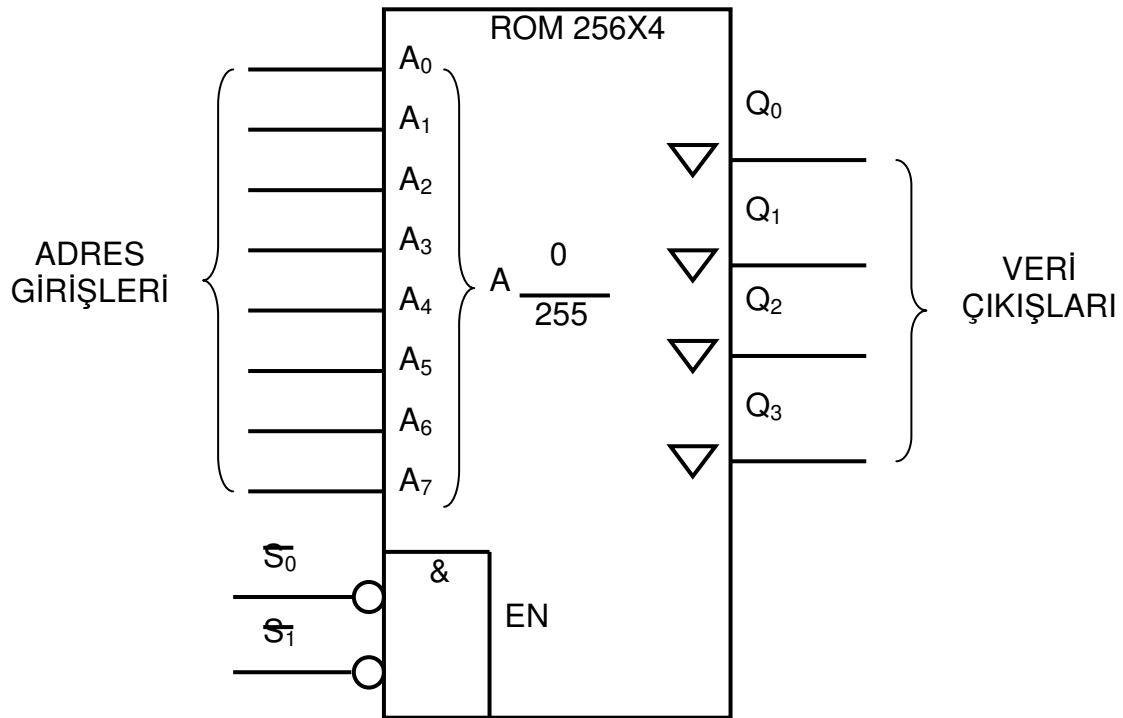
Şekil-12.7 ROM hücrelerinin yapısı.

ROM'UN İÇ ORGANİZASYONU

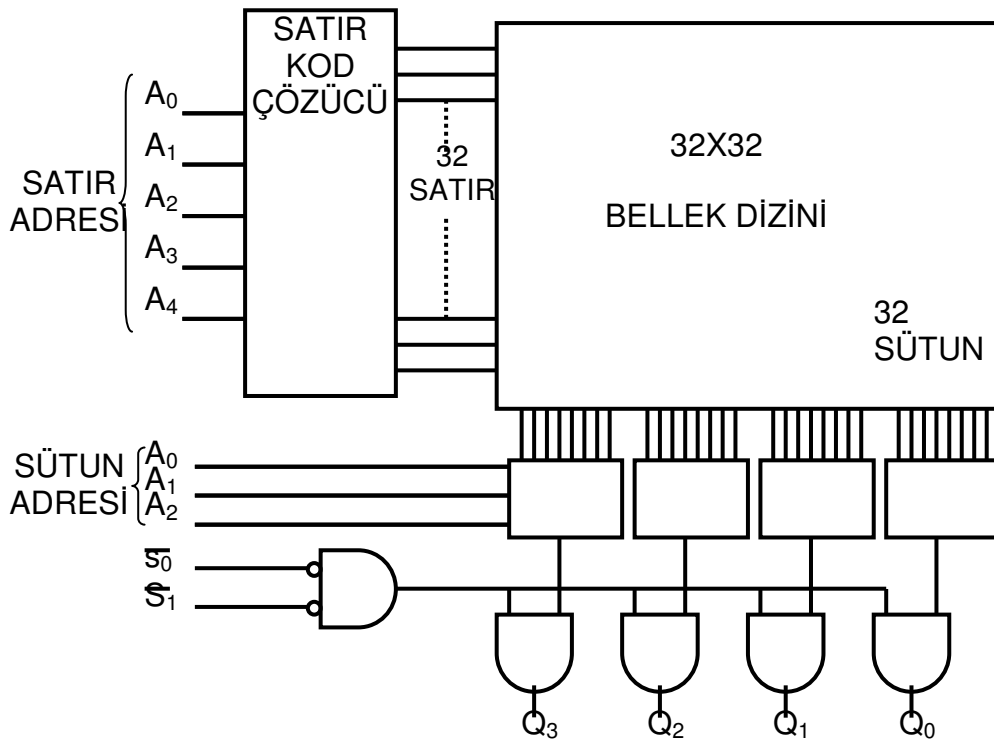
Tümdevre ROM'ların çoğunun iç yapısı biraz önce anlatılandan daha karmaşıktır. Bir tümdevre ROM'un nasıl yapılandığını açıklamak için 256×4 şeklinde düzenlenmiş 1024 bitlik bir elemanı kullanacağız. Mantık sembolü Şekil-12.9'da görülmektedir. 256 ikili kodlardan herhangi biri (8 bit) adres girişlerine

ROM ÖRNEKLERİ

Şekil-12.11'de bir bipolar ROM'a örnek olarak 7488'in mantık sembolü görülmektedir. 32×8 'lik bellek şeklinde düzenlenmiştir; yani, her biri sekiz bitlik depolamaya sahip 32 adrese sahiptir. A_0 'dan A_4 'e kadar olan girişlerdeki beş bitlik bir adres 32 (0'dan 31'e) bellek satırından birini seçer. S girişindeki bir düşük seviye aygıtı etkin hale getirir (EN) ve seçilen veri baytını çıkışlara (Q_1 'den Q_8 'e) yerleştirir. Bu izin girişine yonga seçme (chip select) denir.

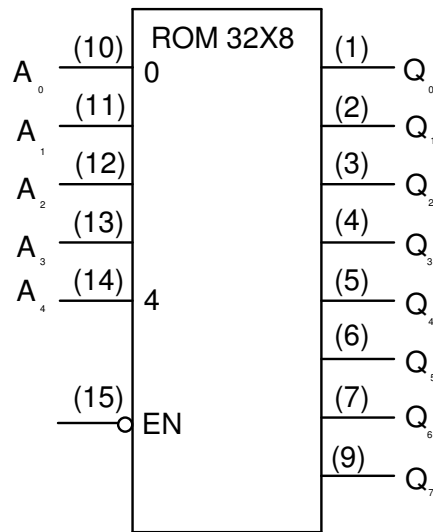


Şekil-12.9 Bir 256x4 ROM' un mantık sembolü.



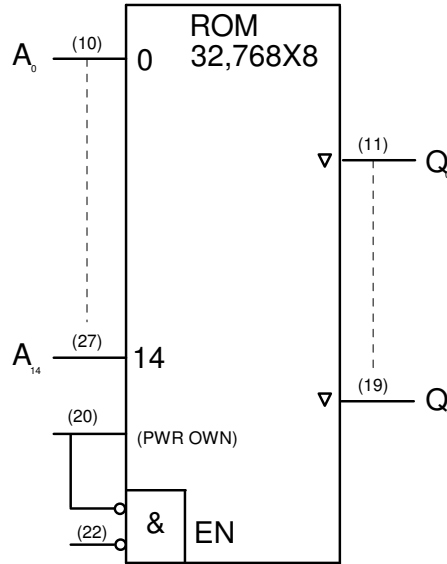
Şekil-12.10 1024 bit ROM'un iç düzeni

Küçük bipolar ROM'la karıştırıldığında, TMS 47256 (TI) yüksek kapasiteli bir MOS ROM'a örnektir. 262,144 biti 32,768x8 (32Kx8)'lik bir düzenleme ile depolayabilir. Her bir yerleşimde 8 bit olan 32,768 adrese sahiptir.



Şekil-12.11 Düşük kapasiteli bir ROM olarak 7488.

Şekil-12.12'de bu aygıtın mantık sembolü görülmektedir. Bunun 15 adres hattı vardır. Bunlara 32,768 yerleşimi adreslemek için gerek duyulmaktadır ($2^{15}=32,768$). Etkin-DÜŞÜK bir yonga seçici girişi, S_1 ve bu özel aygıtta bir yonga izin/güç kesme girişi, E vardır.



Şekil-12.12 TMS 47256 MOS statik ROM'un mantık sembolü

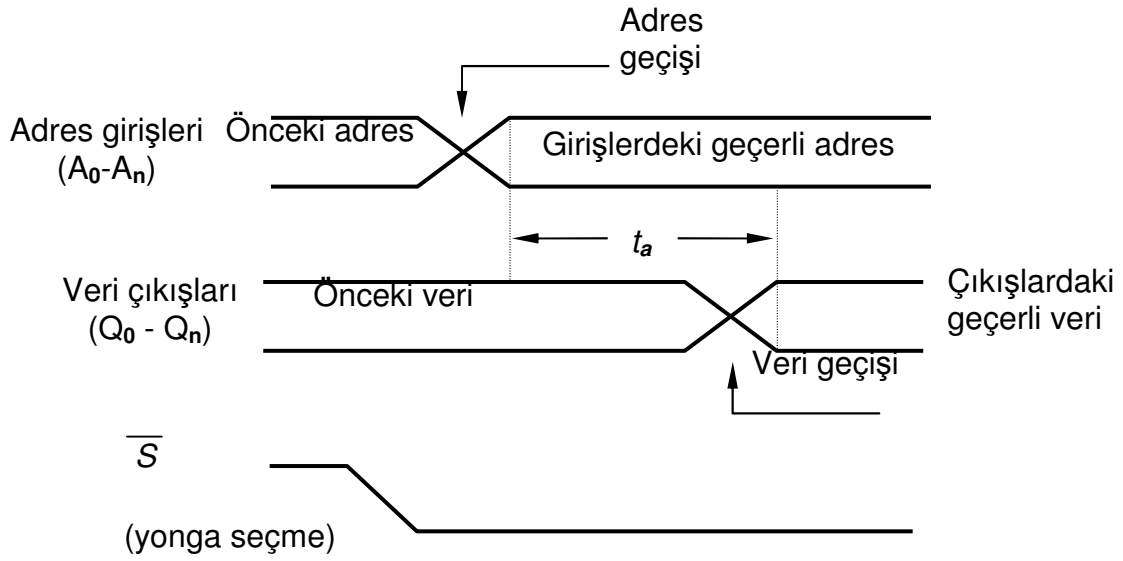
Kontrol girişleri çalışması şöyledir; Bellek çıkışını etkin hale getirmek için S_1 ve E'nin ikisinin de DÜŞÜK seviyede olması gerekmektedir. E YÜKSEK olduğunda aygıt dc güç kaynağından gelen akımı azaltan düşük güçlü bir standby moduna sokulur.

ÜÇ DURUMLU ÇIKIŞLAR VE YOLLAR

Üç durumlu çıkışlar, küçük ters üçgenlerle gösterilirler. Yüksek empedans durumları, yolların ortak kullanımı nedeniyle oluşan yüklenme etkisini yok etmek/azaltmak için birebir olan üç durumlu tamponlar mikroişlemci sistemlerinde çok kullanılırlar.

ROM ERİŞİM SÜRESİ

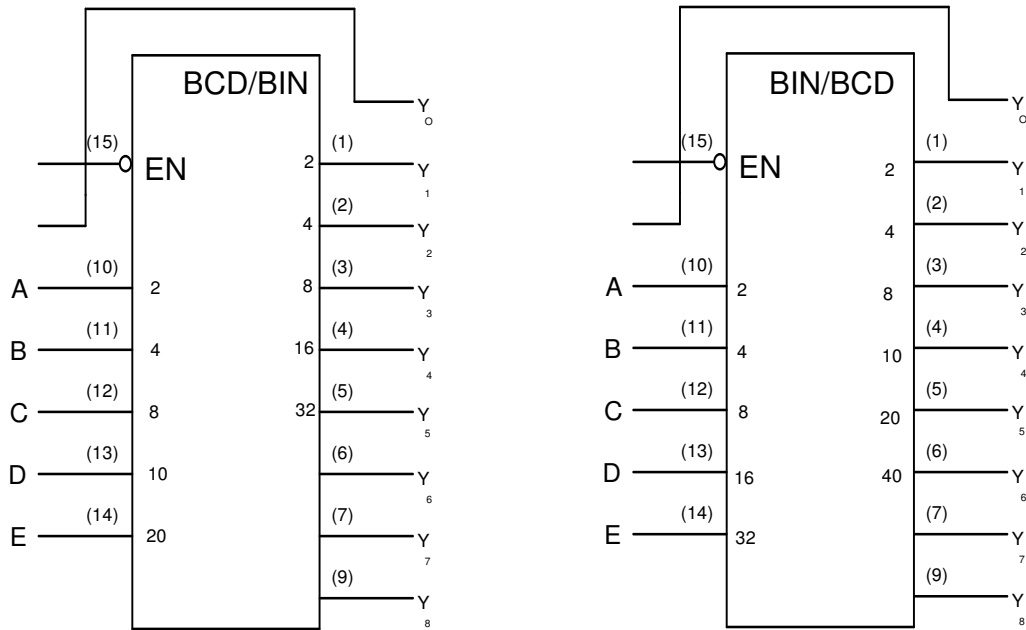
ROM erişim süresini gösteren tipik zamanlama çizgesi şekil 12 de verilmiştir. Bir ROMun erişim süresi, t_a , girişlere geçerli bir adres kodunun uygulanmasından, çıkışlarda geçerli verinin görünmesine dek geçen zamandır. Erişim süresi, adres kodu önceden hazır olmak üzere yonga seçme (\bar{S}) girişinin etkinleştirilmesinden, geçerli verinin alınmasına kadar geçen süre olarak t_a ölçülür.



Şekil-12.13 ROM'un erişim süresi.

ÖRNEK ROM UYGULAMASI

Daha önce söz edildiği gibi bazı ROM'lar yaygın olarak kullanılan işlevleri yerine getirmek için programlanırlar ve piyasada bulunurlar. Örnek olarak, 74184 BCD'den ikilik koda dönüştürücü olarak programlanan bir ROM'dur, ve 74185 ise ikilik kodu BCD'ye dönüştürmek için programlanan bir ROM aygıtıdır. Bunların mantık sembolleri Şekil-12.14'te görülmektedir.



Şekil-12.14 74184 ve 74185'in mantık gösterimi.

Günümüzde bu tür ROM'ları piyasadan elde etmek artık zordur. Çünkü kullanıcıların kolaylıkla kendilerinin istediği gibi programlayabildikleri EEPROM'lar veya PAL, GAL, ve PLD adı verilen programlanabilir ve kolaylıkla elde edilebilmektedir. Bu yeni ürünler ROM'lara oranla daha güvenilir ve enerji tüketimleri daha düşüktür. Aynı zamanda daha ucuz çözüm sunabilmektedirler.

BİLGİSAYAR UYGULAMALARINDA ROM

ROM IBM kişisel bilgisayarlarında, örneğin, BIOS (Temel Giriş /Çıkış Servisleri) olarak adlandırılan servisleri için kullanılır. Bunlar bilgisayar için temel denetim ve destek işlevlerini yerine getiren programlardır. Örneğin ROM' da depolanan BIOS programları belirli video monitör işlevlerini kontrol eder, disk formatlanmasını sağlar, klavyeyi girişler için tarar, ve belirli yazıcı işlevlerini denetler.

PROGRAMLANABİLİR ROM'LAR

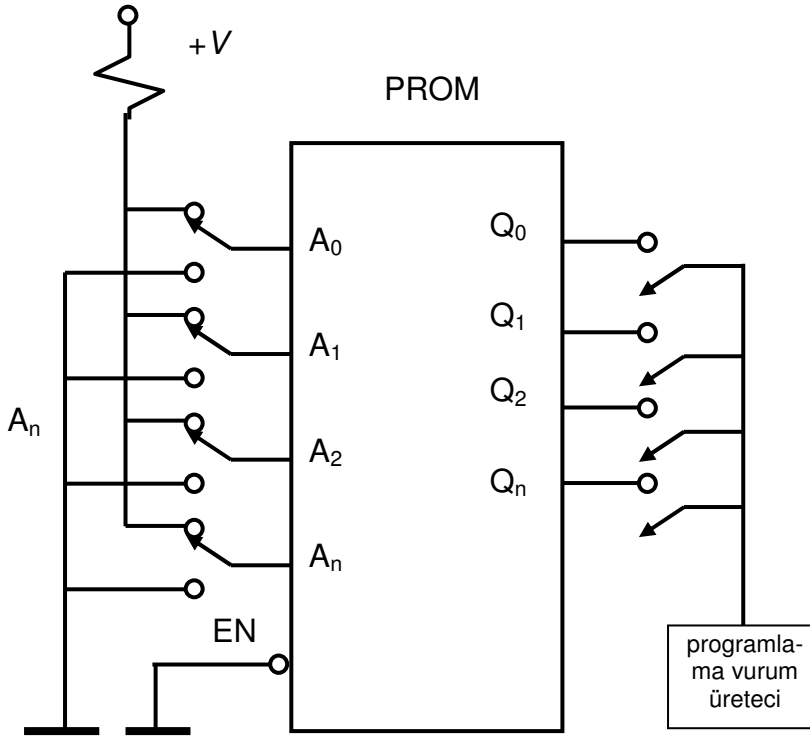
PROM'lar bir kez programlandıktan sonra mask ROM'lar ile ayırdırlar. Tek ayırım, PROM'ların üreticiden tüketiciye programlanmadan ulaşması ve kullanıcının gereksinimleri doğrultusunda programlanmasıdır. PROM'lar, hem bipolar hem de MOS teknoloji ile üretilebilirler. 250.000'i aşan bit kapasitelerinde ve genellikle 4 yada 8 bitlik sözcük formatında olurlar. PROM'larda veri saklama, bir tür sigorta bağlantısı yardımıyla yapılır. Bir gözde sigorta bağlantısının açık olması "0", sağlam olması da "1" bilgisinin saklı olduğunu gösterir. Yazma işlemi geri dönüşü olmayan bir işlemdir ve PROM bir kez programlandığında bir daha değiştirilemezler. Dizi içindeki her transistörün emiteri ile ilgili sütun hattı arasında bir sigorta bağlantısı vardır ve programlama sırasında "0" yüklenmek istenen gözlerdeki sigortalar yakılır. Şekil-12.15'de sigortaların bağlantısı gösterilmiştir. Bu sigortalar üç türde üretilmektedir:

1. Ni-chrome (nikel-krom) gibi metallere,
2. Çok kristal silisyum (Polycrystalline silicon) ile yaratılan çentikli yarıiletken dar yollardan,
3. Birisi yakılıp diğeri diyot olarak kullanılan sırt sırta iki pn eklemesinde.

PROGRAMLAMA

Programlama işlemi, Şekil-12.16'da temel ilkesi verilen özel aygıtlar ile yapılır. Burada, adres girişinden istenilen adresin kodu uygulanır ve bu adreste, "0" yüklenmek istenen (PROM başlangıçta tümüyle "1" yüklüdür) gözlerin çıkışlarına bir vurum uygulanır. Böylece "0" bulunması gereken gözlerdeki sigorta hatları yanar ve istenilen program PROM'a yazılmış olur. Aygıt sonra sıradaki adresi seçer ve işlem yinelenir.

Programlanabilen eleman sayısının artması sonucu piyasaya bu ürünlerin tamamını



Şekil-12.16 Basitleştirilmiş PROM programlama devresi.

UV EPROM'LAR

Bir UV EPROM kılıfındaki saydam quartz pencere ile kolayca tanınır. Morötesi EPROM içindeki FETlerin geytleri yalıtkan oksit malzeme içine gömülüdür. Programlama sırasında elektronlar bu "yüzer" geytlerden alınır. Silme işlemi, bellek dizim yongasına quartz kapak üzerinden yüksek yoğunlukta morötesi ışımaya uygulanmasıyla yapılır. Bir kaç dakika ile bir saat arası değişen süre sonunda geytlerdeki pozitif yük sıfırlanır.

Üretici firmasına ve yapım teknolojisine bağlı olarak EPROM'ların programa gerilimleri farklıdır. NMOS teknolojisi ile üretilen EPROM'lar 25 veya 21 volt ile programlanır. CMOS ve türevleri ile üretilen EPROM'lar 12,5 volt ile programlanır. NMOS olarak EPROM üretimi aslında durdurulmuştur, fakat eski devreler üzerinden sökülen ve silinen EPROM'lar halen piyasada bulunmaktadır. EPROM kodları üretici firmanın kullandığı harfler ile başlar ve 27XX olarak devam eder. 27'den sonra C harfi varsa bu tür EPROM'lar CMOS ve türevleri ile üretildiğini gösterir. XX EPROM'un Kbit cinsinden kapasitesini gösterir.

EEPROM'LAR

EEPROM'lar, elektriksel vurularla hem silinebilen hem de programlanabilen PROM'lardır. Bu aygıtlar EAPROM olarak da adlandırılırlar. EEPROM'lar elektriksel olarak silinip, programlanabildikleri için devreden sökülmeden çabucak

içerikleri değiştirilebilir. EEPROM' un iki türü; yüzer geyt MOS ve metal-nitride-oxide-silicon (MNOS) ile yapılmış olanlarıdır. Yüzer geyt yapıda, yüzer geytlerden yükün boşaltılması yada yüklenmesi, denetim geytine uygulanan gerilim ile yapılır. Günümüzde en yaygın kullanılan PROM ailesi bireyidir. Bazı firmalar üretim teknolojisinde küçük farklılıktan dolayı bu tip PROM'lara FLASH bellek adı vermektedirler. Programlanmaları ve silinmeleri çok kısa sürede gerçekleştirilebilir (birkaç saniye). Yeni üretilen FLASH bellekleri boyutları 4 Gbayt X8 kapasitesindedir. Yakın gelecekte bilgisayarlarda hard diskin yerini alması beklenmektedir.

EEPROM kodları ise yine üretici firma ve 28 ile başlar daha sonra üretim teknolojisini gösteren harf yer alır. Ve en sonunda Kbit cinsinden kapasiteyi gösteren 2 veya 3 rakam yer alır bu rakamlardan sonra yer alan harfler ise paketleri hakkında ve sıcaklık aralığı ile ilgili bilgiler verir. FLASH ROM'ların kodları ise firma kodu ve 29CXX olarak adlandırılmaktadır. Bazı firmalar ise 28FXX kodu ile adlandırmaktadırlar.

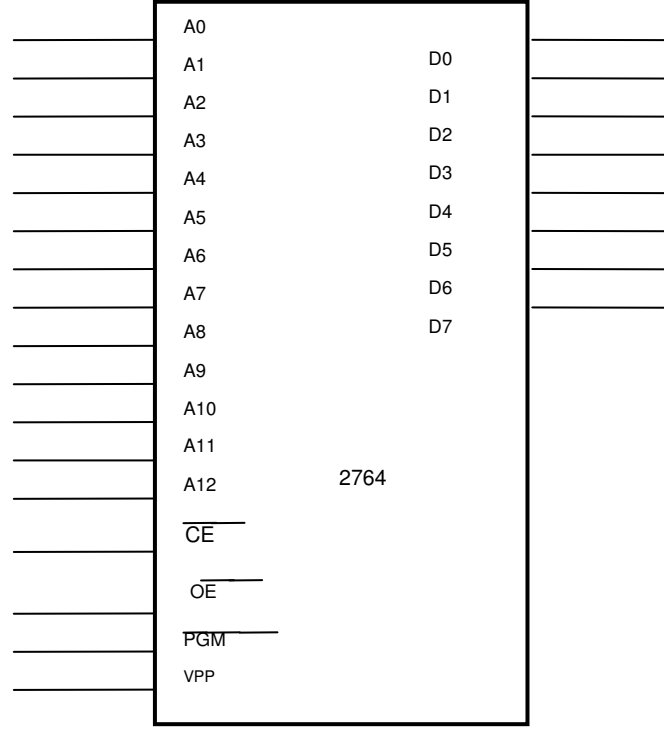
ÖRNEK EPROM 27C64

27C64 birçok firma tarafından üretilen 8KX8 boyutlarında bir EPROM'dur. CMOS teknolojisi ile üretilenler 12,5 Volt ile programlanabilirler silme işlemi ise ultraviyole ışınla 10 dakikada yapılabilir. Okuma için, seçme CE ve OE girişlerinin DÜŞÜK seviye olması ve V_{pp} gerilimi +5 Volt ve power-down/program (PD/ \overline{PGM}) girişleri DÜŞÜK olmalıdır. PD/ \overline{PGM} girişi YÜKSEK olduğunda aygıt kaynaktan az akım çekilen ve az güç harcanan standby (bekleme) durumundadır. silme işleminde $12\text{mW}/\text{cm}^2$ gücünde filtresiz UV lamba kullanılırsa veri 20 - 25 dakika içinde silinir. Çoğu EPROMda olduğu gibi silme işleminden sonra bütün bitler "1"dir. Gün ışığında da silmeyi sağlayan dalga boyunda ışınlar bulunduğu için tümdevrenin üstündeki saydam pencere kapalı olmalıdır.

Programlama için, V_{pp} ucuna +12,5V dc (normalde +5V) uygulanır ve CE girişi YÜKSEK yapılır. Seçilen adrese programlanacak 8-bit, çıkışlara ($Q_0 - Q_7$) verilir. Sonra PD/ \overline{PGM} girişine 10-50ms süreli YÜKSEK seviye uygulanır. Programlama istenilen adres sıralaması ile yapılabilir.

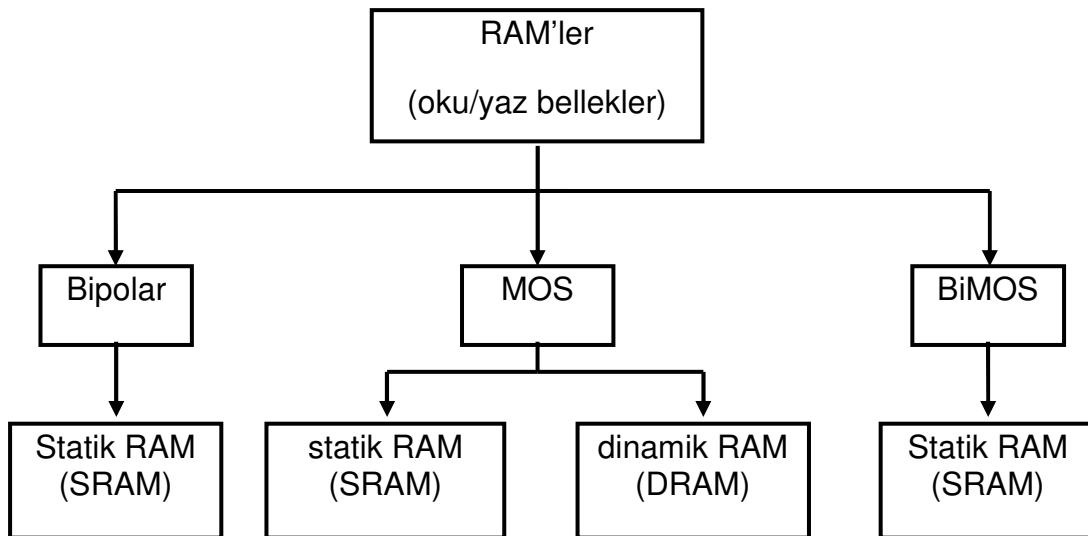
OKU/YAZ DOĞRUDAN ERİŞİMLİ BELLEKLER RAM'LER

RAMlere veri istenilen sırada ve zamanda yazılabilir ve okunabilir. RAM adreslerinden birine bir veri yazılınca önceki veri yok olur. Okuma işlemi ise yüklü veriye zarar vermeyen bir kopyalamadır. Doğrudan erişimli belleklerde devrenin gerilimi kesilince bütün yüklü veri kaybolur.



RAM AİLESİ

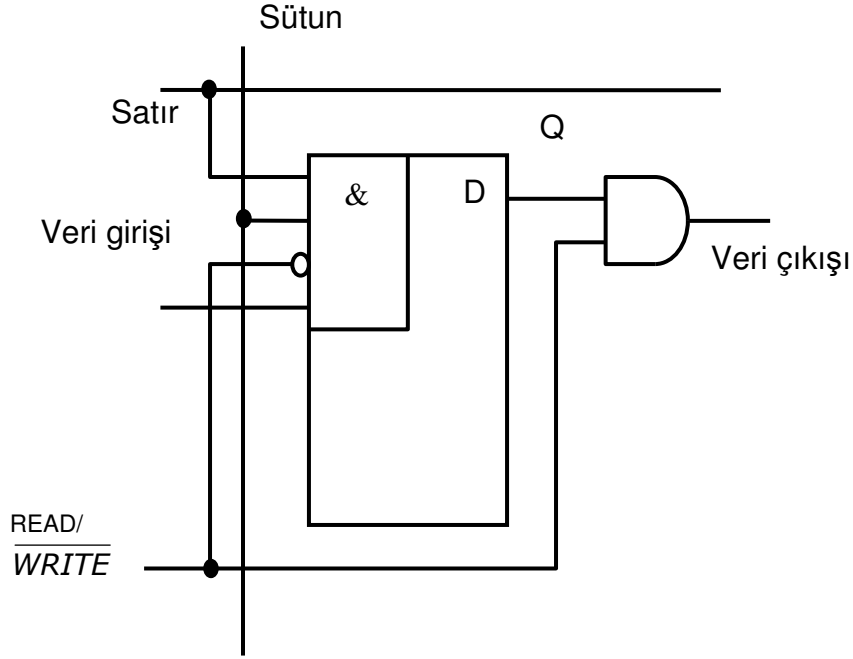
Yarıiletken RAMler bipolar yada MOS olarak üretilmektedirler. Bazı bellek aygıtları BiMOS denilen ve bipolar (TTL ada ECL) ve MOS teknolojilerinin bir birleşimi olan teknikle üretilirler. Bipolar RAMlerin tümü statiktir. Statik bellekte tutucu (latch) benzeri saklama aygıtları kullanıldığından veri, gerilim uygulandığı sürece saklanır. MOS teknolojisinde üretilen dinamik RAMlerde ise veri sığaçlar (capacitor) üzerinde saklandığından belli aralıklarla tazeleme (refreshing /recharging) gerekir.



Şekil-12.17 Yarıiletken RAM ailesi.

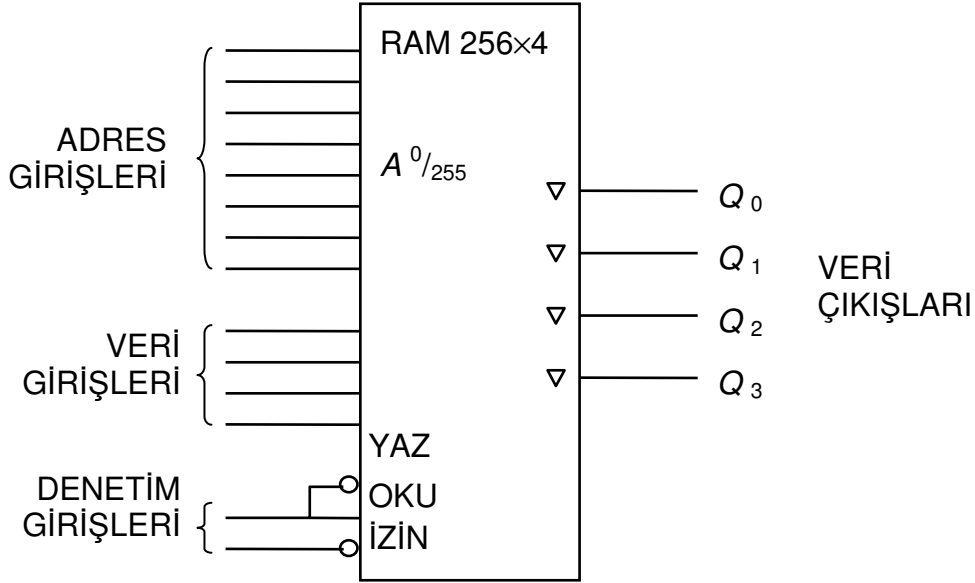
STATİK RAM'LER (SRAM)

SRAM'lerde saklama işlemi bipolar yada MOS tutucular ile yapılır. Veri bir göze yazıldıktan sonra, enerji kesilene yada aynı göze yeni bir veri yazılana dek saklanır. Gerilim kesilince verinin yok olması nedeniyle RAM'lere geçici bellek denir.



Şekil-12.18 Statik RAM bellek gözü. Bellek elemanı olarak tutucu kullanılır.

Şekil 12.18'te verilen SRAM hücresinin çalışması şöyledir: hücre (göz), satır ve sütun hatlarındaki YÜKSEK seviye ile seçilir. OKU/ \overline{YAZ} ($\overline{READ/WRITE}$) girişi "0" olunca (yazma kipi) girişteki veri tutucuya ulaşır. OKU/ \overline{YAZ} girişi "1" olunca tutucu etkilenmez ama yüklü veri biti (Q) çıkış yoluna aktarılır.

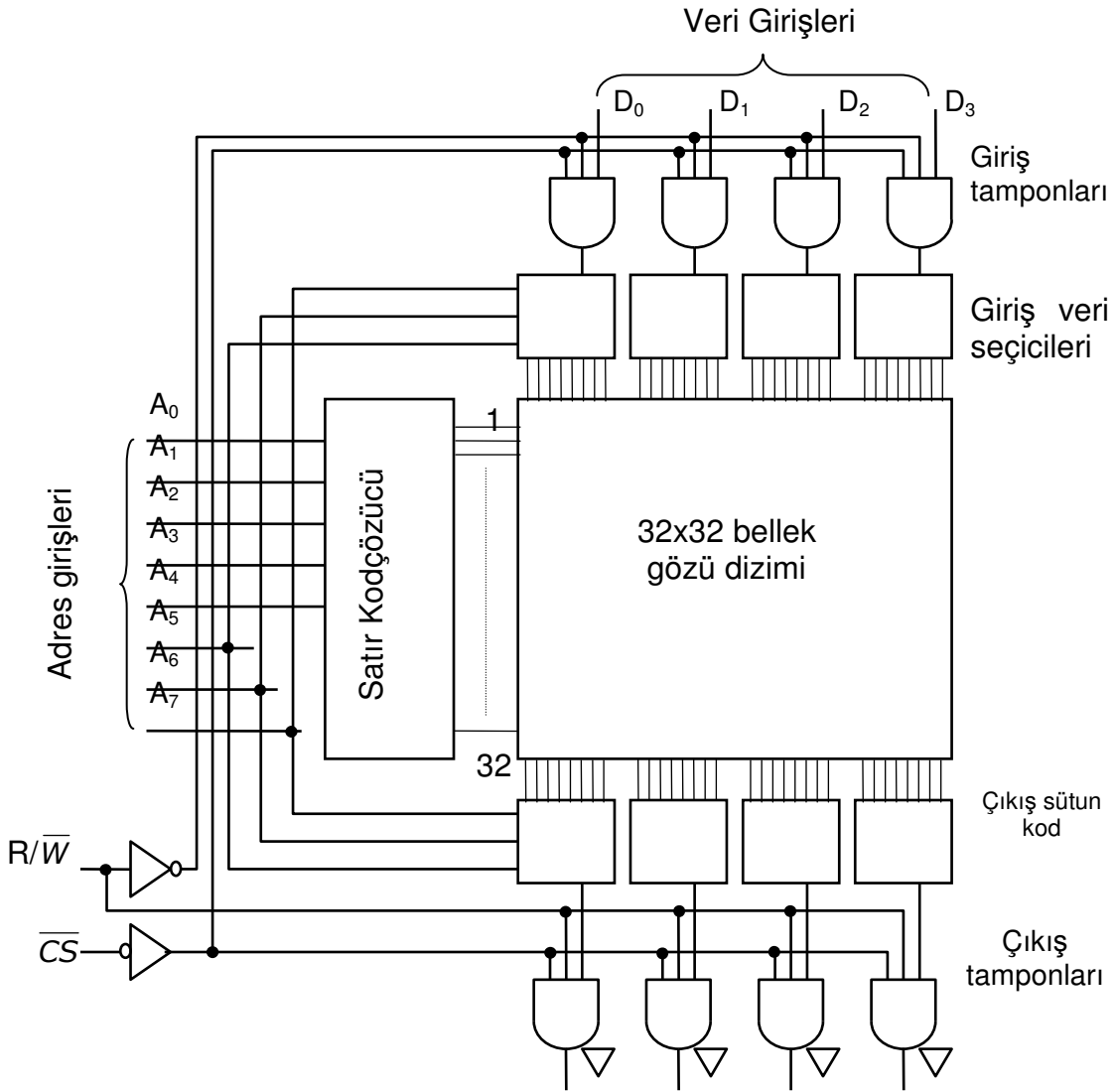


Şekil-12.19 256×4 SRAMın mantık simgesi.

STATİK RAM'İN İÇ DÜZENİ

RAM, ROM ile aynı şekilde adreslenir. Aralarındaki ayırım yalnızca veri giriş uçları ve OKU/ $\overline{\text{YAZ}}$ denetimidir. 1024-bit kapasiteli ve 256×4 düzenli bir SRAMın mantık simgesi şekil-12.19'da, iç yapısı da şekil-12.20'de gösterilmiştir.

Okuma durumunda ($\overline{\text{READ}}/\overline{\text{WRITE}}$ YÜKSEK) seçilen adresteki 4-bit, $\overline{\text{CS}}$ DÜŞÜK seviye yapılarak çıkıştan alınır. Yazma kipinde ($\overline{\text{READ}}/\overline{\text{WRITE}}$ DÜŞÜK) ise veri girişine uygulanan 4-bit, seçilen adrese saklanır. Sekiz adres hattından beşi, satır kodçözücü üzerinden 32 satırdan birini seçer. Geriye kalan üç hat ise, okuma kipinde çıkış sütunlarını, yazma kipinde de veri giriş sütunlarını seçmek için kullanılır. Okuma sırasında çıkış, yazma sırasında da giriş tamponlarına izin verilerek adreslerde veri bozulması önlenir. $\overline{\text{CS}}$ girişi okuma ve yazma durumlarında DÜŞÜK olmalıdır.



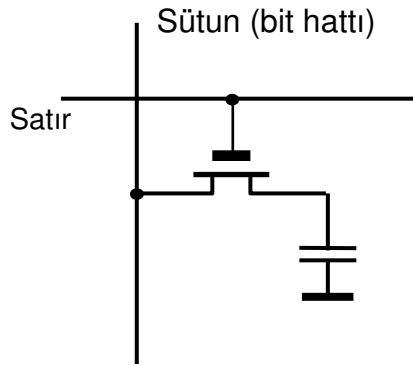
Şekil-12.20 256 × 4 statik RAM için temel düzenleme.

