

T.C.
MİLLÎ EĞİTİM BAKANLIĞI



MEGEP

(MESLEKİ EĞİTİM VE ÖĞRETİM SİSTEMİNİN
GÜÇLENDİRİLMESİ PROJESİ)

ENDÜSTRİYEL OTOMASYON TEKNOLOJİLERİ

LOJİK DEVRELER-1

ANKARA 2007

Milli Eğitim Bakanlığı tarafından geliştirilen modüller;

- Talim ve Terbiye Kurulu Başkanlığının 02.06.2006 tarih ve 269 sayılı Kararı ile onaylanan, Mesleki ve Teknik Eğitim Okul ve Kurumlarında kademeli olarak yaygınlaştırılan 42 alan ve 192 dala ait çerçeve öğretim programlarında amaçlanan mesleki yeterlikleri kazandırmaya yönelik geliştirilmiş öğretim materyalleridir (Ders Notlarıdır).
- Modüller, bireylere mesleki yeterlik kazandırmak ve bireysel öğrenmeye rehberlik etmek amacıyla öğrenme materyali olarak hazırlanmış, denenmek ve geliştirilmek üzere Mesleki ve Teknik Eğitim Okul ve Kurumlarında uygulanmaya başlanmıştır.
- Modüller teknolojik gelişmelere paralel olarak, amaçlanan yeterliği kazandırmak koşulu ile eğitim öğretim sırasında geliştirilebilir ve yapılması önerilen değişiklikler Bakanlıkta ilgili birime bildirilir.
- Örgün ve yaygın eğitim kurumları, işletmeler ve kendi kendine mesleki yeterlik kazanmak isteyen bireyler modüllere internet üzerinden ulaşabilirler.
- Basılmış modüller, eğitim kurumlarında öğrencilere ücretsiz olarak dağıtılır.
- Modüller hiçbir şekilde ticari amaçla kullanılamaz ve ücret karşılığında satılamaz.

İÇİNDEKİLER

AÇIKLAMALAR	iii
GİRİŞ	1
ÖĞRENME FAALİYETİ-1	3
1. SAYI SİSTEMLERİ	3
1.1. İki Tabanlı (Binary) Sayı Sistemi	3
1.2. Basamak Değerleri	5
1.3. Binary Sayının Desimal Sayıya Çevrilmesi.....	6
1.4. Desimal Sayının Binary Sayıya Çevrilmesi.....	8
1.5. Heksadesimal (Onaltı Tabanlı) Sayı Sistemi	9
1.6. Heksadesimal Sayının Binary Sayıya Çevrilmesi	9
1.7. Binary Sayının Heksadesimal Sayıya Çevrilmesi	10
1.8. Heksadesimal Sayının Desimal Sayıya Çevrilmesi.....	10
1.9. Desimal Sayının Heksadesimal Sayıya Çevrilmesi.....	11
1.10. Oktal (Sekiz Tabanlı) Sayı Sistemi	11
1.11. Oktal Sayının Binary Sayıya Çevrilmesi	11
1.12. Binary Sayının Oktal Sayıya Çevrilmesi	12
1.13. Oktal Sayının Desimal Sayıya Çevrilmesi.....	12
1.14. Desimal Sayının Oktal Sayıya Çevrilmesi.....	12
1.15. BCD (Desimal Sayının Binary Olarak Kodlanması).....	13
1.16. ASCII Kod Sistemi (Amerikan Standart Kod).....	14
1.17. Kontrol Tuşları.....	16
1.18. Temel Lojik Kapılar	17
1.18.1. VE (AND) Kapısı (Mantıksal Çarpma).....	17
1.18.2. VEYA (OR) Kapısı (Mantıksal Toplama).....	18
1.18.3. DEĞİL (NOT) Kapısı (Tersleyici)	18
1.18.4. Deneybordunun Yapısı	19
UYGULAMA FAALİYETİ.....	21
ÖLÇME VE DEĞERLENDİRME	22
ÖĞRENME FAALİYETİ-2	23
2. SAYI SİSTEMLERİNDE İŞLEMLER.....	23
2.1. Binary Sayı Sisteminde Toplama İşlemi	23
2.2. Binary Sayı Sisteminde Çıkarma İşlemi.....	24
2.3. Binary Sayı Sisteminde Çarpma İşlemi.....	25
2.4. Binary Sayı Sisteminde Bölme İşlemi.....	25
2.5. Lojik İşlemlerde Sadeleştirme Metodları	26
2.6. Boolean Matematiği	30
2.7. Toplayıcı Devreler	32
UYGULAMA FAALİYETİ.....	36
ÖLÇME VE DEĞERLENDİRME	37
ÖĞRENME FAALİYETİ-3	38
3. DİĞER LOJİK KAPILAR.....	38
3.1. VE DEĞİL (NAND) Kapısı	38
3.2. VEYA DEĞİL (NOR) Kapısı	39
3.3. ÖZEL VEYA (EXOR) Kapısı	39
3.4. DE MORGAN Teoremi	40
3.5. RS Flip Flop.....	40

3.5.1. RS Flip Flop Fonksiyonu.....	41
3.5.3. RS Flip Flop'un Yapılışı	43
UYGULAMA FAALİYETİ.....	45
ÖLÇME VE DEĞERLENDİRME	46
ÖĞRENME FAALİYETİ-4	47
4. ENTEGRE (IC) KULLANIMI	47
4.1. TTL ve CMOS Entegreler	47
4.2. Entegre Girişleri.....	50
4.3. Led (Light Emitting Diode)	51
4.4. 7 Segment LED ve Sürücüsü	52
UYGULAMA FAALİYETİ.....	56
ÖLÇME VE DEĞERLENDİRME	57
MODÜL DEĞERLENDİRME.....	58
CEVAP ANAHTARLARI	59
KAYNAKÇA	61

AÇIKLAMALAR

KOD	522EE0163
ALAN	Endüstriyel Otomasyon Teknolojileri
DAL/MESLEK	Alan Ortak
MODÜLÜN ADI	Lojik Devreler 1
MODÜLÜN TANIMI	Bu modül lojik elemanları ve devreleri tanıtan ve bu elemanlarla lojik tasarım gerçekleştirme yaklaşımlarını geliştirmeye yönelik bilgi ve becerilerin verildiği öğretim materyalidir.
SÜRE	40/32
ÖN KOŞUL	Transistörlü devreler modülünü almış olmak.
YETERLİK	Temel lojik kapıları kullanmak.
MODÜLÜN AMACI	Genel Amaç Gerekli ortam sağlandığında temel lojik devrelerini hatasız olarak kurabileceksiniz. Amaçlar 1. Lojik entegrelerin çıkış kontrolünü doğru olarak yapabileceksiniz. 2. Toplayıcı devrelerini doğru olarak kurabilecek ve gerekli sayısal çıkışı elde edebileceksiniz. 3. Flip – flop devrelerini doğru olarak kurabilecek ve gerekli sayısal çıkışı elde edebileceksiniz. 4. Farklı tipteki lojik entegreleri doğru olarak kullanabileceksiniz.
EĞİTİM ÖĞRETİM ORTAMLARI VE DONANIMLARI	Ortam: Elektronik laboratuvarı Donanım: Lojik entegreler, elektronik elemanlar, lojik deney setleri, güç kaynağı, osilaskop, sinyal jeneratörü, giriş-çıkış işlemleri için hazırlanmış modüller, deneybordu, el takımları.
ÖLÇME VE DEĞERLENDİRME	Her faaliyetin sonunda ölçme soruları ile öğrenme düzeyinizi ölçeceksiniz. Araştırmalarla, grup çalışmaları ve bireysel çalışmalarla öğretmen rehberliğinde ölçme ve değerlendirmeyi gerçekleştirebileceksiniz.



GİRİŞ

Sevgili Öğrenci,

Lojik Devreler - 1 modülü ile farklı tipteki lojik entegreler, toplayıcı devreler ve flip – flop devreleri ile ilgili temel yeterlikleri kazanacaksınız. Tüm bu lojik yapıların elektronik ve otomasyon dünyasında çok önemli yerleri bulunmaktadır.

Bu modülü başarıyla tamamladığınızda elektronik ile bilgisayar alanında kullanılan ürünlerin iç yapısına ve çalışma sistemlerine ait bilgileri daha ayrıntılı değerlendirebileceksiniz. Sonraki yıllarda yapacağınız otomasyon çalışmalarında da bu bilgiler sayesinde endüstri alanında kullanıma uygun ve özel tasarımlar yapabileceksiniz.

Öğreneceğiniz temel bilgiler ile bu modülden sonraki Lojik Devreler – 2 modülüne hazırlık yapmış olacaksınız.

Uygulamaları bireysel olarak tekrar etmeye özen gösteriniz. Çünkü bu modülde lojik sistemlere giriş konuları ve uygulamaları bulunmaktadır. Tekrarlar sayesinde lojik devre tasarım becerinizi geliştirebileceksiniz.

Bu modüldeki uygulamaları yaparken değişik uygulamalar için farklı kitaplardan yararlanabilirsiniz. Böylece düşünme alanınızı genişletebilir ve tasarımlarınızda daha kullanılabilir modeller belirleyebilirsiniz.



ÖĞRENME FAALİYETİ-1

AMAÇ

Lojik entegre çıkış kontrolünü doğru olarak yapabileceksiniz.

ARAŞTIRMA

- Ø Dijital sistemlerin nerelerde kullanıldığını araştırınız.
- Ø Bilgisayarlarda ikili sayı sisteminin kullanılma nedenlerini araştırınız.

1. SAYI SİSTEMLERİ

Lojik sistemlerde dört çeşit sayı sistemi kullanılır.

- İki Tabanlı (Binary) Sayı Sistemi
- On Tabanlı (Desimal) Sayı Sistemi
- Onaltı Tabanlı (Heksadesimal) Sayı Sistemi
- Sekiz Tabanlı (Oktal) Sayı Sistemi

1.1. İki Tabanlı (Binary) Sayı Sistemi

Bilgisayarlar, iki tabanlı sayı sistemi mantığını kullanarak işlem yaparlar. Bu değerler farklı iki voltaj seviyesini gösterir. Lojik **sıfır** için “0” volt ve lojik **bir** için “5” volt ya da **3,3** volt’tur. Bu iki farklı voltaj seviyesi aslında herhangi iki farklı değerle gösterilebilir. Fakat genel olarak 0 ve 1 değerleri kullanılır ki, bu değerler de iki tabanlı sistemde kullandığımız değerlerdir. İkili sayı sisteminin işleyişi ile on tabanlı sayı sisteminin işleyişi aynıdır.

Desimal (on tabanlı) sayı sisteminde 0, 1, 2, 3, 4, 5, 6, 7, 8, 9 sembolleri olmak üzere on adet sembol kullanılır. Bu nedenle **onlu taban sistemi** de denir. Binary (iki tabanlı) sayı sisteminde ise yalnızca iki adet sembol kullanılır ve bunlar 0 ve 1’dir. Bu sisteme de iki adet sembol kullanıldığı için **ikili taban sistemi** de denir.

Şekil 1.1’de siyah daireler madeni parayı ifade etmektedir. Bu tabloda, madeni paraların miktarları desimal sayı sistemi için sol sütunda ve binary sayı sistemi için sağ sütunda sembolize edilmiştir. Örneğin dokuz madeni paranın olduğu satırda bu paraların miktarını desimal sayı sistemine göre belirtmek istediğimizde 9 sembolünü kullanmamız gerekir. Aynı miktarı binary sayı sistemi ile ifade etmek için 1001 sembolünü kullanmalıyız. 0 ve 1 sayıları desimal ve binary sayı sisteminde kullanılan ortak sembollerdir.

MADENİ PARA	DESİMAL SEMBOL	BİNARİ SEMBOL
Madeni para yok	0	0
=	1	1
= =	2	10
= = =	3	11
= = = =	4	100
= = = = =	5	101
= = = = = =	6	110
= = = = = = =	7	111
= = = = = = = =	8	1000
= = = = = = = = =	9	1001

Şekil 1.1: Sayı sembolleri

Binary sayı sisteminde 10 sayısını ‘bir sıfır’ şeklinde, 111 sayısını ‘bir bir bir’, 1001 sayısını ise ‘bir sıfır sıfır bir’ şeklinde söylememiz gerekir.

Binary sayı sistemindeki, her sembol **bit** (basamak) olarak adlandırılır. Her bit **0** ve **1** olmak üzere iki farklı değer alabilir. Buna göre **n** bitlik bir sayı 2^n değişik şekilde oluşturulabilir.

Binary sistemdeki 0 ve 1 sayıları elektrik devrelerinde sinyal var ya da yok anlamında kullanıldığı gibi, bir lambanın yanıp yanmadığını veya bir anahtarın kapalı mı yoksa açık mı olduğunu da gösterir.

Örnek 1.1: Aşağıda 4 bitlik sayılar verilmiştir. Bu sayılardaki her bitin bir led (lamba) olduğunu düşünerek ledlerin yanıp yanmayacağını belirleyiniz.

a. 1010

b. 1000

c. 1111

Çözüm 1.1:

	Binary sayılar	LED’ler
	1010	●○●○
	1000	●○○○
	1111	●●●●

Şekil 1.2: Ledler kullanılarak binary sayıların gösterimi

Örnek 1.2: Binary sayı sisteminde 4 bit, 8 bit ve 16 bit kullanılarak kaç farklı sayı yazılabilir?

Çözüm 1.2:

4 bitlik

$$2^4 = 16$$

8 bitlik

$$2^8 = 256$$

16 bitlik

$$2^{16} = 65536 \text{ değişik sayı yazılabilir.}$$

1.2. Basamak Değerleri

Şekil 1.3'te görüldüğü gibi, desimal sayı sisteminde desimal noktanın solundaki her basamak 10'un pozitif kuvvetleriyle belirtilir. Bir sayının büyüklüğü her basamaktaki rakamın sayı değerine ve basamak değerine bağlıdır. Örneğin şekilde görülen iki binlik, altı yüzlük, yedi onluk ve beş birliğin birleşmesinden $(2675)_{10}$ toplamı elde edilir.

← 10^4	10^3 (binler) 1000	10^2 (yüzler) 100	10^1 (onlar) 10	10^0 (birler) 1	Basamak değeri
←	2	6	7	5	

Şekil 1.3: Desimal (on tabanlı) sayı sistemi

$$\begin{aligned}\text{Sayının Değeri} &= (2 \times 10^3) + (6 \times 10^2) + (7 \times 10^1) + (5 \times 10^0) \\ &= (2 \times 1000) + (6 \times 100) + (7 \times 10) + (5 \times 1) \\ &= 2000 + 600 + 70 + 5 \\ &= (2675)_{10}\end{aligned}$$

Genel olarak; $a_{n-1} 10^{n-1} + \dots + a_2 10^2 + a_1 10^1 + a_0 10^0$ ifadesi ile bir sayının değeri hesaplanabilir.

10^{n-1} , ..., 10^2 , 10^1 , 10^0 basamaklarında taban olan 10 değeri, **radix** olarak adlandırılır.

Desimal (on tabanlı) sayı sisteminde basamakların geçişlerinde 10^n şeklinde olduğu gibi diğer sayı sistemlerinde de aynı kural geçerlidir. Örneğin binary (iki tabanlı) sayı sisteminde taban 10 yerine 2 olduğu için basamak değerleri $2^0=1$, $2^1=2$, $2^2=4$, $2^3=8$, $2^4=16$, $2^5=32$... şeklinde ifade edilir.

Örnek olarak; $(00101011)_2$ sayısının desimal sistemde $(43)_{10}$ a eşit olduğunu şu şekilde kanıtlayabiliriz:

$$\begin{aligned}&= 0 \times 128 + 0 \times 64 + 1 \times 32 + 0 \times 16 + 1 \times 8 + 0 \times 4 + 1 \times 2 + 1 \times 1 = (0 + 0 + 32 + 0 + 8 + 0 + 2 + 1) \\ &= (43)_{10}\end{aligned}$$

En soldaki binary rakam 'En Büyük Dereceli Rakam' (The Most Significant Bit-MSB), en sağdaki rakam ise 'En Küçük Dereceli Rakam' (The Least Significant Bit-LSB) olarak adlandırılır. Bu ifade Şekil 1.4'te MSB ve LSB olarak belirtilmiştir.

← 2^8	2^7	2^6	2^5	2^4	2^3	2^2	2^1	2^0	MSB	LSB
←	128	64	32	16	8	4	2	1		
	1	0	1	0	1	0	1	1		

Şekil 1.4: Binary (2 tabanlı) sayı sistemi

$$\begin{aligned}
\text{Sayının Değeri} &= (1 \times 2^7) + (0 \times 2^6) + (1 \times 2^5) + (0 \times 2^4) + (1 \times 2^3) + (0 \times 2^2) + (1 \times 2^1) + (1 \times 2^0) \\
&= (1 \times 128) + (0 \times 64) + (1 \times 32) + (0 \times 16) + (1 \times 8) + (0 \times 4) + (1 \times 2) + (1 \times 1) \\
&= 128 + 0 + 32 + 0 + 8 + 0 + 2 + 1 \\
&= (171)_{10}
\end{aligned}$$

Desimal			Binary			
100	10	1	8	4	2	1
		0				0
		1				1
		2			1	0
		3			1	1
		4		1	0	0
		5		1	0	1
		6		1	1	0
		7		1	1	1
		8	1	0	0	0
		9	1	0	0	1
	1	0	1	0	1	0
	1	1	1	0	1	1
	1	2	1	1	0	0
	1	3	1	1	0	1
	1	4	1	1	1	0
	1	5	1	1	1	1

Şekil 1.5: Binary sayı sistemi

Desimal sayı sistemi ve binary sayı sisteminde sayıların artışı aynı mantığa dayanır. **Şekil 1.5** binary sayının en yüksek değerine ulaştığında bir üst basamağa nasıl taşındığını ve rakamın son hâlini göstermektedir.

Şekil 1.5’da görüldüğü gibi binary sistem ile desimal sistem karşılaştırıldığında binary sistemde yalnızca iki rakam kullanılır. Bu yüzden küçük sayılardan büyük sayılara gidildikçe kullanılan rakam sayısı artar. Örneğin, desimal sistemde tek rakamla 8 sayısını ifade edebilirken binary sistemde aynı sayıyı $(1000)_2$ şeklinde ifade etmemiz gerekir.

1.3. Binary Sayının Desimal Sayıya Çevrilmesi

Dijital makinelerle çalışma esnasında binary kodlarını desimal sayılara çevirmek zorunda kalacaksınız. Örnek olarak $(1011.101)_2$ rakamı verilmiş olsaydı, bunu desimal olarak nasıl ifade ederdingiz?

1	0	1	1	1	0	1						
↓	↓	↓	↓	↓	↓	↓						
2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}						
8	+	0	+	2	+	1	+	0.5	+	0	+	0.125 = (11.625) ₁₀

Şekil 1.6: Binary sayının desimal sayıya çevrimi

Öncelikle rakamı binary olarak yazarsınız ve sol sütundan başlayarak basamak değerliklerini 2'nin üssü olarak belirledikten sonra birin karşılığındaki rakamları toplayarak desimal sayıya ulaşırsınız. Bu hesaplamayı Şekil 1.6'de görebilirsiniz.

2^6	2^5	2^4	2^3	2^2	2^1	2^0	2^{-1}	2^{-2}	2^{-3}	2^{-4}
64	32	16	8	4	2	1.	0.5	0.25	0.125	0.0625

Şekil 1.7: Binary sayının tam ve kesirli değerleri

Örnek 1.3: Şekil 1.8'deki LED'ler 8 bitlik bir sayının görüntüsü olarak kullanılmıştır. LED sönmük olduğunda binary olarak 0, ışık verirse 1 rakamına karşılık gelir.

Çözüm 1.3:

(a) $00000101_2 = 5$	○ ○ ○ ○ ○ ● ●
(b) $00001101_2 = 13$	○ ○ ○ ○ ● ● ●
(c) $10010001_2 = 145$	● ○ ○ ● ○ ○ ○ ●
(d) $10101010_2 = 170$	● ○ ● ○ ● ○ ● ○

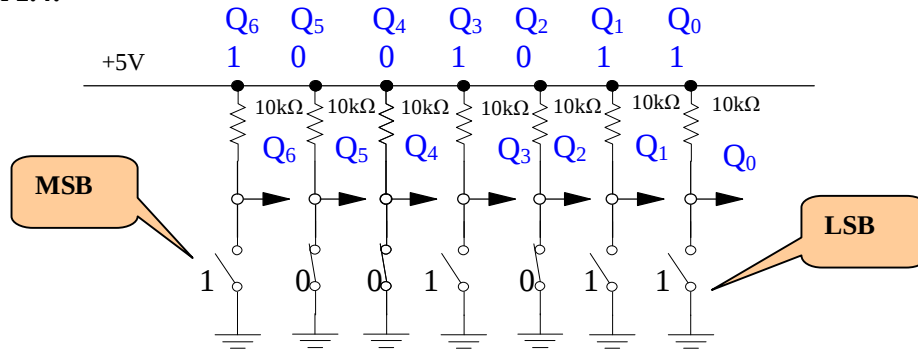
MSB LSB

● : ON
○ : OFF

Şekil 1.8: Binary sayılarda LED'lerin kullanımı

Örnek 1.4: Şekil 1.9'da 7 bitlik kayıt gösterilmiştir. Kayıt; bir binary bit gurubunu tutmak için kullanılan bir devredir. Anahtar kapalıysa birleştirilmiş olan Q çıkışı topraklanacak ve çıkış 0V olacaktır. Diğer taraftan anahtar açık ise $10k\Omega$ 'luk direnç vasıtası ile +5V'a bağlanmasından dolayı birleştirilmiş Q çıkışı yükseğe çekilecektir ve böylece çıkış +5V olacaktır. Bu kayıta tutulan net binary değeri ne olacaktır?

Çözüm 1.4:



Şekil 1.9: 7-Bit'lik butonlu kaydedici

1.4. Desimal Sayının Binary Sayıya Çevrilmesi

Çoğu zaman dijital elektronik cihazlarla çalışırken binary sayıları desimale çevirmek durumunda kalırız. Örneğin desimal olan 45 sayısını binary sayıya çevirmek istediğimizde uygulamamız gereken kural, sayıyı sürekli 2'ye bölmektir. 45'i 2'ye böldüğümüzde bölüm 22, kalan 1'dir. Daha sonra 22'yi yine 2'ye bölersek bölüm 11, kalan ise 0 olur. 11'i tekrar 2'ye böldüğümüzde bölüm 5, kalan 1'dir. 5'in 2'ye bölümünde bölüm 2, kalan 1 olur ve en sonunda 2'yi 2'ye böldüğümüzde bölüm 1 kalan 0'dır. Baştan itibaren bulduğumuz kalanları sonda en başa doğru sırayla yazıp en son bulduğumuz bölümü de eklersek $(101101)_2$ binary sayısını elde ederiz.