DİNAMİK RAMLER (DRAM)

Dinamik RAMlerde veri tutucular yerine küçük sığaçlarda saklanır. Bu yöntemin üstün yanı, çok basit olması nedeniyle aynı boyuttaki bir yonga üzerine daha büyük dizimlerin yapılmasına olanak vererek bit başına maliyeti çok azaltmasıdır. Sakıncası ise, içindeki sığaçların veriyi çok uzun süre saklayamamaları nedeniyle yüklemenin periyodik olarak yenilenmek zorunda olmasıdır. Yenileme işlemi ek bellek devresi gerektirir ve DRAM kullanımını karmaşıklaştırır. Şekil-12.21'de bir MOSFET ve bir sığaçla oluşturulmuş tipik bir DRAM hücresi verilmiştir

Transistör dinamik RAM'larda anahtarlama elemanı olarak kullanılır. Hücrenin çalışması şekil-12.22'de gösterilmiştir. R/W hattındaki DÜŞÜK seviye giriş üç durumlu tamponu izinlerken, çıkış tamponunu izinlemez. Bu moda yazma modu

adı verilir. Bu durumda hücreye “1” bilgisi yazılacak ise D_{IN} hattına bir YÜKSEK seviye, “0” bilgisi yazılacaksa bir DÜŞÜK seviye, satır hattına bir YÜKSEK seviye uygulanmalıdır. (Şekil-2.22 (a)'ya bakınız.) Transistör iletimdedir ve sütundaki yüksek seviye kondansatörü doldurur, eğer dolu ise boşalmasını engelleyerek “1” mantık seviyesinde kalmasını sağlar. D_{IN} hattına bir DÜŞÜK seviye uygulanırsa kondansatör önceden dolu ise boşalır DÜŞÜK seviyeye gelir, önceden boş ise, seviyesini koruyarak mantık “0” saklanmış olur(Şekil-12.22 (b)). R/W hattına YÜKSEK seviye uygulandığında çıkış tamponu izinlenirken giriş tamponu kapalıdır. Bu durumda satır hattına YÜKSEK seviye uygulanırsa transistör iletime geçer ve kondansatörün sahip olduğu yük D_{OUT} çıkışına aktarılır. (bakınız şekil-12.23).

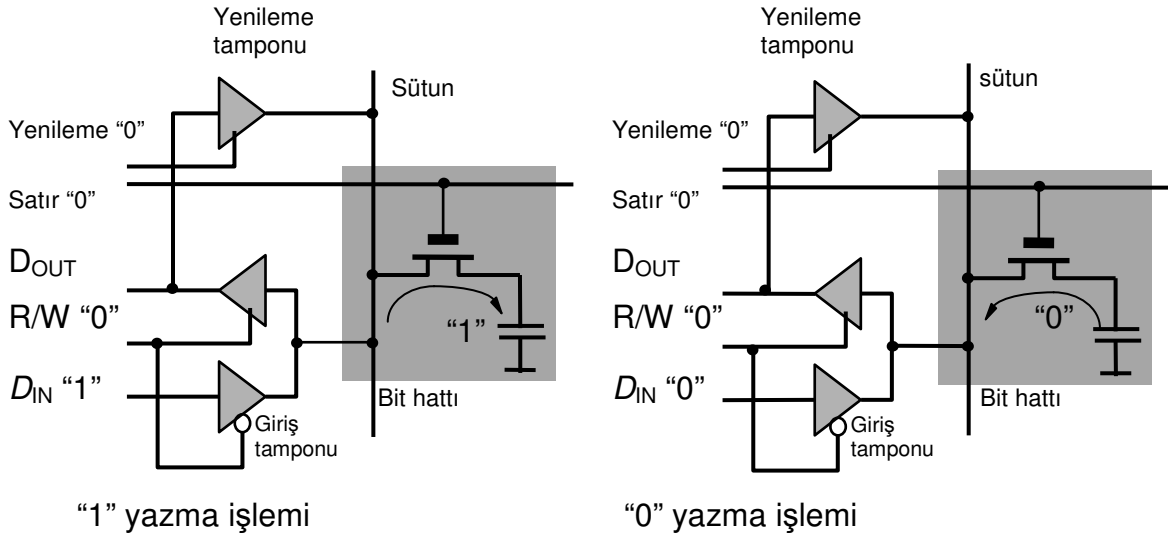


Şekil-12.21 Dinamik MOS RAM hücresi.

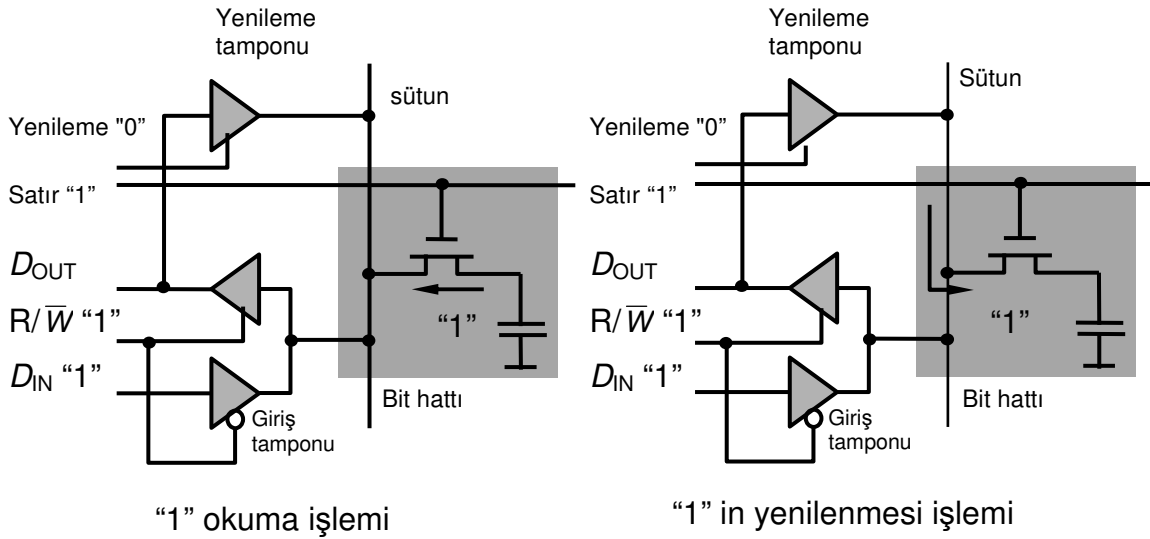
Bellek hücresinin içeriği (tazelenmesi) yenilenmesi işlemi için R/W hattı ve yenileme girişi YÜKSEK seviyeye çekilir. Satır hattına YÜKSEK seviye uygulandığında transistör iletime geçer ve kondansatörün yükü D_{OUT} üzerinden yenileme tampon girişine uygulanır. Tampon tarafından yükseltilecek gerilim tekrar kondansatörü doldurur. Daha doğrusu eksilen kısım tamamlanır. Eğer kondansatörün yükü “0” ise herhangi bir yükseltme yapılmadığı için kondansatör tekrar boş olarak kalır. Şekil-12.23’de hatların seviyeleri ve akım yönleri gösterilmiştir.

DRAMİN TEMEL DÜZENİ

Statik ve dinamik RAMler arasındaki ana ayrım, verinin saklandığı gözlerin yapısıdır. Dinamik RAM’lar veriyi tutucu yerine kondansatör içerisinde saklar. Avantajı ise yapısı çok basittir bir transistör ve kondansatörden bir bellek hücresi elde edebiliriz. Oysa statik RAM’larda transistör sayısı 15-20 arasındadır. Az eleman kullanılması daha küçük alana daha fazla hücrenin yerleştirilmesini ve ucuza üretilmesini sağlamıştır. Dinamik RAM’larda kondansatör yükünün zamanla kaybolması nedeniyle belirli aralıklarla yenileme gerektiğinden ek devreler de gereklidir. DRAM’lerde ayrıca şimdi sayılacak bazı özellikler de vardır.



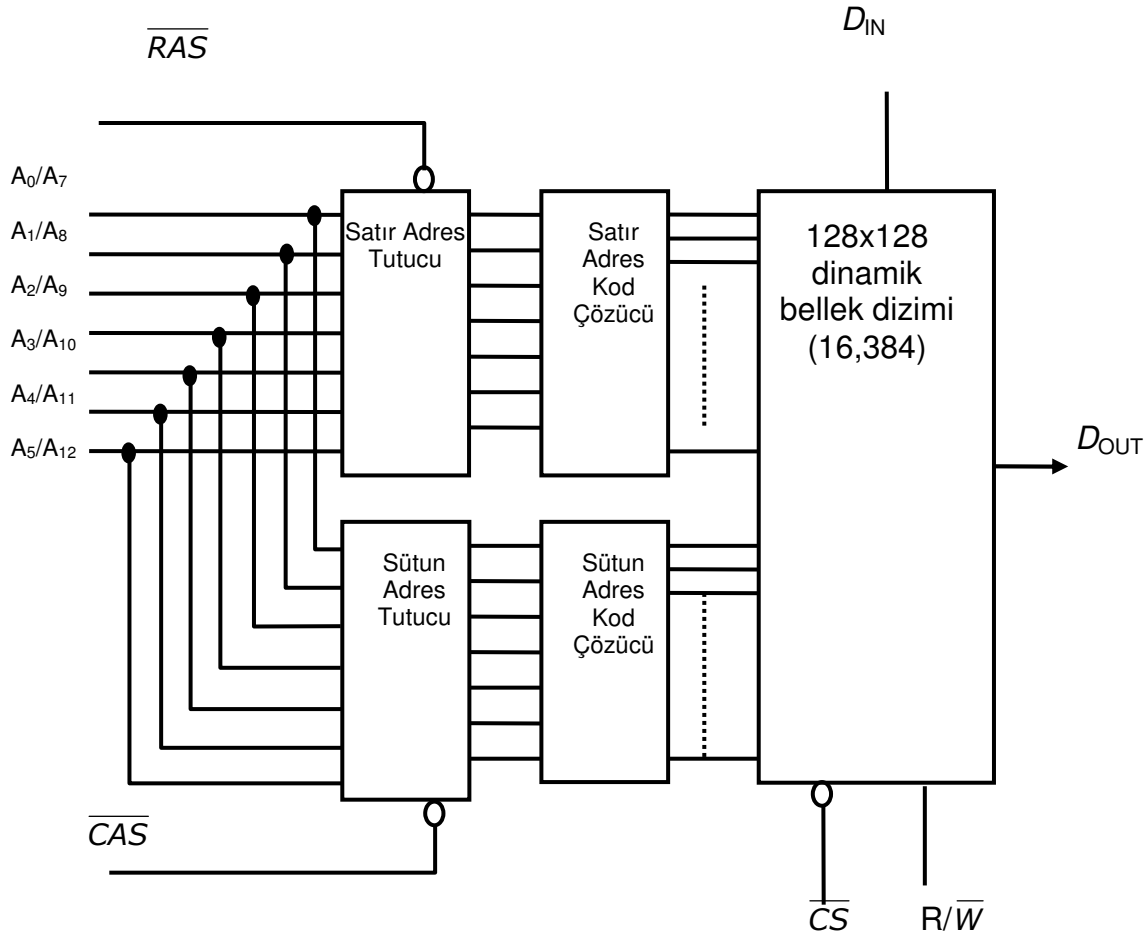
Şekil-12.22 statik RAM'e 0 ve 1 yazma işlemi



Şekil-12.23 Dinamik bellek gözünün çalışması.

ADRES ÇOĞULLAMA

Çoğu DRAMlerde *adres çoğullama* denilen bir teknik kullanılarak adres hattı sayısı, ve böylece de kılıf dışına taşınması gereken bacak sayısı azaltılır. Şekil-12.24'de 16,384 (16 kbit) DRAM blok diyagramı verilmiştir. Diyagram adres çoğullamanın gösterilebilmesi için basitleştirilmiştir ve bellek düzenlemesi 16×1 olarak yapılmıştır.



Şekil-12.24 Adres çoğullamalı 16 kbit dinamik RAM blok diyagramı.

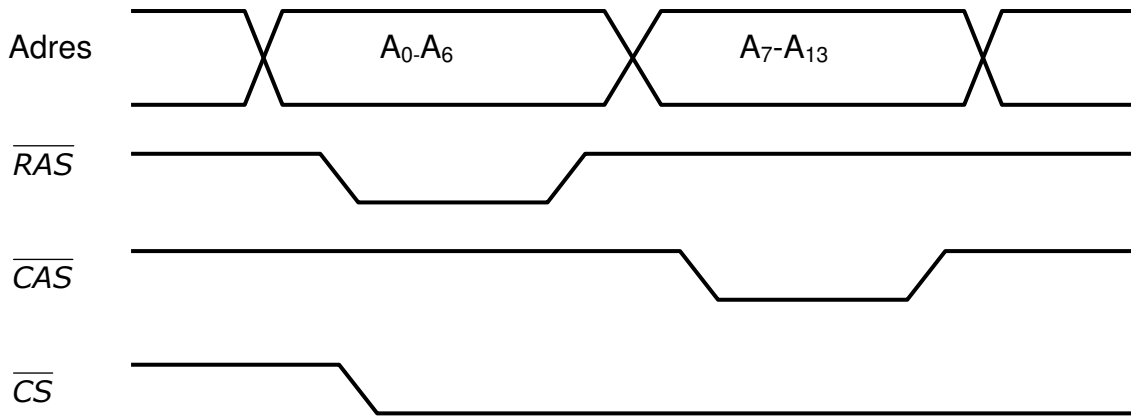
14-bit adres kodu ($2^{14}=16,384$) adres girişine sırayla verilir. Önce 7-bit satır adresi verilir ve RAS (row address strobe) girişi "0" yapılarak satır adresi satır adres tutucusuna yazılır. Sonra, 7-bit sütun adresi verilir ve CAS (column address strobe) girişi "0" yapılarak sütun adresi sütun adres tutucusuna yazılır. 7-bit satır adresi ve 7-bit sütun adresi kod açılarak yazma yada okuma için seçilen adrese ulaşılır.

BELLEK İÇERİĞİNİN TAZELENMESİ

Tüm dinamik ram hücreleri kondansatörün yükünü zamanla kaybetmesinden dolayı belirli aralıklarla içeriklerinin tazelenmesi (refresh) (bazı kaynaklarda yenileme olarak kullanılır) gerekir. Tazeleme devresi bellek tümdevresi içerisinde yer alabilir veya dışarıdan bu iş için üretilmiş olan denetleyici tümdevreler kullanılarak yapılabilir. Tazeleme işlemi kondansatör yükünün mantık "1" saklandı ise, mantık "0" okunacak seviyeye gelmeden yapılması gerekir.

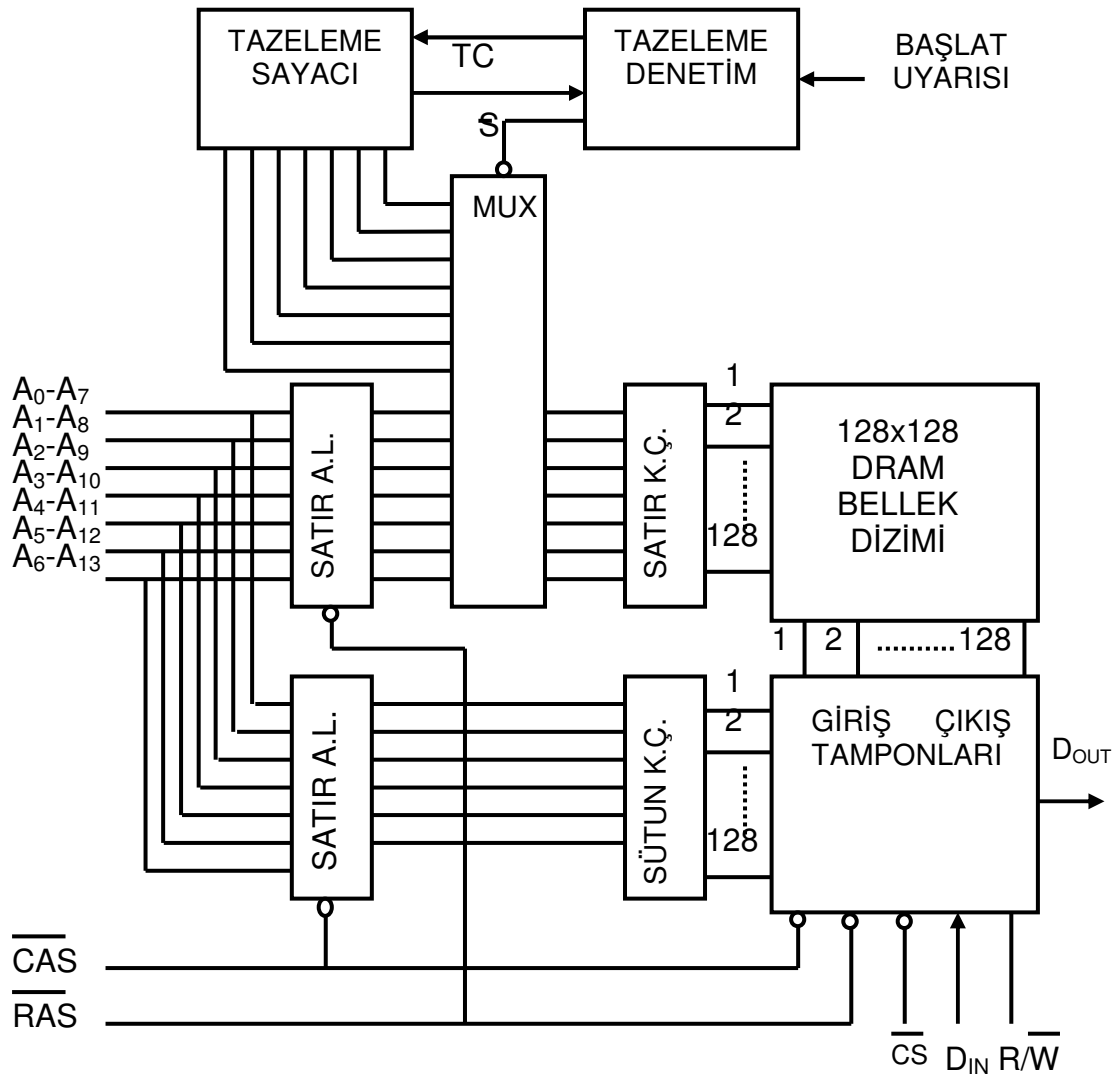
Tazeleme işlemi başlangıç satırından başlayarak her satırdaki hücreler aynı anda

olacak şekilde sırayla yapılır. Bu tazeleme yöntemine BURST tazeleme adı verilir. Tazeleme işlemi her 2ms ile 4ms arasında mutlaka yapılmalıdır. Tazeleme sırasında okuma veya yazma yapılamaz. Diğer bir tazeleme yöntemi ise satır tazelemesini okuma ve yazma işlemlerinin arasına serpiştirerek yapar. Bu yöntemde öncelik bellekten veri okuma veya yazmaya verilmiştir. Şekil-12.26'daki tazeleme devresinin çalışmasını inceleyelim. Tazeleme işlemi başlat girişi ile başlatılır. Tazeleme zamanlayıcısı ve denetleme birimi tazeleme kendisine gelen uyarı ile tazeleme sayacının değerini bir arttırmasını ve çıkışlarını satır adres kod çözücü girişine uygulanmasını sağlar, buraya ulaşan satır adresi tazelenen satırı seçer. daha önce bahsedildiği gibi aslında bellekler içeride her satırda sekiz bellek hücresi içermezler. Satır ve sütun sayısı eşittir. Şekil-12.26'daki örnek DRAM'ın boyutu 128X128'dir.

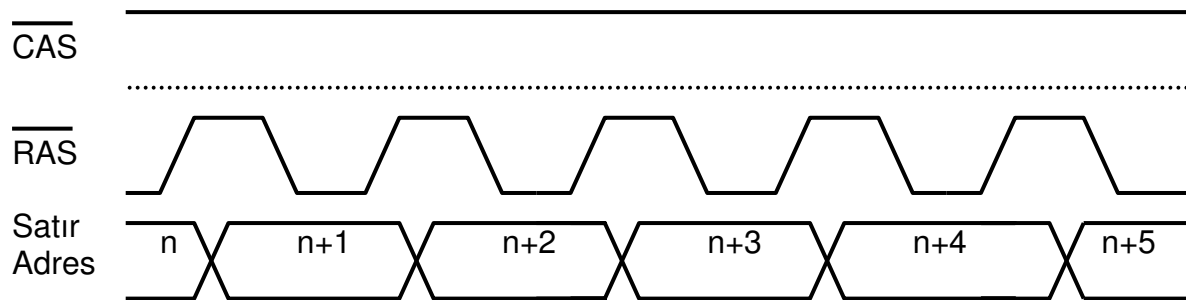


Şekil-12.25 Adres çoğullama için zamanlama diyagramı.

Bir satır tazelenildiğinde 128 bellek hücresi tazelenmiş olur. Satırın tazelenmesi için RAS işaretinin etkin yapılması gerekir. Bir sonraki satırın tazelenmesi için tekrar uyarı girişinin ve RAS işaretinin etkin olması gerekir. BURST tazeleme yönteminde uyarı girişine işlemin başlangıcında işaret gönderilir ve zamanlayıcı sabit frekansta sayar bu frekansla orantılı olarak RAS işareti dış birim tarafından etkin yapılır. Böylece tazeleme bir defada gerçekleştirilir. Fakat bu tazeleme süresince belleği kullanmak isteyen birimler beklemek zorundadır. Şekil-12.27'de tazelemenin zamanlama diyagramı verilmiştir. Tazeme sırasında CAS işareti yüksek seviyede bırakılarak içerideki veri değişimlerinin dışarıdaki birimleri etkilemesi engellenir. Tazeleme işlemi okuma ve yazma işlemlerinin arasına serpiştiren yöntemde ise her satır tazelemenin öncesi uyarı işaretinin gönderilmesi zorunludur.



Şekil-12.26 Örnek dinamik RAM'ın iç yapısı ve tazeleme devresi.



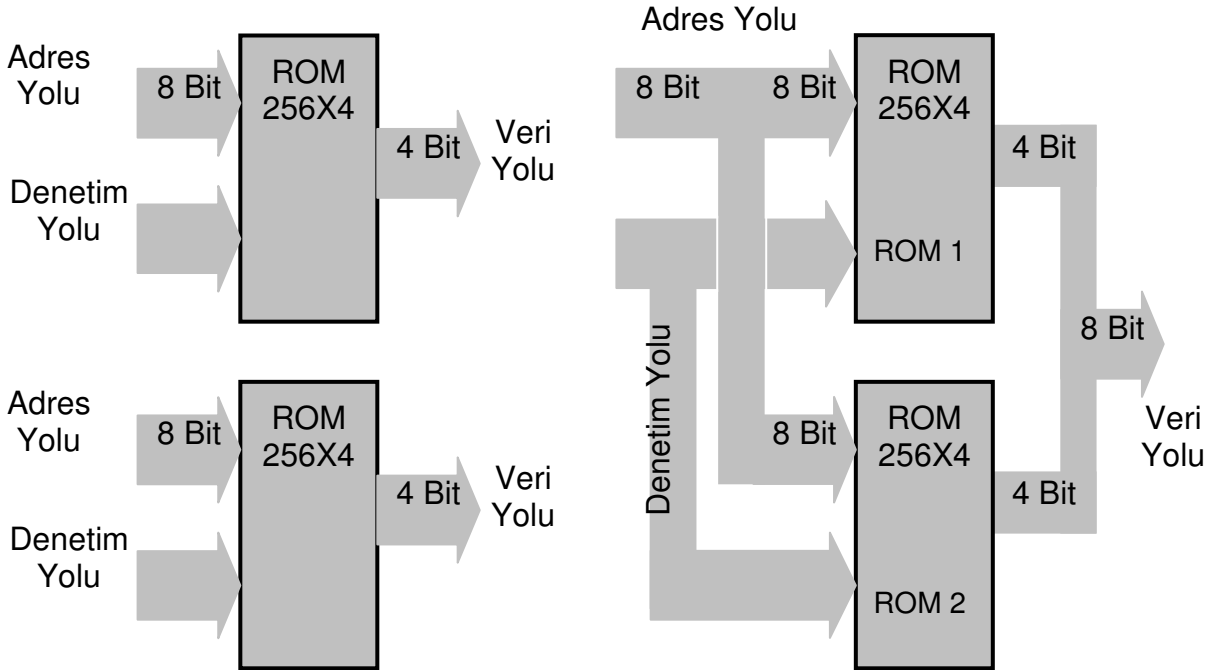
Şekil-12.27 Tazeleme zamanlama diyagramı.

BELLEK GENİŞLETME

Her zaman gereksinim duyulan bellek kapasitesi tek bir tümdevre olarak üretilmez. İstenilen boyutta bellek elde etmek için üretilen belleklerden yola çıkarak sözcük veya satır genişletmesi yapılarak elde edilebilir. Bellek genişletme, adres, veri ve denetim yollarına uygun sayıda bellek yongası eklenerek gerçekleştirilir. Genişletme yapılırken tüm devrenin girişleri veya çıkışları birbirine birleştirilebilir. Birleştirme yapılırken genişletme tipine göre öncelik sırası olmakla beraber adres hatları, veri hatları, denetim hatları olarak sınıflandırılması bağlantıları kolaylaştırır. Daha önce veri hatlarına kısaca veri yolu, adres hatlarına adres yolu ve kontrol hatlarına kısaca kontrol yolu olarak adlandırmıştık bundan sonra bu terimleri kullanacağız.

SÖZCÜK UZUNLUĞUNU ARTIRMA

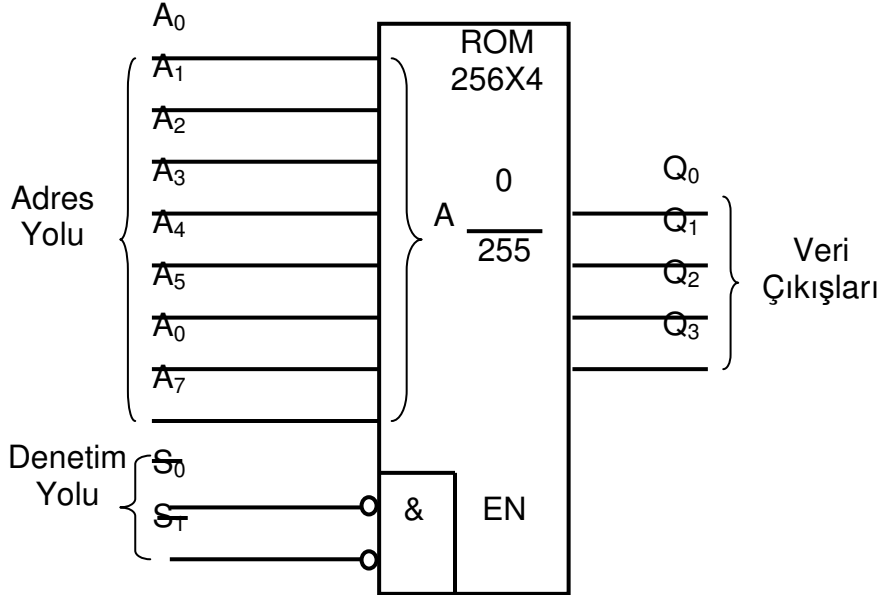
Belleğin sözcük uzunluğunu arttırmak için veri yolundaki bit sayısı arttırılmalıdır. Örneğin 8-bitlik sözcük uzunluğu için her biri 4-bit uzunlukta iki bellek kullanılabilir. Bu genişletmeye enine genişletme adı da verilir.



Şekil-12.28 Sözcük sayısını arttırmak için "enine" genişletmenin temel ilkesi.

Şekil-12.28'de görüldüğü gibi sözcük genişletmesi için giriş kısmında bulunan hatlar birleştirilerek giriş sayısı sabit kalıyor, çıkış kısmındaki hatlar ise birleştirilmiyor ve çıkıştaki hat sayısı iki katına çıkıyor. Girişteki hatlar birleştirilirken aynı isimli olmalarına dikkat edilmelidir. Adres hatlarının sayısı artmadığı için satır sayısı sabit kalacaktır. ($2^8=256$).

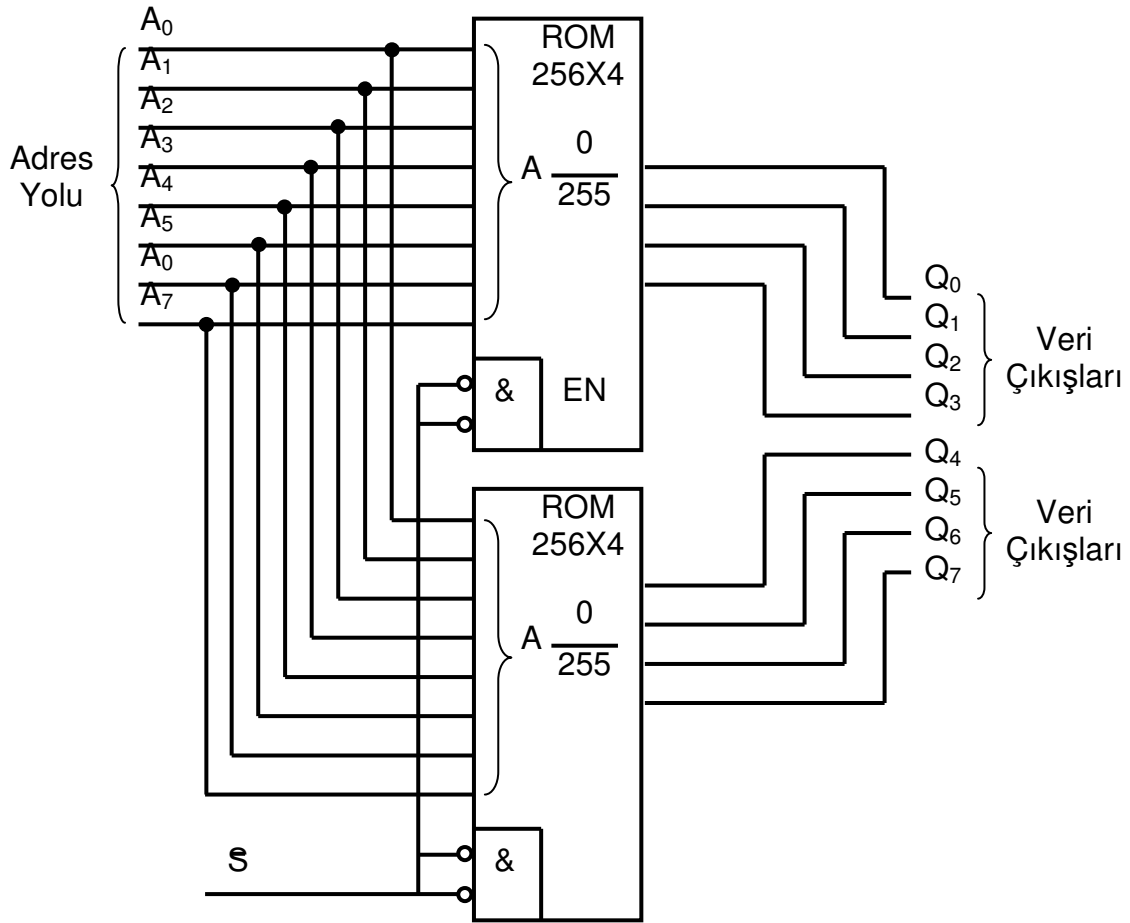
ÖRNEK: Şekil-12.29'da mantık simgesi verilen ROM'dan kullanarak 256X8 boyutlarında bir bellek dizimi elde edin.



Şekil-12.29 256X4 ROM'un mantık simgesi

ÇÖZÜM:

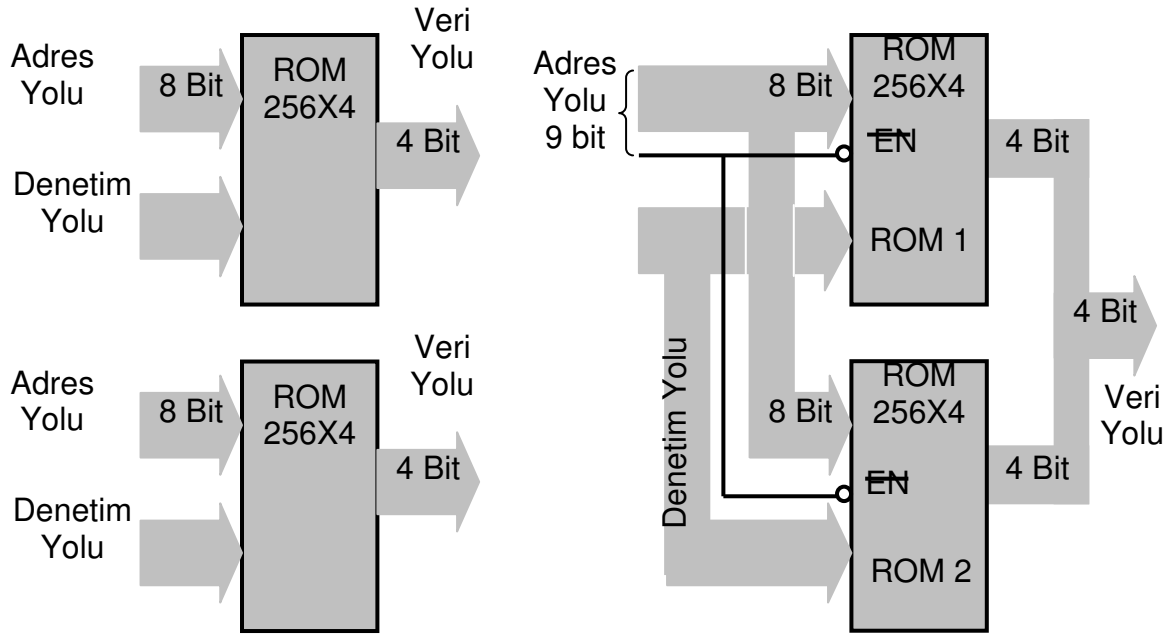
İki adet 256X4 boyutundaki ROM kullanılarak 256X8 boyutunda bellek dizimi Şekil-12.30'da gösterildiği gibi elde edilir.



Şekil-12.30 Kelime uzunluğunun 8'e çıkarılması.

SATIR SAYISI ARTIRMA

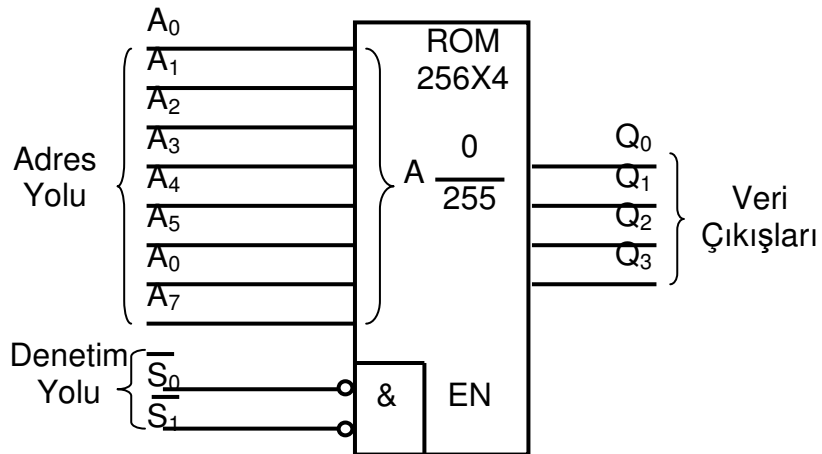
Bellekte saklanabilecek sözcük sayısının artırılması, adres sayısının artırılmasıyla olasıdır. Bu artışı sağlamak için adres bitlerinin sayısı artırılmalıdır. Şekil-12.31'de boyuna artırılma işlemi gösterilmiştir. Adres yolunda bir hat tümdevre adres girişlerinden fazla olacaktır. Bu hat tümdevre seçme görevi üstlenecektir. DÜŞÜK seviye olduğunda ROM 1 etkin olur ve bu tümdevreden satır seçilir. YÜKSEK seviye olduğunda ise ROM 2 etkin olur. Adres yolunda artan hat sayısı 2 veya daha fazla olduğunda sadece DEĞİL geçidi yetmeyecek ve kod çözücü kullanılacaktır.



Şekil-12.31 Satır sayısını arttırmak için "boyuna" genişletmenin temel ilkesi.

ÖRNEK:

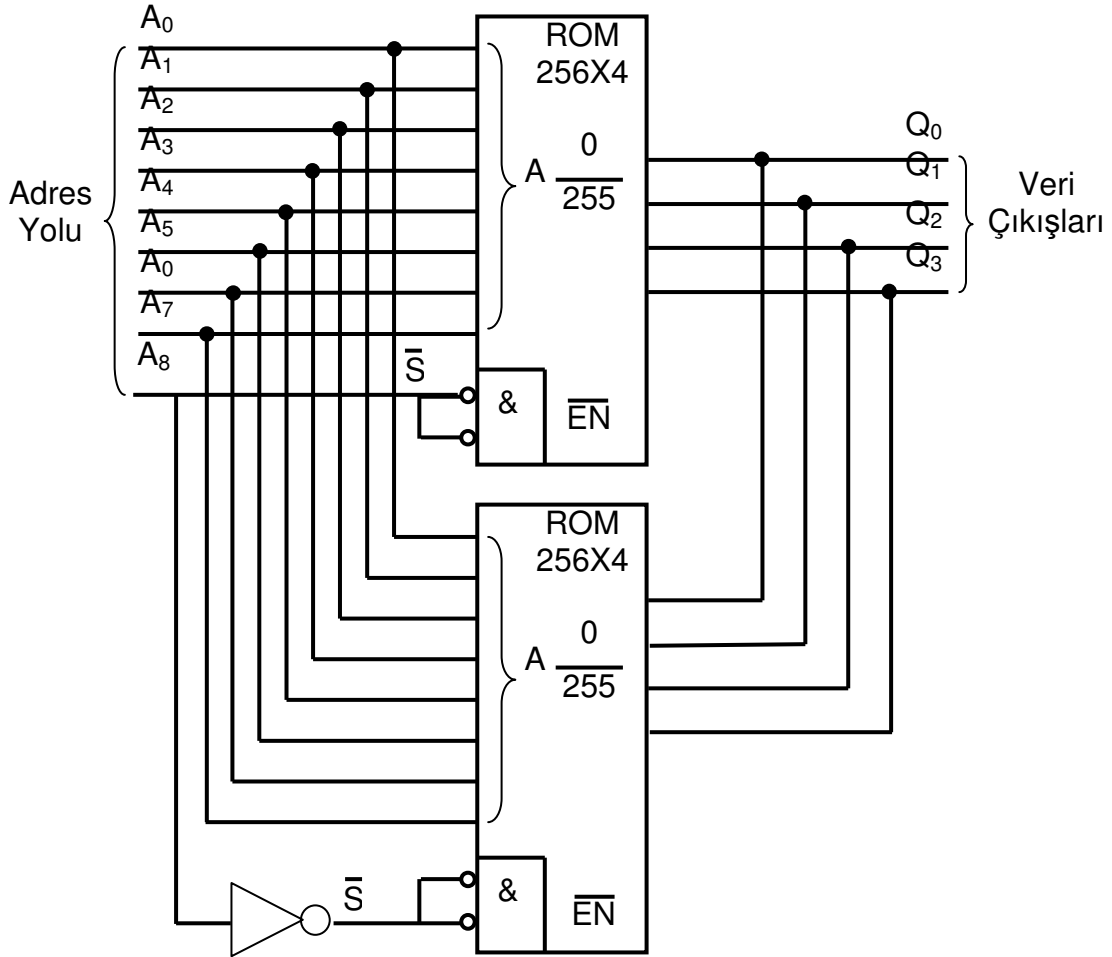
Şekil-12.32'de mantık simgesi verilen belleklerden kullanarak 512X4 kapasite de bellek elde eden bağlantıyı yapınız.



Şekil-12.32 256X4 ROM'un mantık simgesi

ÇÖZÜM:

Satır sayısının iki katına çıkarmak için tümdevrelerin adres bilgisi için dokuzuncu adres girişi olması gerekir. Fakat bu ROM'larda sadece 8 adres girişi vardır. Dokuzuncu adres bilgisi EN (S) girişine uygulanacaktır. İlk 256 satırı oluşturan ROM 1 satırları adreslenirken dokuzuncu bit "0" olacaktır. EN girişi DÜŞÜK seviyede etkin olduğu için doğrudan bağlanabilir. Adres yoluna 00-255 arası adres bilgisi geldiğinde ROM 1 tümdevresi etkin olacaktır. Dokuzuncu bit ROM 2'ye doğrudan bağlanamaz, çünkü bu tümdevre dokuzuncu bitin "1" olduğu durumlarda etkin olacaktır. EN girişi ile uyuşabilmesi için DEĞİL'lenerek bu girişe bağlanması gerekir. Adres yoluna 256-511 arası bilgi geldiğinde ROM 2 etkin olacaktır.

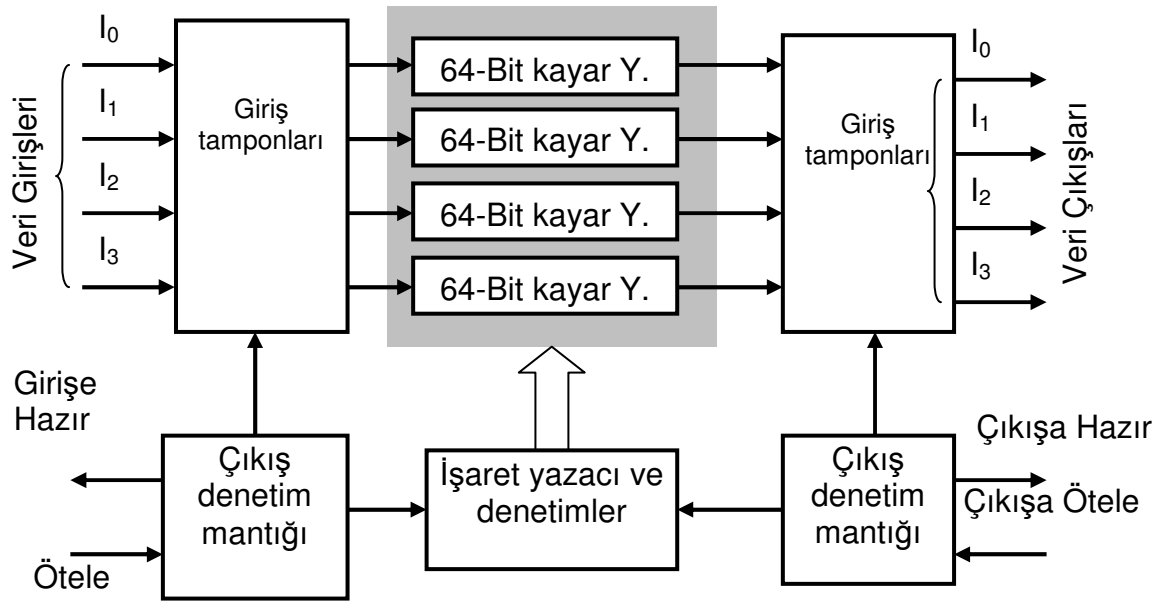


Şekil-12.33 Satır sayısının 512 yapılması.

ÖZEL BELLEK TÜRLERİ

İLK GİREN İLK ÇIKAR (FIFO-FIRST IN FIRST OUT) BELLEKLER

Bu bellek türü kayar yazaçlar ile oluşturulmuştur. FIFO terimi, bu bellek türünün temel çalışma biçiminden kaynaklanır: Belleğe ilk yazılan bit ilk önce okunacaktır. Kayan yazaç ile FIFO yazaç arasındaki en önemli ayırım, tabloda gösterilmiştir. Alışılacağı yazaçta veri biti, yalnızca yeni veri girildikçe kayar. FIFO yazaçta ise girilen veri hemen yazacın boş olan en sağ ucuna kaydırılır.



Şekil-12.34 Tipik FIFO seri belleğin blok diyagramı.

Normal kayar yazaç

Giriş	X	X	X	X	Çıkış
0	0	X	X	X	→
1	1	0	X	X	→
1	1	1	0	X	→
0	0	1	1	0	→

FIFO kayar yazacın

Giriş	—	—	—	—	Çıkış
0	—	—	—	0	→
1	—	—	1	0	→
1	—	1	1	0	→
0	0	1	1	0	→

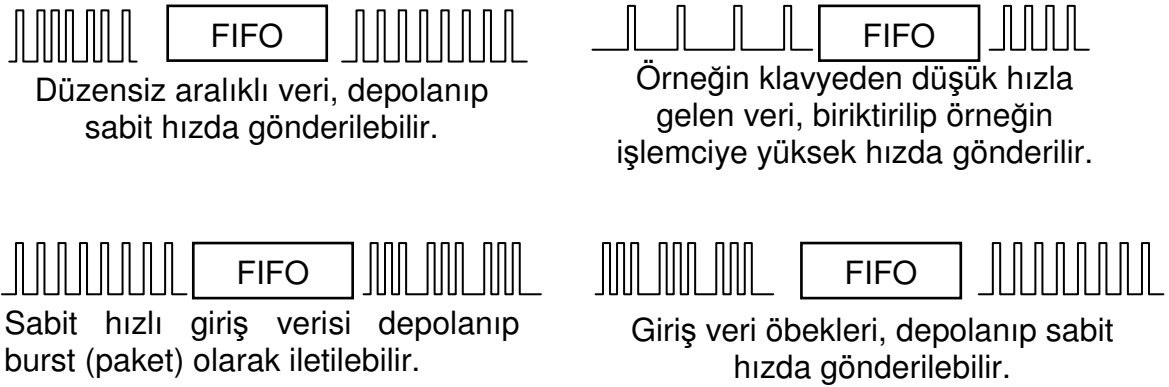
Tablo-12.1 Normal kayar yazaç ile FIFO kayar yazacın çalışması.

Şekil-12.34'te, dört 64-bit seri veri yazacı ve bir 64-bit işaret (denetim) yazacı içeren bir FIFO seri bellek görülmektedir. Veri, içeri ötele (shift in) komutuyla

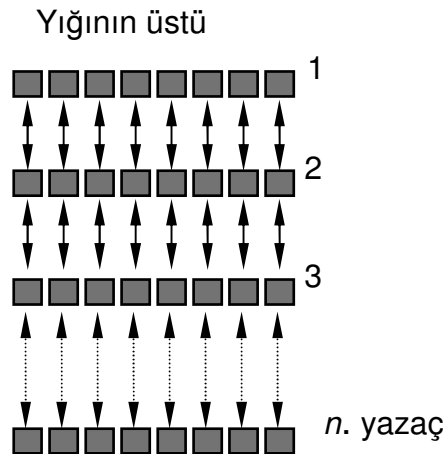
girildiğinde, işaret yazacı denetiminde çıkışa en yakın boş yere kaydırılır. Veri dolu gözlere geçemez. Çıkışa ötele (shift out) komutu ile bir veri dışarı verince, içerideki diğer bitler hemen birer basamak sağa kaydırılır. Eşzamansız bir FIFO bellekte veri, kullanılan iki ayrı saat işareti yardımıyla, veri girişinden bağımsız olarak dışarı verilir.

FIFO UYGULAMALARI

En önemli uygulamalardan biri, veri hızları birbirinden değişik olan iki sistem arasında iletişim sağlamaktır. Veri FIFO belleğe bir hızda girilip başka bir hızda alınabilir. Şekil-12.35'de böyle durumlarda FIFO yazacın nasıl kullanılabildiği açıklanmaktadır.



Şekil-12.35 FIFO yazacın veri-hızı tamponlama uygulamaları.

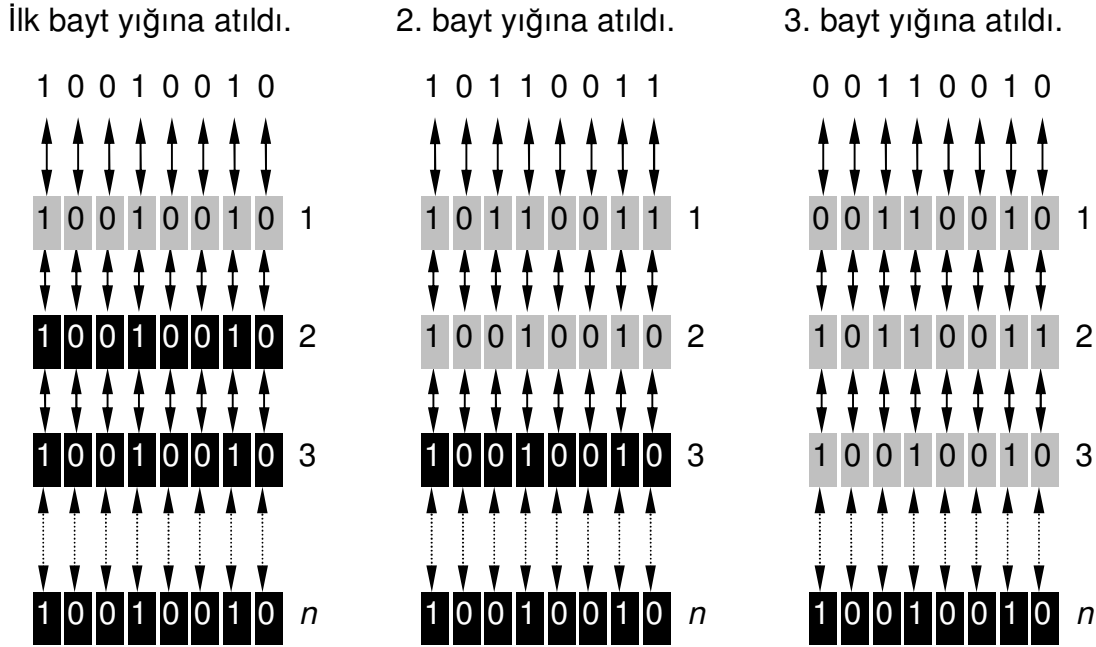


Şekil 12.36 Yazac yığını.

SON GİREN İLK ÇIKAR BELLEKLER

LIFO (Last In First Out) bellekler mikroişlemci ve diğer hesaplama sistemlerinde

bulunurlar. Verinin saklanması, ters sırayla alınmasını sağlarlar yani, yüklenen son veri ilk önce okunur. *Yığınlar (stack)* LIFO bellekler yaygın olarak push-down (aşağı-bastır) yığınlar olarak adlandırılır. Bazı sistemlerde şekil 5-36'daki gibi bir grup yazaçla gerçekleştirilirler. Bir yığında istenen sayıda yazaç olabilir ve en üstteki yazaç, yığın tepesi (top-of-stack) olarak adlandırılır.



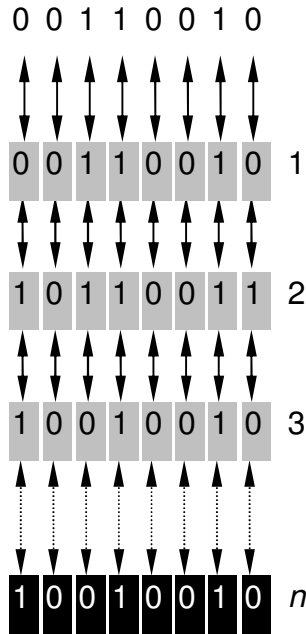
Şekil 12.37 LIFO yığına veri yazımı.

Bu tür yığınlara veri paralel olarak yığın üstünden yerleştirilir, gelen her veri bir öncekini bir adım dibe doğru iter. Bu tür yığına veri yazmanın mantığı şekil-12.37'de gösterilmiştir. Öncelikle yığına arka arkaya 3 bayt veri atılmıştır. Bu veriler okunmak istendiğinde en son yazılandan başlayarak sırayla okunabilir. Üsten veri okundukça altta kalan veride okunan verinin boşalttığı bir üst satıra yerleşir. Şekil-12.38'de LIFO yığından verinin okunması adım adım gösterilmiştir.

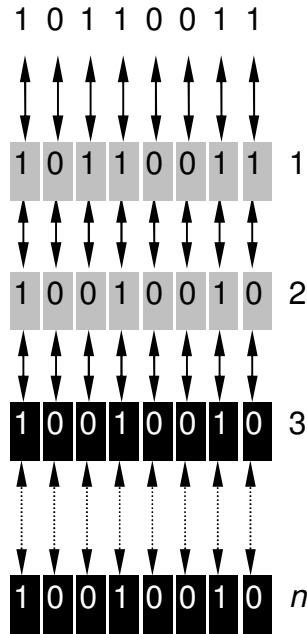
RAM YIĞINLAR

Bazı mikroişlemci tabanlı sistemlerde LIFO bellek elde etmek için değişik bir yaklaşım kullanılır. Yazaçlar kullanmak yerine, sistem RAM'inin bir bölümü yığın olarak kullanılır. Şekil-12.39'daki gibi bayt düzenli (8-bit sözcük) bir belleğin onaltı bitlik adres kodunu onaltılık (hexadecimal) olarak gösteririz. İkili `0000000000001111` adresi, onaltılık `000F` olarak gösterilir. 16-Bit adres en küçük `000016`, en büyük `FFFF16` değerlerini alabilir. RAM'in bir bölümünün yığın olarak kullanıldığını düşünelim. Bu durumda *yığın göstergesi* (stack pointer) olarak adlandırılan özel ayrı bir yazaç, yığın tepesinin adresini saklar. Verinin yığına atılması şu şekilde açıklanabilir: Veri baytı olağan yazma işlemiyle `00FF` adresine yazılır. Bu işlemden sonra yığın göstergesinin içeriği bir azaltılarak `00FE` değerine getirilir. Şekil-12.40 (a)'da bu durum gösterilmiştir. Bu anda yığının üstü

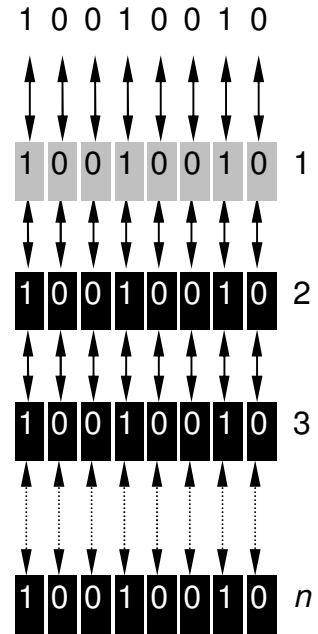
Önceden 3 bayt veri saklanmış durumda. 3. Bayt yığının üstünde.



3. bayt yığından çekilince yığının üstüne 2. bayt yerleşir.

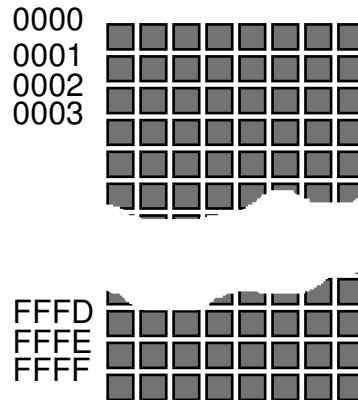


2. bayt yığından çekilince yığının üstüne 1. bayt yerleşir.



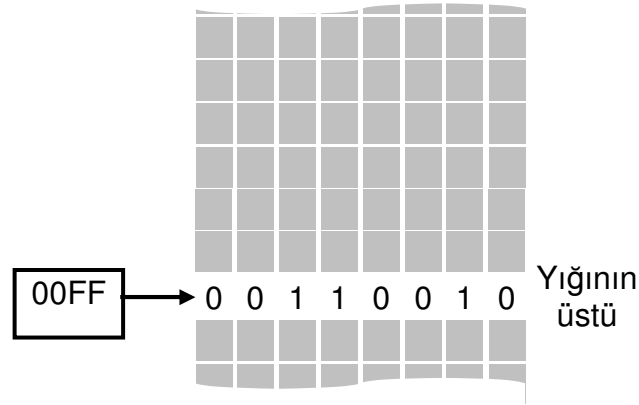
Şekil 12.38 LIFO yığından veri okunması.

16-Bit adres

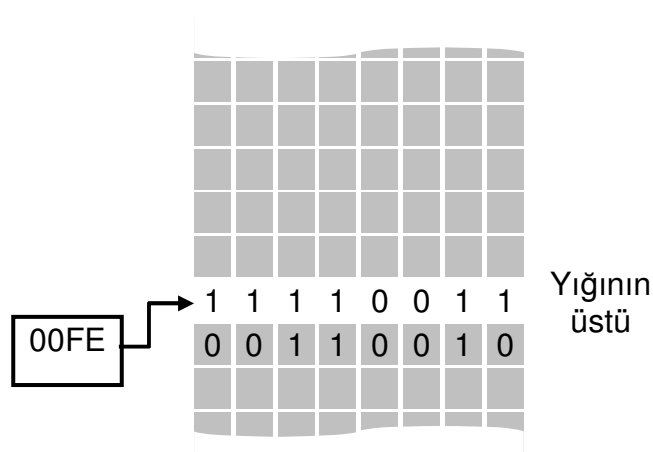


Şekil 12.39 64 Kbaytlık belleğin yapısı.

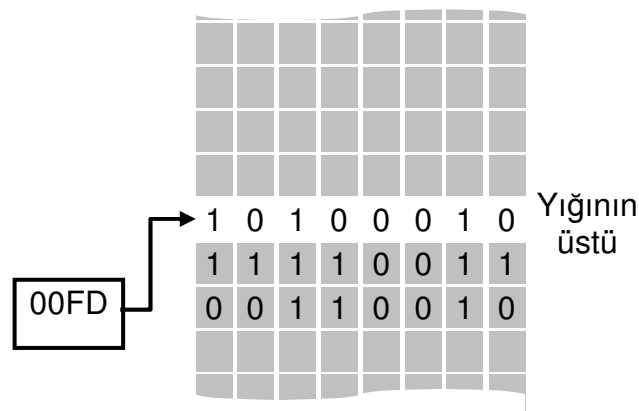
Şekil-12.40'da yığına atılan üçüncü veriden sonra yığın göstergesinin içeriği 00FC değerine gelir. Okuma işlemi yapılırken ilk önce yığın göstergesinde yazılı adres okunur ve yığın göstergesinin içeriği bir artırılır, veri yığından çekilir. Unutulmamalı ki RAMden okuma yok edici değildir ve okunan veri bellekte kalır. Bu verinin silinmesi ancak aynı bellek yerine yeni bir veri yazılmasıyla olur.



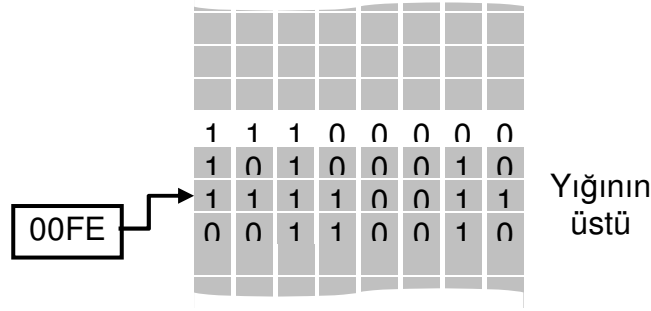
(a) Yığın göstericinin içeriği yığının üstünü gösteriyor ve ilk bayt yığına atılıyor.



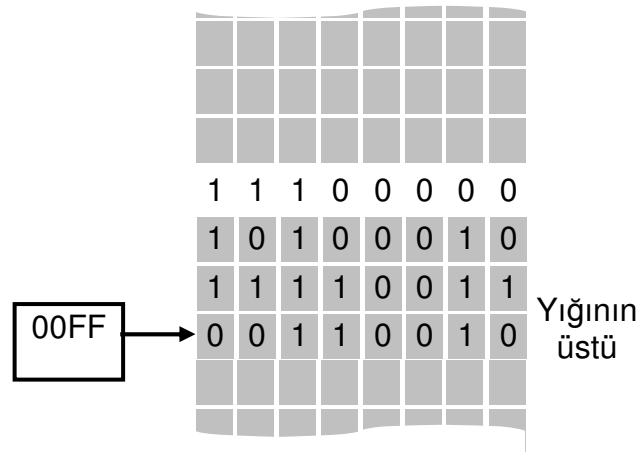
(b) Yığın göstericinin içeriği bir azaltılır ve 2. bayt yığına atıldığında yığının yeni üstünü işaretler.



(c) Yığın göstericinin içeriği bir azaltılır 3. bayt yığına atıldığında yığının yeni üstünü işaretler.



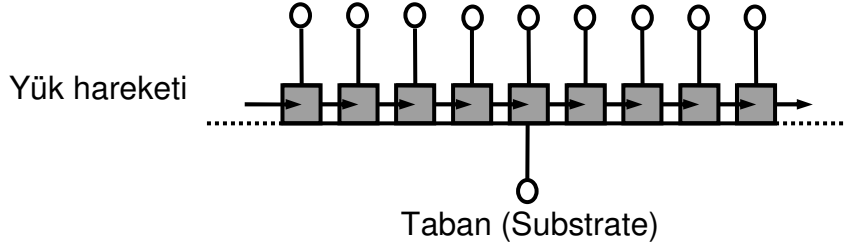
(c) Yığın göstericinin içeriği bir arttırılır 2. bayt yığının üstünden çekilir



Şekil 12.41 RAM yığından veri çekilmesi.

CCD BELLEKLER

CCD (charge-coupled device/yük bağımlı aygıt) bellek veriyi sığaçlar üzerinde yük olarak saklar. Dinamik RAMden ayrılan yanı olarak bellek gözünde transistör bulunmamasını söyleyebiliriz. Yüksek yoğunluklarda üretilebilmesi CCDlerin en önemli üstünlüğüdür. CCD bellek içinde *kanal* denilen uzun, yarıiletken sığaç satırları vardır. Veri kanala, "0" için küçük, "1" için büyük bir yük mantığıyla seri olarak girilir. Yeni veri girildikçe bu yük paketleri, saat vurularıyla kanal boyunca kaydırılır. RAMde olduğu gibi yükler periyodik olarak yenilenmelidir. Bu işlem yük paketlerinin yenileme devresinden seri olarak geçirilmesiyle yapılır. Şekil-12.42'de mantık şeması gösterilen bu bellek türünde veri seri olarak tutulduğundan erişim süresi görece uzundur.



Şekil 12.42 CCD kanalı.

Son yıllarda üretilen sayısal fotoğraf makineleri ve kameralarda veri geçici olarak CCD'lerde saklanmaktadır. Yoğun olarak kullanmaları sonucu teknolojileri geliştirilmiş ve maliyetleri dinamik RAM ile aynı seviyededir.

BİLGİSAYARDA KULLANILAN BELLEKLER

Rasgele erişimli bellekler (RAM) bilgisayarlarda işletim sırasında programların ve verilerin işleyişini takip etmek için kullanılan bellek türüdür. Bilgisayar kapatıldığında RAM'in içeriği silinir. Yani, RAM bellek geçici bir bellek çeşididir. Bugün kullanılmakta olan farklı RAM teknolojileri bulunmaktadır, fakat bu farklı rasgele erişimli bellek türleri bir defada ortaya çıkmamıştır. Bugünkü bellek teknolojisinde belirgin bir gelişme olmuştur. Bu gelişme bilgisayar tasarımcılarının ve kullanıcılarının sürekli gereksinim duyduğu artan hız isteğini karşılamak için bir çalışma sonucunda ortaya çıkmıştır. Bilgisayar işlemcilerinin hızları çok çabuk arttırılmakta olduğundan, bellek teknolojisi de bu hıza ayak uydurmaya zorlanmıştır. Üretici ve tüketicilerin artan isteğine ek olarak, işlemci hızlarındaki bu sürekli artış eğilimi bellek tasarımcılarını da gelecek yıllarda gelişmeye devam etmelerine ve bellek teknolojilerini yeniden tasarlamaya zorlayacaktır.

DİNAMİK RAMLAR

Bilgisayarlarda ana bellek olarak en yaygın kullanılan RAM çeşitlerinden biri Dinamik RAM (DRAM) olarak bilinen RAMdir. Dinamik RAM verisini korumak için bir tazeleme devresine gereksinim duyan RAM türüdür. Tazeleme devresi, hücrenin kullanılıyor olup olmadığına bakmaksızın saniyede yüzlerce defa her bellek gözünün (hücresinin) içeriğini okuyarak yeniden yazar. *Dinamik* isminin verilmesinin nedeni, bellek hücrelerinin içindeki verinin sürekli tazelenmesidir. Her bellek hücresinin sürekli tazelenmesi DRAMin statik RAMden (SRAM) daha yavaş işlemesine neden olmaktadır. Bu olguya rağmen, neredeyse tüm bilgisayarlarda DRAMin ana bellek olarak kullanılmasının nedeni belirgin olarak SRAMden daha ucuz olması ve daha az yer kaplamasıdır. Bunun nedeni, DRAMler her bit için yalnızca bir transistör ve bir kondansatöre gereksinim duyarken SRAMler her bit için 4 ile 6 transistör kullanmaktadır.

DRAMin yıllar içinde geliştirilmiş farklı türevleri bulunmaktadır. En çok kullanılan DRAM çeşitlerinden bazıları şunlardır:

- Page Mode
- Fast Page Mode (FPM)
- Extended Data Output (EDO)
- Burst Extended Data Output (BEDO)

Bilgisayar endüstrisinde yıllarca kullanılan yalnızca bir DRAM çeşidi vardı. Page mode RAM, ve daha sonra DRAM piyasasına bir hayli süre hakim olan FPM RAM. Page mode ve fast page mode arasındaki temel fark fast page mode mantığının sonraki RAM erişiminin önceki erişimle aynı satırdan yapılacağını varsaymasıdır. Fast page mode RAM 70ns veya 60ns erişim süresine sahiptir, ki daha düşük erişim süresi daha hızlı veri iletimi anlamına gelmektedir. 66MHz'lik işlemci yolu hızı ile 60ns'lik FPM RAM kullanılmalıdır. MİB'nin FPM RAME en hızlı erişim süresi dört veri grubu okuması 5-3-3-3'tür. Bunun anlamı, ilk okuma 5 saat çevrimi ve diğer üç okuma ise tamamlamak için 3 saat çevrimi sürmektedir.

Geliştirilen diğer RAM türü ise EDO RAMdir. Genişletilmiş veri çıkışlı RAM FPM RAMin değişik bir şeklidir. İkisi arasındaki ana farklılık EDO'nun FPM'den daha iyi hızlı sıralı okuma yapabiliyor olmasıdır. Hızlı MİBler ile çalışırken bu küçük fark önemlidir. EDO DRAMler 70ns, 60ns veya 50ns'lik erişim sürelerine sahiptir. Doğru yonga seti (chipset) ile, EDO dört okuma başına saat çevrimi sayısı açısından FPM'nin başarısını tamamen elinden almıştır. Örneğin, Triton yonga seti ile, fark X-3-3-3'e göre X-2-2-2'dir. Bunun sonucu olarak EDO RAMin başarımı FPM RAME oranla belirgin bir şekilde artmıştır. EDO RAMin ana sorunu ise veri yolu hızı 66MHz'ten yüksek olan MİBlerle çalışmasının zor olmasıdır.

BEDO RAM, EDO RAMin başarımını arttırmak için tasarlanmış yeni bir şeklidir. Adından da anlaşılacağı gibi, BEDO RAM verileri grup olarak okumaktadır. Adres sağlandıktan sonra, diğer üç veri elemanı her birini üç saat çevriminde tamamlayabilmektedir. Bu da 5-1-1-1'lik erişim hızı demektir. Bu RAM türünde de bazı çekinceler bulunmaktadır. Örneğin, BEDO RAM sınırlı sayıda yonga setleri tarafından desteklenmektedir. BEDO RAMin diğer sorunu, EDO RAMde olduğu gibi, 66MHz'ten yüksek veri yolu hızlarında doğru çalışmamasıdır.

Bellek piyasasına giren en yeni DRAMlerden biri Senkron DRAMdir (SDRAM). Bu yeni RAM çeşidi diğer DRAM çeşitlerine göre daha uyumlu ve yüksek çalışma verimi ile çalışmaktadır. Örneğin, SDRAM çok geniş bir aralıktaki yonga setlerini desteklemektedir. Sistem saatiyle eşzamanlı bütün giriş ve çıkış sinyallerini de idare edebilmektedir. Bu özellik daha önce yalnızca statik RAMlere özgüydü. SDRAMlerin erişim süresi dört veri grubunun okunması için 5-1-1-1'de EDO RAMlerle karşılaştırılabilir düzeydedir. SDRAMin en güçlü özelliklerinden biri 100MHz'e kadar yol hızlarını idare edebilmesidir. Buna göre, SDRAM yakın gelecekte ana DRAM formlarından biri olacaktır.

STATİK RAMLAR

Statik RAM (SRAM) bilgisayarlarda yaygın olarak kullanılan diğer bir RAM şeklidir. Saniyede yüzlerce kez tazelenmesi gereken DRAME zıt olarak SRAM dışarıdan tazelenmesine gerek kalmadan verisini koruyabilmektedir. Başarım açısından DRAMden üstündür. Buna karşın, SRAM boyutunun büyük olması ve pahalı

olmasından dolayı bilgisayarlarda ana bellek olarak pek kullanılmaz. Her SRAM biti dört ile altı arası transistörden oluşmaktadır, ki bu da DRAMe oranla boyutunun neden büyük olduğunu açıklamaktadır. Bunun sonucu olarak, statik RAM bilgisayar içinde düzey 1 ve düzey 2 cache (ön) bellekler olarak kullanılır.

Ön bellek başarımını arttırmak için geliştirilen birkaç SRAM çeşidi de bulunmaktadır. Bunlar üç sınıfta toplanabilir:

- Asenkron Statik RAM (Async SRAM)
- Senkron Grup (Burst) Statik RAM (Synch SRAM)
- Pipelined Burst Statik RAM (PLB SRAM)

Asenkron statik RAM yukarıda bahsedilen SRAM tasarımları içinde en eski olanıdır. Bu RAM çeşidi ilk düzey 2 ön belleklerin kullanıldığı 386'ların zamanından beri yıllarca kullanılmıştır. *Asenkron* terimi ön bellek içindeki veri iletiminin sistem saatine bağlı olmadığını ifade etmektedir. Bunun sonucu olarak, yaşına ek olarak, Async SRAM en yavaş SRAMdir. Tüm SRAMler içinde en yavaş olmasının aksine, asenkron SRAM hala DRAMden daha hızlı erişime sahiptir. Async SRAMler 20, 15 veya 12ns'lik erişim sürelerine sahiptir. Ne yazık ki, bu erişim süreleri senkron veri erişimine izin verecek kadar hızlı değildir. Bu nedenle, MİB bu cache RAMde veri hazır olana kadar beklemeye zorlanmaktadır. Bu bekleme süresi daha hızlı erişim sürelerine sahip olmasından dolayı DRAMinkinden daha azdır. 33MHz'e kadar olan yol hızlarında bu SRAM çeşidi 2-1-1-1'lik zamanlama hızına sahiptir. Bundan yüksek yol hızları için SRAM 3-2-2-2'ye yükselmektedir. Çoğu yol hızları 66MHz ve daha yüksek olduğundan, asenkron SRAMlerin kullanımında zamanlama hızındaki bu kötüleşme belirgin bir çekince yaratır.

Senkron burst statik RAM asenkron veri iletimindeki bu bekleme süresini gidermek için geliştirilmiştir. Asenkron statik RAMin aksine, senkron RAM bellek yol saatine bağlıdır. Sistem saatinin her tetiklemesinde, Synch RAM varsa veriyi ön belleğe iletebilir veya ön bellekten alabilir. Bu demektir ki, senkron RAM yavaşlamadan asenkron RAME göre daha hızlı sistem hızlarını idare edebilmektedir. Gerçekten, 66MHz'lik yol hızında, senkron RAM burst çevrimi 3-1-1-1 olan pipelined burst statik RAMden bile 2-1-1-1'lik burst çevrimiyle daha iyi başarımla sergileyebilmektedir. Sync Burst SRAM 8.5'tan 12ns'ye kadar adres/veri sürelerine sahiptir. Bu belirgin olarak asenkron RAMden daha hızlıdır. Ne yazık ki, 66MHz'ten yüksek yol hızlarında, çalışması 3-2-2-2 burst çevrimine kadar yavaşlamaktadır. Pipelined burst statik RAMin SRAM piyasasının çoğunluğunu ele geçirmesinin birincil nedeni de bu olabilir.

Pipelined burst statik RAM senkron SRAM ile yaklaşık aynı zamanlarda piyasaya çıkmıştır. İkisi arasındaki en temel farklılık pipelined RAMin çok daha yüksek sistem yolu hızlarını idare edebilir olmasıdır. Örneğin, pipelined burst statik RAM 133MHz'e kadar olan yol hızlarında çalışabilir. Bu RAMin pipeliningi giriş veya çıkış yazaçlarının kullanımı aracılığıyla yapılabilmektedir. Yukarıda bahsedildiği gibi, bu RAM kullanılarak dört verinin okunması için burst çevrimleri 3-1-1-1 çevrim sürmektedir. İlk okumadaki ek çevrim yazaçların yüklenmesinden kaynaklanmaktadır. PLB SRAM 75MHz ve daha yukarı yol hızlarında

kullanılabilecek en hızlı RAMdir. Bu RAM için adres/veri süreleri yalnızca 4.5 ile 8ns arasındadır. Her iki durumda 3-1-1-1'lik çevrim süresini sağladığından daha yavaş yol hızlarında da başarımı iyidir. Kararlı olarak artan işlemcilerle birlikte, sistem yol hızlarında da kararlı bir artış olmalıdır. Çoğu yeni yol hızları en azından 66MHz'tir. Bu nedenle, pipelined burst SRAM önümüzdeki birkaç yıl içinde en fazla potansiyele sahip olan RAMdir.

RAMLARIN GELECEĞİ

Rambus DRAM (RDRAM) piyasaya girmek üzere olan tamamen yeni bir RAM çeşididir. RDRAM *Rambus* adıyla bilinen bir firma tarafından geliştirilmiştir. Halen Nintendo 64TM video oyun sistemleri gibi PC multimedia uygulamalarında kullanılmaktadır, ancak bilgisayarlar için henüz kullanıma girmemiştir. Rambus 1999 yılında belleğini 1.6GB/s'ye kadar bellek veri iletimi hızlarına çıkarmayı hedeflemekteydi. Hızının yanı sıra, Rambus DRAMi diğer RAM çeşitlerinden ayıran ana etmen aşırı yüksek yol hızlarını idare edebilme yeteneğidir. Şu andaki DRAM tasarımları bellek denetleyicisine 500MHz'ten 600MHz'e kadar olan hızlara yükselmiştir. İşlemci yol hızları daha yüksek düzeylere çıkmaya devam ettikçe, bu etmenler Rambus DRAMinin şu anki RAM piyasasını ele geçirmesini garantilemektedir.

Rambus tasarımı, veriyi Rambus Kanalı adı verilen 8 bitlik bir yoldan ileten yüksek hızlı arabirim üzerine kurulmuştur. Bunun yanı sıra, DRAM altında yatan temel çekirdek teknolojisi halen geçerlidir. Şu anki 5-1-1-1'lik en hızlı DRAMe benzer veri burst çevrimine sahiptir. Ancak, Rambus DRAMin bellek denetleyicisine veri iletim hızı önceki diğer tüm DRAMleri geçmektedir. Örneğin, Rambus sistemi dahili olarak 533MHz'lik bir hızda salındığında, veri her 3.75ns'de iletilmektedir. Bu da demektir ki, Rambus başarımı yüksek yol hızlarında SRAMin erişim süresini de geçmektedir. Buna ek olarak, aynı çekirdek yapısını paylaştıklarından dolayı, Rambus DRAMin maliyeti şu anki DRAM kadar uygundur. Rambus yongasının veri grubu okumada azalma olmadan hızlı yol hızlarını idare edebilmesindeki kolaylığa daha önce erişilememiştir. RAM yongalarının yeni nesline doğru kendilerine yol bulacaklardır.

SONUÇ

Bugün halihazırda çok sayıda farklı RAM çeşidi bulunmaktadır. DRAMin eski versiyonları az sayıda eski bilgisayarlarda bulunabilir. Çoğu bilgisayar satıcıları daha yeni SDRAM yongalarına geçmiştir. Bunun nedeni daha yeni bilgisayarlarda yüksek yol hızlarında verinin verimli olarak işleme yeteneğine olan belirgin taleptir. Bu talep sürekli olarak artan bilgisayar hızlarının bir sonucudur, ki bu da hem üreticilerin hem de tüketicilerin daha hızlı işlem hızına olan artan taleplerinden kaynaklanmaktadır. Kullanılmakta olan bilgisayar işlemcileri 100MHz'ten daha yüksek yol hızlarına gereksinim duymamaktadırlar, fakat önümüzdeki birkaç yıl içinde daha büyük ve daha iyi işlemciler geliştirildikçe bu istek artacaktır. Bu hızlar 200MHz'i geçmeye başladığında, SDRAM yetmemeye başlayacaktır. Bu yüksek yol hızı taleplerini karşılayacak tek uygun çözüm

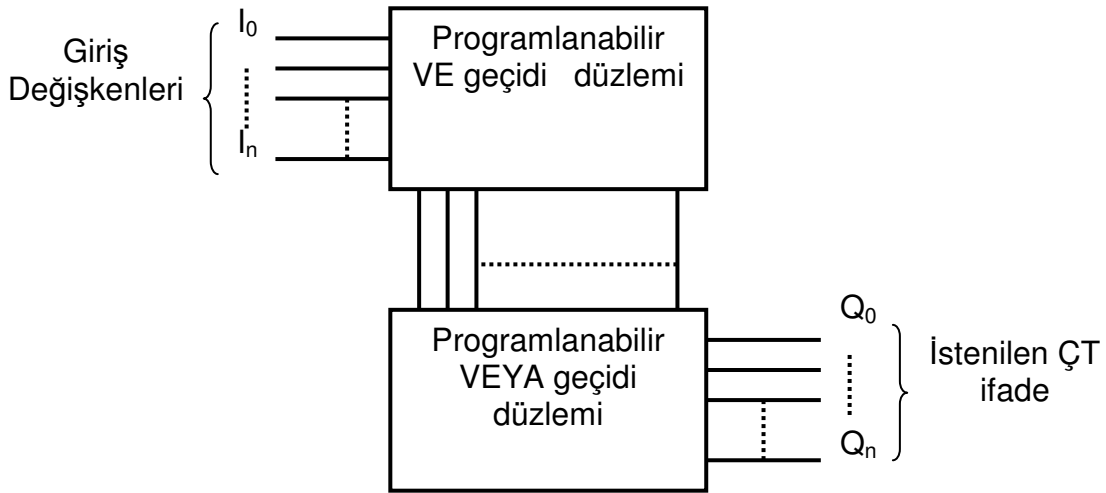
alternatif DRAM teknolojilerine geçmek olacaktır. Piyasaya sürüldüğünde, RDRAMın bellek piyasasında belirgin bir etki yaratacak olmasının nedeni de budur. Bu çok yüksek yol hızlarını kaldıracak yeteneğe sahip tek RAM budur. Rambus DRAM veya bunun bir türevi tüm RAM piyasasında gelecek yıllarda baskın gelecektir.