Örnek 1.5: Desimal olarak verilen 46,3125 sayısını binary sayıya çeviriniz.

Tam kısım			Ondalık kısım		
Desimal sayı	Bölen	Kalan	Desimal sayı	Çarpan	Sonuç
Tam kısım			Ondalık kısım		
46	: 2=>	0	0,3125	×2 =	0,6250 0
23	: 2=>	1	0,625	×2 =	1,2500 1
11	: 2=>	1	0,2500	×2 =	0,5000 0
5	: 2=>	1	0,5000	×2 =	1,0000 1
2	: 2=>	0			
1	=>	1			
$(46)_{10} = (1\ 0\ 1\ 1\ 1\ 0)_2$			$(0,3125)_{10} = (0\ 1\ 0\ 1)_2$		

Şekil 1.10: Desimal sayının binary sayıya çevrilmesi

Çözüm 1.5: Tam kısım ve ondalık kısım olarak iki bölümde ayrı ayrı işlem yapılır.

Tam kısım: Sürekli 2'ye bölünür. 2'ye bölünemeyecek hâle geldikten sonra kalanlar sondan başa doğru yazılarak binary sayının tam kısmı oluşturulmuş olur.

Ondalık kısım: Sürekli 2 ile çarpılır. Her çarpma işleminden sonra elde edilen sayının tam kısmı bir sonraki çarpım işleminde ihmal edilir. Sonuç 1,000 oluncaya kadar bu işleme devam edilir. Bire ulaşıldıktan sonra çarpım sonuçlarının ihmal edilen tam kısımları üstten başlayarak yazılır ve binary sayının ondalık kısmı oluşturulur. **Şekil 1.10'**dan takip edebilirsiniz.

SONUÇ: $(46,3125)_{10} = (101110,0101)_2$

1.5. Heksadesimal (Onaltı Tabanlı) Sayı Sistemi

Binary sayı sistemi bilgisayarın anladığı tek sayı sistemidir. Bilgisayardan girdiğiniz tüm yazı, sayı ve işlemler binary sayıya çevrilerek bilgisayar tarafından algılanır. Fakat binary sayı sisteminde yalnızca 2 rakam olduğu için büyük sayıları ifade etmek oldukça fazla rakamla mümkün olur. Örneğin desimal sayı olan 202 sayısını 3 rakam kullanarak ifade edebilirken aynı sayıyı binary sayı sisteminde 11001010 şeklinde yazarız ki bu bizim 8 rakam kullandığımızı gösterir.

Bilgisayar üreticileri bu sorunu heksadesimal sayı sistemini geliştirerek çözmüşler. Bu sayı sisteminde sayılar daha az rakam kullanılarak ifade edilebilmektedir. Ayrıca bu sayı sisteminin ayrı bir üstünlüğü de binary sayıya geçiş ve binary sayıdan heksadesimal sayıya geçişin kolay olmasıdır. Heksadesimal sayı sisteminde 16 sembol kullanılır. Bunlar 0,1,2,3,4,5,6,7,8,9,A,B,C,D,E,F dir.

‘A’ harfi 10 sayısına, ‘B’ harfi 11 sayısına , ‘C’ harfi 12 sayısına, ‘D’ harfi 13 sayısına, ‘E’ harfi 14 sayısına ve ‘F’ harfi de 15 sayısına karşılık gelir.

Bunun yanında heksadesimal sayıyı yazarken diğer sistemlerde de olduğu gibi tabanı belirtmek zorundayız. Bu belirtme örneğin; (1A)₁₆ şeklinde olabileceği gibi 1Ah şeklinde de olabilir.

Heksadesimal sayılar 4’er bit şeklinde ayrılırlar. Örneğin 0000 0101 binary sayıyı heksadesimal olarak ifade ederken ilk dört bitin değeri sıfır olsa dahi 05h olarak yazmak zorundayız. Örnek olarak heksadesimalde (F)₁₆ sayısı, binary sistemde (1111)₂ olarak gösterilir. Ya da (9)₁₆ sayısı (1001)₂ dir.

Diğer bir örnek ise (4C)₁₆ rakamı 8 bit binary rakamı ifade eder. Binary yazılımı ise (0100 1100)₂ dir. Heksadesimal sayı sistemi mikroişlemci temel uygulamalarında binary sayı sistemindeki 8, 16, 32, 64 biti gösterebilmek için sıklıkla kullanılır.

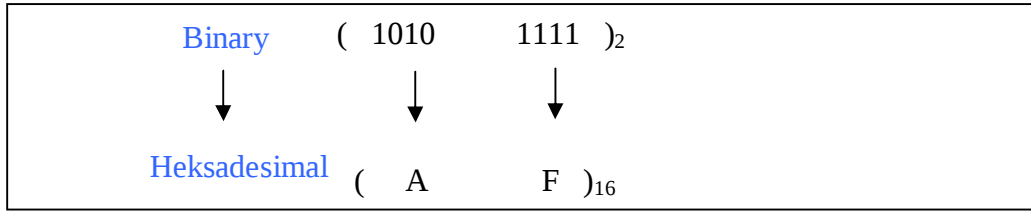
1.6. Heksadesimal Sayının Binary Sayıya Çevrilmesi

Mikrobilgisayar ve mikroişlemciler ile çalışılırken heksadesimal sayının binary sayıya çevrilmesi kadar binary sayının da heksadesimale çevrilmesi gerekir. Örneğin **Şekil 1.11**’de heksadesimal olarak verilen (D6)₁₆ sayısının binary çevrimi gösterilmiştir. Her bir heksadesimal rakam binaryde 4 bitlik bir yer kaplar. Heksadesimal D rakamının binary de karşılığı 1101 ve diğer heksadesimal rakam olan 6’nın binary karşılığı 0110 dir. Bu iki grup birleştirilerek (D6)₁₆ = (11010110)₂ elde edilir.

1.7. Binary Sayının Heksadesimal Sayıya Çevrilmesi

Şimdi ise bir önceki konumuzun tersine binary olarak verilmiş bir sayının heksadesimale çevrilişini göreceğiz. $(10101111)_2$ sayısını nasıl heksadesimal olarak çevirdiğimizi **Şekil 1.11**'den takip edebilirsiniz. Öncelikle sayı gurubunu 4 bitlik guruplara bölmek ve daha sonra onların karşılıklarını bulmak gerekmektedir.

1010 olan ilk gurubun karşılığı; $0.2^0 + 1.2^1 + 0.2^2 + 1.2^3 = A$ ve 1111 olan ikinci grubun karşılığı $1.2^0 + 1.2^1 + 1.2^2 + 1.2^3 = F$ dir. Böylece $(10101111)_2 = (AF)_{16}$ eşitliği elde edilir.

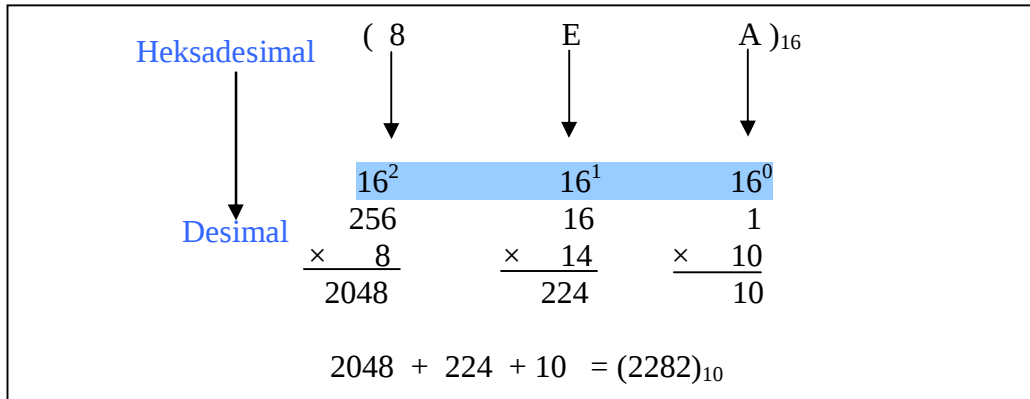


Şekil 1.11: Binary sayıyı heksadesimal sayıya çevirme

1.8. Heksadesimal Sayının Desimal Sayıya Çevrilmesi

$(8EA)_{16}$ sayısını desimal sayıya çevirme yöntemi de yine aynıdır. A rakamının desimalde 10'a ve E rakamının da 14'e karşılık geldiğini önceden biliyorduk. Bu durumda 8EA sayısı desimal sayıya;

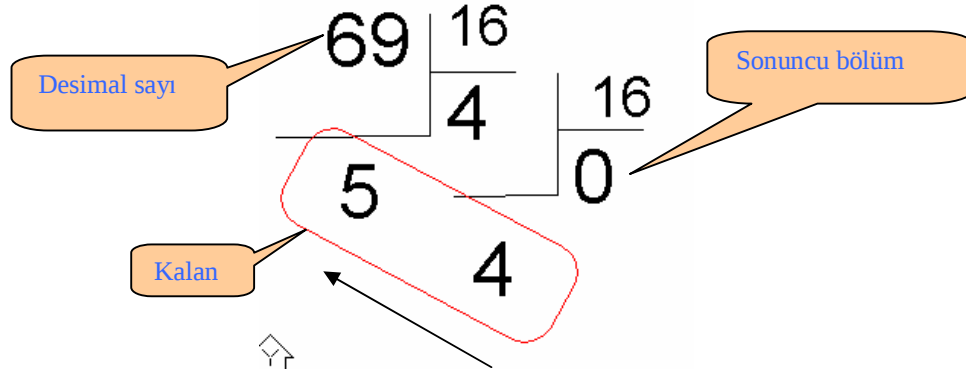
$8.16^2 + E.16^1 + A.16^0 = (2282)_{10}$ şeklinde çevrilir. Bu hesaplamayı **Şekil 1.12**'den de takip edebilirsiniz.



Şekil 1.12: Heksadesimal sayıyı desimal sayıya çevirme

1.9. Desimal Sayının Heksadesimal Sayıya Çevrilmesi

Desimal olarak verilen sayı sürekli 16'ya bölünür kalan 0 oluncaya dek devam edilir. Bu kalanlar sondan başlanarak yazıldığında heksadesimal sayı elde edilir. Şekil 1.13'te 69 sayısının heksadesimale çevrilmesi gösterilmiştir. 69'u 16'ya böldüğümüzde bölüm 4, kalan 5 tir. 4'ün 16'ya bölümünde, bölüm 0 kalan 4 tür. Kalanları sondan başlayarak yazarsak heksadesimal (45)₁₆ sayısı elde edilir. $(69)_{10} = (45)_{16}$



Şekil 1.13: Desimal sayıyı heksadesimal sayıya çevirme

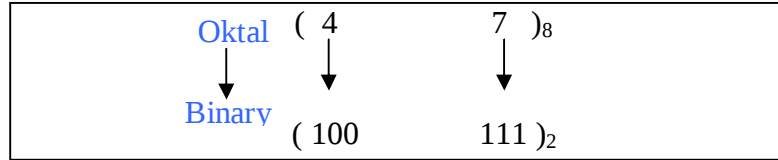
1.10. Oktal (Sekiz Tabanlı) Sayı Sistemi

Kimi eski tip bilgisayarlarda bu sayı sistemi kullanılmaktadır. Oktal sayı sisteminde 8 adet sembol vardır. Bunlar, 0 1 2 3 4 5 6 7'dir. Bu sembollerin dışında sembol kullanılmaz. Taban da 8 olarak belirtilir. Oktal sayı sisteminin avantajı doğrudan 3 bite ayrılan binary rakamların kolaylıkla çevrilebilmesidir.

1.11. Oktal Sayının Binary Sayıya Çevrilmesi

Bilgisayar sistemlerinde oktal sayıları binary sayıya çevirirken her rakam 3 bitlik binary sayı ile ifade edilir. Örneğin $(47)_8$ sayısını çevirirken oktal sayı sistemindeki 4 rakamı binary de 100'a ve 7 rakamı 111'e karşılık gelir ve sonuç,

$(47)_8 = (100111)_2$ olur.



Şekil 1.14: Oktal sayıyı sayıya çevirme

1.12. Binary Sayının Oktal Sayıya Çevrilmesi

Şekil 1.15'te $(101011110)_2$ sayısının oktal sayıya çevrilmesi görülmektedir. Binary rakamlar öncelikle binary noktadan sola doğru üçlü guruplara bölünür ve bu gruplar oktal sayılara çevrilir.

Binari	(	101	011	110	) ₂
		↓	↓	↓	
Oktal	(	5	3	6	) ₈

Şekil 1.15: Binary sayıyı oktal sayıya çevirme

1.13. Oktal Sayının Desimal Sayıya Çevrilmesi

Şekil 1.16'da $(573)_8$ oktal sayısını desimal sayıya çevirme yöntemi gösterilmiştir. Bunu örneğimizde takip edersek; üç tane 8^1 in, yedi tane 8^2 nin, beş tane 8^3 ün toplanması ile $(379)_{10}$ sayısına ulaşılmış olur.

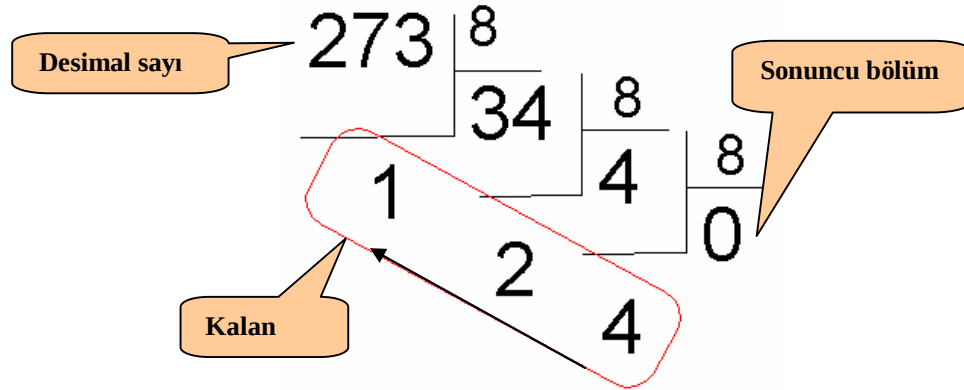
Oktal	(	5	7	3	) ₈
		↓	↓	↓	
		8^2	8^1	8^0	
		64	8	1	
Desimal		$\times 5$	$\times 7$	$\times 3$	
		320	56	3	
		$320 + 56 + 3 = (379)_{10}$			

Şekil 1.16: Oktal sayıyı desimal sayıya çevirme

1.14. Desimal Sayının Oktal Sayıya Çevrilmesi

Şekil 1.17'de desimal 273 sayısının oktal sayıya çevrilmesi gösterilmektedir. Buradaki yöntem yine sayıyı bölüm 0 olana kadar 8'e bölmektir. 273 sayısının 8'e bölümünden bölüm 34, kalan 1 dir.

34 sayısının 8'e bölümünden de bölüm 4, kalan 2'dir. 4'ün 8'e bölünmesinden bölüm 0, kalan 4'tür. Sondan başlayarak kalanları yazarsak oktal sayı sistemindeki $(421)_8$ sayısı bulunur. Bu durumda $(273)_{10}=(421)_8$ olur.



Şekil 1.17: Desimal sayıyı oktal sayıya çevirme

1.15. BCD (Desimal Sayının Binary Olarak Kodlanması)

Binary sayı sistemi desimal sayı sistemine göre daima daha az kullanışlıdır. Örneğin, $(11110011100)_2$ sayısının desimal $(1948)_{10}$ sayısına çevrilmesi desimalden binary sayıya çevirmekten daha az vaktimizi alır. Tasarımcılar da bu dezavantajı düşünerek daha pratik olması için desimal rakamları binary olarak belirlemişler ve BCD 'Binary Coded Decimal' olarak adlandırmışlardır. Bu kodlama sayesinde kol saati, hesap makineleri, monitör ve klavye gibi desimal tabanlı girdilerin binary koda dönüştürülmesi oldukça kısa zaman almış ve daha kullanışlı hâle gelmiştir.

Desimal sistemdeki her rakam BCD kodda 4 bit olarak ifade edilir. Aşağıdaki örnekte desimal 1948 sayısı BCD kodda 0001 1001 0100 1000 a dönüşür. Her 4 bitlik grup desimalde bir basamağa karşılık gelir. Böylece $(1948)_{10} = (0001\ 1001\ 0100\ 1000)_{BCD}$ elde edilmiş olur.

1				9				4				8			
8	4	2	1	8	4	2	1	8	4	2	1	8	4	2	1
0	0	0	1	1	0	0	1	0	1	0	0	1	0	0	0
Binler BCD Rakam				Yüzler BCD Rakam				Onlar BCD Rakam				Birler BCD Rakam			

Şekil 1.18: Desimal sayının BCD kod sistemi ile gösterilmesi

1.16. ASCII Kod Sistemi (Amerikan Standart Kod)

Genellikle programlama dilleri, klavyeden yazdığımız karakter ve sembolleri bilgisayarın kendi diline (binary) çevirir.

Bilgi aktarımı için Amerikan Standart Kod (ASCII), mikrobilgisayarlarda bilgi aktarımında oldukça geniş yer almaktadır. ASCII kodu klavyeden aldığı bilgileri yazıcı ve bilgisayar ekranına aktarma sırasında 7-bit kod kullanır. Klavyeden girilen her bir harf, sembol veya rakam karşılığında bir ASCII kod vardır ve bu kodlara çevrildikten sonra bilgisayar kendi hafızasında karşılığı olan istediğimiz harfi ya da rakamı ekrana yansıtır.

ASCII kod sisteminde 128 karakter standartlaştırılmıştır ve bunlar 4 grupta toplanır.

Birinci grupta 32 karakter bulunur ve fonksiyon tuşlarının çalışması ile ilgili kodlardır. Bunlara kontrol karakterleri denir. Çünkü bu karakterler, ekran sembollerinden ziyade çeşitli ekran ve yazıcı kontrol işlemlerini yürütürler. Örneğin, kursörün sol tarafa taşınması (ASCII kodu 0Dh)

İkinci grup 32 karakter çeşitli noktalama işaretlerini, sayıları ve özel karakterleri içerir. Örneğin, boşluk karakteri (ASCII kodu 20h)

Üçüncü grup ASCII karakterleri klavyedeki üst seviye karakterleri kapsar. 26 tanesi alfabetik karakterler, geri kalan 6'sı ise özel sembollerdir.

Dördüncü grup ASCII karakterleri, klavyedeki alt seviye karakterleri, 5 özel sembol ve diğer kontrol karakterleridir (DEL). Küçük karakterler 61h ve 7Ah arasındaki ASCII kodlarını kullanır.

Üst seviye karakterler Caps Lock veya Shift tuşları ile elde edilir. Alt seviye karakterlerde bu tuşlar kullanılmaz. Alt seviye karakterlerle, üst seviye karakterler arasında 1 bitlik pozisyon farkı vardır. Bu kodun değiştiği tek yer ASCII kod tablosunun 6'ncı bitidir. Üst seviye karakterleri 6'ncı bitte daima sıfır, alt seviye karakterleri ise 6'ncı bitte daima birdir.

7'nci bit	6'nci bit	Grup
0	0	Kontrol karakterleri
0	1	Rakamlar ve noktalama
1	0	Üst seviye ve özel
1	1	Alt seviye ve özel

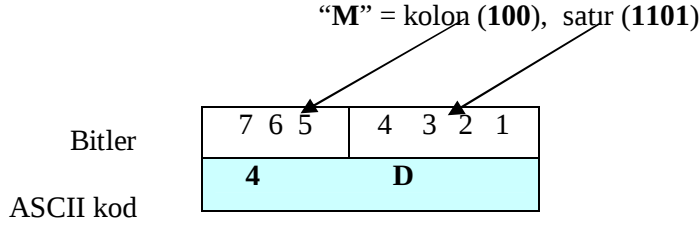
Şekil 1.19: Alt ve üst seviye bitleri

KOLON BİTLERİ															
HEX								0	1	2	3	4	5	6	7
								0	0	0	0	1	1	1	1
								0	0	1	1	0	0	1	1
								0	1	0	1	0	1	0	1
SATIR BİTLERİ	HEX	Bit 7	Bit 6	Bit 5	Bit 4	Bit 3	Bit 2	Bit 1	Kontrol						
									Fonksion						
	0				0	0	0	0	NUL	DLE	SP	0	@	P	p
	1				0	0	0	1	SOH	DC1	!	1	A	Q	q
	2				0	0	1	0	STX	DC2	“	2	B	R	r
	3				0	0	1	1	ETX	DC3	#	3	C	S	s
	4				0	1	0	0	EOT	DC4	\$	4	D	T	t
	5				0	1	0	1	ENQ	NAK	%	5	E	U	u
	6				0	1	1	0	ACK	SYN	&	6	F	V	v
	7				0	1	1	1	BEL	ETB	'	7	G	W	w
	8				1	0	0	0	BS	CAN	(	8	H	X	x
	9				1	0	0	1	HT	EM	)	9	I	Y	y
	A				1	0	1	0	LF	SUB	*	:	J	Z	z
	B				1	0	1	1	VT	ESC	+	;	K	[	k
	C				1	1	0	0	FF	FS	,	<	L	\	l
	D				1	1	0	1	CR	GS	-	=	M	]	m
	E				1	1	1	0	SO	RS	.	>	N	^	n
	F				1	1	1	1	SI	US	/	?	O	_	o
															DEL

Şekil 1.20: ASCII Kodlar

Şekil 1.20’de ASCII kodlarının tümü verilmiştir. Bu kodlamada klavye üzerindeki rakamlar, kontrol tuşu ile kullanılan karakterler, noktalama işaretleri, harflerin hepsinde kullanılır. Örneğin, 7 bit ASCII kodu olan 1111111’e klavyedeki DEL tuşu karşılık gelir. Yani DEL tuşuna bastığımızda bilgisayarın CPU’su bu komutu 1111111 olarak algılar. Diğer karakterleri Şekil 1.20’den görebilirsiniz.