PROGRAMLANABİLİR MANTIK AYGITLARI

Programlanabilir mantık aygıtları (PLD), sigortalı programlama biçimleriyle PROMlara benzerler ama uygulamaları oldukça değişiktir. PLD, boole mantık işlevlerini gerçekleştirmede kullanılır ve genellikle ayrı ayrı kullanılacak bir kaç geçit ve ikili tümdevrelerinin yerini alabilir. Böylece PLD kullanımıyla, baskı devre kartının küçülmesi ve kullanılan devre elemanı sayısının azalması nedeniyle, üretim sırasında yerden ve maliyetten oldukça kazanç sağlanır.

PROGRAMLANABİLİR MANTIK DİZİM (PLA)

PLA, programlanabilir bir VE düzlemi ve programlanabilir bir VEYA düzlemi içerir. Bazı firmalar yapısında çok az değişiklik yaparak PLA kısaltması yerine PAL kısaltmasını kullanır. VE düzlemi, giriş değişkenlerinin çarpımlarını üretmek üzere, VEYA düzlemi de bu çarpanları birleştirerek çarpımların toplamı/ÇT (sum-of-products/SOP) işlevlerini oluşturmak üzere programlanabilir.



Şekil-12.43 PLA'nın blok şeması.

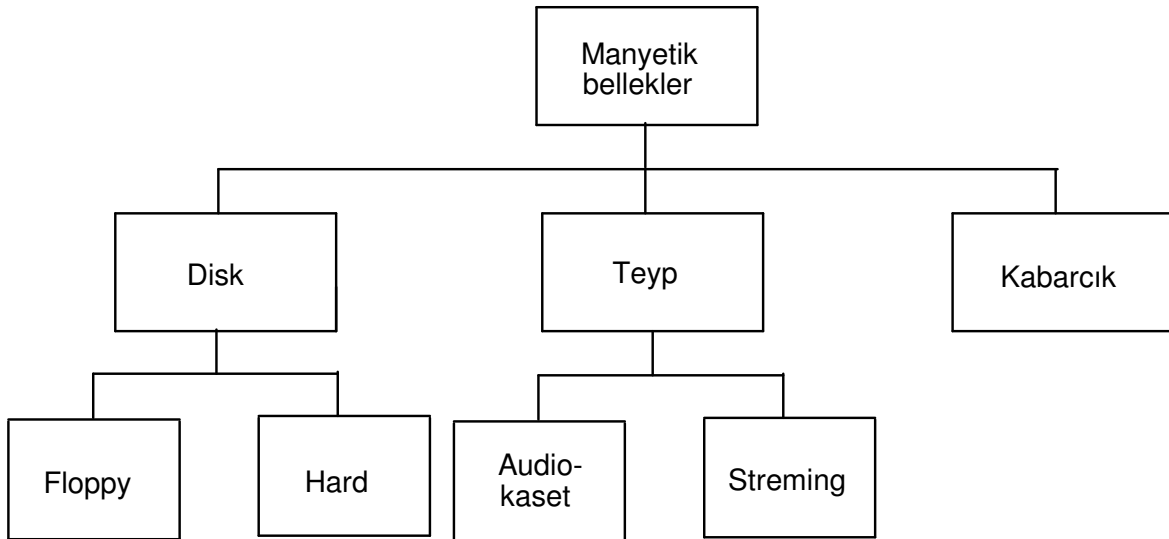
VE ve VEYA düzlemlerinde, kesişim noktaları sigortalarla birleştirilmiştir. Bu sigortalar PROM programlama benzer yöntemle atılarak veya sağlam bırakılarak istenilen devre bağlantısı yapılabilir. PLA'lar da universal programlayıcı kullanarak programlanabilir. Şekil-12.43'de PLA'nın blok şeması gösterilmiştir.

MANYETİK BELLEK ÇEŞİTLERİ

Manyetik depolama aygıtı bilgisayar sistemlerinde öncelikle kütle veri depolamada

kullanılır. Bu aygıtlar bazen harici veya ikincil bellekler olarak adlandırılır. Manyetik bellekler yarıiletken belleklerden çok daha yavaş erişim zamanına sahiptir; dolayısıyla bunların uygulamaları çok miktardaki verinin kalıcı veya yarı kalıcı depolanmasıyla sınırlıdır. Manyetik bellekler 3 temel kategoride toplanmaktadır; disk, teyp ve bubble (kabarcık).

Bilgisayar sistemlerinde kullanılan 2 temel çeşit disk vardır: floppy veya esnek diskler ve hard veya sabit diskler. Temel olarak, floppy disk, hard diskten daha az depolama kapasitesine sahiptir. Fakat floppy taşınabilir ve 3.5 inç'lik çeşidi gömlek cebinde taşınabilir. Günümüzde floppy disklerin yerini birkaç GB mertebesindeki taşınabilir bellekler almıştır. Hard disk nispeten hızlı olarak erişilebilen daha geniş depolama kapasitesine sahiptir. Bilgisayar sistemlerinde bazen kullanılan 2 temel teyp depolama çeşidi audio kaset teyp ve streaming teyptir. Eskiden olduğu kadar yaygın olarak kullanılmamalarına rağmen, audio kasetler eski kayıtların uzun süreli depolanmaları için büyük ve orta ölçekli işletmeler tarafından yaygın bir şekilde kullanılmıştır. "Mass" depolama ve disk-tabanlı verinin yedeklenmesi için günümüzde GB mertebesinde kapasiteye sahip bilgisayara dışarıdan monte edilebilen hard diskler kullanılmaktadır.



Şekil-12.44 Manyetik bellek ailesi

Manyetik kabarcık bellek disk veya teypten nispeten daha yeni bir gelişmedir ve daha hızlıdır. Ancak, pahalı olmasından dolayı kullanımı şimdilik sınırlıdır. Teknolojik olarak gelişimini tamamlamış değildir. Hatta son yıllarda bu konudaki araştırmalar rafa kaldırılmıştır. Araştırmalar daha çok mekanik bileşen içermeyen elektronik ve optik sistemler üzerine yönelmiştir. Şekil-12.44'te manyetik belleğin sınıflandırılması görülmektedir.

BÖLÜM

13

DAC VE ADC

Sayısal sistemler kendi içerisinde işlem yaparken ikilik sayıları kullanır. Dış çevre birimleri ise bu sayılar ile çalışmaz, çoğunlukla analog işaret ile çalışırlar. Bu iki sistemin birlikte kullanılabilmesi için dönüşüm zorunludur. Çevre birimi çıkışındaki verinin ikilik olarak işlenebilmesi için ikilik sayı sistemine dönüştürülmesi için kullanılan devreye ANALOG'tan SAYISALA dönüştürücü adı verilir. Kısaca ADC olarak adlandırılır. İşlenen verinin tekrar çevre birimine gönderilebilmesi için analog değere dönüştürülmesi gerekir. Bu işleme yapan devreye SAYISAL'dan ANALOG'a dönüştürücü adı verilir kısaca DAC olarak kullanılır. Bu bölümde ADC ve DAC'ların yapısı, çalışması ve kullanımı öğrenilecektir. İncelemeye DAC'tan başlanacaktır.

SAYISALDAN ANALOĞA ÇEVİRİCİ

Sayısalan analoğa çevirici (Digital to Analog Converters) DAC devrelerini incelerken OPAMP'ın yapısının bilinmesi gerekir. OPAMP girişine gelen gerilimi çıkışından yükseltmiş olarak verir. Kazancı girişe ve geri beslemeye bağlanan dirençler yardımıyla değiştirilebilir. Şekil-13.1'de OPAMP'ın sembolü ve yükselteç olarak kullanılması gösterilmiştir. OPAMP yükseltme dışında karşılaştırma işlemi de yapar.

Yükselteç olarak kullanıldığında giriş ile çıkış arasında aşağıdaki eşitlik geçerlidir. Negatif işaret ise girişin değiştiğini belirtir.

$$\frac{V_{\text{çıkış}}}{V_{\text{giriş}}} = \frac{R_f}{R_{\text{giriş}}}$$

Karşılaştırıcı olarak kullanıldığında iki giriş arasındaki çok küçük fark girişlerden büyük olana göre çıkış gerilimi pozitif doyum gerilimine veya negatif doyum gerilimine kurar. Doyum gerilimi yaklaşık olarak besleme gerilimine eşittir.

İKİLİK AĞIRLIKLIL GİRİŞLİ SAD

Giriş dirençleri ikinin ağırlıklarına göre belirlenen bu tip SAD'nin devresi şekil-13.2'de gösterilmiştir. Bu yöntemde OPAMP toplayıcı ve yükselteç olarak

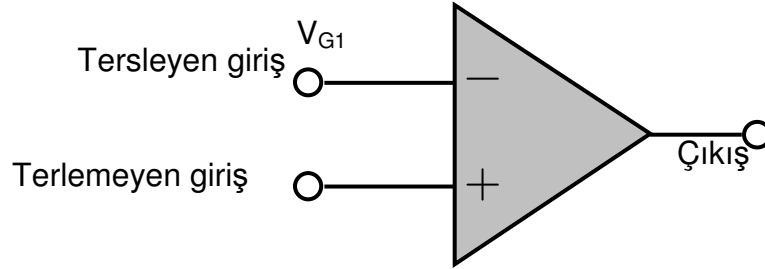
kullanılmıştır. Bu devrenin sayısal işaretin bit adedi kadar girişi olacaktır. Giriş direncinin değeri bu girişin temsil ettiği bitin ağırlığına bakılarak belirlenir. Ağırlığı yüksek olan girişe devrenin çıkış gerilimine etkisinin fazla olması için giriş direnci küçük seçilir. Ağırlık düştükte aynı oranda direnç değeri artar.

Çıkış geriliminin değeri toprak ile $V_{\text{çıkış}}$ uçlarından ölçülen gerilimdir. OPAMP'ın pozitif ve negatif girişlerinin birleşim noktası da toprak olduğuna göre çıkış geriliminin değeri aşağıdaki gibi yazılabilir.

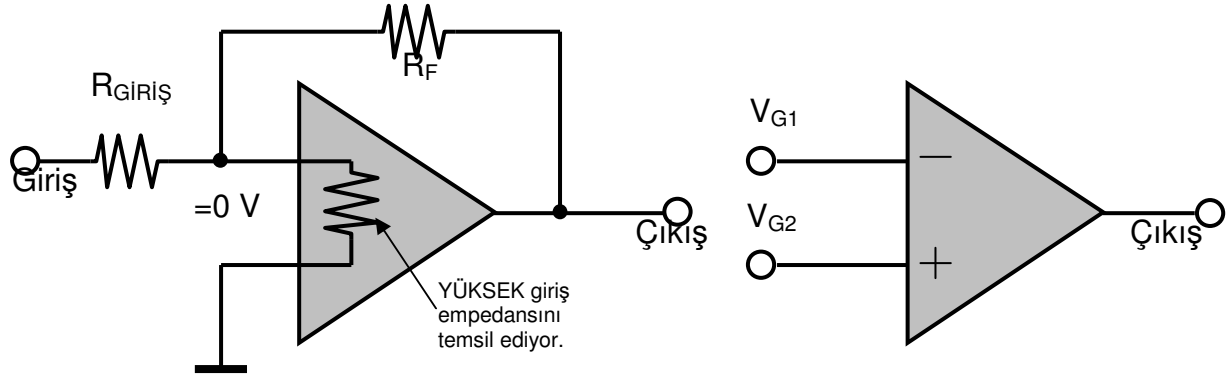
$$V = I_F R_F$$

I_F akımının değeri tüm giriş akımlarının toplamıdır. Giriş akımları ise girişe uygulana gerilimlerden bulunabilir.

$$I_F = I_0 + I_1 + I_2 + I_3$$



(a) OPAMP simgesi.



(b) Yükselteç.

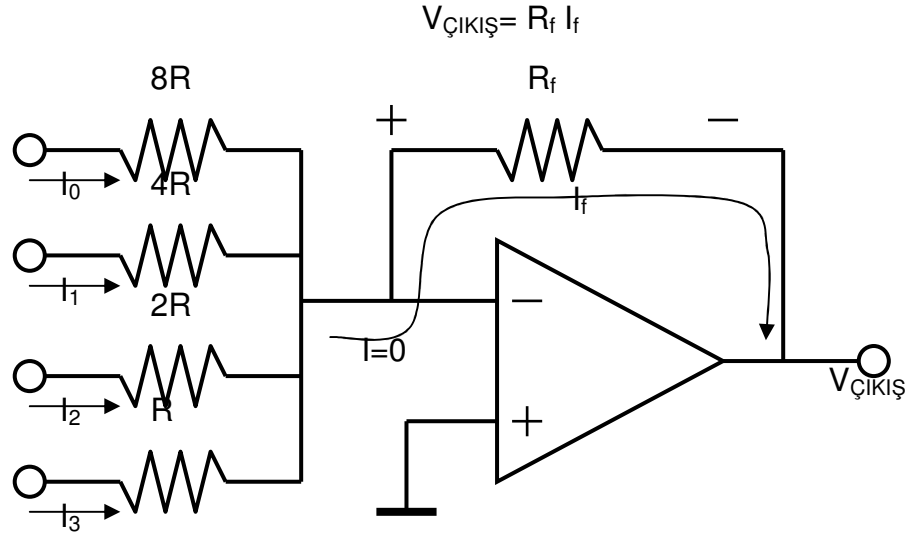
(c) Karşılaştırıcı.

Şekil-13.1 OPAMP'ın sembolü, yükselteç ve karşılaştırıcı olarak kullanılması.

Girişlere sayısal devrelerde ya 0 volt yada 5 volt uygulandığına göre tüm girişlere gelen gerime V diyebiliriz. Buna göre I_0, I_1, I_2, I_3 akımlarını şöyle hesaplayabiliriz.

$$I_0 = \frac{V}{8R}, \quad I_1 = \frac{V}{4R}, \quad I_2 = \frac{V}{2R}, \quad I_3 = \frac{V}{R}$$

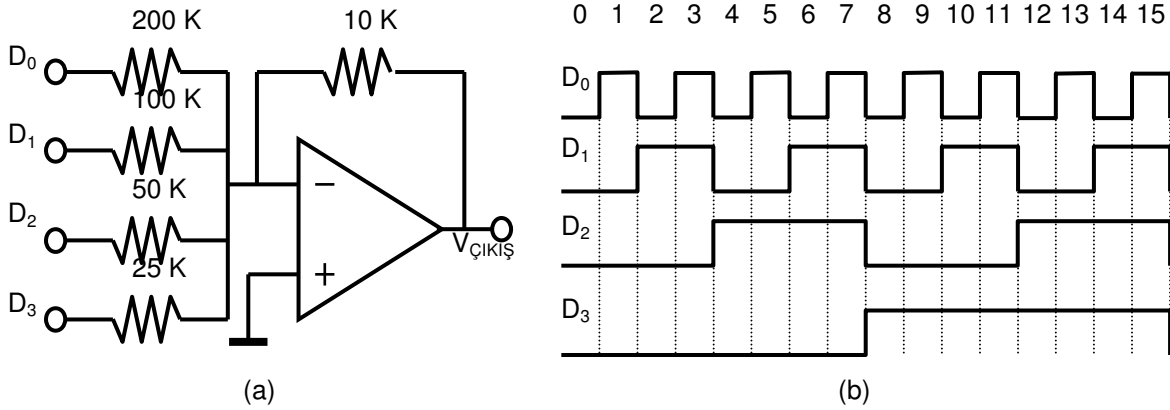
Düşük değerli bitin çıkış gerilimi üzerindeki değerinin çok küçük olduğuna dikkat ediniz.



Şekil-13.2 İkilik ağırlıklı girişli SAD.

ÖRNEK:

Şekil-13.3.(a)'deki devrenin girişine uygulanan sayısal işaret şekil-13.3.(b)'de verilmiştir. Verilen zaman aralığında çıkışı hesaplayarak dalga şeklini çizin.



Şekil-13.3

ÇÖZÜM:

Öncelikle girişlerden gelen akımlar belirlenir. Akımlar belirlenirken girişin 0 ve 1 olmalarına göre akım hesaplanır. Aşağıda sadece 1 için hesaplama yapılmıştır 0 olduğunda akımlar sıfır olacağı için hesaplama yapılmamıştır.

$$I_0 = \frac{5 \text{ V}}{200 \text{ K}} = 0,025 \text{ mA}$$

$$I_1 = \frac{5 \text{ V}}{100 \text{ K}} = 0,05 \text{ mA}$$

$$I_2 = \frac{5 \text{ V}}{50 \text{ K}} = 0,1 \text{ mA}$$

$$I_3 = \frac{5 \text{ V}}{25 \text{ K}} = 0,2 \text{ mA}$$

OPAMP giriş empedansının çok yüksek olması tersleyen girişe giden akımı yaklaşık olarak sıfır olmasına neden olacaktır. Sonuç olarak giriş akımlarının tamamı geri besleme direnci üzerinden geçecektir. Çıkış gerilimini değeri geri besleme direnci ile giriş akımlarının toplamına eşittir. Sayısal girilir 15 zaman aralığına bölüdüğü için her zaman aralığı için çıkış gerilimi tekrar hesaplanmalıdır. Bu çok fazla hesaplama gerektireceğinden her akımın diğerinden bağımsız çıkış gerilimine etkisi hesaplanacak eğer giriş sayısal değer mantık 1 ise çıkış gerilimi bulunurken toplanacak eğer mantık 0 ise toplanmayacaktır.

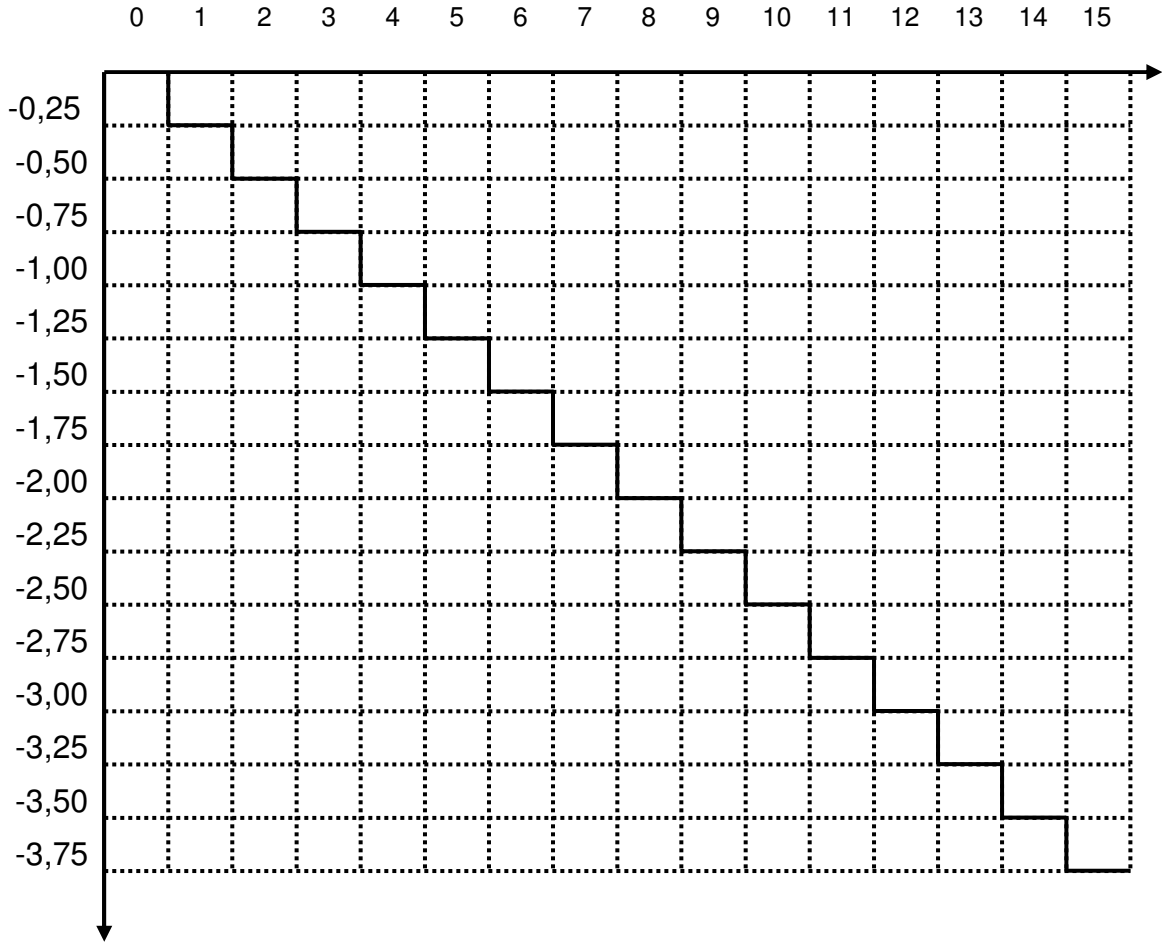
$$V_{\text{ÇIKIŞ}(D_0)} = (10 \text{ k}\Omega)(-0,025 \text{ mA}) = -0,25 \text{ V}$$

$$V_{\text{ÇIKIŞ}(D_1)} = (10 \text{ k}\Omega)(-0,05 \text{ mA}) = -0,5 \text{ V}$$

$$V_{\text{ÇIKIŞ}(D_2)} = (10 \text{ k}\Omega)(-0,1 \text{ mA}) = -1 \text{ V}$$

$$V_{\text{ÇIKIŞ}(D_3)} = (10 \text{ k}\Omega)(-0,2 \text{ mA}) = -2 \text{ V}$$

0 nolu zaman aralığından başlayarak çıkış gerilimlerini her zaman aralığı için hesaplayalım. Şekil-13.3 (b)'de 0 zaman aralığında sayısal girişlerin değerleri 0000 dır, çıkış gerilimi de 0V'tur. 1 nolu zaman aralığında sayısal girişler 0001 ve çıkış gerilimini sadece D₀ girişi belirler, çıkış gerilimi -0,25 Volttur. 2 nolu zaman aralığında sayısal giriş 0010 ve çıkış gerilimini sadece D₁ girişi belirler, çıkış gerilimi -0,5 Volttur. 3 nolu zaman aralığında sayısal giriş 0011 ve çıkış gerilimini D₁ ve D₀ girişleri birlikte belirler, çıkış gerilimi (-0,5) + (-0,25) = -0,75 Volttur. 4 nolu zaman aralığında sayısal giriş 0100 ve çıkış gerilimini D₂ girişi belirler, çıkış gerilimi -1 Volttur. 5 nolu zaman aralığında sayısal giriş 0101 ve çıkış gerilimini D₂ ve D₀ girişleri birlikte belirler, çıkış gerilimi (-0,25) + (-1) = -1,25 Volttur. 6 nolu zaman aralığında sayısal giriş 0110 ve çıkış gerilimini D₂ ve D₁ girişleri birlikte belirler, çıkış gerilimi (-0,5) + (-1) = -1,5 Volttur. 7 nolu zaman aralığında sayısal giriş 0111 ve çıkış gerilimini D₁, D₁ ve D₀ girişleri birlikte belirler, çıkış gerilimi (-1) + (-0,5) + (-0,25) = -1,75 Volttur. 8 nolu zaman aralığında sayısal giriş 1000 ve çıkış gerilimini D₃ girişi belirler, çıkış gerilimi -2 Volttur. Diğer adımlar izlenerek hesaplanıp grafiği çizildiğinde , şekil-13.4'teki eğri elde edilir.

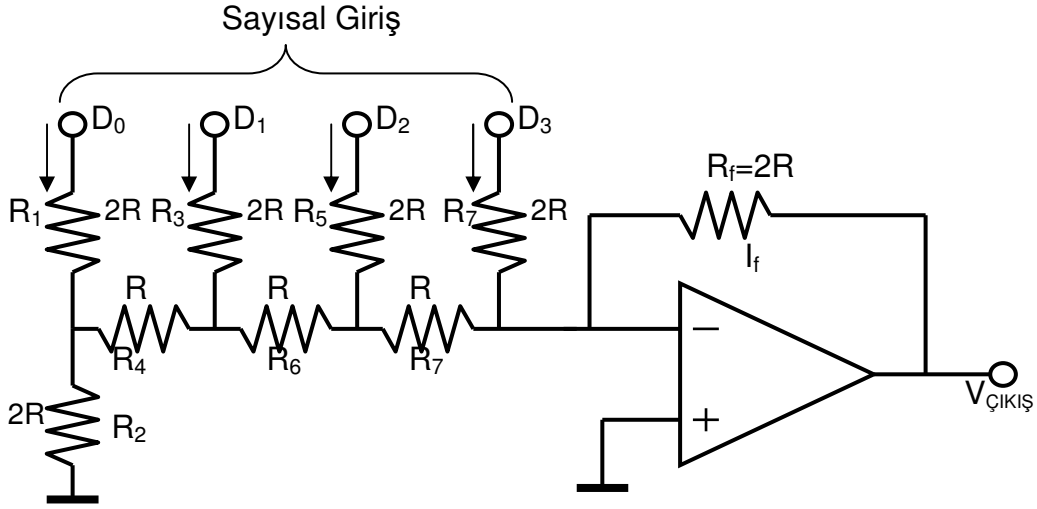


Şekil-13.4

R/2R MERDİVEN SAD

Sayısal analog çevirme işleminde kullanılan diğer bir yöntem de $R/2R$ merdiven yöntemidir. Bu sistemin ikilik ağırlıklı sayısalan analoğa yeğlenmesinin nedeni, yalnızca iki ayrı değerde direnç gerektirmesidir. Devrenin bağlantı şeması Şekil-13.5'te gösterilmiştir.

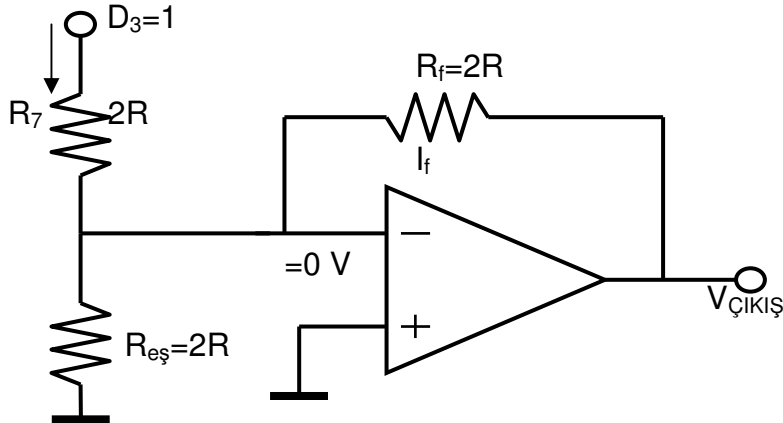
Devrenin çözümünde devre analizi dersinde öğrendiğiniz yöntemleri kullanacağız. Sayısal girişleri birer gerilim kaynağı gibi düşünebilirsiniz. Her adımda sadece birinin var olduğunu diğerlerinin sıfır olduğunu varsayacağız. Bu yöntem süperpozisyon yöntemi ile aynıdır.

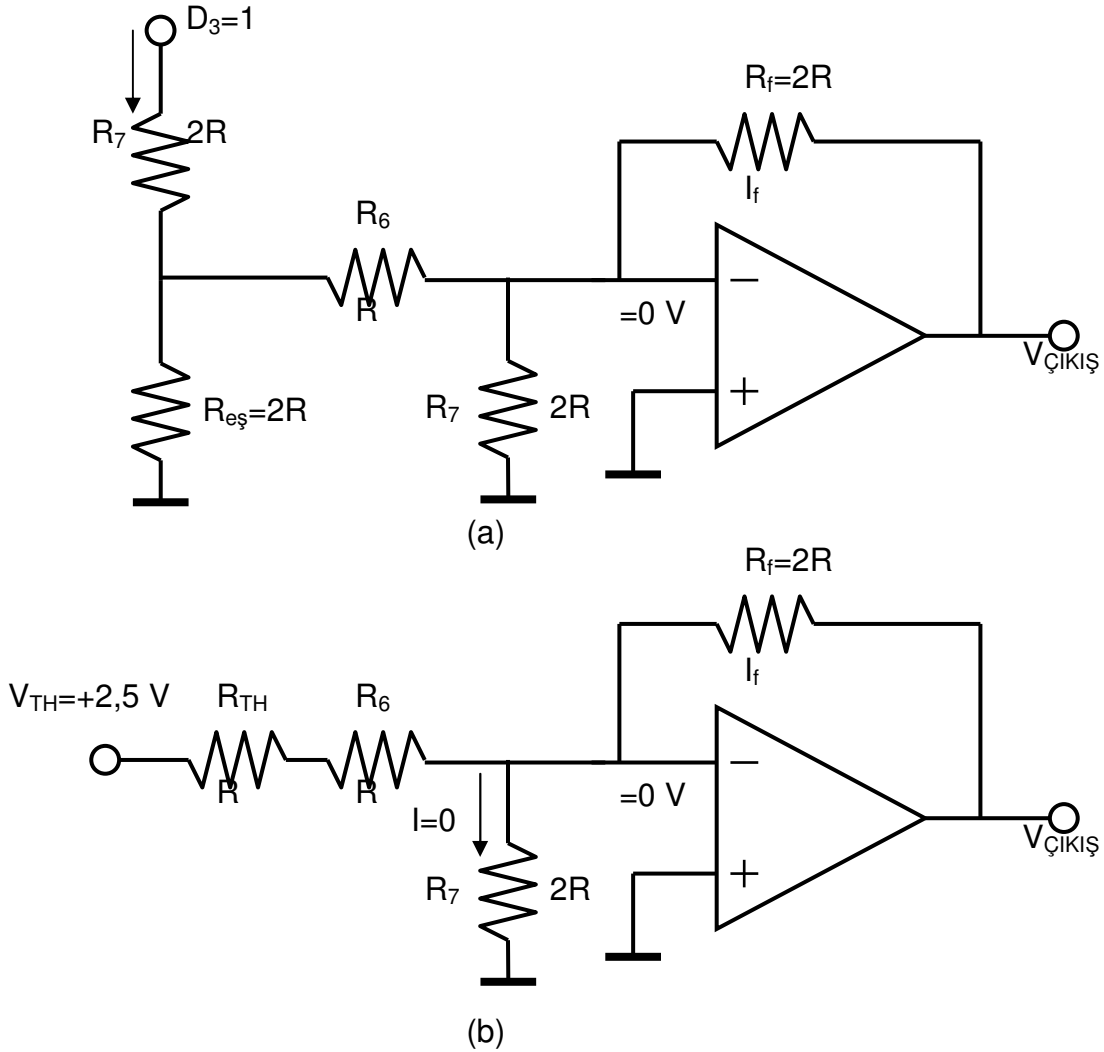


Şekil-13.5 R/2R merdiven SAD

D_3 bitinin YÜKSEK seviye (+5V) ve diğerlerinin DÜŞÜK seviye (toprak) olduğunu varsayarak çözüme başlayalım. Bu durum sayısal olarak 1000 durumu ile belirtilir. Devrede D.A. çözümleme yapılıncaya eşdeğer devre Şekil-13.6'daki gibi olur. Her iki ucu aynı potansiyelde olan direncin üzerinden akım geçmez ve devrenin çözümünde etkisi yoktur. $2R$ değerindeki eşdeğer dirençten hiç akım geçmez çünkü işlemel yükseltecin + girişi sanal topraktır. R_7 üzerinden gelen bütün akım, R_F direnci üzerinden geçer ve çıkış gerilimi aşağıdaki eşitlik kullanılarak -5 Volt elde edilir.

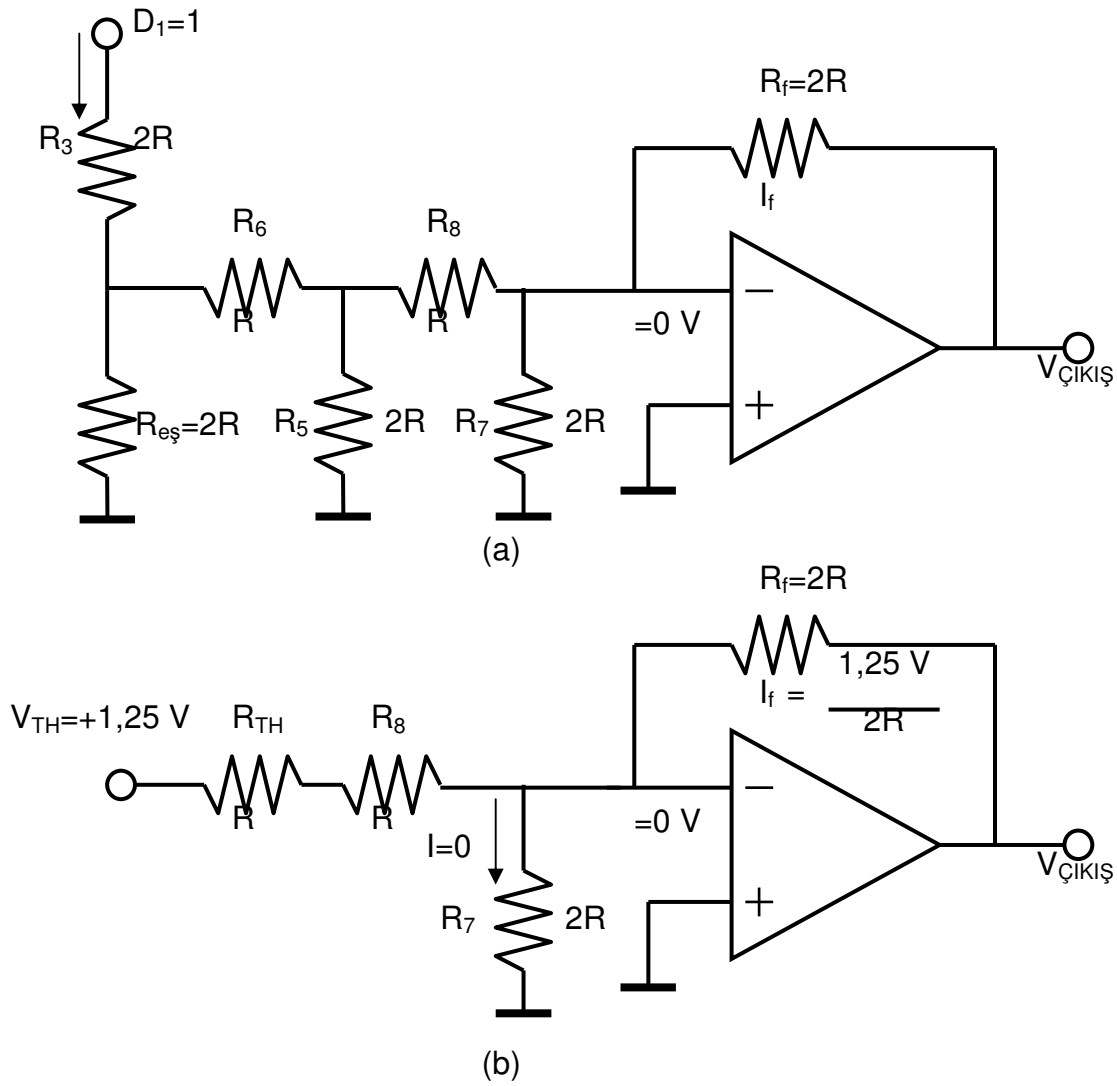
$$V_{\text{ÇIKIŞ}} = IR_F = \left(-\frac{5 \text{ V}}{2R} \right) 2R = -5 \text{ V}$$

Şekil-13.6 $D_3=1$ diğer girişler 0 olduğunda eşdeğer devre.



Şekil-13.7 $D_2=1$ diğer girişler 0 olduğunda eşdeğer devre.

Şekil-13.7'de D_2 girişinin YÜKSEK seviye ve diğer tüm girişlerin DÜŞÜK seviye olduğu 0100 ikilik verisi uygulanmışken eşdeğer devre çizilmiştir. R_8 den bakarak devrenin Thevenin eşdeğeri alınır, R değerindeki R_{TH} direncine seri, 2,5 voltluk bir V_{TH} kaynağı elde edilir. Bu kaynağın sağladığı akım $(2,5V/2R)$, R_7 uçlarındaki gerilim 0 V olduğu için tümüyle R_f üzerinden geçer ve çıkışta aşağıdaki eşitlikte belirtildiği gibi $-2,5$ voltluk bir gerilim oluşturur.