Örnek olarak ‘M’ harfinin ASCII kodunu bulalım: Tablodan önce M’nin üstündeki kolona (4.kolon) daha sonra yanındaki satıra (D satırı) bakıp okuduğumuz değerleri yazarsak istediğimiz ‘M’ = 100 1101 kodunu bulmuş oluruz ve CPU’ya 100 1101 gönderilmiş olur.



1.17. Kontrol Tuşları

Kimi zaman aşağıdaki özel işlemli tuşları kullanmak zorunda kalırız. Fakat dikkat edilmesi gereken bir nokta, bu tabloda açıklaması verilen tuşlar kimi özel işlevli bilgisayarlarda farklı görevler üstlenmiş olabilirler. Tabii tüm bilgisayarlarda oldukça sık kullanılan DEL (delete), SP (space), BEL (bell), BS (back space), FD (line feed) tuşları ortak tuşlardır.

NUL	Null	DLE	Data Link Escape
SOH	Start of Heading	DC1	Device Control 1
STX	Start of Text	DC2	Device Control 2
ETX	End of Text	DC3	Device Control 3
EOT	End of Transmission	DC4	Device Control 4
ENQ	Enquiry	NAK	Negative Acknowledge
ACK	Acknowledge	SYN	Synchronous Idle
BEL	Bell(audible signal)	ETB	End of Transmission Block
BS	Backspace	CAN	Cancel
HT	Horizontal Tabulation (pouched card skip)	EM	End of Medium
LF	Line Feed	SUB	Substitute
VT	Vertical Tabulation	ESC	Escape
FF	Form Feed	FS	File Separator
CR	Carriage Return	GS	Group Separator
SO	Shift Out	RS	Record Separator
SI	Shift In	US	Unit Separator
SP	Space (blank)	DEL	Delete

Şekil 1.20: Kontrol tuşları

Örnek 1.7: 'Okul' kelimesini ASCII kodu ile belirtiniz.

Çözüm 1.7:

O=1001111= 4Fh

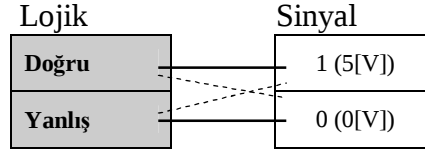
k=1101011= 6Bh

u=1110101= 75h

l=1101100 = 6Ch şeklinde bulunur.

1.18. Temel Lojik Kapılar

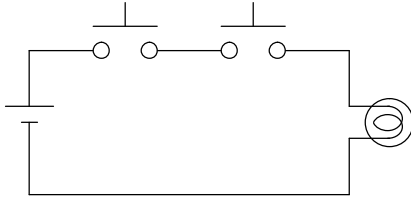
Dijital devrelerde genellikle lojik 0 yanlış, lojik 1 ise doğruyu ifade eder. Bu şekilde kullanılması pozitif lojik olarak adlandırılır. Negatif lojikte ise lojik 1 yanlış, lojik 0 doğru olarak kabul edilir.



Şekil 2.1

1.18.1. VE (AND) Kapısı (Mantıksal Çarpma)

Mantıksal çarpma işlemini Şekil 2.2. ile daha iyi açıklayabiliriz. Burada birbirine seri bağlı iki anahtar mevcuttur.



Şekil 2.2

Şekilde anahtarlar giriş, lamba ise çıkış elemanlarıdır.

Giriş için;

Eğer anahtar basılı ise Doğru (1), aksi takdirde Yanlış (0)'dır.

Çıkış için;

Eğer lamba yanıyorsa Doğru (1), yanmıyorsa Yanlış (0) olmaktadır.

Devrenin buna göre işlevini Şekil 2.3. (a)'daki gibi kolayca tablo şeklinde yazabiliriz. Bu tabloda girişin mümkün olabilecek her farklı durumu giriş sütununa belirtilmiştir.

Giriş		Çıkış
0	0	0
0	1	0
1	0	0
1	1	1

(a) Doğruluk tablosu

(b) Mantıksal ifade

(c) Devre sembolü

Şekil 2.3: VE kapısı

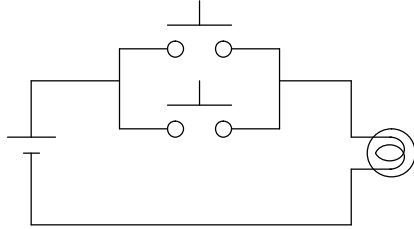
Mantıksal çarpma ifadesi doğruluk tablosunun sonucundan gelmektedir. VE olarak adlandırılır. Giriş değerlerinin çarpımı çıkışa eşittir.

A VE B doğru ise X de doğrudur.

Şekil 2.3 (b) ve (c), VE'nin mantıksal ifadesini ve sembolünü göstermektedir.

1.18.2. VEYA (OR) Kapısı (Mantıksal Toplama)

Mantıksal toplama işlemini Şekil 2.4'deki devre ile daha iyi açıklayabiliriz. Burada birbirine paralel iki anahtar mevcuttur.



Doğruluk tablosu, mantıksal ifade ve devre sembolü Şekil 2.5'te gösterilmiştir.

Şekil 2.4

Giriş		Çıkış
A	B	X
0	0	0
0	1	1
1	0	1
1	1	1

(a) Doğruluk tablosu

(b) Mantıksal ifade

(c) Devre sembolü

Şekil 2.5: VEYA kapısı

Doğruluk tablosunda giriş değerlerinin toplamının çıkışa eşit olduğu görülür. Buna göre VEYA kapısının 'mantıksal toplama' işlemi yaptığını söyleyebiliriz.

A VEYA B doğru ise X de doğrudur.

1.18.3. DEĞİL (NOT) Kapısı (Tersleyici)

Değil kapısının işlevini daha iyi anlamak için Şekil 2.6'yı inceleyiniz. Bu devrede anahtar normalde kapalıdır. Mekanik temas olduğu zaman (Doğru mantık) kontakları açılır. Mekanik temasın olması mantık 1 olarak algılanmalıdır. Doğruluk tablosu, mantıksal ifade ve devre sembolü Şekil 2.7'te gösterilmiştir. Giriş sinyalini tersine çevirdiği için çevirici yada invertör de denir.

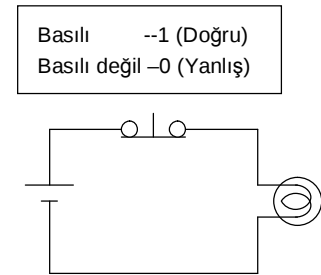
Giriş	Çıkış
A	X
0	1
1	0

(a) Doğruluk tablosu

(b) Mantıksal ifade

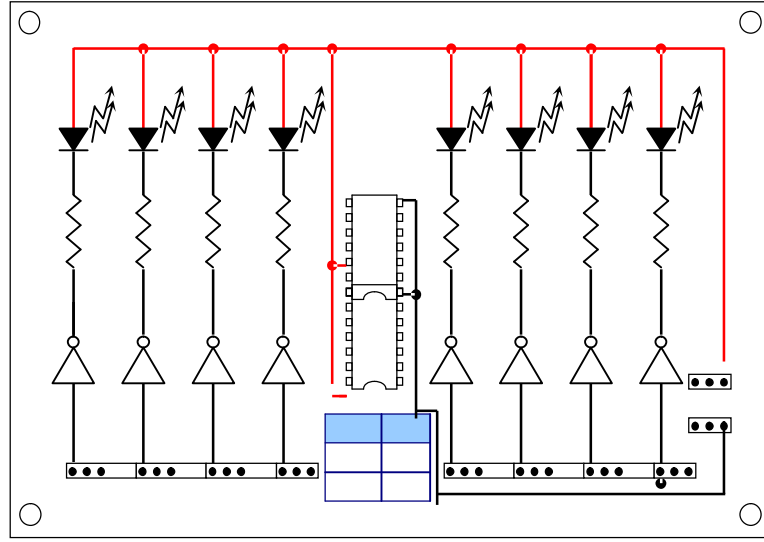
(c) Devre sembolü

Şekil 2.7: DEĞİL kapısı



Şekil 2.6

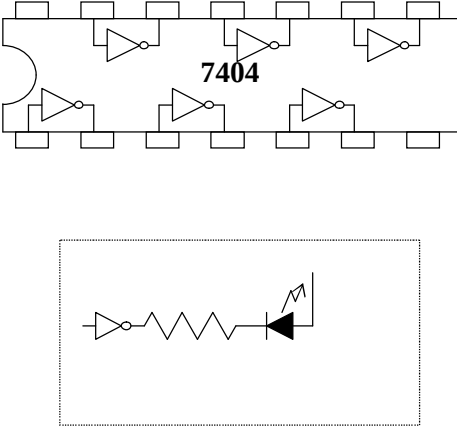




Şekil 2.10: Çıkış Ünitesi

UYGULAMA FAALİYETİ

İşlem basamaklarında belirtilen adımları izleyerek uygulama faaliyetini yapınız. Bu uygulama faaliyetiyle deneybordu kullanarak lojik kapı entegresini kontrol edebileceksiniz.

İşlem Basamakları	Öneriler
Ø Deneybordu üzerinde Vcc ve GND bağlantılarını yapınız.	Ø Bağlantı için zil teli kullanınız.
Ø Giriş – çıkış modülünde Vcc ve GND bağlantılarını yapınız.	Ø Bağlantı için zil teli kullanınız.
Ø 7404 lojik entegrenin Vcc ve GND bağlantılarını yapınız.	Ø Entegrenin karakteristik çalışmasını, özelliklerini inceleyiniz.
Ø Şekildeki devreyi kurunuz. 	Ø Bağlantı için zil teli kullanınız.
Ø DA güç kaynağı ile devreye gerilim uygulayınız.	Ø Maksimum 5V uygulayınız.
Ø Giriş modülündeki butonlara basarak çıkış durumunu gözleyiniz.	
Ø Sonuçları bir rapor hâlinde hazırlayınız.	Ø Sonuç raporunda devrenin kurulum aşamasını, çalışmasını ve devrede kullanılan malzemelerle ilgili bilgiler veriniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları cevaplayarak bu faaliyette kazandığınız bilgileri ölçünüz.

ÖLÇME SORULARI

1. Binary 1010 sayısının desimal karşılığı aşağıdakilerden hangisidir?

- A. 8 B. 9 C. 10 D. 11

2. 16 girişi kodlayabilmek için en az kaç adet çıkışa ihtiyaç vardır?

- A. 1 B. 2 C. 3 D. 4

3. Heksadesimal A5 sayısının binary karşılığı aşağıdakilerden hangisidir?

- A. 1010 0101 B. 1010 0100 C. 1000 0100 D. 1110 0101

4. Desimal 17 sayısının BCD kod karşılığı aşağıdakilerden hangisidir?

- A. 1000 1000 B. 0001 0111 C. 0011 0011 D. 0001 0001

5. 'u' harfinin ASCII kod sistemindeki karşılığı aşağıdakilerden hangisidir?

- A. 111 0101 B. 7 5 C. 0111 111 D. 5 7

6. Dijital devrelerde pozitif doğru ifadesi aşağıdakilerden hangisi ile tanımlanabilir?

- A. Yanlış B. 0 C. 0V D. 5V

7. Veya ifadesi aşağıdakilerden hangisini tanımlamaktadır?

- A. Toplama B. Çıkarma C. Çarpma D. Bölme

8. Ve ifadesi aşağıdakilerden hangisini tanımlamaktadır?

- A. Toplama B. Çıkarma C. Çarpma D. Bölme

DEĞERLENDİRME

Soruların tamamını doğru olarak çözebildiyseniz bir sonraki faaliyete geçiniz. Cevaplarınız yanlış ya da eksik ise ilgili bölümü tekrar ediniz.

ÖĞRENME FAALİYETİ-2

AMAÇ

Tam toplayıcı devresini doğru olarak kurabilecek ve gerekli sayısal çıkışı elde edebileceksiniz.

ARAŞTIRMA

- Ø Tam toplayıcı devresinin hangi sistemlerde kullanıldığını araştırınız.
- Ø Tam toplayıcı devresi için özel olarak üretilen entegreleri araştırınız.

2. SAYI SİSTEMLERİNDE İŞLEMLER

Toplama, binary sayı sisteminin ana işlemidir. Diğer işlemler ise çıkarma, çarpma ve bölmedir. Bunlar yalnızca toplama prensibine göre hesaplanırlar. Binary sayılarla hesaplamalar desimal sistemle karşılaştırılmalı olarak yapıldığında oldukça kolaydır. Tek fark ise on tabanlı sayı sisteminde 10 rakam varken iki tabanlı sayı sisteminde sadece 2 rakam vardır.

2.1. Binary Sayı Sisteminde Toplama İşlemi

Basamak		10,000	1,000	100	10
Değerleri					
Tasman	0	1	1	1	1
Toplanan:		1	5	2	3
Toplanan:	+	4	8	9	6
Toplam:		6	4	2	0

6+7=13 ise birler basamağındaki 3 yazılır, onlar basamağındaki 1 bir üst basamağa taşınır.

Şekil 2.1: Desimal sayıların toplanması

Şekil 2.2’de binary sayılarda toplama işleminin temeli gösterilmiştir. İlk üç örnek desimal sistemdeki toplama işlemi ile aynı olduğu için basittir. Diğer iki örnekte ise elde 1 bir üst basamağa taşınmıştır.

$\begin{array}{r} 0 \\ + 0 \\ \hline 0 \end{array}$	$\begin{array}{r} 0 \\ + 1 \\ \hline 1 \end{array}$	$\begin{array}{r} 1 \\ + 0 \\ \hline 1 \end{array}$	$\begin{array}{r} 1 \\ + 1 \\ \hline 10 \end{array}$	$\begin{array}{r} 1 \\ 1 \\ + 1 \\ \hline 11 \end{array}$
$0 + 0 = 0$	$0 + 1 = 1$	$1 + 0 = 1$	$1 + 1 = (10)_2$	$1 + 1 + 1 = (11)_2$

Gelen elde

Şekil 2.2: Binary sayıların toplanması

Örnek 2.1: Verilen sayıları binary sistemde toplayınız ve daha sonra desimale çevirerek toplamayı desimal olarak da yapıp işlem sonuçlarını karşılaştırmamız?

a-101+10

b-1011+10

c-101011+111

Çözüm 2.2:

a.

$$\begin{array}{r} 101 \\ + 10 \\ \hline 111 \end{array}$$

b.

$$\begin{array}{r} 1011 \\ + 10 \\ \hline 1101 \end{array}$$

c.

$$\begin{array}{r} 101011 \\ + 111 \\ \hline 110010 \end{array}$$

2.2. Binary Sayı Sisteminde Çıkarma İşlemi

	5	17	11	13
Eksilen	6	8	2	3
Çıkan	- 4	9	3	7

Şekil 2.3: Desimal sayılarda çıkarma işlemi

Burada desimal sistemdeki kurallara benzerliğin yanında $0-1=1$ kuralı da vardır. Sonucun 1 olmasının sebebi üst basamaktan ödünç alınan $(10)_2$ dir.

$\begin{array}{r} 0 \\ - 0 \\ \hline 0 \end{array}$	$\begin{array}{r} 1 \\ - 0 \\ \hline 1 \end{array}$	$\begin{array}{r} 1 \\ - 1 \\ \hline 0 \end{array}$	$\begin{array}{r} 0 \\ - 1 \\ \hline 1 \end{array}$	Eksilen
				Çıkan
				Fark
$0 - 0 = 0$	$1 - 0 = 1$	$1 - 1 = 0$	$0 - 1 = 1$	Ödünç 1

Şekil 2.4: Binary sayılarda çıkarma işlemi kuralları

2.3. Binary Sayı Sisteminde Çarpma İşlemi

Şekil 2.5'te desimal sayılarda çarpma işlemi gösterilmiştir.

8 Çarpılan	28 Çarpılan
x 3 Çarpan	x 13 Çarpan
24 Çarpım	84
	+ 28
	364 Çarpım

Şekil 2.5: Desimal sayılarda çarpma işlemi

Şekil 2.6'da ise binary kurallarına göre çarpma işlemi verilmiştir.

0	0	1	1	Çarpılan
x 0	x 1	x 0	x 1	Çarpan
0	0	0	1	Çarpım
0 x 0 = 0	0 x 1 = 0	1 x 0 = 0	1 x 1 = 1	

Şekil 2.6: Binary sayılarda çarpma işlemi kuralları

2.4. Binary Sayı Sisteminde Bölme İşlemi

Şekil 2.7'de desimal sayı sistemindeki bölme kuralları yer almaktadır.

Bölünen	459	23	Bölen
	23	19	
	229		Bölüm
	207		
	022		

Şekil 2.7: Desimal sayı sisteminde bölme kuralları

Örnek 2.2: Desimal 10 sayısını desimal 4 sayısına, binary sayıya çevirerek bölünüz.
a) $10 \div 4$ b) $459 \div 23$

Çözüm 2.2:

$$\begin{aligned} \text{a. } (10)_{10} &= (1010)_2 \\ (4)_{10} &= (100)_2 \end{aligned}$$

$$\begin{array}{r} 1010 \overline{) 100} \\ \underline{- 100} \\ 00100 \\ \underline{- 100} \\ 000 \end{array}$$

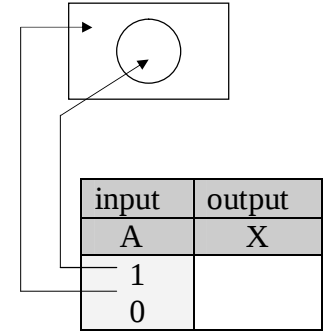
2.5. Lojik İşlemlerde Sadeleştirme Metodları

Ø Venn Diyagramı

Venn diyagramı doğruluk tablosunda görebildiklerinizi ifade etmek için kullanılan farklı bir yöntemdir. Dikdörtgen ve daireler kullanılır. N devrenin giriş sayısıdır ve dairelerle gösterilir. Dikdörtgen üzerinde 2^N alan vardır ve her bir alan doğruluk tablosundaki bir satıra karşılıktır. Eğer sinyal 1 (doğru) ise karşılık gelen alan karalanmıştır.

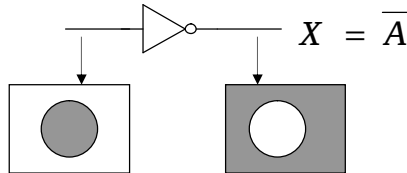
(1) Bir girişli Venn diyagramı

Devre bir girişli ise Venn diyagramında bir daire olur. Şekil 2.8'de doğruluk tablosunun satırlarına karşılık gelen alanlar gösterilmektedir. Sinyal 1 (doğru) değeri alırsa daire içi gölgeli olmalıdır.



Şekil 2.8

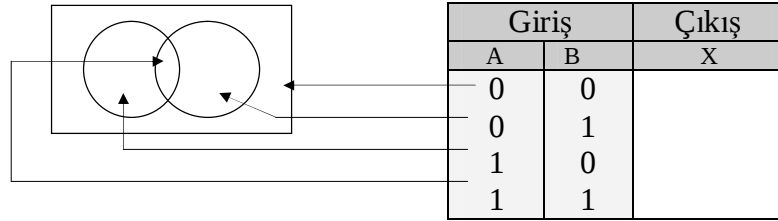
Şekil 2.9'da DEĞİL kapısının giriş ve çıkış sinyal değerleri Venn diyagramıyla gösterilmektedir.



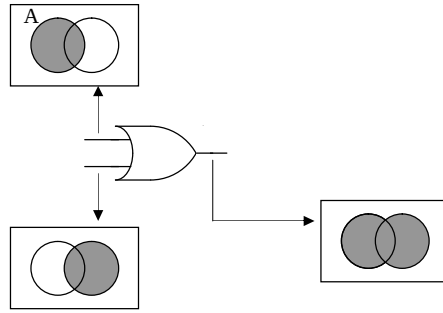
Şekil 2.9

(2) İki girişli Venn diyagramı

Devrenin iki girişli olması durumunda Venn diyagramında A ve B değişkenlerine karşılık birbiriyle kesişen iki daire çizilir. Şekil 2.10 doğruluk tablosunun satırları ile alanlar arasındaki ilişkiyi göstermektedir.



Şekil 2.10



Şekil 2.11

Şekil 2.11’de VEYA kapısının giriş ve çıkış sinyalleri Venn diyagramıyla gösterilmektedir.

Ø Karnaugh Haritaları

(1) İki değişkenli karnaugh haritası

İki değişkenli karnaugh haritasında kutu sayısı $2^N = 2^2 = 4$ adettir. Şekil 2.12’de iki değişkenli karnaugh haritalarındaki kutuların nasıl doldurulacağı gösterilmiştir. Şekil 2.12 (a)’da kutuların içerisine yazılan değerlerin hangi ifadelerle belirtileceği, Şekil 2.12 (b) de ise giriş ifadelerinin hangi sırayla kutulara yazılacağı gösterilmiştir.

Karnaugh haritalarında çözüm yaparken “1” değerleri gruplanır. Gruplama yaparken “1” sayısının ikinin katları şeklinde olması ve yanyana değerlerin gruplanması gerekir. Herhangi bir grupta yer alan değer diğer gruplarda da kullanılabilir.

Q	B	A	
		0	1
0	0	$\bar{A}\bar{B}$	$\bar{A}B$
1	0	$A\bar{B}$	AB

(a)

Şekil 2.12

(b)

Örnek 2.3: Yandaki karnaugh haritasının çıkış ifadesini yazınız.

Çözüm 2.3:

Q	B	A	
		0	1
0	0	1	
1	0		1

$Q_1 = \bar{A}\bar{B}$
 $Q_2 = A.B$
 $Q = \bar{A}\bar{B} + A.B$

Q	B	A	
		0	1
0	0	1	
1	0		1

Örnek 2.4: Yandaki karnaugh haritasının çıkış ifadesini yazınız.

Çözüm 2.4:

Q	B	A	
		0	1
0	0		1
1	0	1	1

$Q_1 = B$
 $Q_2 = A$
 $Q = A + B$

Q	B	A	
		0	1
0	0		1
1	0	1	1

(2) Üç değişkenli karnaugh haritası

Üç değişkenli karnaugh haritasında kutu sayısı $2^N = 2^3 = 8$ adettir. Şekil 2.13’te üç değişkenli karnaugh haritalarındaki kutuların nasıl doldurulacağı gösterilmiştir. Şekil 2.13 (a)’da kutuların içerisine yazılan değerlerin hangi ifadelerle belirtileceği, Şekil 2.13 (b) de ise giriş ifadelerinin hangi sırayla kutulara yazılacağı gösterilmiştir.