Şekil-13.8 $D_1=1$ diğer girişler 0 olduğunda eşdeğer devre.

$$V_{\text{ÇIKIŞ}} = IR_F = \left(-\frac{2,5 \text{ V}}{2R} \right) 2R = -2,5 \text{ V}$$

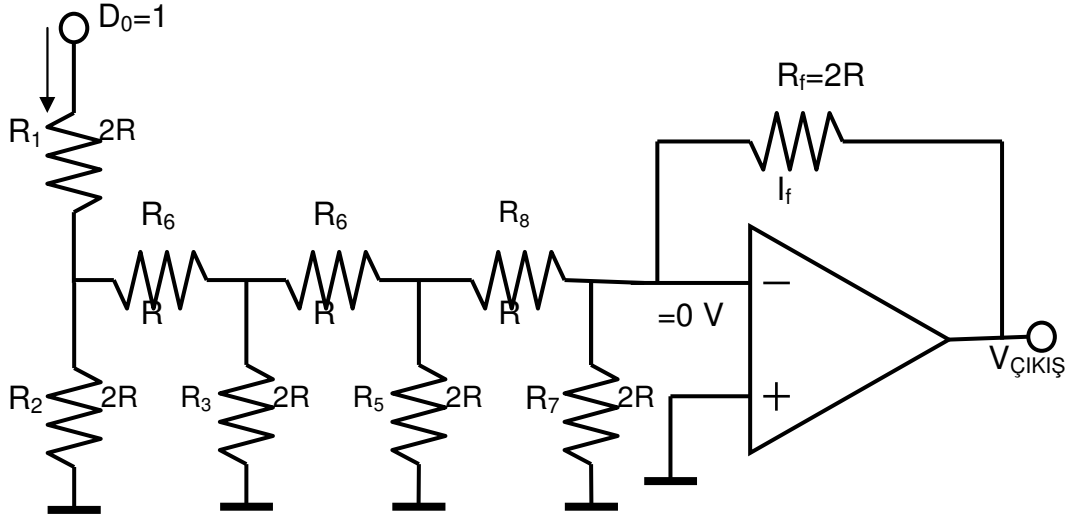
0010 sayısal verisi ile oluşan eşdeğer devre de Şekil-13.8'da verilmiştir. R_8 den bakarak devrenin Thevenin eşdeğeri alınır, R değerindeki R_{TH} direncine seri, 1,25 voltluk bir V_{TH} kaynağı elde edilir. Bu kaynağın ürettiği akım, çıkışta aşağıdaki eşitlikte belirttiği gibi $-1,25$ voltluk bir gerilim oluşturur.

$$V_{\text{ÇIKIŞ}} = IR_F = \left(-\frac{1,25 \text{ V}}{2R} \right) 2R = -1,25 \text{ V}$$

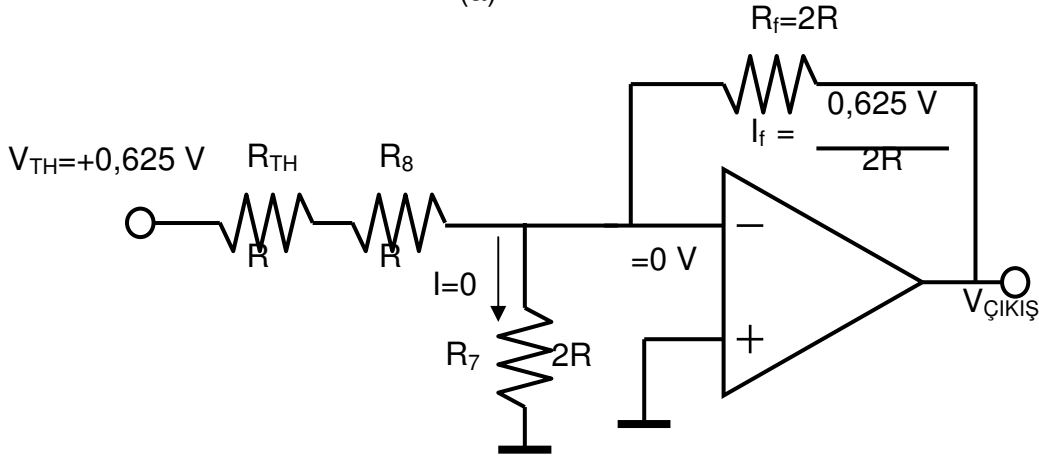
0001 sayısal verisi ile oluşan eşdeğer devre de Şekil-13.9'da verilmiştir. R_8 den

bakarak devrenin Thevenin eşdeğeri alınır, R değerindeki R_{TH} direncine seri, $0,625$ voltluk bir V_{TH} kaynağı elde edilir. Bu kaynağın ürettiği akım, çıkışta aşağıdaki eşitlikte belirtildiği gibi $-0,625$ voltluk bir gerilim oluşturur.

$$V_{\text{ÇIKIŞ}} = IR_F = \left(-\frac{0,625 \text{ V}}{2R} \right) 2R = -0,625 \text{ V}$$



(a)



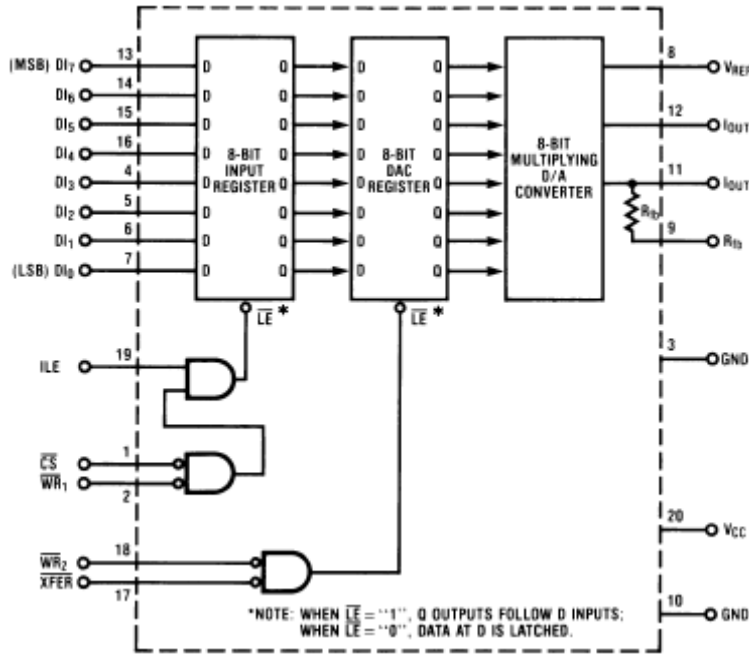
(b)

Şekil-13.9 $D_0=1$ diğer girişler 0 olduğunda eşdeğer devre.

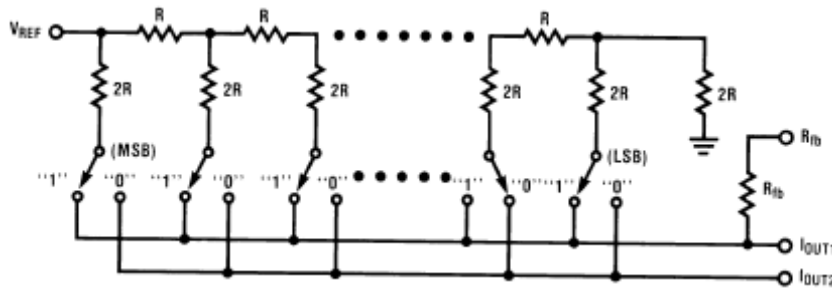
ÖRNEK DAC

Örnek olarak verilen DAC0832 DAC'ı, sekiz ayrı ikilik biti giriş olarak kullanmaktadır. Her bit, değişik değerlerde bir akım değeri üretir ve bu değerler de kendini oluşturan bitin ikilik ağırlığı ile doğru orantılıdır. 8-Bit ikilik bir sistemde ilk bit 1, ikincisi 2, üçüncüsü 4, ve diğerleri de sırayla 8, 16, 32, 64, 128 ağırlıklarındadır. Bu çeviricide akımı denetleyen sistem, R-2R merdivenidir. Bu devrenin genel yapısı Şekil-13.10'da verilmiştir. Devrede yalnızca R ve $2R$ değerlerinde olmak üzere iki ayrı değerde direnç kullanıldığından bu ad verilmiştir.

Çeviricinin en önemli bölümü merdivendir. Bu sistemin çalışması, her direnç birleşme noktasında (A, B ve C), akımın eşit olarak bölünmesine dayanır. Akım bölme işlemi Şekil-13.11'de ayrıntılı olarak verilmiştir. A noktasından akım, eşit olarak (I_1 ve I_2) ikiye ayrılır. Devre akımı her birleşme noktasında eşit olarak ikiye bölecek biçimde tasarlandığından, I_2 akımı da, B noktasında I_3 ve I_4 olarak ikiye bölünür. R_4, R_5, R_6 seri-paralel devresinin direnci R_3 direncine eşit olduğu için, I_3 ve I_4 akımları eşit ve I_2 nin yarı değerindedir.



(a) DAC0830'un işlevsel blok şeması

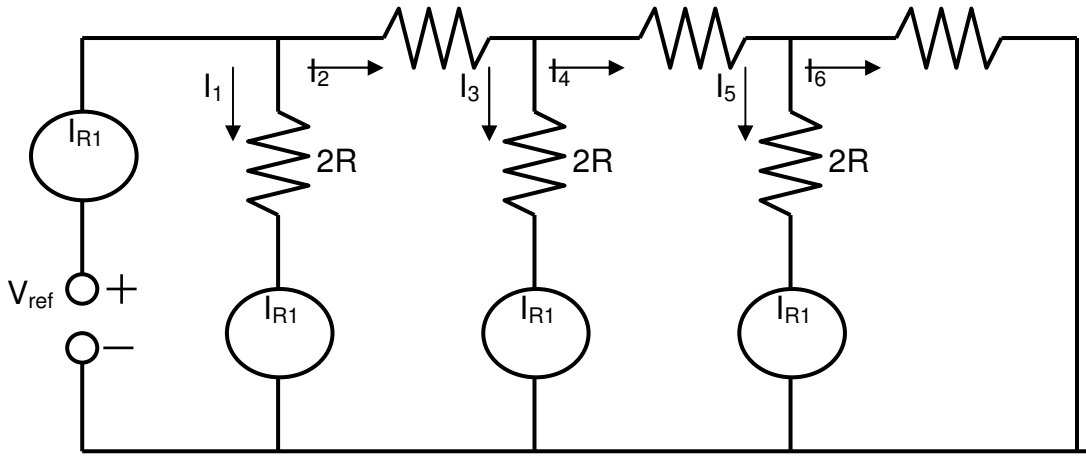


(b) DAC0830'un iç yapısı

Şekil - 13.10

Önceki akımı iki eşit akıma bölme işlemi en son birleşme noktasına dek sürer. Akım sürekli ikiye bölündüğü için, ilk birleşimde giriş akımının yarısı, ikinci birleşimde giriş akımının dörtte biri, üçüncü birleşimde giriş akımının sekizde biri, olarak bölme işlemi yapılarak ikilik basamak değerleri (1, 2, 4, 8, 16, 32, 64, 128) elde edilir.

İkilik ağırlıklar ile bölünen akım değerleri R_1, R_3, R_5, \dots ile ve bir elektronik anahtar üzerinden toplama devresine iletilir. Sözü geçen elektronik anahtar, Şekil-13.11'de görülmektedir. İkilik giriş değerine bağlı olarak davranan bu anahtarlar yardımıyla her bitin akımı ya toprağa yada toplama devresine uygulanarak, girişteki sayısal koda bağlı toplam akım elde edilir. Toplama işlemi için kullanılan işlemel yükseltecin girişi sanal toprak olduğundan, anahtarlama işlemi merdivenin toplam direncini ve akım bölme değerlerini etkilemez. OP-AMP girişine gelen akımlar toplanarak, sayısal kodun değeri ile doğru orantılı bir çıkış gerilimi oluşturulur.



Şekil-13.11 Akım bölücü devre

Deneyde kullanılan DAC0830 TD, 20 bacaklı DIL kılıfta standart bir ticari aygıttır ve özellikleri şöylece açıklanabilir:

DAC0830 gelişmiş CMOS/Si-Cr 8 bitlik çoğullanabilir (multiplying) bir DAC entegresidir. R-2R direnç merdiven devresinde referans akımını böler. Devre, CMOS akım anahtarları ile düşük güç tüketimi ve düşük çıkış sızıntı akımını sağlayan kontrol devresini kullanır. TTL mantık girişi gerilim seviyeleri ile uyumlu özel bir devreye sahiptir.

Özellikler:

- ✓ $\pm 10V$ referans ile çalışma
- ✓ Mikroişlemcisiz (tek başına) çalışma
- ✓ Akım durulma süresi: $1\mu s$
- ✓ Çözünürlük: 8 bit
- ✓ Doğrusallık: 8, 9 veya 10 bit
- ✓ Kazanç sıcaklık katsayısı: $\%0.0002 \text{ FS}/^\circ\text{C}$
- ✓ Düşük güç tüketimi: 20mW
- ✓ Tek güç kaynağı ile besleme: $+5V \dots +15V_{DC}$
- ✓ Doğrusallıktan maksimum sapma ($-10V \leq V_{REF} \leq +10V$ için): $\%0.05 \dots \%0.2$; $\%FS$
- ✓ Monotonluk: 8 bit
- ✓ Maksimum kazanç hatası ($-10V \leq V_{REF} \leq +10V$ için): $\pm\%0.2$ (tipik), $\pm\%1$ (maks.)

Bacakların Tanımları:Kontrol sinyalleri:

\overline{CS} : Chip Select (etkin DÜŞÜK). ILE ile birlikte \overline{CS} , \overline{WR}_1 'i izinler.

ILE: Input Latch Enable (etkin YÜKSEK). \overline{CS} ile birlikte ILE, \overline{WR}_1 'i izinler.

\overline{WR}_1 : Write 1. Etkin DÜŞÜK \overline{WR}_1 dijital giriş veri bitlerini (DI) giriş tutucusuna yüklemek için kullanılır. \overline{WR}_1 YÜKSEK olduğunda giriş tutucusundaki veri tutulur. Giriş tutucusunu güncellemek için, ILE YÜKSEK iken, \overline{CS} ve \overline{WR}_1 DÜŞÜK olmalıdır.

\overline{WR}_2 : Write 2 (etkin DÜŞÜK). Bu sinyal \overline{XFER} ile birlikte giriş tutucusunda bulunan 8 bitlik veriyi DAC yazacına iletir.

\overline{XFER} : Transfer Control Signal (etkin DÜŞÜK). \overline{XFER} , \overline{WR}_2 'yi izinler.

Diğer Bacakların İşlevleri:

DI₀...DI₇: Dijital girişler. DI₀ en az değerlikli bit (LSB) ve DI₇ en değerlikli bittir (MSB).

I_{out1}: DAC akım çıkışı 1. I_{out1} DAC yazacındaki dijital kodun tümü "1"lerden oluştuğunda maksimum değerini alır ve DAC yazacındaki tüm "0"lar için sıfırdır.

I_{out2}: DAC akım çıkışı 2. I_{out2} bir sabit eksi I_{out1} 'dir, veya I_{out1} + I_{out2} = sbt.

R_{fb}: Feedback resistor. Tümleşik devre üzerinde bulunan ve DAC gerilim çıkışı sağlamak için bağlanacak opamp için paralel geribesleme direnci. Yonga üzerindeki bu direnç R-2R merdiven devresindeki dirençler ile özdeş olduğu için kullanılmalıdır.

V_{REF}: Reference Voltage Input. Bu giriş dahili R-2R merdiven devresini hassas harici bir gerilim kaynağına bağlar. V_{REF}, -10V ile +10V aralığında seçilebilir.

V_{CC}: Dijital kaynak gerilimi. Entegrenin güç kaynağı bacağıdır. V_{CC} +5V ile +15V_{DC} aralığında olabilir. Optimum çalışma +15V'ta gerçekleşir.

GND: 10 nolu bacak akım anahralaması uygulamaları için I_{out1} ve I_{out2} ile aynı toprak potansiyelinde olmalıdır. Fark olması durumunda doğrusallıktan sapma olur (bkz. DAC0830 veri yaprağı, s.7).

DAC çeviricinin çıkış gerilimi, iki koşula bağlıdır: 8-bit giriş sayısının büyüklüğü ve referans geriliminin değeri. Çıkış geriliminin en yüksek değeri ise kaynak gerilimidir. TD 5V ile en fazla 17V arasındaki kaynak gerilimleri ile çalışabildiği ve çıkış gerilimi kaynak gerilimini aşamayacağı için, örneğin 5V kaynak gerilimi kullanıyorsanız 10V'luk referans gerilimi ile 0-10V aralığında çıkış elde edemezsiniz.

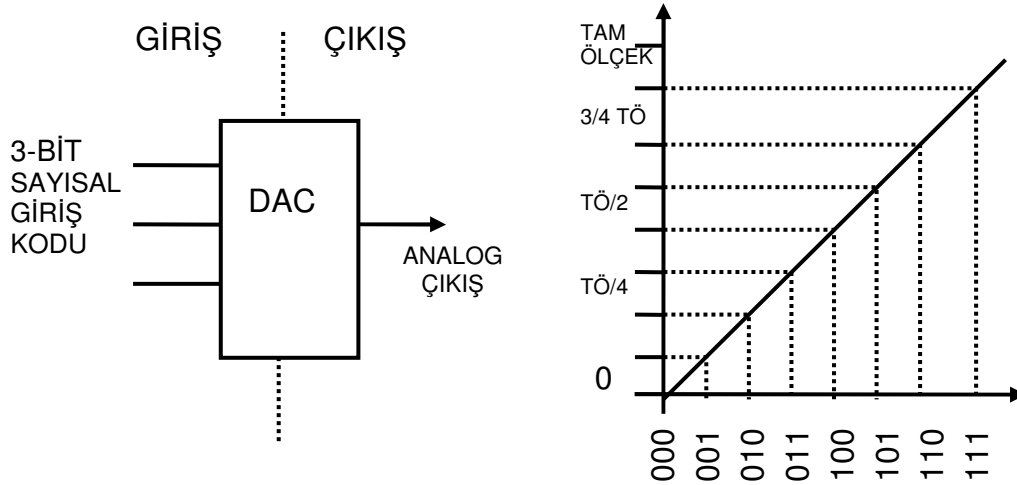
DAC, sayısal giriş kodunu alarak, analog bir gerilim yada akım olarak çıktılar. Şekil-13.12'de 3-bitlik bir çeviricinin sayısal koda karşı tam ölçeğe göre çıkış gerilimi verilmiştir. Tam ölçeğin 4V olduğunu varsayarsak, her kod 0,5 V ($4/2^3 = 0,5$) değerinde bir aralıkta gösterilir. Dikkat ederseniz 4V'luk tam ölçek çıkış değerine tam olarak ulaşamamaktadır çünkü, 000 koduna sıfır volt değeri atanmıştır ve geri kalan 7 kod içinde en yüksek olanı 111 ile elde edilebilecek gerilim $7 \times 0,5 = 3,5$ V olur. Duyarlılığı arttırmak ve tam ölçek çıkış gerilimine iyice yaklaşmak olasıdır. Örneğin 6-bit giriş kodu kullanılırsa, 2^6 değişik kod oluşacak ve bu kodlardan her biri çıkış geriliminde 0,0625V'luk ($4/2^6 = 0,0625$) bir değişim oluşturacaktır. Görüldüğü gibi en yüksek analog çıkış gerilimi, tam ölçek değerine daha da yaklaşmıştır. Çıkış geriliminin en düşük değeri (resolution - ayırma), nicemleme boyutu (quantization size - Q) olarak adlandırılır ve tam ölçeğin 2^n değerine ($n = \text{bit sayısı}$) bölünmesiyle elde edilir.

Burada kullanılan DAC için nicemleme boyutu Q,

$$Q = \frac{2,56}{2^8} = \frac{2,56}{256} = 0,01\text{volt} = 10\text{mV}$$

olarak bulunur.

Buna göre giriş verisinin her biti, analog çıkış geriliminde, bitin ikilik ağırlığı çarpı on milivoltluk bir değişime yol açar. 00000000 verisi sıfır volt olarak alınacağına göre, sistemin en yüksek çıkışı $255 \times 10 \text{ mV} = 2,55\text{V}$ olur.



Şekil-13.12 DAC simgesi ve aktarım eğrisi.

SAD BAŞARIM ÖZELLİKLERİ

Çözünürlük (resolution)

Bir DAC'nin ayırma değeri, çıkışındaki basamak sayısının tersidir ve giriş bitlerinin sayısına bağlıdır. Örneğin 4-bitlik bir S/Ö nin ayırması, $2^4 - 1$ de birdir (onbeşte bir). Yüzde olarak belirtilecekse, $(1/15) \times 100 = 6,67\%$ olarak yazılmalıdır.

Nicemleme boyutu (quantitization size)

Girişteki ardışık iki veri arasında, çıkıştaki gerilim değişimine nicemleme boyutu denir ve tam ölçek çıkış geriliminin toplam basamak sayısına bölümü ile gösterilir. Örneğin çıkış gerilim değeri 10V olan 8-bitlik bir çeviricinin nicemleme boyutu, $10/2^8 = 0,039$ V olarak bulunur.

Doğruluk (accuracy)

Doğruluk, gerçek DAC çıkışı ile umulan çıkışın karşılaştırılmasıdır. Tam ölçek yada en yüksek çıkış geriliminin yüzdesi olarak gösterilir. Örneğin bir çeviricinin tam ölçek çıkışı 10 V ve doğruluğu da $\pm 10\%$ ise, herhangi bir çıkış gerilimindeki en büyük hata, $(10) \times (0,001) = 10$ mV olacaktır. Doğruluğun en azından $\pm 1/2$ LSB (en az önemli bit) değerinde olması istenir. 8-Bit bir çeviricide LSB $1/256 = 0,0039$ (tam ölçeğin 0,39%) değerindedir. Buna göre doğruluk yaklaşık $\pm 0,2\%$ dolayında olmalıdır.

Doğrusallık (linearity)

Doğrusal bir hata, DAC'nin beklenen düz-doğru çıkışından sapmadır. Bu sapmanın özel bir durumu, bütün giriş bitleri sıfır iken çıkışta görülen gerilimdir ve kayıklık hatası (offset error) olarak adlandırılır.

Tekdüzelik (monotonicity)

Giriş verisi sırayla artarken, DAC çıkışının geri adım atmamasıdır.

Durulma süresi (settling time)

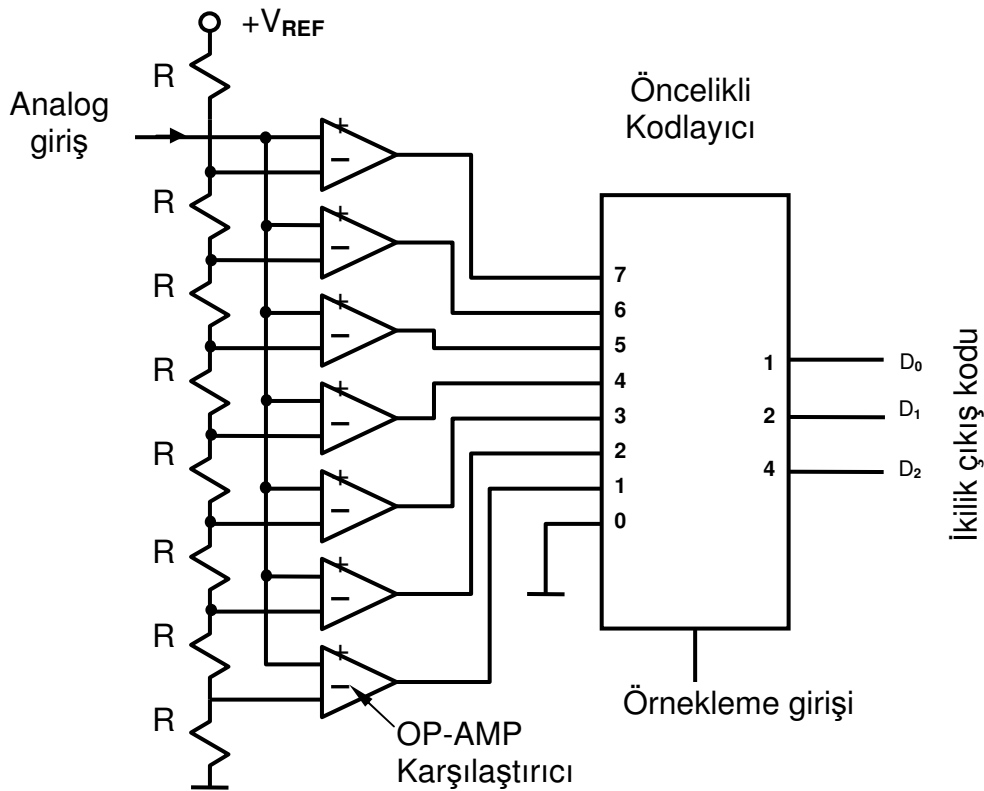
Giriş kodunda bir değişiklik oluştuktan sonra DAC çıkışının $\pm 1/2$ LSB sapma ile oturması arasında geçen süreye denir.

ANALOGDAN-SAYISALA ÇEVİRİCİ (ADC)

Analog-sayısal çevirici (ADC yada A/S), ölçülen yada elde edilen analog büyüklüklerin ikilik kodlar biçiminde sayısal sistemlere aktarılarak işlenmesi ve saklanması gerektiğinde kullanılır. Yaygın olarak kullanılan ADC yöntemleri altı tanedir.

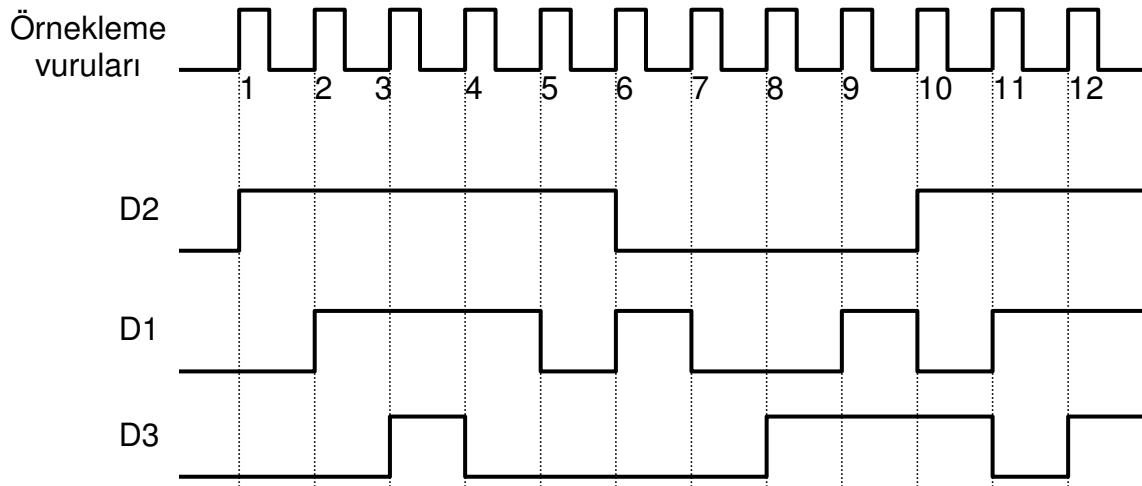
ANINDA ANALOG SAYISAL ÇEVİRİCİ

Anında analog sayısal çevirici (Flash (simultaneous) A/D Converter) Bu yöntemde analog giriş işareti, referans gerilimleriyle karşılaştırılır. Girişe ulaşan analog gerilim karşılaştırıcılardan birinin referans gerilimini aştığında, karşılaştırıcı çıkışında bir YÜKSEK seviye oluşur ve öncelikli kodlayıcı yardımıyla işaretin sayısal kodu üretilir. Öncelikli kodlayıcı girişine birden fazla yüksek seviye gelebilir fakat öncelik en büyüğe verildiği için bu girişin sayısal kodu çıkıştan elde edilir. Şekil-13.13'de görülen 3-bitlik ADC devresinde yedi karşılaştırıcı bulunmaktadır çünkü, 000 durumu için karşılaştırıcı kullanılmamıştır. Benzer biçimde 4-bitlik bir çeviricide de $2^4-1=15$ karşılaştırıcı kullanılır.

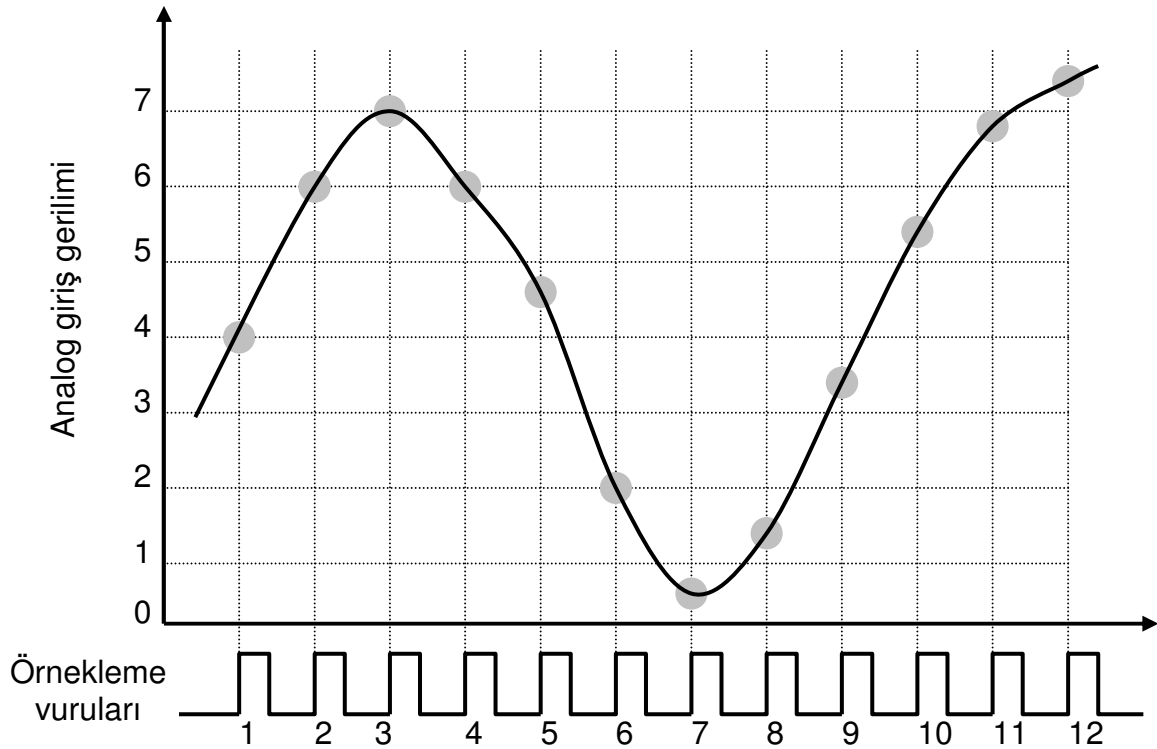


Şekil-13.13 3-bit anında (flash) ADC

Bu tür çeviricinin en önemli üstünlüğü, çevirme hızının çok yüksek olmasıdır. Bununla birlikte, hassas çevirme işlemleri için çok sayıda karşılaştırıcı kullanmak gerekeceği için kullanışlı değildir. Örneğin 8-bitlik bir ADC devresinde anında çevirme yöntemi kullanılacaksa, $2^8-1=255$ tane karşılaştırıcı gereklidir ki, bu da devreyi çok masraflı ve karmaşık duruma getirir. ADC girişindeki analog işareti belirten sayısal kodların doğruluğu, örnekleme hızına bağlıdır. Birim zamanda alınan örnek sayısı arttıkça, analog işaretin sayısal gösterimi de giderek gerçeğini en iyi temsil eder duruma gelir.



Şekil-13.15 Analog işaretin sayısala çevirmek için örneklenmesi.

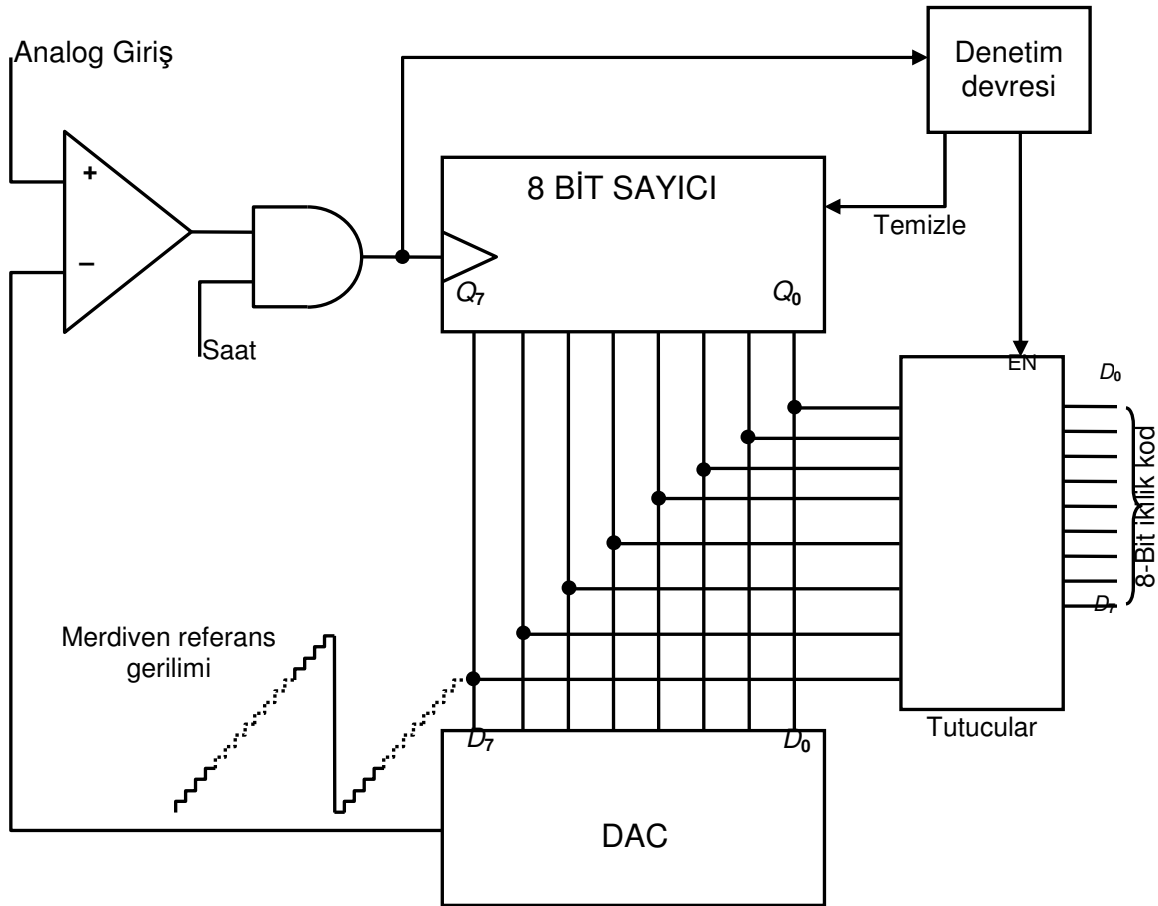


Şekil-13.14 Analog işaretin sayısala çevirmek için örneklenmesi.

SAYISAL-YOKUŞ ADC (DİGİTAL-RAMP A/D)

Bu çevirme yöntemine, *sayıcı yöntemi* adı da verilir. Devrede, analog işarete karşılık gelen sayısal kodun üretiminde bir ikilik sayaç ve DAC, birlikte kullanılır. Başlangıçta sayacın ve dolayısıyla DAC ün de sıfır ürettiğini varsayalım. Bu durumda girişe analog bir işaret uygulanırsa, uygulanan gerilim referans gerilimini (DAC çıkışı) aşar aşmaz karşılaştırıcı çıkışı YÜKSEK duruma geçerek sayacı ikilik

olarak saydırır. İkilik sayılar ilerledikçe DAC çıkışındaki referans gerilimi, basamak basamak artar. Referans geriliminin analog giriş işaretini geçmesiyle, karşılaştırıcı çıkışında üretilen DÜŞÜK, sayacı durdurur. Sayacın bu andaki içeriği, referans gerilimini analog giriş işaretinden daha büyük yapan en az basamak sayısına eşittir ve doğal olarak analog giriş değerini temsil eder. Denetim devresi bu sayıyı tutuculara yükler ve sayacı sıfırlayarak, girişi bir kez daha örnekleyecek yeni bir sayma sürecini başlatır. Bu yöntem flash yönteminden daha yavaştır çünkü, giriş işaretinin en yüksek olduğu durumda çevirme işlemi yapılmadan önce sayacın sıfırdan başlayarak bütün sayıları sayması gerekir. Bu, 8-Bit kod üreten bir sistemde 256 sayaç durumu demektir. Şekil-13.13 te 4-bitlik bir dönüşüm gösterilmiştir. Çizimden de görüleceği gibi, her örnek için sayaç sıfırdan başlayarak, referans gerilimi analog girişten büyük olana dek saymakta ve çevirme süresi de uygulanan gerilimin değerine göre değişmektedir.

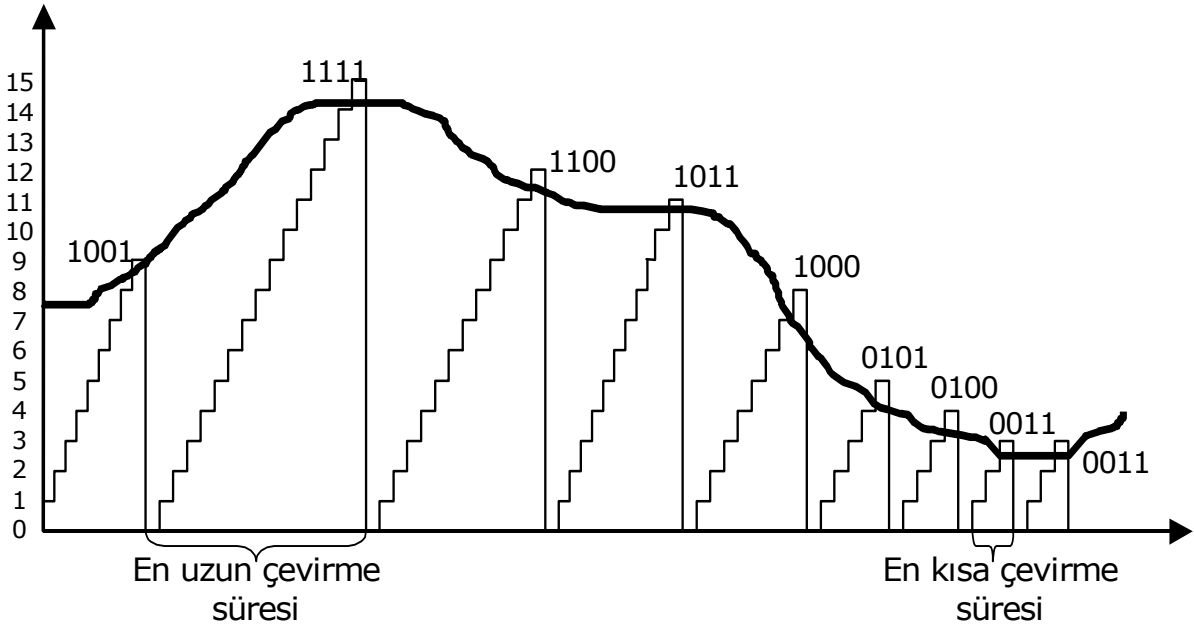


Şekil-13.16 8-bitlik sayısal-yokuş ADC

İZLEYİCİ ANALOG-SAYISAL ÇEVİRİCİ

İzleme (tracking) yönteminde de yokuş yöntemine benzer olarak bir ileri-geri sayaç ve bir DAC kullanılmıştır. İzleme yönteminde sayaç, alınan her örnekten sonra sıfırlanmadığı, bunun yerine giriş işaretini izleme eğiliminde olduğu için çevirme süresi yokuş yöntemine göre daha kısadır. Şekil-13.18'de izleyici ADC'nin blok şeması gösterilmiştir.