Q	BC	A			
		00	01	11	10
0	0	$\bar{A}\bar{B}\bar{C}$	$\bar{A}\bar{B}C$	$\bar{A}B\bar{C}$	$\bar{A}BC$
1	0	$A\bar{B}\bar{C}$	$A\bar{B}C$	$AB\bar{C}$	ABC

(a)

Şekil 2.13

Q	BC	A			
		00	01	11	10
0	0	0	1	3	2
1	0	4	5	7	6

(b)

Örnek 2.5: Aşağıdaki karnaugh haritasının çıkış ifadesini yazınız.

Q \ BC	00	01	11	10
A 0				1
A 1	1	1	1	1

Çözüm 2.5:

Q \ BC	00	01	11	10
A 0				1
A 1	1	1	1	1

$Q_1 = B \cdot \bar{C}$
 $Q_2 = A$
 $Q = B \cdot \bar{C} + A$

(3) Dört değişkenli karnaugh haritası

Dört değişkenli karnaugh haritasında kutu sayısı $2^N = 2^4 = 16$ adettir. Şekil 2.14'te dört değişkenli karnaugh haritalarındaki kutuların nasıl doldurulacağı gösterilmiştir. Şekil 2.14 (a)'da kutuların içerisine yazılan değerlerin hangi ifadelerle belirtileceği, Şekil 2.14 (b) de ise giriş ifadelerinin hangi sırayla kutulara yazılacağı gösterilmiştir.

Q \ CD	00	01	11	10
AB 00	$\bar{A}\bar{B}\bar{C}\bar{D}$	$\bar{A}\bar{B}\bar{C}D$	$\bar{A}\bar{B}C\bar{D}$	$\bar{A}\bar{B}CD$
AB 01	$\bar{A}B\bar{C}\bar{D}$	$\bar{A}B\bar{C}D$	$\bar{A}BC\bar{D}$	$\bar{A}BCD$
AB 11	$A\bar{B}\bar{C}\bar{D}$	$A\bar{B}\bar{C}D$	$AB\bar{C}\bar{D}$	$AB\bar{C}D$
AB 10	$A\bar{B}C\bar{D}$	$A\bar{B}CD$	$ABC\bar{D}$	$ABCD$

(a)

Q \ CD	00	01	11	10
AB 00	0	1	3	2
AB 01	4	5	7	6
AB 11	12	13	15	14
AB 10	8	9	11	10

(b)

Şekil 2.14

Örnek 2.6: Aşağıdaki karnaugh haritasının çıkış ifadesini yazınız.

Q \ CD	00	01	11	10
AB				
00	1		1	1
01		1		
11		1		
10	1		1	1

Çözüm 2.6:

Q \ CD	00	01	11	10
AB				
00	1		1	1
01		1		
11		1		
10	1		1	1

$Q_1 = \overline{B} \cdot \overline{D}$
 $Q_2 = B \cdot C \cdot \overline{D}$
 $Q_3 = \overline{B} \cdot D$
 $Q = \overline{B} \cdot \overline{D} + BCD + \overline{B}D$

2.6. Boolean Matematiği

Boolean matematiğinde ikili sayı sistemi üzerine bazı kurallar geliştirilmiştir. Yazılan lojik ifadeler, içeriği bozulmadan kurallar çerçevesinde değiştirebilir veya sadeleştirilebilir. Burada sadece eşitlikleri ezberlemek önemli değildir. Yeteneklerinizi ve hayal gücünüzü de ortaya koymanız önemlidir.

Ø Kurallar

Şekil 2.15 Boolean matematiğinin 10 kuralını göstermektedir. Bunlar Boolean matematiğinin temel kurallarıdır.

Şu ana kadar iki farklı değer alabilen A ve B değişkenlerini kullandık. Fakat bu tabloda A ve B değişkenleri yerine 0 ve 1 değerleri vardır.

VE	VEYA	DEĞİL
$0 \cdot 0 = 0$	$0 + 0 = 0$	$\overline{0} = 1$
$0 \cdot 1 = 0$	$0 + 1 = 1$	$\overline{1} = 0$
$1 \cdot 0 = 0$	$1 + 0 = 1$	
$1 \cdot 1 = 1$	$1 + 1 = 1$	

Şekil 2.15

Bir deęişkenli Boolean matematięinin kuralları:

Bir deęişken olması durumunda Boolean matematięinin kuralları Őekil 2.16’da gősterilmektedir.

- | |
|-----------------------------|
| (Kural 1) $0 + A = A$ |
| (Kural 2) $1 + A = 1$ |
| (Kural 3) $A + A = A$ |
| (Kural 4) $A + A = \bar{1}$ |
| (Kural 5) $0 \cdot A = 0$ |
| (Kural 6) $1 \cdot A = A$ |
| (Kural 7) $A \cdot A = A$ |
| (Kural 8) $A \cdot A = 0$ |
| (Kural 9) $A = A$ |

Őekil 2.16: Boolean matematięinin kuralları (1)

İki veya daha fazla deęişkenli Boolean matematięinin kuralları:

İki veya daha fazla deęişkenli ifadeler iin Boolean matematięinin kuralları Őekil 2.17’de gőrőlmektedir. Venn diyagramı veya doęruluk tablosunu kullanarak bu kuralların saęlamasını elde edebiliriz.

$A + B = B + A$	(Yer deęiřtirme kuralı)
$A \cdot B = B \cdot A$	(Yer deęiřtirme kuralı)
$A + (B + C) = (A + B) + C$	(Toplamada birleřme kuralı)
$A \cdot (B \cdot C) = (A \cdot B) \cdot C$	(arpmada birleřme kuralı)
$A \cdot (B + C) = A \cdot B + A \cdot C$	(Daęılma kuralı)
$A + B \cdot C = (A + B) \cdot (A + C)$	(Daęılma kuralı)
$A + A \cdot B = A$	(Gereksizlik kuralı)
$A \cdot (A + B) = A$	(Gereksizlik kuralı)

Őekil 2.17: Boolean matematięinin kuralları (2)

Örnek 2.7: Aşağıdaki ifadeyi Boolean matematiği kurallarını kullanarak sadeleştiriniz.

$$A.B + A.B.C$$

Çözüm 2.7:

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

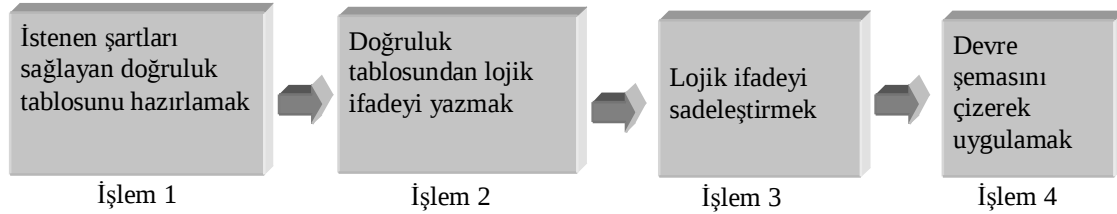
veya

$$A.B + A.B.C = A.B.1 + A.B.C = A.B.(1+C) = A.B.1 = A.B$$

2.7. Toplayıcı Devreler

Bir ihtiyaca yönelik lojik devre tasarımı yaparken doğrudan devre şemasının çizimi bazen tasarımı güçleştirir ya da hatalara neden olabilir. Bu nedenle devre tasarımına yeni başlayanlar için bazı işlemlerin adım adım gerçekleştirilmesi hata oranını azaltır ve işlemleri kolaylaştırır.

Lojik devre tasarımında izlenmesi gereken işlemler Şekil 2.18’de basitçe gösterilmiştir.



Şekil 2.18

Ø İstenen şartları sağlayan doğruluk tablosunun hazırlanması

Bir problemin çözümüne başlamadan önce problemin detaylarının çok iyi anlaşılması gerekir. Daha sonra çözüme yönelik çalışma yapabilirsiniz.

Örnek 2.8: 3 girişi olan ve bu girişlerden herhangi iki tanesi veya üçü birden 1 yapıldığında çıkıştaki lambanın yanmasını sağlayan sistemin doğruluk tablosunu yapınız.

Çözüm 2.8: Üç girişli bir devre olduğundan doğruluk tablosunda üç adet giriş ve bunların farklı olasılıkları gösterilmelidir. Girişte buton kullanılacaktır. Çıkışta ise yalnızca bir lamba vardır.

input			output
A	B	C	X
0	0	0	
0	0	1	
0	1	0	
0	1	1	
1	0	0	
1	0	1	
1	1	0	
1	1	1	

Giriş: 0—basılı değil 1—basılı
Çıkış: 0—lamba sönük 1—lamba yanıyor

Hangi şartlarda çıkışın 1 yada 0 olması gerektiğini doğruluk tablosunda gösterebiliriz.

Yalnızca istenen şartlar sağlandığında çıkışın 1 olduğunu doğruluk tablosunda inceleyiniz.

input			output
A	B	C	X
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	1

Şekil 2.19

Örnek 2.9: Yarım toplayıcının (half adder) doğruluk tablosunu gösteriniz. Yarım toplayıcı devreler iki bitlik ikili sayıların toplanması için uygundur.

Çözüm 2.9: Burada, A ve B toplanacak binary sayılardır. S toplama işlemi sonucu ve C ise (carry) eldedir. Buna göre doğruluk tablosu şekildeki gibi olmalıdır.

input		output	
0	0	0	0
0	1	0	1
1	0	0	1
1	1	1	0

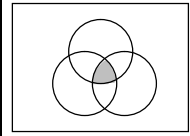
Ø Doğruluk tablosundan lojik ifadenin yazılması

Şekil 2.20’de verilen tablolarda yalnızca bir durumda çıkış 1 olmaktadır.

Örnek (a): A **VE** B **VE** C lojik 1 olduğunda çıkış 1 olmaktadır. Lojik ifade buna göre yazılır.

$$X=A.B.C$$

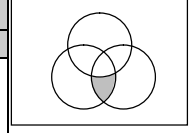
Giriş			Çıkış
A	B	C	X
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	1



Örnek (b): A'nın 0, B ve C'nin 1 olduğu durumda çıkışın 1 olması isteniyor. Burada tek dikkat edeceğimiz nokta A yerine A'nın değilini kullanmamız gerektiğidir.

$$X = \bar{A}.B.C$$

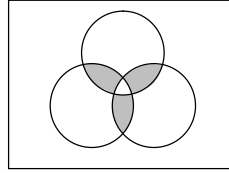
Giriş			Çıkış
A	B	C	X
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	0
1	1	0	0
1	1	1	0



Şekil 2.20

Doğruluk tablosunda birden fazla farklı durumda çıkış 1 oluyorsa her bir durum için lojik ifade ayrı ayrı yazılır. Daha sonra her bir durum için yazılan lojik ifadeler **VEYA** işlemine tabi tutulur. Şekil 2.21'de A,B,C girişlerinden herhangi ikisi '1' olduğunda çıkışı da '1' yapan devrenin doğruluk tablosu görülmektedir.

Giriş			Çıkış
A	B	C	X
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	1
1	0	0	0
1	0	1	1
1	1	0	1
1	1	1	0



Adım 1: Her 1 çıkışı için lojik ifadeler yazılır.