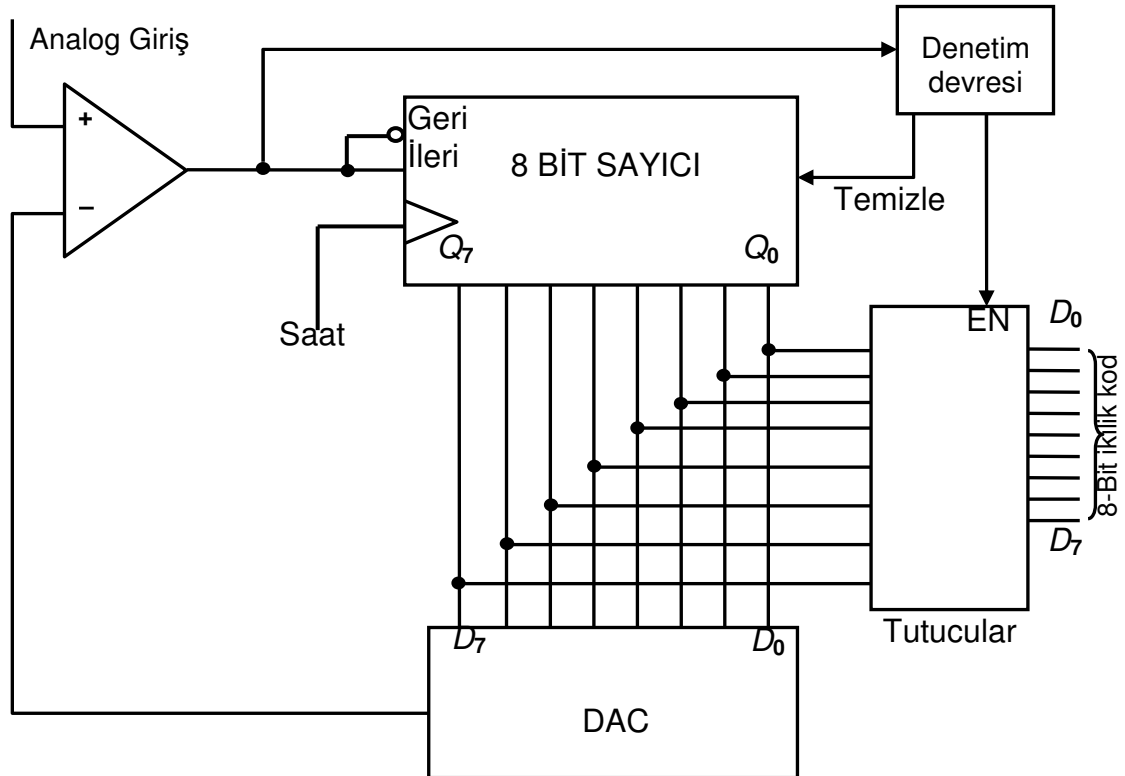
DAC çıkışındaki referans gerilimi analog girişten daha az olduğunda karşılaştırıcı çıkışında YÜKSEK vardır ve sayaç ileri yönde saydırılır. Sayacın ilerlemesi, referans gerilimi analog işarete yetişene dek sürer ve bu anda karşılaştırıcı çıkışı DÜŞÜK olarak sayacı geri saydırmaya başlar. Analog giriş gerilimi azalıyor, sayaç ta onun değerini izleyerek geri sayar. Giriş geriliminin değeri artıyorsa sayaç bu kez ileri sayar. Analog giriş değişmiyorsa, sayaç bir ileri bir geri sayarak yine bu değeri izler. İki ikilik değer arasında oluşan bu salınım, yöntemin en belirgin sakıncasıdır. İzleme yöntemiyle çevirme işlemi ile üretilen dalga biçimi, Şekil-13.19'da 4-bitlik bir çevirici için çizilmiştir. Bazı bölgelerde analog işaret ile sayısal işaretin oldukça farklı olduğuna dikkat ediniz.



Şekil-13.17 Yokuş yöntemi kullanılan 4-bit ADC de çevirme sırasında üretilen basamak dalgaları.



Şekil-13.19 İzleyici ADC'de izleme işlemi ile üretilen dalga biçimi.

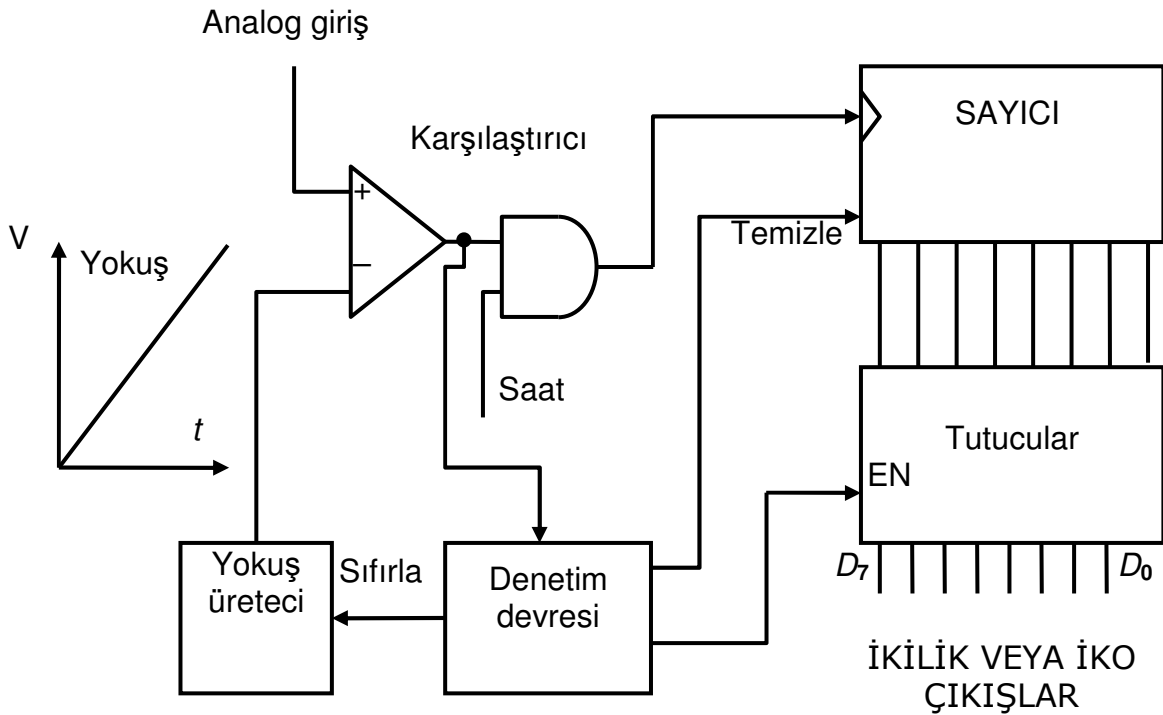


Şekil-13.18 8-bitlik izleyici ADC

TEK-EĞİMLİ ANALOG-SAYISAL ÇEVİRİCİ

Tek eğimli (single-slope) çevirme devresinde, yokuş ve izleyici yöntemlerinde olduğu gibi DAC kullanılması gerekmez. Bu devre gereksindiği sabit eğimli referans gerilimini, doğrusal bir yokuş üretici ile sağlar. Yöntemin devresi Şekil-13.20'de verilmiştir. Çevirme döngüsünün başlangıcında, sayaç sıfırlanmış durumda ve yokuş üretici çıkışı da 0 V değerindedir. Bu noktada analog giriş gerilimi referans geriliminden yüksektir ve karşılaştırıcı çıkışında YÜKSEK seviye vardır. Karşılaştırıcı çıkışındaki YÜKSEK seviye, sayacı saatler ve yokuş üreticini başlatır.

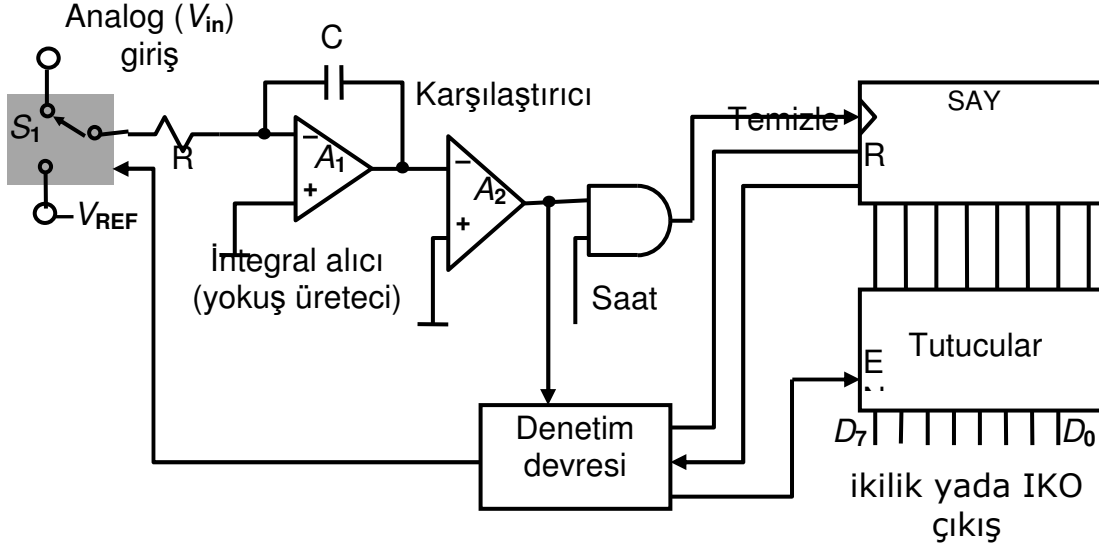
Yokuş, analog girişe eşit olana dek yükselecek ve eşitlik sağlanınca sıfırlanarak bu andaki ikilik yada İKO sayı, denetim devresi tarafından tutuculara yüklenecektir. Yokuş eğiminin 1V/ms ve analog giriş geriliminin de karşılaştırma noktasında 2V olduğunu varsayalım. Karşılaştırma anında yokuş gerilimi de 2V olacak ve üreteç de 2ms'dir çalışıyor olacaktır. Karşılaştırıcı çıkışı 2ms'dir YÜKSEK olduğuna göre, sayaca da 200 saat vurumu ($f_{\text{saat}}=100\text{kHz}$ varsayılarak) gönderilmiş olacaktır. Tam karşılaştırma noktasında, sayaç çıkışı onluk 200 değerindedir. Uygun bir ölçekleme ve kod çözme ile bu ikilik sayı, 2.00V olarak gösterilebilir. Bazı sayısal voltmetrelerde bu temel yöntem kullanılmaktadır.



Şekil-13.21 Tek eğimli ADC

ÇİFT EĞİMLİ ANALOG SAYISAL ÇEVİRİCİ

Bu yöntemin çalışma ilkesi, biri değişken biri sabit eğimli olmak üzere iki ayrı yokuş kullanılması dışında bir önceki devre ile benzerdir. Bu yöntem sayısal voltmetrelerde ve diğer ölçme aygıtlarında yaygın olarak kullanılmaktadır.

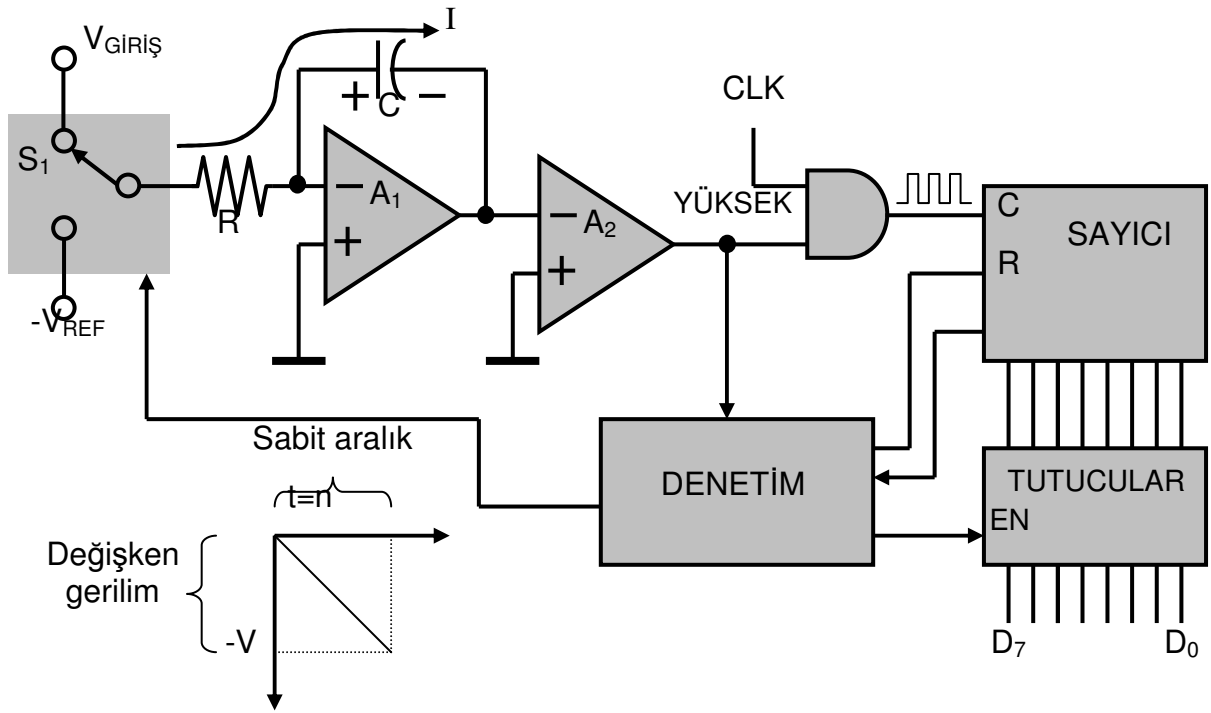


Şekil-13.22 Çift-eğimli ADC.

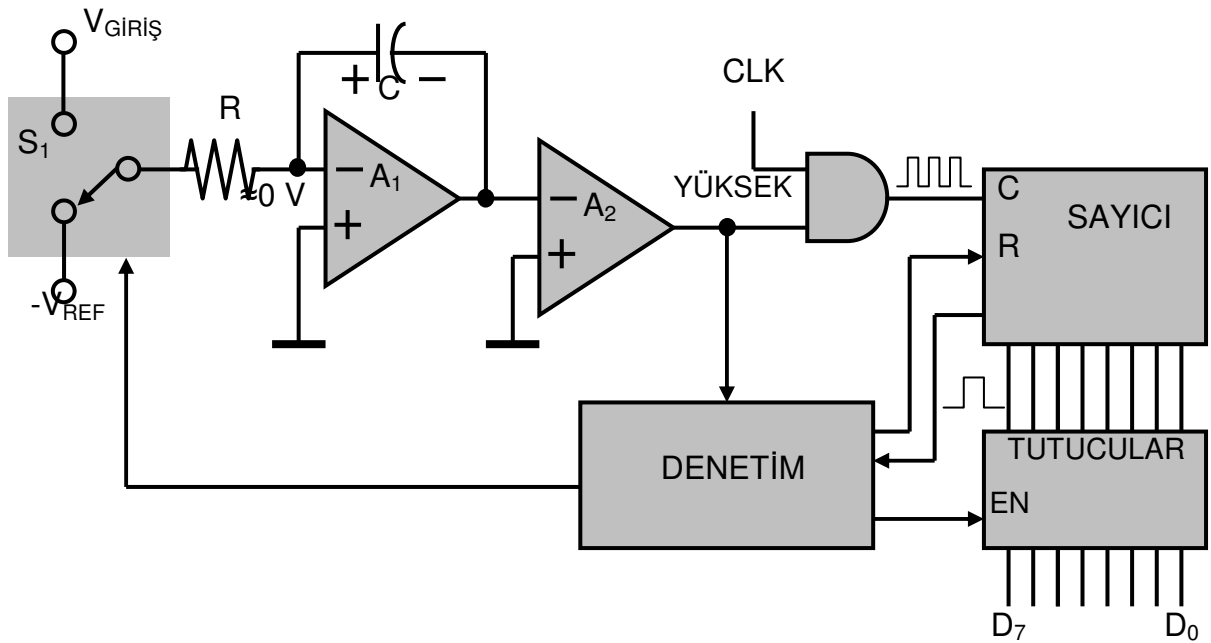
Çift-eğimlilik özelliğini oluşturmak için bir yokuş üretici-integral alıcı (ramp generator-integrator) kullanılır. Şekil-13.22'de çift-eğimli bir analo-sayısal çevirme işleminin nasıl gerçekleştiği açıklanmıştır. Sayıcı integral alıcı çıkışlarının sıfır olduğu durumda, giriş seçme anahtarı S_1 üzerinden artı değerli bir analog giriş işareti uygulandığını düşünelim.

A_1 OPAMP'ının negatif girişi sanal toprak olduğundan ve V_1 geriliminin bir süre değişmeyeceğini varsayarak R direnci ve C sığacı üzerinden sabit bir akım geçeceğini söyleyebiliriz. Bu akım sabit olduğundan, C sığacı doğrusal olarak dolacak ve sonuçta A_1 in çıkışında eksiye giden doğrusal bir gerilim yokuşu belirecektir (Şekil-13.23) Sayaç belirlenen sayıya ulaştığı zaman sıfırlanacak ve denetim devresi eksi referans gerilimini ($-V_{REF}$) A_1 OPAMP'ının negatif girişine verecektir. Bu noktada sığaç, girişteki analog değer ile orantılı bir gerilime ($-V$) dolmuştur (Şekil-13.24).

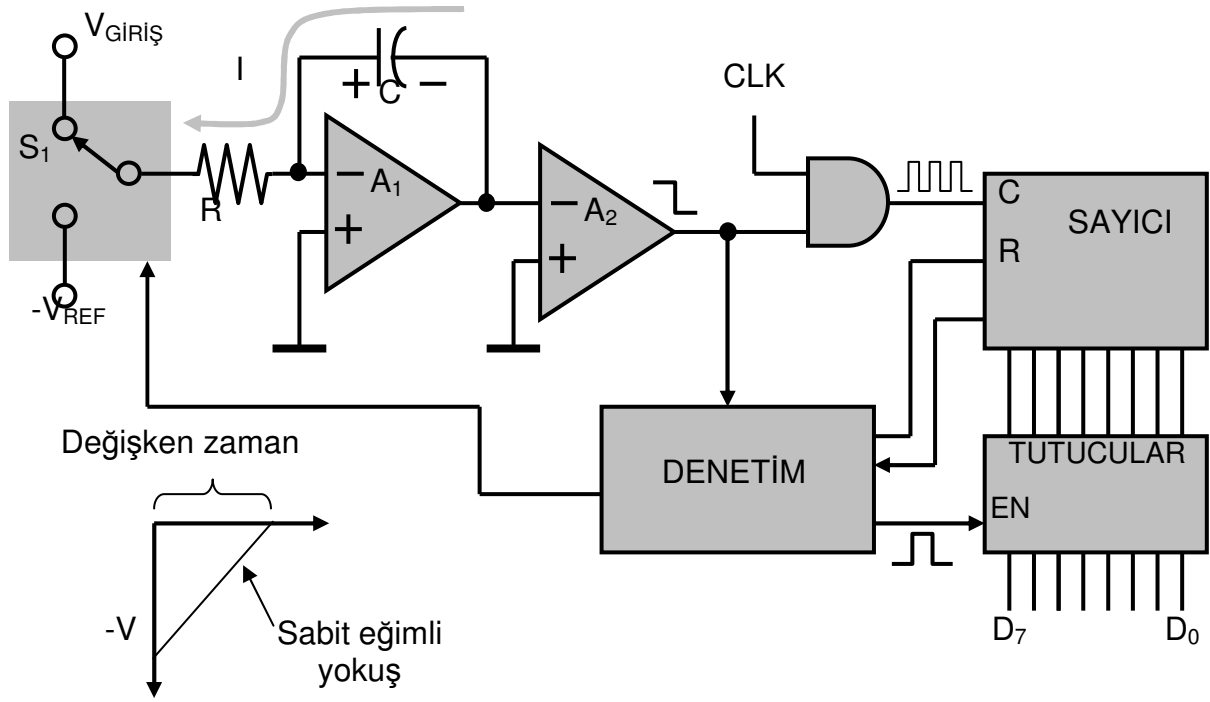
Sığaç bu kez de, $-V_{REF}$ geriliminden kaynaklanan sabit bir akımla doğrusal olarak boşalır (şekil-13.24). Bu doğrusal boşalma, A_1 in çıkışında $-V$ den başlayıp artıya giden ve eğimi, doldurma geriliminden bağımsız ve sabit olan doğrusal bir gerilim yokuşu oluşturur. Sığaç boşalırken, sayaç ta sıfırdan başlayarak sayar. Boşalma hızı (eğimi) sabit olduğu için, sığacın boşalma süresi, dolmuş olduğu $-V$ gerilim değerine (V_{in} ile orantılı) bağlıdır. İntegral alıcı (A_1) çıkış gerilimi sıfır volta ulaşınca karşılaştırıcı (A_2) çıkışında DÜŞÜK seviye belirir ve sayaç durdurulur. Bu anda sayaçtaki ikilik sayı tutulur ve bir çevirme döngüsü tamamlanmış olur.



Şekil-13.24 Sabit aralıkta değişken gerilime bağlı, negatif yokuşu elde edilmesi.



Şekil-13.25 Sabit aralıkta negatif yokuşun sonu sayıcı sıfırlanır, S_1 anahtarın konumu değiştirilir.



Şekil-13.26 Sabit aralıkta değişken gerilime bağlı, negatif yokuşu elde edilmesi.

ARDIŞIK YAKLAŞIM ADC

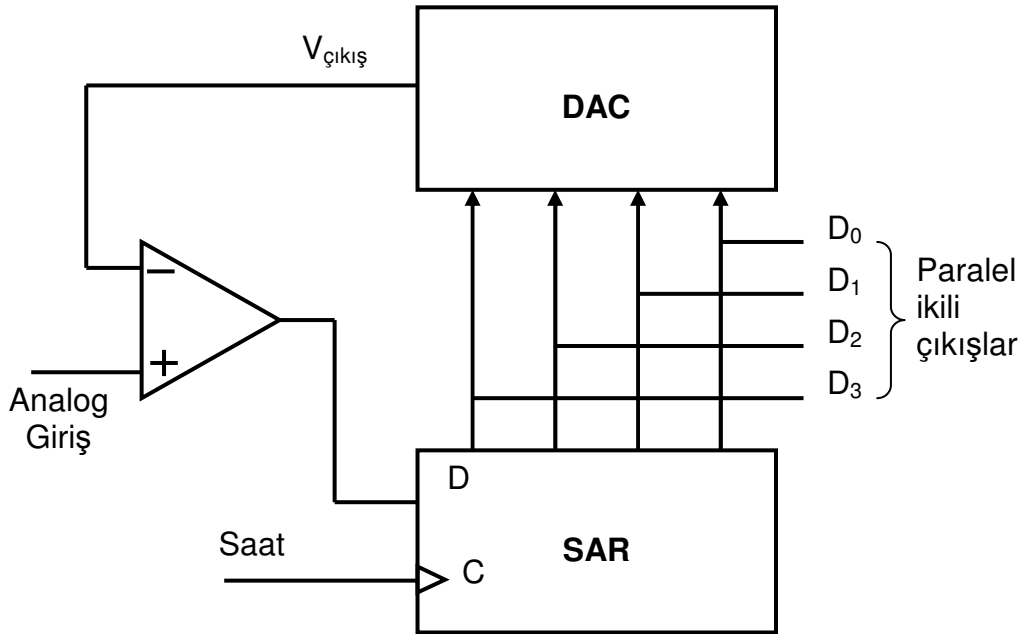
Ardışık-yaklaşım analog-sayısal (Successive-approximation) çeviriciler tümdevre ADC'lerde en yaygın kullanılan yöntemdir. Bu yöntemin çevirme süresi, flash yöntemi dışındaki tüm diğer yöntemlerden çok daha kısadır. Ayrıca çevirme süresi girişteki analog işaretin değerinden bağımsız ve sabittir.

Şekil-13.27'de 4-bitlik ardışık-yaklaşımli analog-sayısal çeviricinin devresi verilmiştir. Burada bir DAC, bir karşılaştırıcı ve bir de ardışık-yaklaşım yazacı (successive-approximation register-SAR) kullanılmıştır.

Sistemin çalışmasını 4 bitlik çevirme işlemiyle açıklayacağız. SAR yazacı 4 bitlik yaklaşım sayısının belirlendiği ve kesin sonucun elde edildiği yazacıdır. Bu yazacın kelime uzunluğu sistemin bit sayısını belirler. DAC giriş bitleri, YDB'den (MSB) başlanarak her adımda sadece sırası gelen bit bir yapılır. DAC çıkışından elde edilen analog işaret giriş işareti ile karşılaştırıcı kullanılarak karşılaştırılır. Karşılaştırıcı çıkışından YÜKSEK seviye alınırsa bu bit bir olarak korunur, eğer DÜŞÜK seviye elde edilirse bu bit temizlenir. Tüm bitler yaklaşım işlemine sokulduktan sonra analog giriş işaretinin karşılığı SAR içerisinde elde edilir. DAC çıkışı daha büyükse karşılaştırıcı çıkışı DÜŞÜK seviye, daha küçükse YÜKSEK seviye olur.

4 bit sistemin çalışması Şekil-13.28'de gösterilmiştir. Şekil-13.28 (a)'da analog giriş işaretinin büyüklüğü 5 Volt kabul edilerek en yüksek değerlikli bit "1" yapılmıştır. DAC çıkışından bu durumda 8 V çıkış elde edilir, DAC çıkışı analog

girişten daha büyük olduğu için karşılaştırıcı çıkışı DÜŞÜK seviye olur ve 2^3 değerlikli bit temizlenir. Bunun anlamı girişteki gerilimin 8 volttan küçük olduğudur.

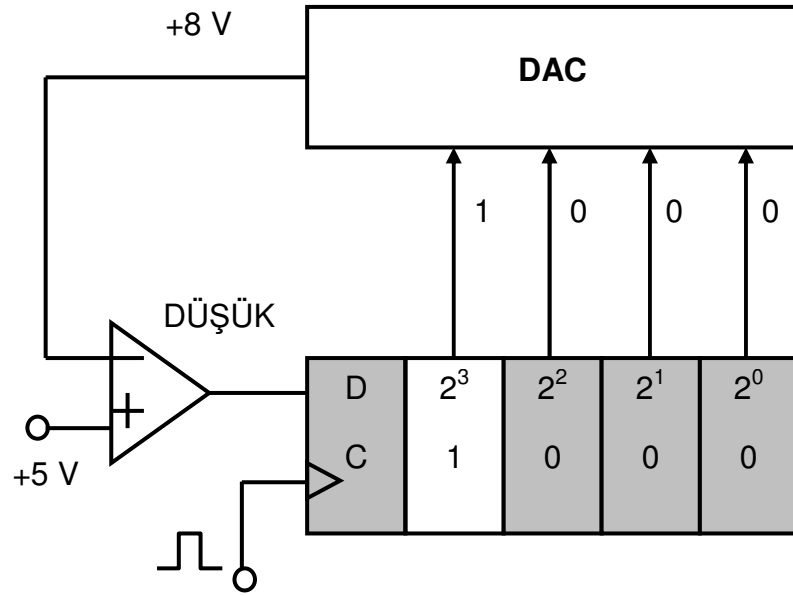


Şekil-13.27 Ardışık-yaklaşım çevirme işlemi.

İkinci adımda 2^2 ağırlıklı bit "1" yapılarak DAC girişine uygulanır, çıkıştan elde edilen gerilim analog giriş işaretinden küçük olduğu için karşılaştırıcı çıkışı YÜKSEK seviye olur ve bu bit kurulu olarak kalır. Bakınız şekil-13.28 (b). Üçüncü adımda 2^1 ağırlıklı bit kurulur, DAC girişine ikilik 6 sayısı uygulandığı için çıkışından elde edilen 6 Volt giriş işaretinden büyük olduğu için karşılaştırıcı çıkışı DÜŞÜK seviye olur ve bu bit temizlenir. Bakınız şekil-13.28 (c). Son olarak da 2^0 ağırlıklı bit kurulur, DAC çıkışından elde edilen 5 volt girişteki analog işaret ile eşit olduğu için karşılaştırıcı çıkışı YÜKSEK seviye olur ve bit kurulu olarak bırakılır. Bakınız şekil-13.28 (d). Dört adımın sonunda paralel çıkışlardan analog girişin karşılığı olan ikilik sayı elde edilir. Aslında dönüşüm sırasında çıkışa ikilik kodlar şekil-13.27'de olduğu gibi doğrudan verilmez. Çıkışta diğer tüm devreler ile birlikte çalışabilmesi için üç konumlu tamponlar yer alır. Bu tamponlar her dönüşüm sonunda izinlenir ve ikilik sayısal object dosya sadece çevrim işlemi bittikten sonra çıkışa aktarılır. Blok şemada basit olması açısından gösterilmemiştir. Dönüşüm süresinin sabit olması mikroişlemcilerle birlikte kullanımı kolaylaştırır.

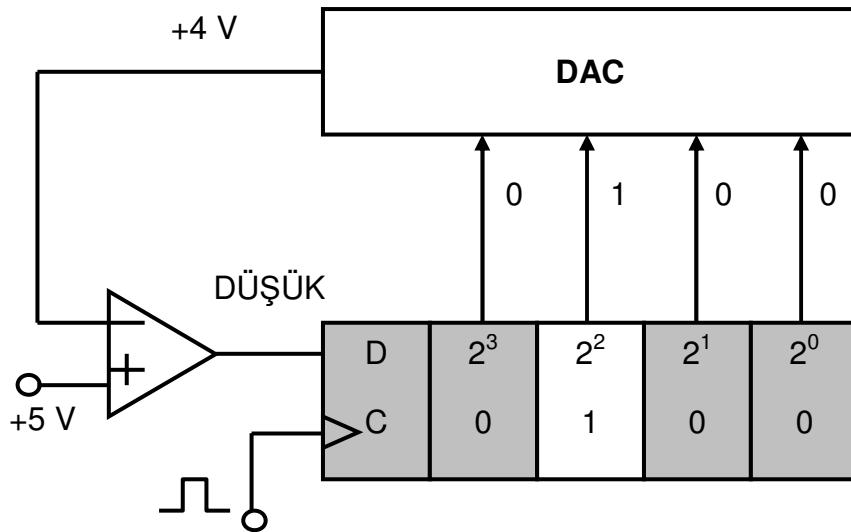
ÖRNEK ADC

Bu kısımda National Semiconductor firması tarafından üretilen ADC0804 analog-sayısal çeviricisi incelenecektir. Tümdevrenin mantık simgesi Şekil-13.29'da gösterilmiştir. Tek bir +5 Volt kaynaktan beslenen bu ADC'de tümdevre içerisinde osilatör devresi yer almaktadır. Çözünürlüğü 8 bittir ve çevrim süresi 100 μ s'dir. Aynı zamanda bu ADC'nin çıkışları üç konumlu tamponludur ve mikroişlemci ile birlikte kullanılabilir. Tekdüzeliği de iyidir.

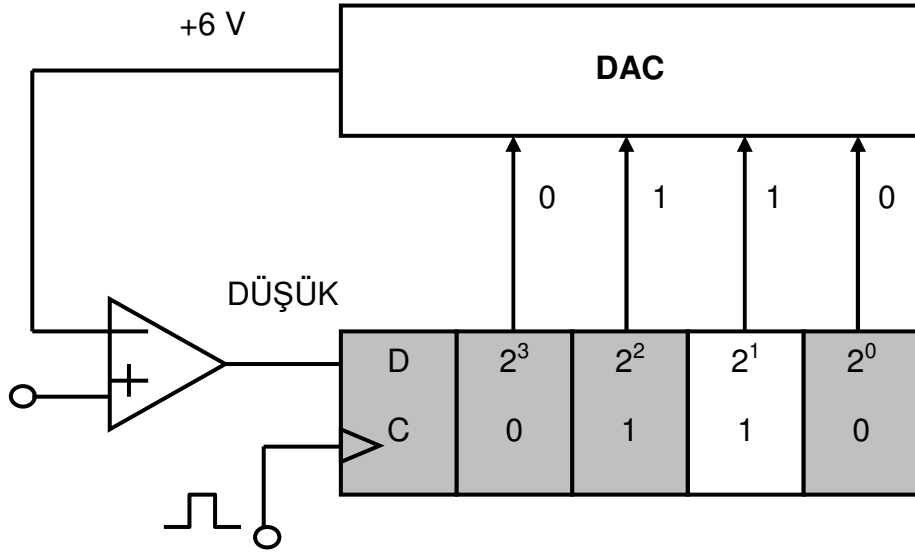


(a)

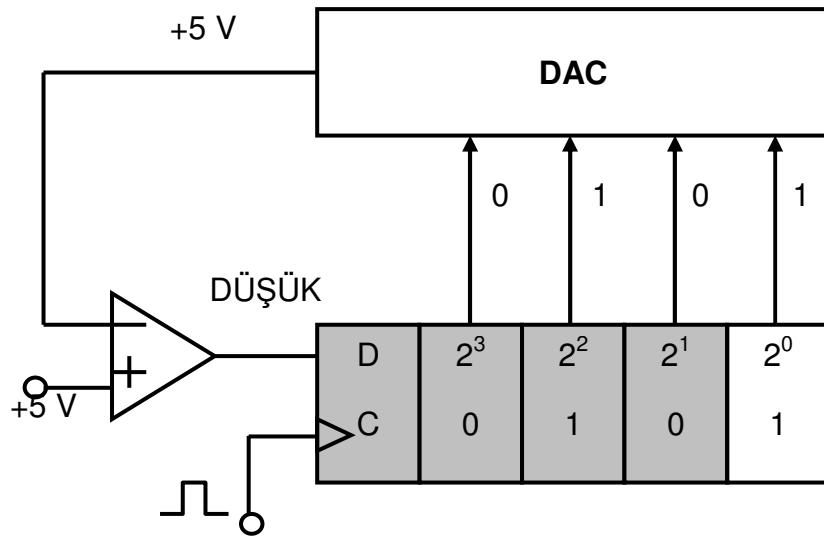
ADC0804'ün çalışması şöyledir; 256 dirençten oluşan bir DAC, bir karşılaştırıcı ve bir 8 bit SAR'dan oluşur. 2^8 değerlikli bitten başlayarak tüm 8 biti kurarak SAR'ın içeriğini belirler. Bu işlem 64 saat saykılı gerektirir. Çevrimin sonunda SAR'ın içeriği çıkış tutucularına verilir. INT/ çıkışını DÜŞÜK seviyeye çekerek işlemin tamamlandığını diğer birimlere bildirir. Her dönüşüm işlemi başında SAR WR/ girişine uygulanan bir düşen kenar uygulanarak temizlenir. Mikroişlemci ile birlikte kullanılmayacaksa INT/ çıkışı WR/ girişine bağlanarak uyarma işlemini kendi kendine yapması sağlanır. Tüm bu işlemlerin olabilmesi için CS/ girişini sürekli DÜŞÜK seviyede olması gerekir. RD/ girişi ise eğer CS/ DÜŞÜK seviyede ise çıkış üç durumlu tamponları yetkilemek için kullanılır.



(b) 2. adım.

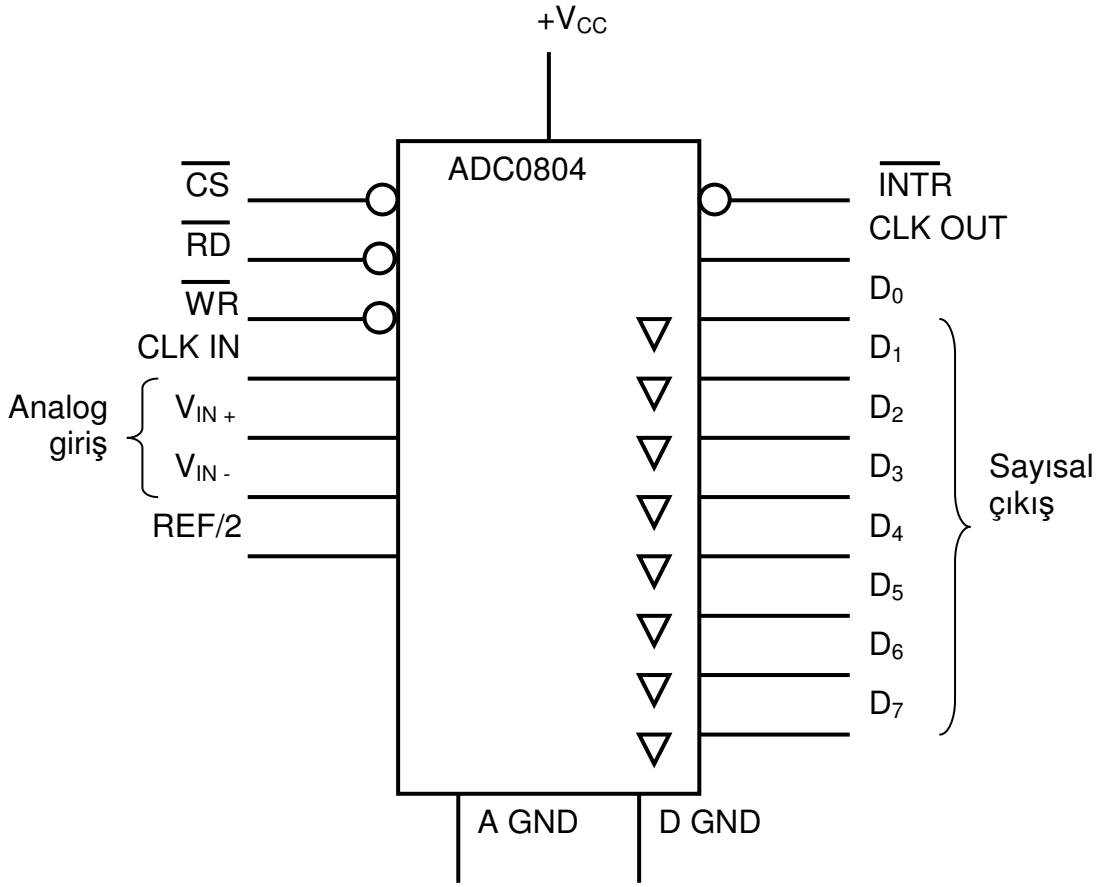


(c) 3. adım.



(d) 4. adım.

Şekil-13.28 Ardışık-yaklaşımın adımları.



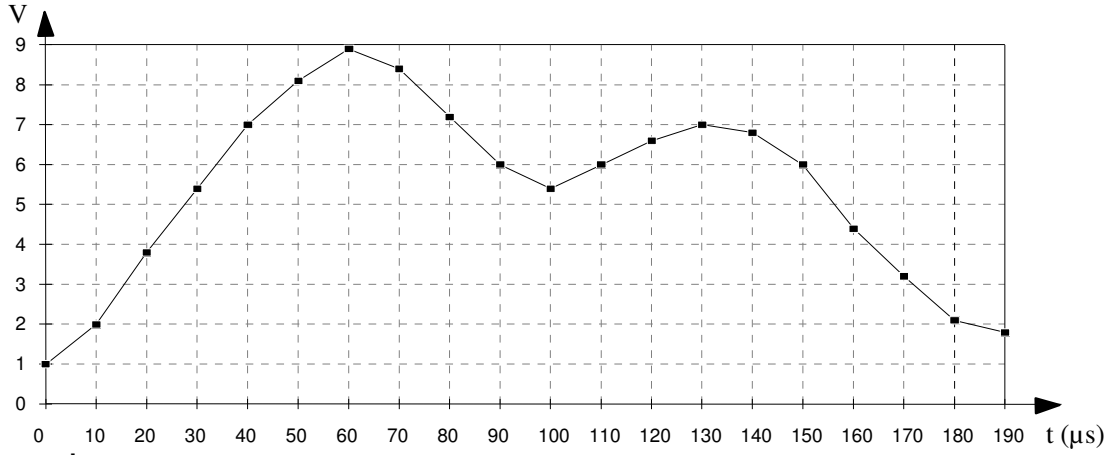
Şekil-13.29 ADC0804 analog-sayısal dönüştürücü.

REF/2 girişine analog girişe bağlanacak maksimum analog değerin yarısı kadar bir gerilim uygulanır. Genellikle ayar için bu girişe uygulanan gerilime ince ayar yapmak için bir ayar trimpotu yerleştirilir. Bu girişe önerilen değer 2,55 Volttur. Böylece girişten bunun iki katı olan 5,10 voltu uyguladığınızda tüm sayısal çıkışların YÜKSEK seviye olur. ADC'nin çözünürlüğünü hesaplamak da kolay olur. Gerekli hesaplamalar için ADC'nin veri yapraklarına bakınız.

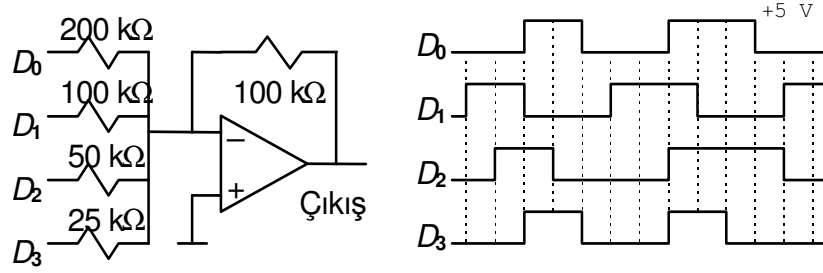
CLK OUT çıkışı ve CLK IN girişleri arasında bir direnç bağlanır, ayrıca CLK IN girişi ile toprak arasına bir kondansatör bağlanarak tümdevre içerisinde yer alan osilatör devresi için gerekli salınım devresi elde edilir. Tetikleme frekansı 500 kHz'e kadar arttırılabilir.

SORULAR

1. Çift eğimli A/D çeviricinin blok şemasını çizerek çalışmasını anlatınız.
2. 4 bitlik bir R/2R merdiven D/A çeviricinin çalışma ilkesini açıklayarak 1001 girdisinin analog değerini devrenin analizini yaparak hesaplayın. ($V_{CC} = +5V$)
3. Aşağıda verilen analog gerilim, örnekleme hızı 100 kHz olan 3-bitlik bir flaş çevirici ile sayısallaştırıldığında elde edilecek kodları yazın. ($V_{REF} = 8V$)



4. İkili ağırlıklı 6-bitlik bir DAC devresinde LSB ye bağlı direncin değeri $10\text{k}\Omega$ ise diğer dirençlerin değerleri ne olur?
5. Aşağıdaki çeviricinin dört girişine, sağda görülen +5V ile 0V arasında değişen giriş işaretleri verilmiştir. Çıkış gerilimini ölçekli olarak çizin.



6. 4 bitlik bir R/2R merdiven D/A çeviricinin çalışma ilkesini açıklayarak 1010 girdisinin analog değerini devrenin analizini yaparak hesaplayın. ($V_{CC} = +5\text{V}$)
7. 3, 10 ve 18 bitlik DAC'ların, yüzde olarak ayrımları ve doğrulukları ne kadardır?

TÜMDEVRE VERİ YAPRAKLARI

GİRİŞ

Bu bölümde derste ve deneyde kullanılan tümdevre veri yapraklarının özeti verilmiştir. Ayrıntılı bilgi WWW.philipslogic.com ve www.ti.com adreslerinden alınabilir.

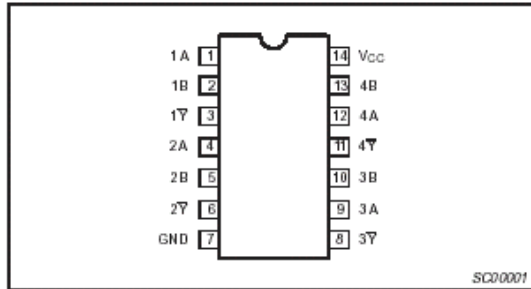
7400 DÖRTLÜ İKİ GİRİŞLİ VED GEÇİDİ

Quad 2-input NAND gate

74ALS00A

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74ALS00A	4.0ns	1.0mA

PIN CONFIGURATION



ORDERING INFORMATION

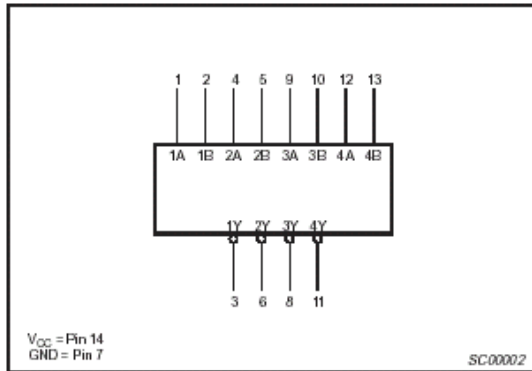
DESCRIPTION	ORDER CODE	DRAWING NUMBER
	COMMERCIAL RANGE V _{CC} = 5V ±10%, T _{amb} = 0°C to +70°C	
14-pin plastic DIP	74ALS00AN	SOT27-1
14-pin plastic SO	74ALS00AD	SOT108-1
14-pin plastic SSOP Type II	74ALS00ADB	SOT337-1

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

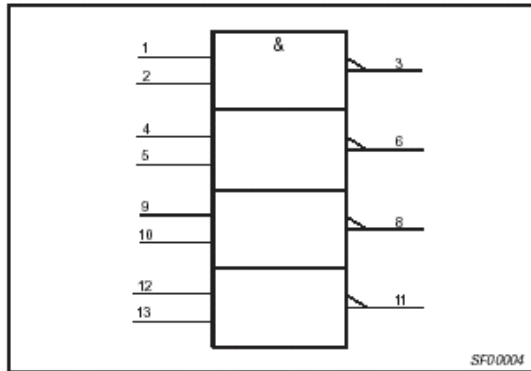
PINS	DESCRIPTION	74ALS (U.L.) HIGH/LOW	LOAD VALUE HIGH/LOW
nA, nB	Data inputs	1.0/1.0	20µA/0.1mA
nY	Data output	20/80	0.4mA/8mA

NOTE: One (1.0) ALS unit load is defined as: 20µA in the High state and 0.1mA in the Low state.

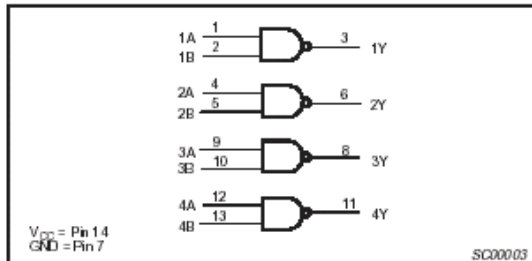
LOGIC SYMBOL



IEC/IEEE SYMBOL



LOGIC DIAGRAM



FUNCTION TABLE

INPUTS		OUTPUT
nA	nB	nY
H	H	L
L	X	H
X	L	H

H = High voltage level
L = Low voltage level
X = Don't care

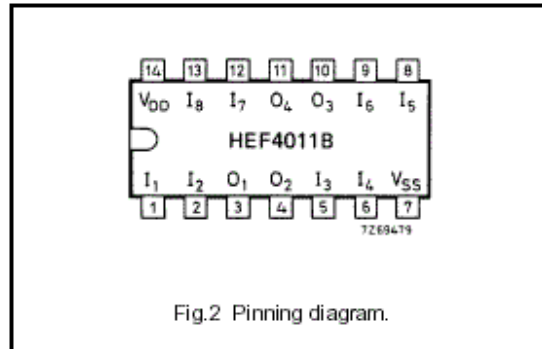
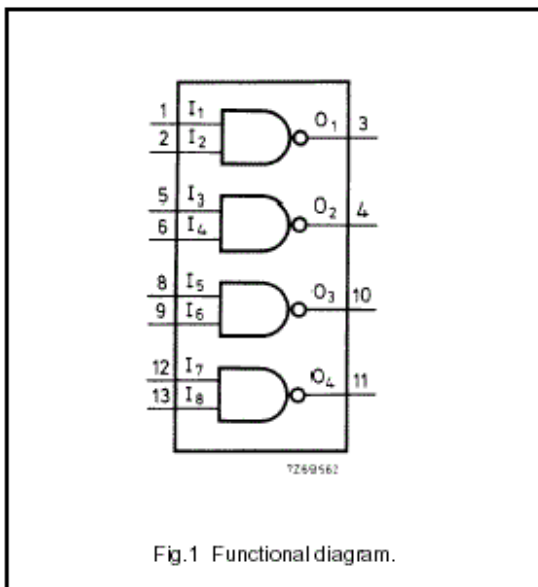
4011 DÖRTLÜ İKİ GİRİŞLİ CMOS VED GEÇİDİ

Quadruple 2-input NAND gate

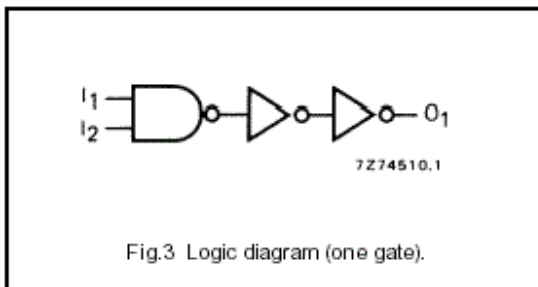
HEF4011B gates

DESCRIPTION

The HEF4011B provides the positive quadruple 2-input NAND function. The outputs are fully buffered for highest noise immunity and pattern insensitivity of output impedance.



- HEF4011BP(N): 14-lead DIL; plastic (SOT27-1)
- HEF4011BD(F): 14-lead DIL; ceramic (cerdip) (SOT73)
- HEF4011BT(D): 14-lead SO; plastic (SOT108-1)
- (): Package Designator North America



74LS02 DÖRTLÜ İKİ GİRİŞLİ VEYAD GEÇİDİ

Quad 2-input NOR gate

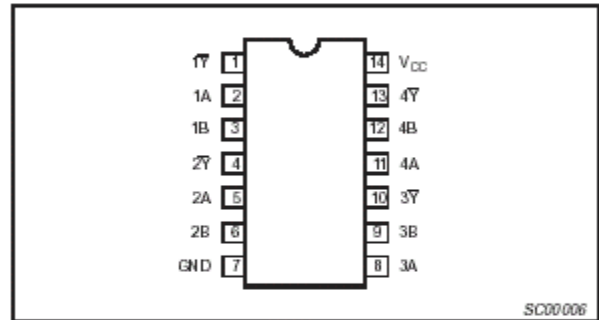
74ALS02

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74ALS02	4.0ns	1.0mA

ORDERING INFORMATION

DESCRIPTION	ORDER CODE	DRAWING NUMBER
	COMMERCIAL RANGE V _{CC} = 5V ±10%, T _{amb} = 0°C to +70°C	
14-pin plastic DIP	74ALS02N	SOT27-1
14-pin plastic SO	74ALS02D	SOT108-1

PIN CONFIGURATION

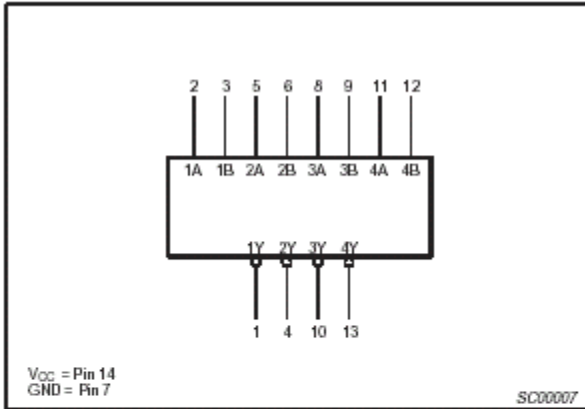


INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

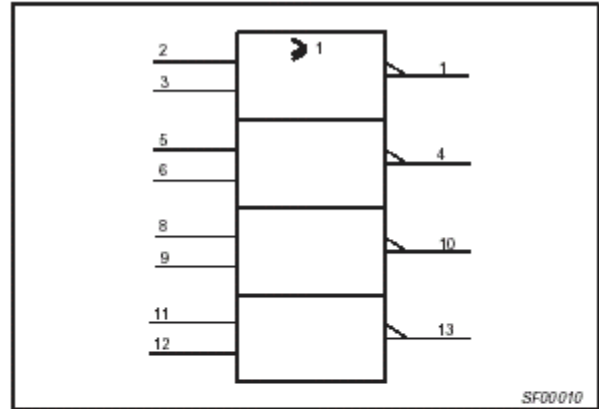
PINS	DESCRIPTION	74ALS (U.L.) HIGH/LOW	LOAD VALUE HIGH/LOW
nA, nB	Data inputs	1.0/1.0	20µA/0.1mA
nY	Data output	20/80	0.4mA/8mA

NOTE: One (1.0) ALS unit load is defined as: 20µA in the High state and 0.1mA in the Low state.

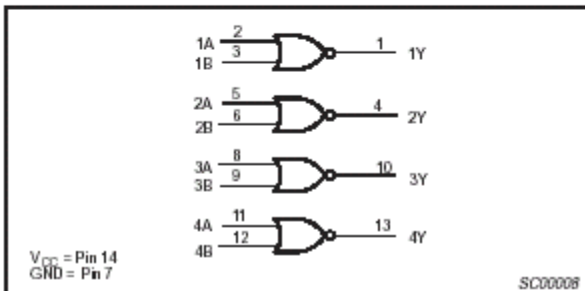
LOGIC SYMBOL



IEC/IEEE SYMBOL



LOGIC DIAGRAM



FUNCTION TABLE

INPUTS		OUTPUT
nA	nB	nY
H	H	L
L	X	H
X	L	H

H = High voltage level
L = Low voltage level
X = Don't care

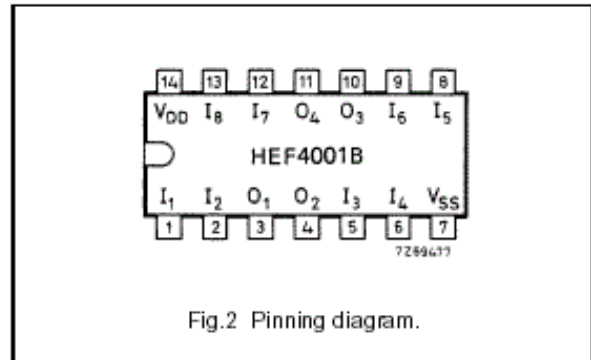
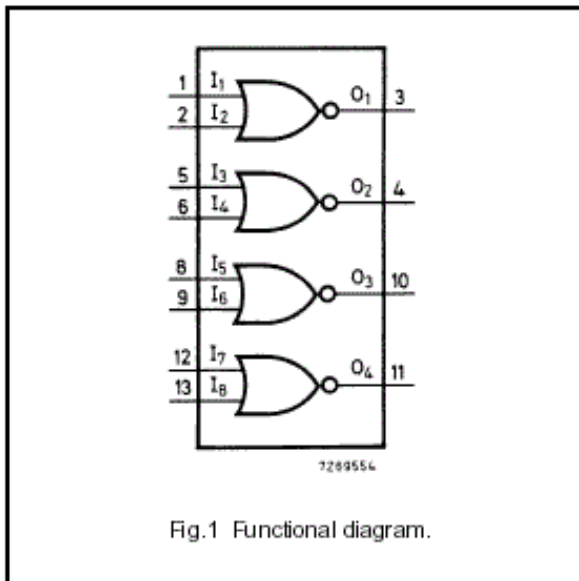
4001 DÖRTLÜ İKİ GİRİŞLİ CMOS VEYAD GEÇİDİ

Quadruple 2-input NOR gate

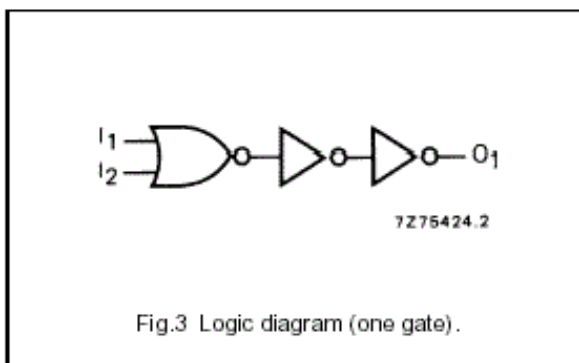
HEF4001B gates

DESCRIPTION

The HEF4001B provides the positive quadruple 2-input NOR function. The outputs are fully buffered for highest noise immunity and pattern insensitivity of output impedance.



- HEF4001BP(N): 14-lead DIL; plastic (SOT27-1)
- HEF4001BD(F): 14-lead DIL; ceramic (cerdip) (SOT73)
- HEF4001BT(D): 14-lead SO; plastic (SOT108-1)
- (): Package Designator North America



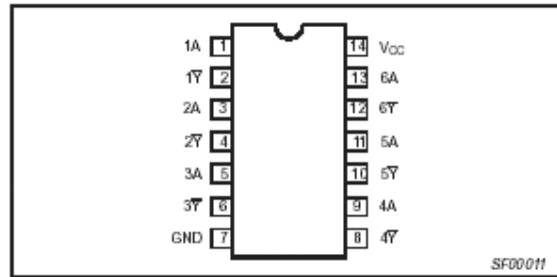
74LS04 ALTILI DEĞİL GEÇİDİ

Hex inverter

74ALS04B

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74ALS04B	3.5ns	2.0mA

PIN CONFIGURATION



ORDERING INFORMATION

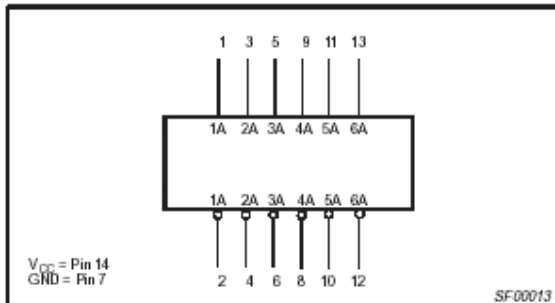
DESCRIPTION	ORDER CODE	DRAWING NUMBER
	COMMERCIAL RANGE V _{CC} = 5V ±10%, T _{amb} = 0°C to +70°C	
14-pin plastic DIP	74ALS04BN	SOT27-1
14-pin plastic SO	74ALS04BD	SOT108-1
14-pin plastic SSOP Type II	74ALS04BDB	SOT337-1

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

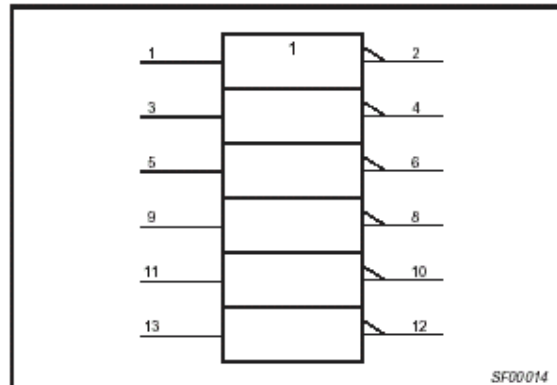
PINS	DESCRIPTION	74ALS (U.L.) HIGH/LOW	LOAD VALUE HIGH/LOW
nA	Data Input	1.0/1.0	20µA/0.1mA
nY	Data output	20/80	0.4mA/8mA

NOTE: One (1.0) ALS unit load is defined as: 20µA in the High state and 0.1mA in the Low state.

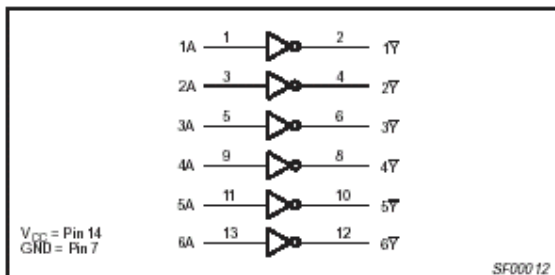
LOGIC SYMBOL



IEC/IEEE SYMBOL



LOGIC DIAGRAM



FUNCTION TABLE

INPUT	OUTPUT
nA	nY
L	H
H	L

H = High voltage level
L = Low voltage level

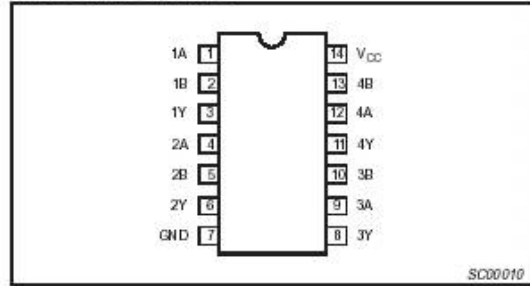
74LS08 DÖRTLÜ İKİ GİRİŞLİ VE GEÇİDİ

Quad 2-input AND gate

74ALS08

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74ALS08	5.0ns	1.8mA

PIN CONFIGURATION



ORDERING INFORMATION

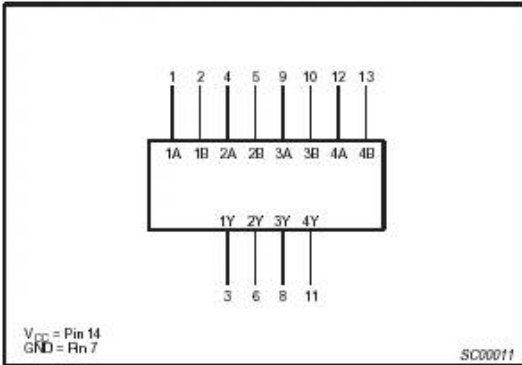
DESCRIPTION	ORDER CODE	DRAWING NUMBER
	COMMERCIAL RANGE V _{CC} = 5V ±10%, T _{amb} = 0°C to +70°C	
14-pin plastic DIP	74ALS08N	SOT27-1
14-pin plastic SO	74ALS08D	SOT108-1
14-pin plastic SSOP Type II	74ALS08DB	SOT337-1

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

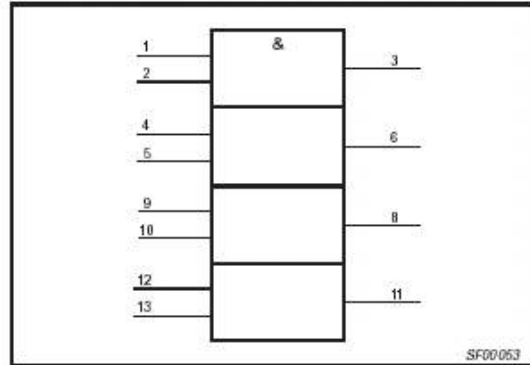
PINS	DESCRIPTION	74ALS (U.L.) HIGH/LOW	LOAD VALUE HIGH/LOW
nA, nB	Data inputs	1.0/1.0	20µA/0.1mA
nY	Data outputs	20/80	0.4mA/8mA

NOTE: One (1.0) ALS unit load is defined as: 20µA in the High state and 0.1mA in the Low state.

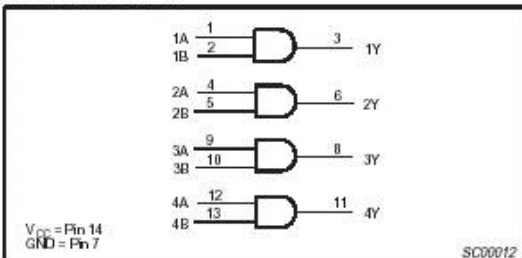
LOGIC SYMBOL



IEC/IEEE SYMBOL



LOGIC DIAGRAM



FUNCTION TABLE

INPUTS		OUTPUT
nA	nB	nY
H	H	L
L	X	H
X	L	H

H = High voltage level
L = Low voltage level
X = Don't care

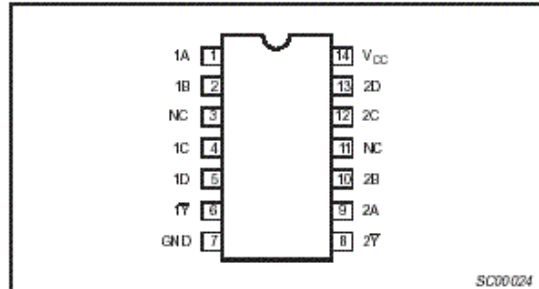
74LS20 İKİLİ DÖRT GİRİŞLİ VED GEÇİDİ

Dual 4-input NAND gate

74ALS20A

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74ALS20A	4.5ns	0.65mA

PIN CONFIGURATION



ORDERING INFORMATION

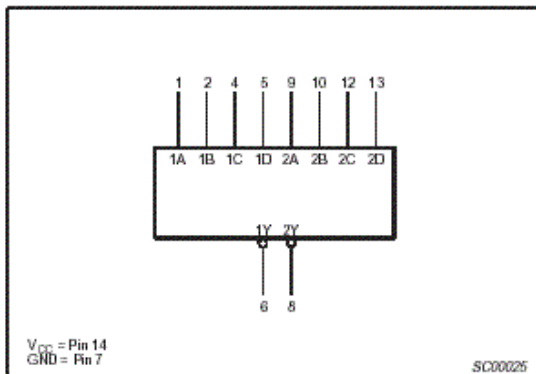
DESCRIPTION	ORDER CODE	DRAWING NUMBER
	COMMERCIAL RANGE V _{CC} = 5V ±10%, T _{amb} = 0°C to +70°C	
14-pin plastic DIP	74ALS20AN	SOT27-1
14-pin plastic SO	74ALS20AD	SOT108-1

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

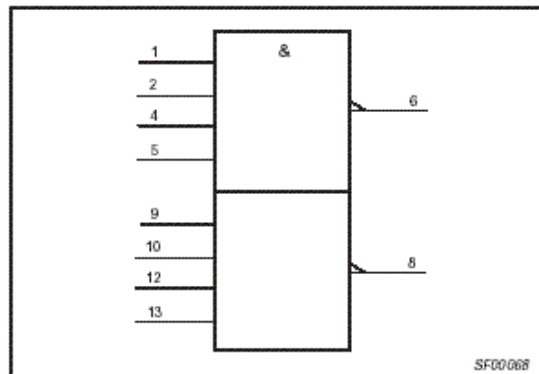
PINS	DESCRIPTION	74ALS (U.L.) HIGH/LOW	LOAD VALUE HIGH/LOW
nA, nB, nC, nD	Data Inputs	1.0/1.0	20µA/0.1mA
nY	Data outputs	20/80	0.4mA/8mA

NOTE: One (1.0) ALS unit load is defined as: 20µA in the High state and 0.1mA in the Low state.

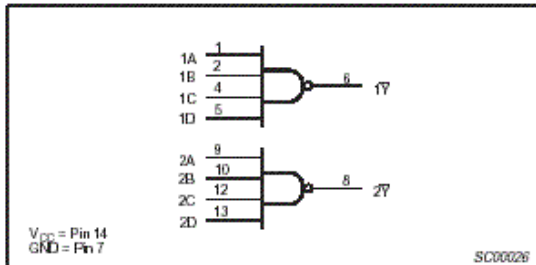
LOGIC SYMBOL



IEC/IEEE SYMBOL



LOGIC DIAGRAM



FUNCTION TABLE

INPUTS				OUTPUT
nA	nB	nC	nD	nY
H	H	H	H	L
L	X	X	X	H
X	L	X	X	H
X	X	L	X	H
X	X	X	L	H

H = High voltage level
L = Low voltage level
X = Don't care

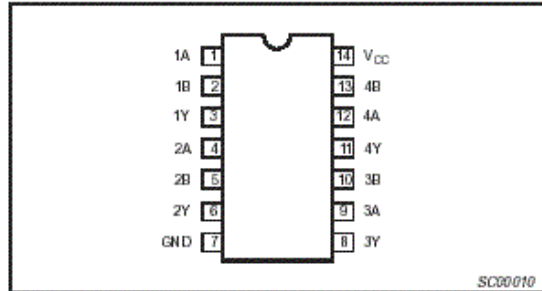
74LS32 DÖRTLÜ İKİ GİRİŞLİ VEYA GEÇİDİ

Quad 2-input OR gate

74ALS32

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74ALS32	5.0ns	2.3mA

PIN CONFIGURATION



ORDERING INFORMATION

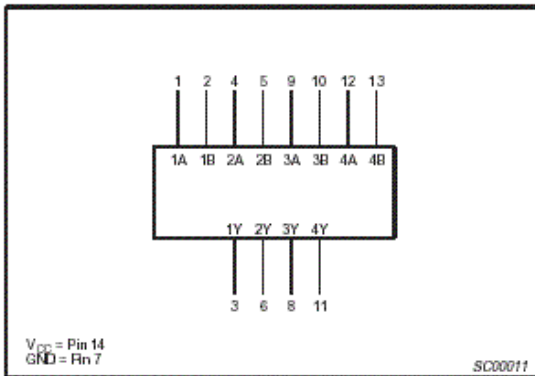
DESCRIPTION	ORDER CODE	DRAWING NUMBER
	COMMERCIAL RANGE V _{CC} = 5V ±10%, T _{amb} = 0°C to +70°C	
14-pin plastic DIP	74ALS32N	SOT27-1
14-pin plastic SO	74ALS32D	SOT108-1
14-pin plastic SSCP Type II	74ALS32DB	SOT337-1

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

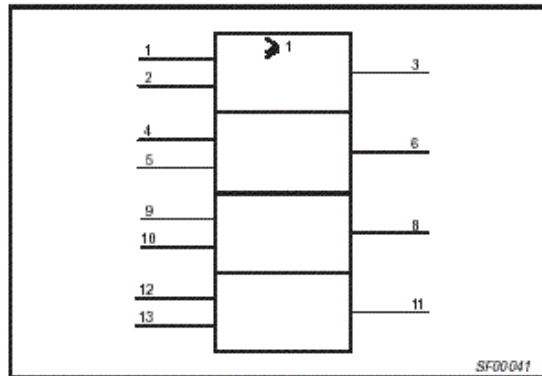
PINS	DESCRIPTION	74ALS (U.L.) HIGH/LOW	LOAD VALUE HIGH/LOW
nA, nB	Data Inputs	1.0/1.0	20µA/0.1mA
nY	Data output	20/80	0.4mA/8mA

NOTE: One (1.0) ALS unit load is defined as: 20µA in the High state and 0.1mA in the Low state.

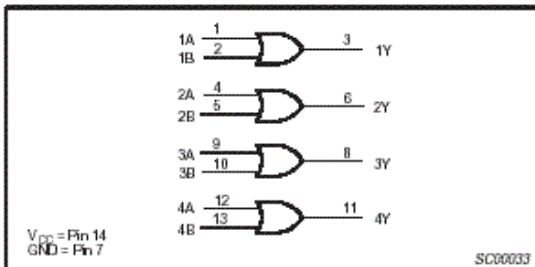
LOGIC SYMBOL



IEC/IEEE SYMBOL



LOGIC DIAGRAM



FUNCTION TABLE

INPUTS		OUTPUT
nA	nB	nY
H	X	H
X	H	H
L	L	L

H = High voltage level
L = Low voltage level
X = Don't care

74LS86 DÖRTLÜ EXOR GEÇİDİ

Quad 2-input exclusive-OR gate

74ALS86

DESCRIPTION

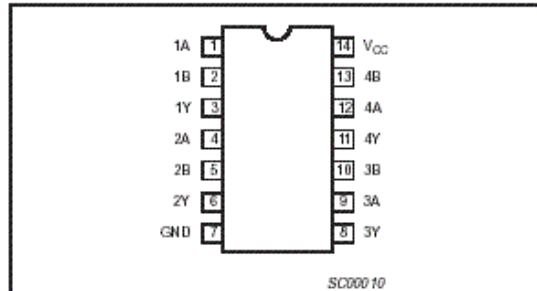
The 74ALS86 contain four independent 2-input Exclusive-OR gates. A common application is a true/complement element. If one input is held Low, the signal on the other input will be reproduced in true form at the output. If one input is held High, the signal on the other input will be reproduced inverted at the output.

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74ALS86	6.0ns	3.9mA

ORDERING INFORMATION

DESCRIPTION	ORDER CODE		DRAWING NUMBER
	COMMERCIAL RANGE V _{CC} = 5V ±10%, T _{amb} = 0°C to +70°C		
14-pin plastic DIP	74ALS86N	SOT27-1	
14-pin plastic SO	74ALS86D	SOT108-1	

PIN CONFIGURATION

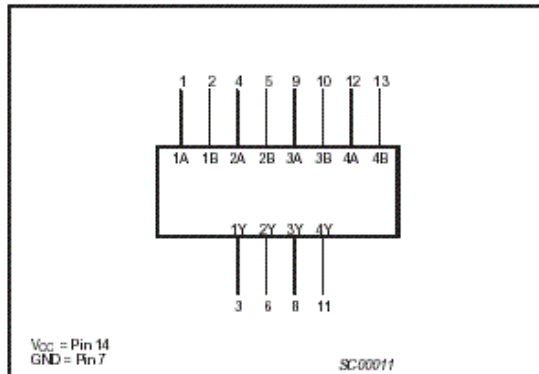


INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

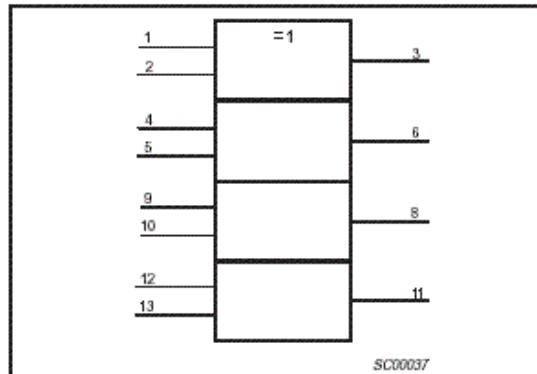
PINS	DESCRIPTION	74ALS (U.L.) HIGH/LOW	LOAD VALUE HIGH/LOW
nA, nB	Data inputs	1.0/1.0	20µA/0.1mA
nY	Data outputs	20/80	0.4mA/8mA

NOTE: One (1.0) ALS unit load is defined as: 20µA in the High state and 0.1mA in the Low state.

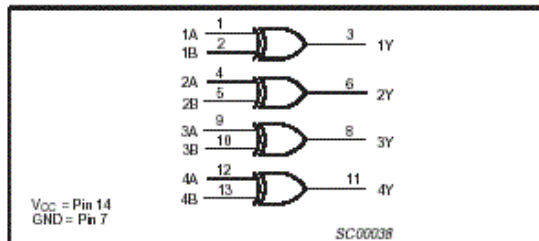
LOGIC SYMBOL



IEC/IEEE SYMBOL



LOGIC DIAGRAM



FUNCTION TABLE

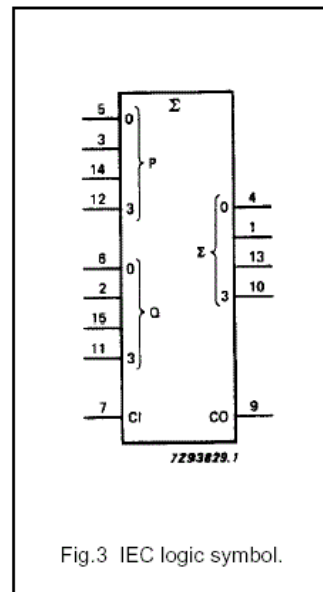
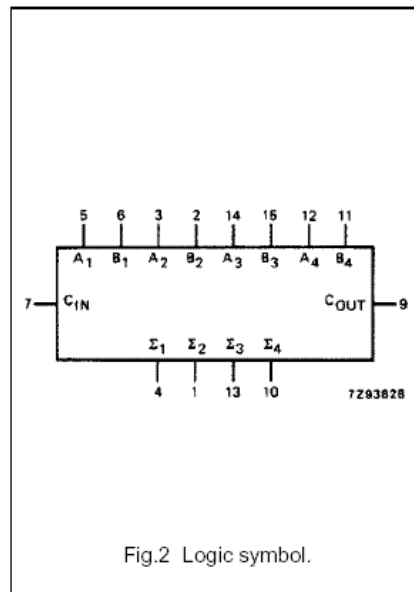
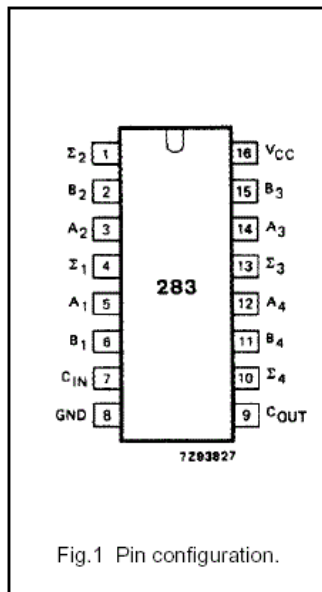
INPUTS		OUTPUT
nA	nB	nY
L	L	L
L	H	H
H	L	H
H	H	L

H = High voltage level
L = Low voltage level

74LS283 (74LS83) 4 BİT TAM TOPLAYICI

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
4, 1, 13, 10	Σ_1 to Σ_4	sum outputs
5, 3, 14, 12	A_1 to A_4	A operand inputs
6, 2, 15, 11	B_1 to B_4	B operand inputs
7	C_{IN}	carry input
8	GND	ground (0 V)
9	C_{OUT}	carry output
16	V_{CC}	positive supply voltage



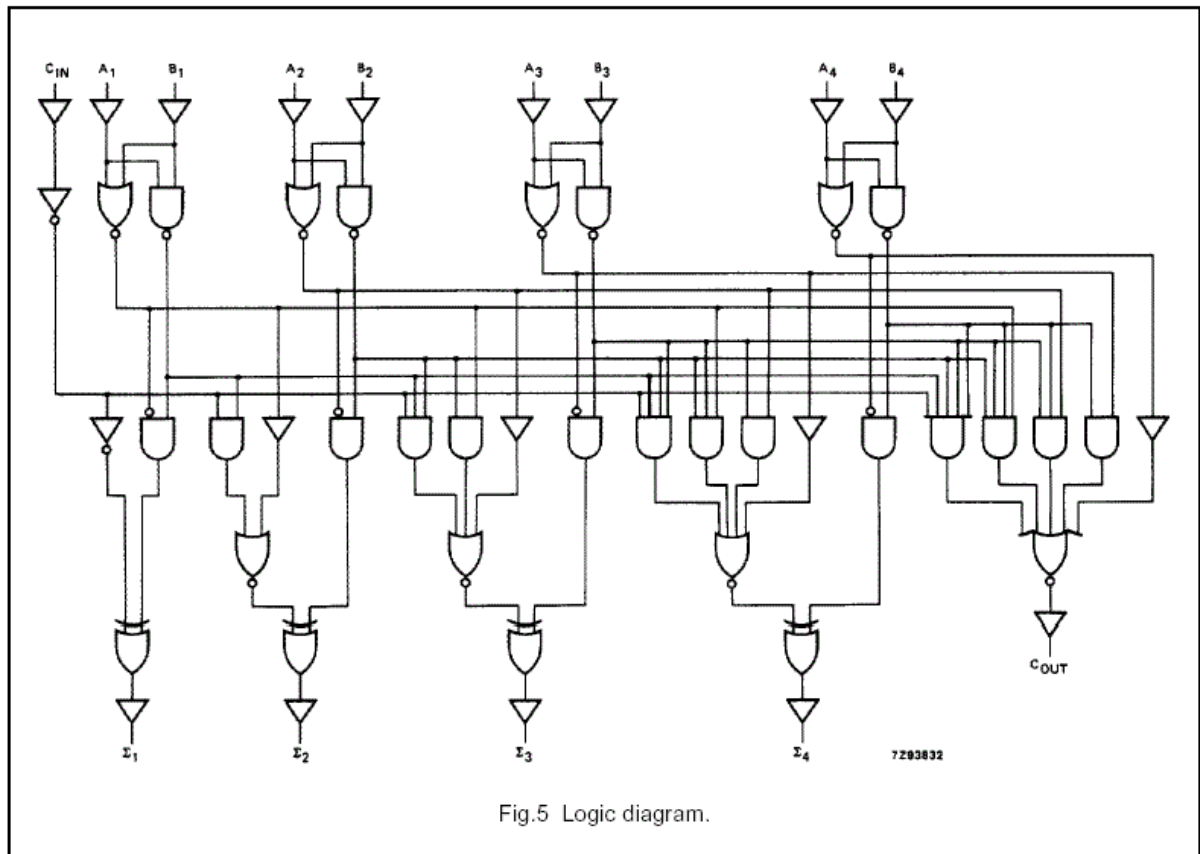


Fig.5 Logic diagram.

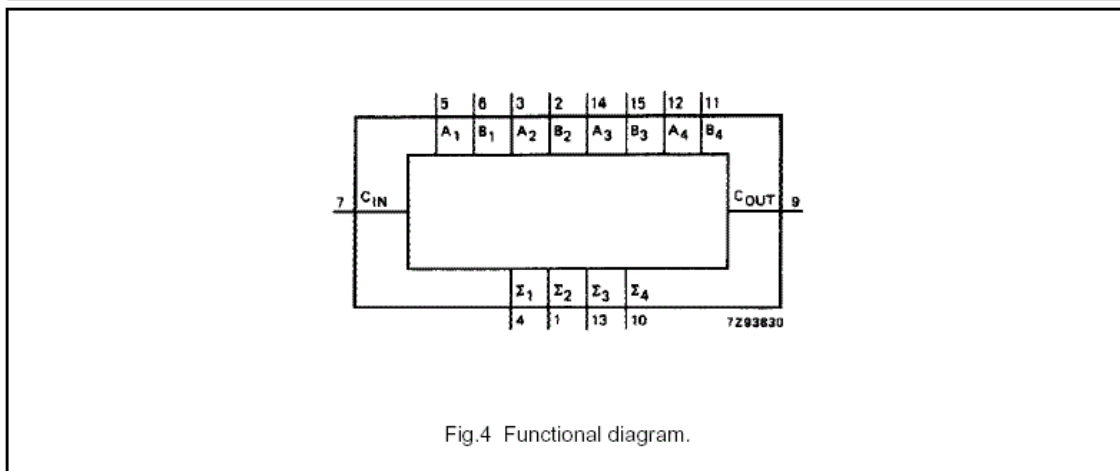


Fig.4 Functional diagram.

FUNCTION TABLE

PINS	C _{IN}	A ₁	A ₂	A ₃	A ₄	B ₁	B ₂	B ₃	B ₄	Σ ₁	Σ ₂	Σ ₃	Σ ₄	C _{OUT}	EXAMPLE ⁽²⁾
logic levels	L	L	H	L	H	H	L	L	H	H	H	L	L	H	
active HIGH	0	0	1	0	1	1	0	0	1	1	1	0	0	1	(3)
active LOW	1	1	0	1	0	0	1	1	0	0	0	1	1	0	(4)

Note

- H = HIGH voltage level
L = LOW voltage level

2. example

1001
1010

10011

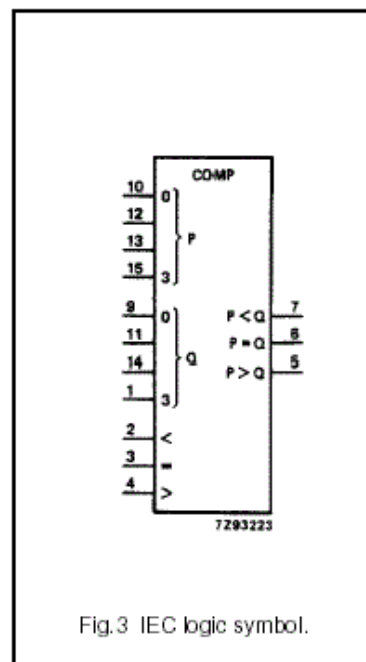
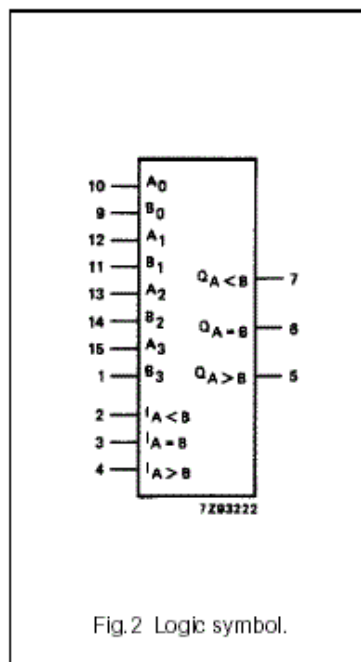
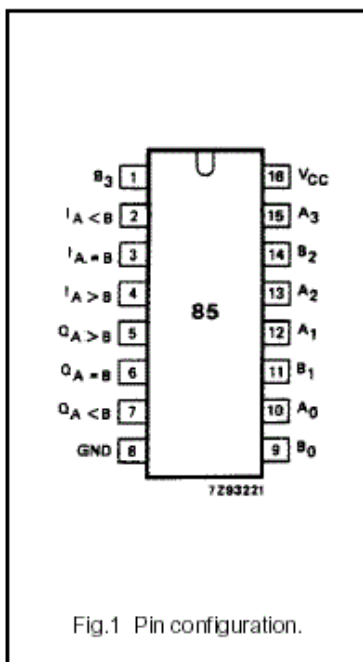
74HC85 4 BİT BÜYÜKLÜK KARŞILAŞTIRICI

4-bit magnitude comparator

74HC/HCT85

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
2	$I_{A<B}$	A < B expansion input
3	$I_{A=B}$	A = B expansion input
4	$I_{A>B}$	A > B expansion input
5	$Q_{A>B}$	A > B output
6	$Q_{A=B}$	A = B output
7	$Q_{A<B}$	A < B output
8	GND	ground (0 V)
9, 11, 14, 1,	B_0 to B_3	word B inputs
10, 12, 13, 15	A_0 to A_3	word A inputs
16	V_{CC}	positive supply voltage



4-bit magnitude comparator

74HC/HCT85

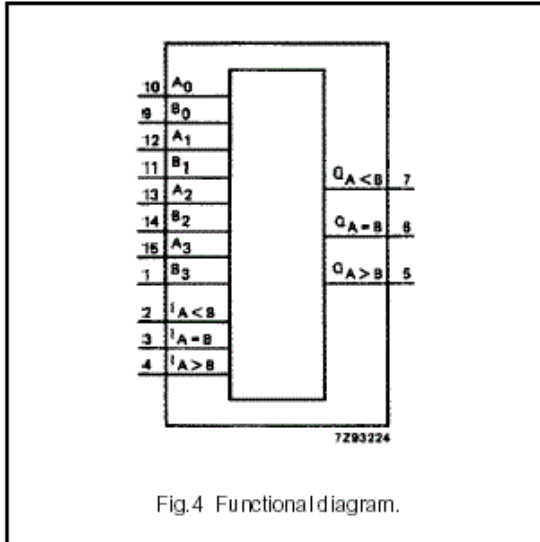


Fig.4 Functional diagram.

APPLICATIONS

- Process controllers
- Servo-motor control

FUNCTION TABLE

COMPARING INPUTS				CASCADING INPUTS			OUTPUTS		
A ₃ , B ₃	A ₂ , B ₂	A ₁ , B ₁	A ₀ , B ₀	I _{A>B}	I _{A<B}	I _{A=B}	Q _{A>B}	Q _{A<B}	Q _{A=B}
A ₃ >B ₃	X	X	X	X	X	X	H	L	L
A ₃ <B ₃	X	X	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ >B ₂	X	X	X	X	X	H	L	L
A ₃ =B ₃	A ₂ <B ₂	X	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ >B ₁	X	X	X	X	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ <B ₁	X	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ >B ₀	X	X	X	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ <B ₀	X	X	X	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	H	L	L	H	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	L	H	L	L	H	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	L	L	H	L	L	H
A ₃ >B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	X	X	H	L	L	H
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	H	H	L	L	L	L
A ₃ =B ₃	A ₂ =B ₂	A ₁ =B ₁	A ₀ =B ₀	L	L	L	H	H	L

Notes

1. H = HIGH voltage level
 L = LOW voltage level
 X = don't care

74LS138 3-8 KODÇÖZÜCÜ/VERİ DAĞITICI

1-of-8 decoder/demultiplexer

74ALS138

FEATURES

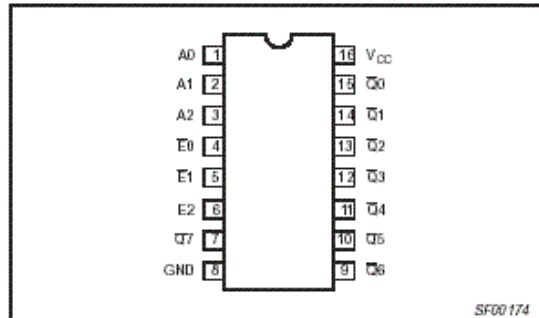
- Demultiplexing capability
- Multiple Input enable for easy expansion
- Ideal for memory chip select decoding

DESCRIPTION

The 74ALS138 decoder accepts three binary weighted inputs (A0, A1, A2) and when enabled, provides eight mutually exclusive, active-Low outputs ($\overline{Q}0 - \overline{Q}7$). The device features three Enable inputs; two active-Low ($\overline{E}0, \overline{E}1$) and one active-High (E2). Every output will be High unless $\overline{E}0$ and $\overline{E}1$ are Low and E2 is High. This multiple enable function allows easy parallel expansion of the device to 1-of-32 (5 lines to 32 lines) decoder with just four 74ALS138s and one inverter. The device can be used as an eight output demultiplexer by using one of the active-Low Enable inputs as the data input and the remaining Enable inputs as strobes. Enable inputs not used must be permanently tied to their appropriate active-High or active-Low state.

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74ALS138	12.0ns	4.0mA

PIN CONFIGURATION



ORDERING INFORMATION

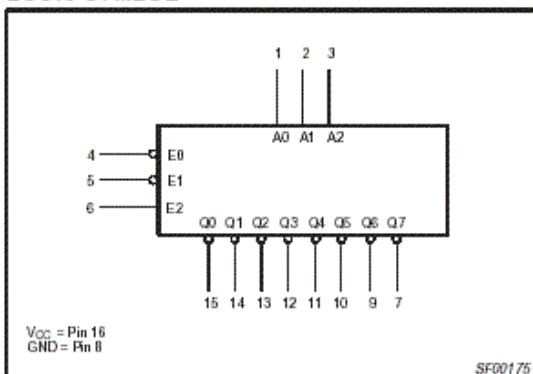
DESCRIPTION	ORDER CODE	DRAWING NUMBER
	COMMERCIAL RANGE $V_{CC} = 5V \pm 10\%$, $T_{amb} = 0^{\circ}C$ to $+70^{\circ}C$	
16-pin plastic DIP	74ALS138N	SOT38-4
16-pin plastic SO	74ALS138D	SOT109-1

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

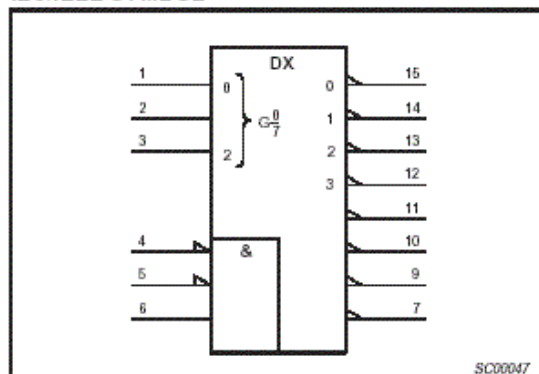
PINS	DESCRIPTION	74ALS (U.L.) HIGH/LOW	LOAD VALUE HIGH/LOW
A0 – A2	Address Inputs	1.0/1.0	20µA/0.1mA
$\overline{E}0, \overline{E}1$	Enable Inputs (active-Low)	1.0/1.0	20µA/0.1mA
E2	Enable Input (active-High)	1.0/1.0	20µA/0.1mA
$\overline{Q}0 - \overline{Q}7$	Data outputs (active-Low)	50/33	1.0mA/20mA

NOTE: One (1.0) ALS unit load is defined as: 20µA in the High state and 0.1mA in the Low state.

LOGIC SYMBOL



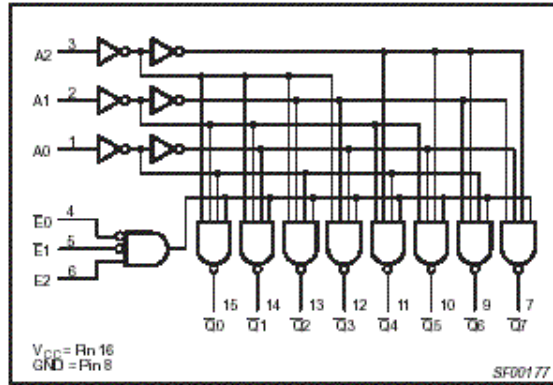
IEC/IEEE SYMBOL



1-of-8 decoder/demultiplexer

74ALS138

LOGIC DIAGRAM



FUNCTION TABLE

INPUTS						OUTPUTS							
$\overline{E_0}$	$\overline{E_1}$	$\overline{E_2}$	A0	A1	A2	$\overline{Q_0}$	$\overline{Q_1}$	$\overline{Q_2}$	$\overline{Q_3}$	$\overline{Q_4}$	$\overline{Q_5}$	$\overline{Q_6}$	$\overline{Q_7}$
H	X	X	X	X	X	H	H	H	H	H	H	H	H
X	H	X	X	X	X	H	H	H	H	H	H	H	H
X	X	L	X	X	X	H	H	H	H	H	H	H	H
L	L	H	L	L	L	L	H	H	H	H	H	H	H
L	L	H	H	L	L	H	L	H	H	H	H	H	H
L	L	H	L	H	L	H	H	L	H	H	H	H	H
L	L	H	H	H	L	H	H	H	L	H	H	H	H
L	L	H	L	L	H	H	H	H	H	L	H	H	H
L	L	H	H	L	H	H	H	H	H	H	L	H	H
L	L	H	L	H	H	H	H	H	H	H	H	L	H
L	L	H	H	H	H	H	H	H	H	H	H	H	L

H = High voltage level
 L = Low voltage level
 X = Don't care

74HC154 4-16 KODÇÖZÜCÜ/VERİ DAĞITICI

4-to-16 line decoder/demultiplexer

74HC/HCT154

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 13, 14, 15, 16, 17	\bar{Y}_0 to \bar{Y}_{15}	outputs (active LOW)
18, 19	\bar{E}_0, \bar{E}_1	enable inputs (active LOW)
12	GND	ground (0 V)
23, 22, 21, 20	A_0 to A_3	address inputs
24	V_{CC}	positive supply voltage

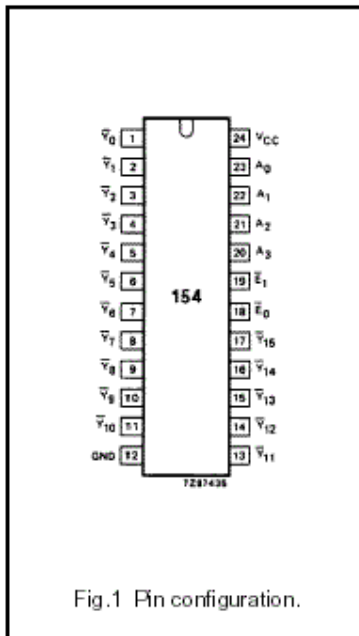


Fig.1 Pin configuration.

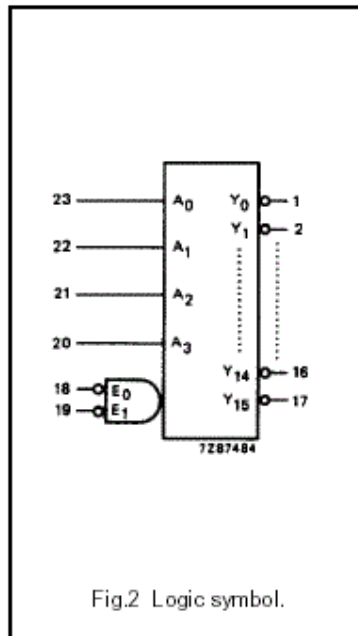


Fig.2 Logic symbol.

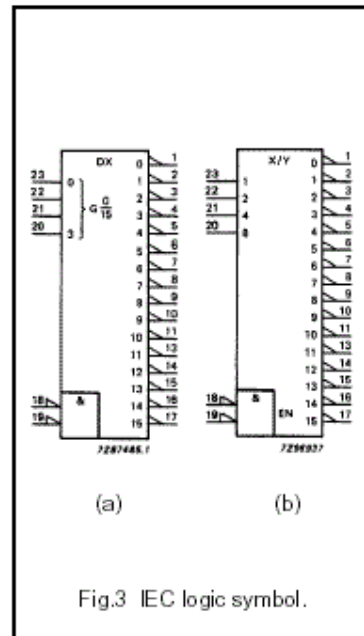


Fig.3 IEC logic symbol.

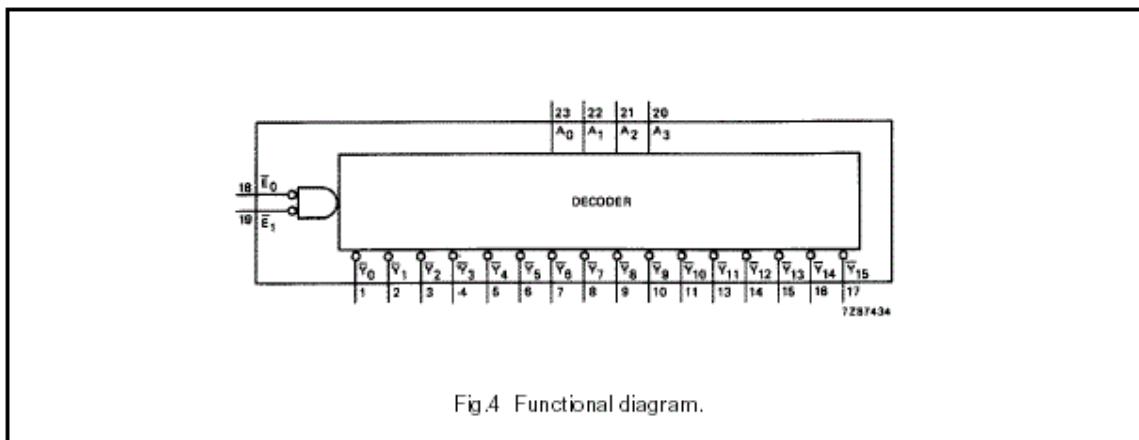


Fig.4 Functional diagram.

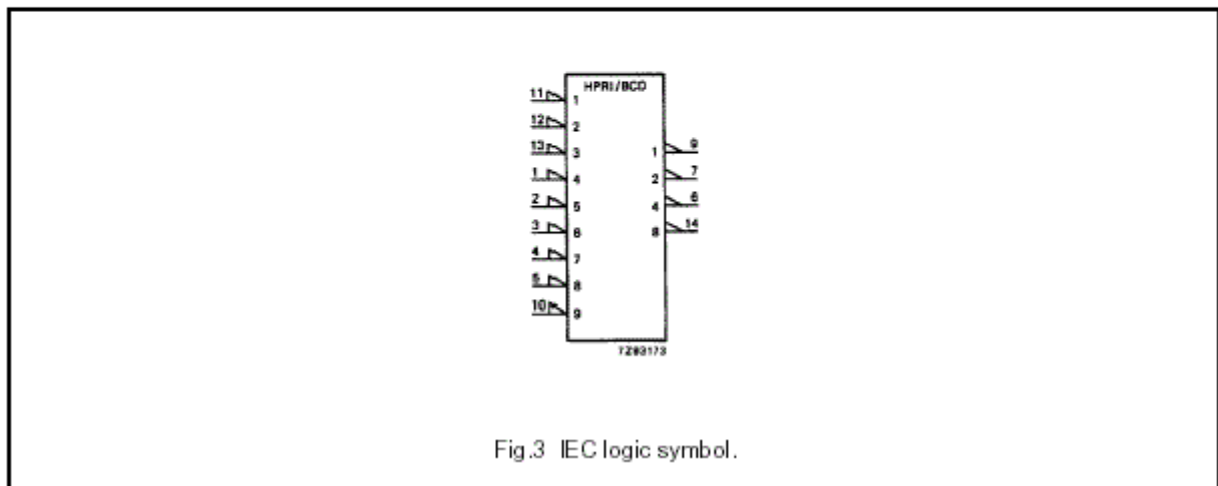
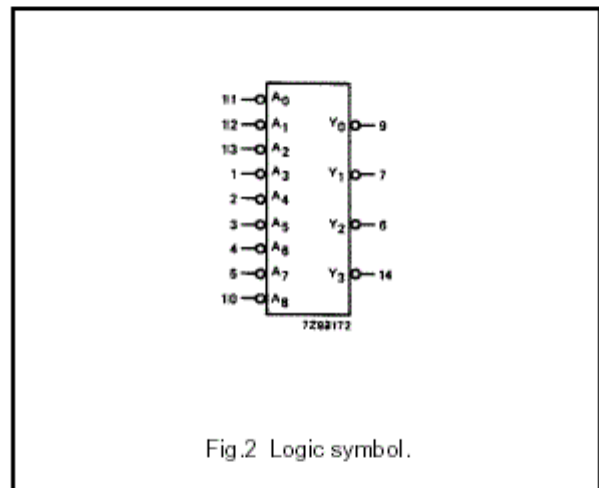
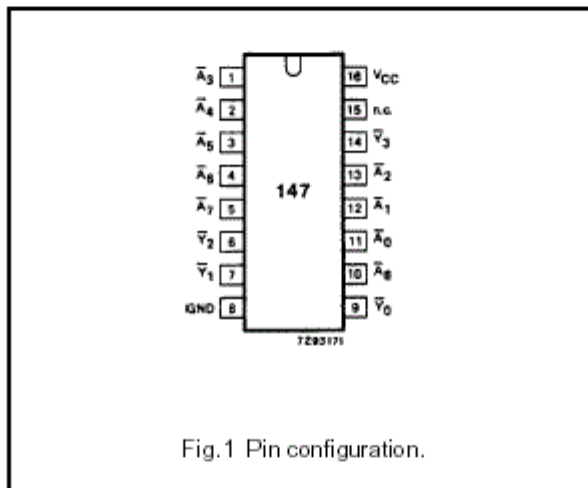
74HCT147 10 GİRİŞLİ 4 BİT ÇIKIŞLI YÜKSEK GİRİŞ ÖNCELİKLİ KODLAYICI

10-to-4 line priority encoder

74HC/HCT147

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
8	GND	ground (0 V)
9, 7, 6, 14	\bar{Y}_0 to \bar{Y}_3	BCD address outputs (active LOW)
11, 12, 13, 1, 2, 3, 4, 5, 10	\bar{A}_0 to \bar{A}_9	decimal data inputs (active LOW)
15	n.c.	not connected
16	V _{CC}	positive supply voltage



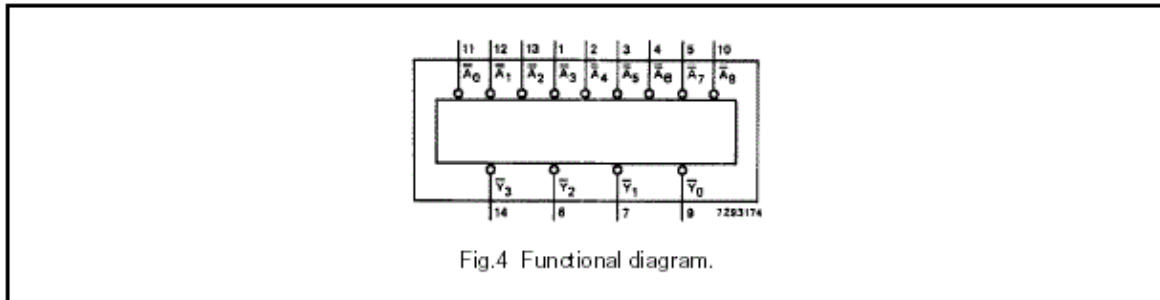


Fig.4 Functional diagram.

FUNCTION TABLE

INPUTS									OUTPUTS			
\bar{A}_0	\bar{A}_1	\bar{A}_2	\bar{A}_3	\bar{A}_4	\bar{A}_5	\bar{A}_6	\bar{A}_7	\bar{A}_8	\bar{Y}_3	\bar{Y}_2	\bar{Y}_1	\bar{Y}_0
H	H	H	H	H	H	H	H	H	H	H	H	H
X	X	X	X	X	X	X	X	L	L	H	H	L
X	X	X	X	X	X	X	L	H	L	H	H	H
X	X	X	X	X	L	H	H	H	H	L	L	L
X	X	X	X	L	H	H	H	H	H	L	L	H
X	X	X	L	H	H	H	H	H	H	L	H	L
X	X	L	H	H	H	H	H	H	H	H	L	L
X	L	H	H	H	H	H	H	H	H	H	L	H
L	H	H	H	H	H	H	H	H	H	H	H	L

Notes

1. H = HIGH voltage level
 L = LOW voltage level
 X = don't care

74LS151 8 GİRİŞLİ VERİ SEÇİCİ

8-input multiplexer

74ALS151

FEATURES

- 8-to-1 multiplexing
- On chip decoding
- Multi-function capability
- Complementary outputs
- See 74ALS251 for 3-State version

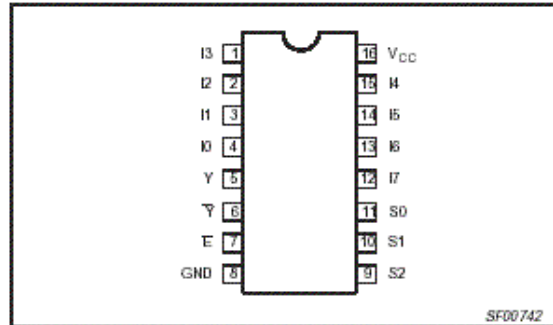
DESCRIPTION

The 74ALS151 is a logic implementation of a single 8-position switch with the switch position controlled by the state of three select (S0, S1, S2) inputs. True (Y) and complementary (\bar{Y}) outputs are both provided.

The enable (\bar{E}) is active-Low. When \bar{E} is High, Y output is Low and the \bar{Y} output is High regardless of all other inputs.

TYPE	TYPICAL PROPAGATION DELAY	TYPICAL SUPPLY CURRENT (TOTAL)
74ALS151	8.0ns	8.0mA

PIN CONFIGURATION



ORDERING INFORMATION

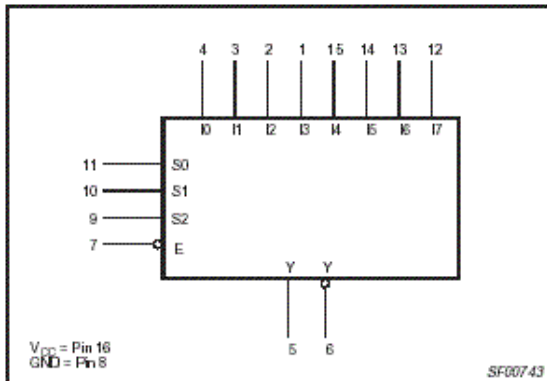
DESCRIPTION	ORDER CODE	DRAWING NUMBER
	COMMERCIAL RANGE V _{CC} = 5V ± 10%, T _{amb} = 0°C to +70°C	
16-pin plastic DIP	74ALS151N	SOT38-4
16-pin plastic SO	74ALS151D	SOT109-1

INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

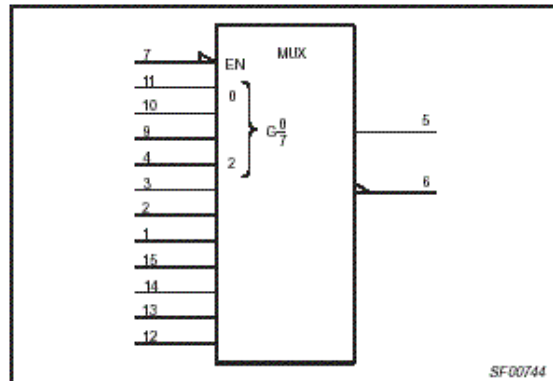
PINS	DESCRIPTION	74ALS (U.L.) HIGH/LOW	LOAD VALUE HIGH/LOW
I0 – I7	Data Inputs	1.0/1.0	20µA/0.1mA
S0 – S2	Select Inputs	1.0/1.0	20µA/0.1mA
\bar{E}	Enable Input (active-Low)	1.0/1.0	20µA/0.1mA
Y, \bar{Y}	Data outputs	130/240	2.6mA/24mA

NOTE: One (1.0) ALS unit load is defined as: 20µA in the High state and 0.1mA in the Low state.

LOGIC SYMBOL



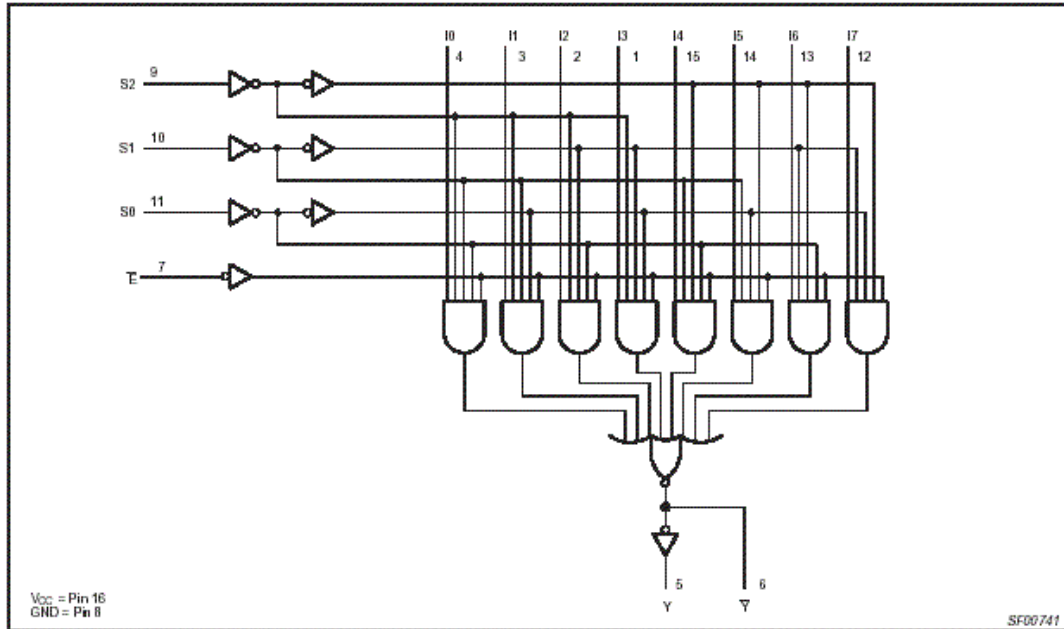
IEC/IEEE SYMBOL



8-input multiplexer

74ALS151

LOGIC DIAGRAM



FUNCTION TABLE

INPUTS				OUTPUTS	
S2	S1	S0	E	Y	\bar{Y}
X	X	X	H	L	H
L	L	L	L	10	$\bar{10}$
L	L	H	L	11	$\bar{11}$
L	H	L	L	12	$\bar{12}$
L	H	H	L	13	$\bar{13}$
H	L	L	L	14	$\bar{14}$
H	L	H	L	15	$\bar{15}$
H	H	L	L	16	$\bar{16}$
H	H	H	L	17	$\bar{17}$

H = High voltage level
 L = Low voltage level
 X = Don't care

74LS74A DUAL D TİPİ FLİP-FLOP

Dual D-type flip-flop with set and reset

74ALS74A

DESCRIPTION

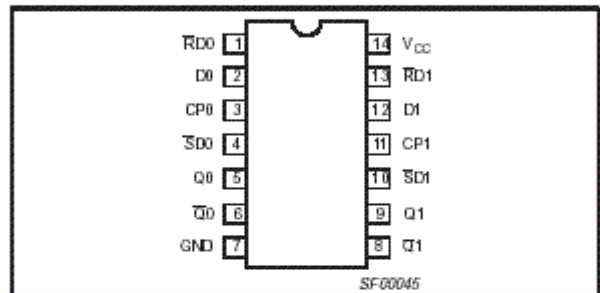
The 74ALS74 is a dual positive edge-triggered D-type flip-flop featuring individual data, clock, set, and reset inputs; also true and complementary outputs. Set (\overline{SD}) and reset (\overline{RD}) are asynchronous active-Low inputs and operate independently of the clock input. When set and reset are inactive (High), data at the D input is transferred to the Q and \overline{Q} outputs on the Low-to-High transition of the clock. Data must be stable just one setup time prior to the Low-to-High transition of the clock for predictable operation. Clock triggering occurs at a voltage level 1 and is not directly related to the transition time of the positive-going pulse. Following the hold time interval, data at the D input may be changed without affecting the levels of the output.

TYPE	TYPICAL f_{MAX}	TYPICAL SUPPLY CURRENT (TOTAL)
74ALS74A	150MHz	3.0mA

ORDERING INFORMATION

DESCRIPTION	ORDER CODE	DRAWING NUMBER
	COMMERCIAL RANGE $V_{CC} = 5V \pm 10\%$ $T_{amb} = 0^\circ C \text{ to } +70^\circ C$	
14-pin plastic DIP	74ALS74AN	SOT27-1
14-pin plastic SO	74ALS74AD	SOT108-1
14-pin plastic SSOP Type II	74ALS74ADB	SOT337-1

PIN CONFIGURATION

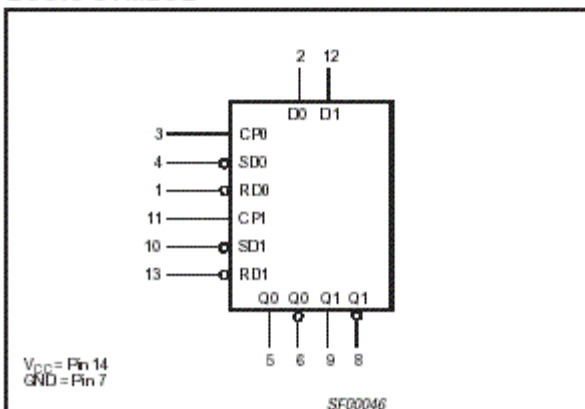


INPUT AND OUTPUT LOADING AND FAN-OUT TABLE

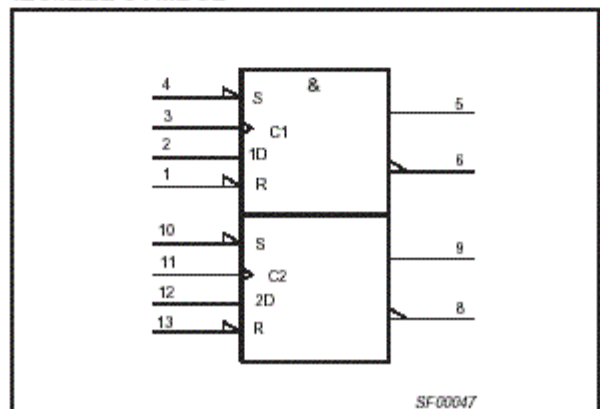
PINS	DESCRIPTION	74ALS (U.L.) HIGH/LOW	LOAD VALUE HIGH/LOW
D0, D1	Data Inputs	1.0/2.0	20µA/0.2mA
CP0, CP1	Clock Inputs (active rising edge)	1.0/2.0	20µA/0.2mA
$\overline{SD0}$, $\overline{SD1}$	Set Inputs (active-Low)	2.0/4.0	40µA/0.4mA
$\overline{RD0}$, $\overline{RD1}$	Reset Inputs (active-Low)	2.0/4.0	40µA/0.4mA
Q0, Q1, $\overline{Q0}$, $\overline{Q1}$	Data outputs	20/80	0.4mA/8mA

NOTE: One (1.0) ALS unit load is defined as: 20µA in the High state and 0.1mA in the Low state.

LOGIC SYMBOL



IEC/IEEE SYMBOL



74HCT75 DÖRTLÜ TUTUCU

Quad bistable transparent latch

74HC/HCT75

PIN DESCRIPTION

PIN NO.	SYMBOL	NAME AND FUNCTION
1, 14, 11, 8	$1\bar{Q}$ to $4\bar{Q}$	complementary latch outputs
2, 3, 6, 7	1D to 4D	data inputs
4	LE_{3-4}	latch enable input, latches 3 and 4 (active HIGH)
5	V_{CC}	positive supply voltage
12	GND	ground (0 V)
13	LE_{1-2}	latch enable input, latches 1 and 2 (active HIGH)
16, 15, 10, 9	1Q to 4Q	latch outputs

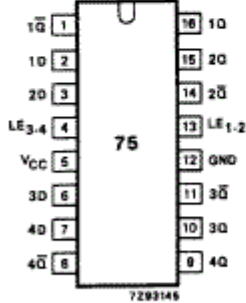


Fig.1 Pin configuration.

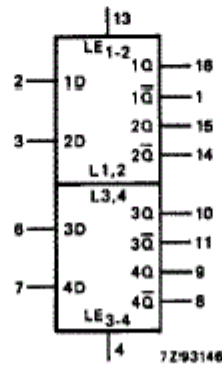


Fig.2 Logic symbol.

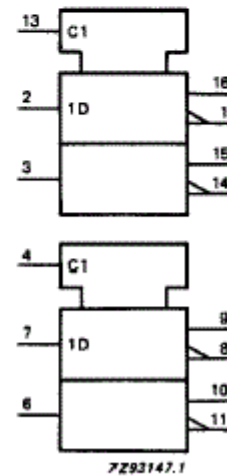


Fig.3 IEC logic symbol.

Octal buffer/line driver; 3-state

74AHC244; 74AHCT244

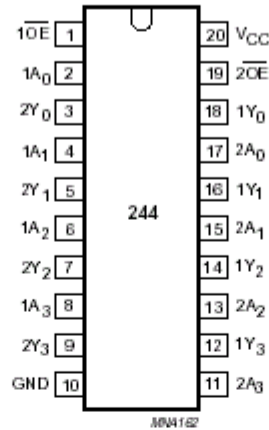


Fig.1 Pin configuration.

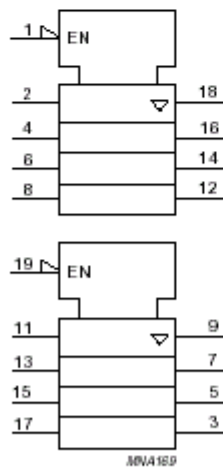


Fig.2 IEEE/IEC logic symbol.

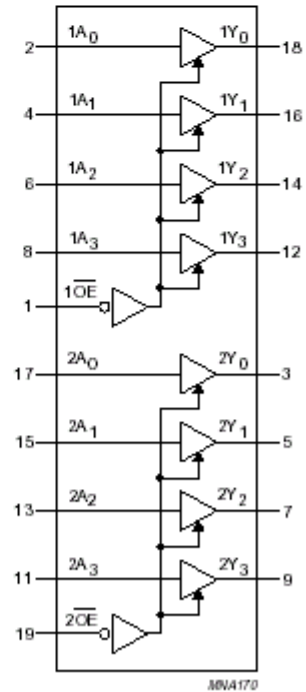


Fig.3 Logic diagram.

LM555 Timer Connection Diagram