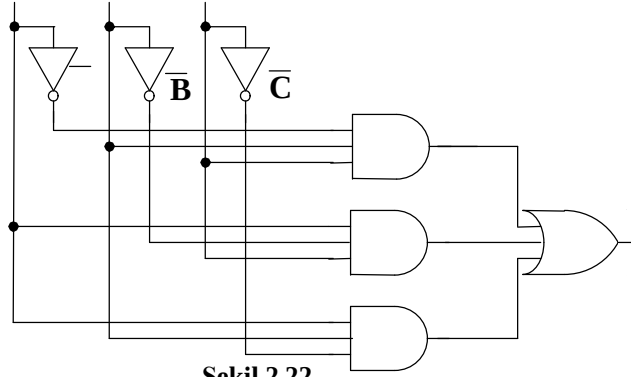


Adım 2: İfadeler VEYA işlemi ile birleştirilir.

Şekil 2.21

Şekil 2.22’te yukarıdaki doğruluk tablosuna uygun devre şeması verilmiştir.

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



Şekil 2.22

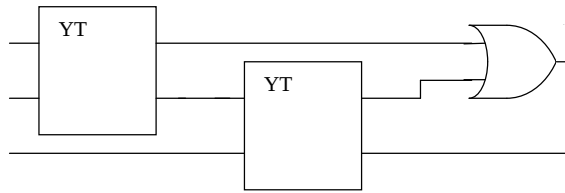
Ø Tam Toplayıcı

Şekil 2.23 yarım ve tam toplayıcı modüllerinin sembollerini göstermektedir.



Şekil 2.23

Tam toplayıcı (FA) yapmak için iki adet yarım toplayıcı (HA) kullanabiliriz (Şekil 2.24). Önce A ve B’nin ilk bitleri yarım toplayıcı ile toplanır. Sonra başka bir yarım toplayıcı kullanarak S_i^* ara toplam ile C_{i-1} toplanır. Elde bitlerini (C_i^* veya C_i^{**}) sonuca yansıtmak için VEYA kapısı kullanılır. C_i^* ve C_i^{**} bitlerinin her ikisinin de aynı anda 1 olamayacağına dikkat ediniz.



Şekil 2.24

UYGULAMA FAALİYETİ

İşlem basamaklarında belirtilen adımları izleyerek uygulama faaliyetini yapınız. Bu uygulama faaliyeti ile toplayıcı devre tasarımı yapabileceksiniz.

İşlem Basamakları	Öneriler
Ø Yarım toplayıcı devre doğruluk tablosunu çiziniz.	
Ø Tam toplayıcı devre doğruluk tablosunu çiziniz.	
Ø Tam toplayıcı devre şemasını çizip, kullanacağınız entegreleri seçiniz.	Ø Entegrenin çalışmasını, karakteristik özelliklerini inceleyiniz.
Ø Entegre bağlantı şemasını çiziniz.	Ø Enerji vermeyiniz.
Ø Deneybordunun bağlantısını yapınız.	
Ø Giriş – çıkış modülü ile deneybordunun bağlantısını yapınız.	Ø Bağlantı için zil teli kullanınız.
Ø DA güç kaynağı ile entegreye gerilim uygulayınız.	Ø Maksimum 5V uygulayınız.
Ø Giriş modülündeki butonlara basarak çıkış durumunu gözleyiniz.	
Ø Sonuçları bir rapor hâlinde hazırlayınız.	Ø Sonuç raporunda devrenin kurulum aşamasını, çalışmasını ve devrede kullanılan malzemelerle ilgili bilgiler veriniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları cevaplayarak bu faaliyette kazandığınız bilgileri ölçünüz.

ÖLÇME SORULARI

1. Binary 1010 sayısı ile 1000 sayısının toplam karşılığı aşağıdakilerden hangisidir?

- A. 10000 B. 10010 C. 10001 D. 11000

2. Binary 1010 sayısından 1000 sayısının çıkarılmasından kalan sonuç kaçtır?

- A. 001 B. 0010 C. 0100 D. 0001

3. Binary 1010 sayısı ile 10 sayısının çarpım sonucu aşağıdakilerden hangisidir?

- A. 10100 B. 10101 C. 10001 D. 11100

4. $A + B = B + A$ ifadesini aşağıdakilerden hangisi açıklamaktadır?

- A. Değişme B. Birleşme C. Toplama D. Çarpma

Aşağıdaki soruları doğru yanlış yöntemiyle cevaplayarak bu faaliyette kazandığınız bilgileri ölçünüz.

5. Hesap makinesi yarım toplayıcı devreye bir örnektir.
6. Dört bitlik sayıları toplarken yarım toplayıcı devresi kullanmak daha uygundur.
7. Bir lojik ifadede “1” açık olanı temsil ediyorsa “0” kapalı olanı temsil eder.
8. Bir lojik ifadede “0” açık olanı temsil ediyorsa “1” kapalı olanı temsil eder.

DEĞERLENDİRME

Soruların tamamını doğru olarak çözebildiyseniz bir sonraki faaliyete geçiniz. Cevaplarınız yanlış ya da eksik ise ilgili bölümü tekrar ediniz.

ÖĞRENME FAALİYETİ-3

AMAÇ

Flip-flop devresini şemaya uygun olarak kurabilecek ve gerekli sayısal çıkışı elde edebileceksiniz.

ARAŞTIRMA

- Ø Flip flopların hangi amaçlarla kullanıldıklarını araştırınız.
- Ø RS flip flopların çalışmasında diğer flip floplara göre farkını inceleyiniz.

3. DİĞER LOJİK KAPILAR

3.1. VE DEĞİL (NAND) Kapısı

VE DEĞİL kapısı kullanılarak tüm kapıların özelliklerini elde etmek mümkündür. Şekil 3.1’de VE DEĞİL kapı sembolü, lojik ifadesi ve doğruluk tablosu gösterilmiştir.

Giriş		Çıkış
A	B	X
0	0	1
0	1	1
1	0	1
1	1	0

(a) Doğruluk tablosu

(b) Mantıksal ifade

(c) Devre sembolü

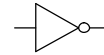
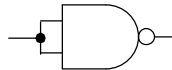
Şekil 3.1

Mantıksal ifadede VE DEĞİL için özel bir sembol yoktur. Bu nedenle VE ile DEĞİL sembollerinin birleşimiyle ifade edilir.

Örnek 3.1: VE DEĞİL kapısı kullanarak DEĞİL kapısı elde etmek mümkündür. Bunu Boolean matematik kurallarını kullanarak açıklayınız ve sonra da lojik devresini çiziniz.

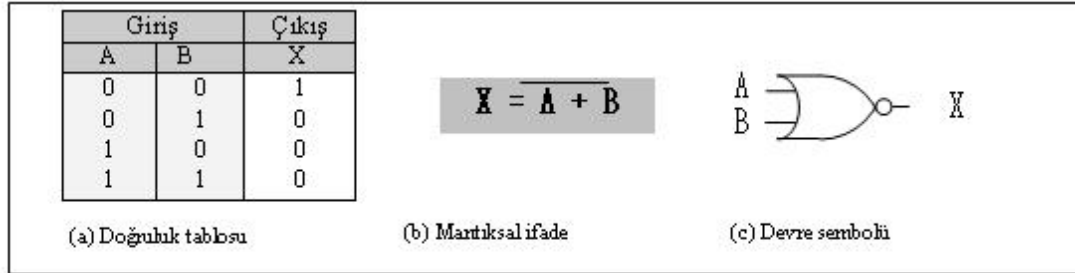
Çözüm 3.1:

$$\bar{A} = \overline{A.A}$$



3.2. VEYA DEĞİL (NOR) Kapısı

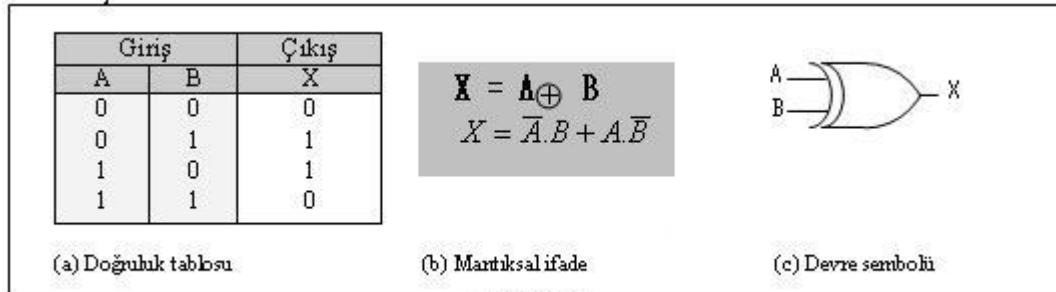
Şekil 3.2’de VEYA DEĞİL kapı sembolü, lojik ifadesi ve doğruluk tablosu gösterilmiştir. VEYA ve DEĞİL’in birleşiminden meydana gelmiştir.



Şekil 3.2

3.3. ÖZEL VEYA (EXOR) Kapısı

Şekil 3.3’te ÖZEL VEYA kapı sembolü, lojik ifadesi ve doğruluk tablosu gösterilmiştir.



Şekil 3.3

ÖZELVEYA kapısının özellikleri doğruluk tablosunda belirtilmiştir. VE, VEYA, ÖZEL VEYA işlemleri çok kullanışlı olduğundan bazı program dillerinde $\oplus$ olarak kullanılmaktadır.

3.4. DE MORGAN Teoremi

De Morgan teoremi kullanılarak VEYA işlemi ile VE işlemi arasında dönüşüm yapılabilir. De Morgan teoremi ile işlemler sadeleştirilerek daha basit hâle getirilebilir.

De Morgan teoremi

$$\begin{aligned}\overline{A+B} &= \overline{A} \cdot \overline{B} \\ \overline{A \cdot B} &= \overline{A} + \overline{B}\end{aligned}$$

Örnek 3.2: Aşağıdaki ifadeyi sadeleştiriniz.

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

Çözüm 3.2:

$$\begin{aligned}\overline{\overline{A} \cdot \overline{B}} + A \cdot B &= \overline{\overline{A} \cdot \overline{B}} \cdot \overline{\overline{A} \cdot \overline{B}} \\ &= (\overline{\overline{A}} + \overline{\overline{B}}) \cdot (\overline{\overline{A}} + \overline{\overline{B}}) \\ &= (A + B) \cdot (\overline{A} + \overline{B}) \\ &= (A + B) \cdot \overline{A} + (A + B) \cdot \overline{B} \\ &= A \cdot \overline{A} + B \cdot \overline{A} + A \cdot \overline{B} + B \cdot \overline{B} \\ &= B \cdot \overline{A} + A \cdot \overline{B} \\ &= A \oplus B\end{aligned}$$

3.5. RS Flip Flop

Flip-Flop, birbirine göre ters değerler alan iki çıkışlı bir modüldür. Uygun şartlar sağlandığında çıkıştaki '1' yada '0' değerini saklı tutabilir ve bu özelliği gereği birçok amaç için kullanılır.

3.5.1. RS Flip Flop Fonksiyonu

Şekil 3.4 RS-FF sembolünü ve doğruluk tablosunu göstermektedir. Set ve Reset girişleri vardır.

Giriş	Çıkış	
	S	R
0 0	Önceki durum	
0 1	0	1
1 0	1	0
1 1	Belirsizlik	

Şekil 3.4

(i) Her iki giriş '0' olduğu zaman çıkış değişmez. Diğer bir ifadeyle önceki durumunu korur.

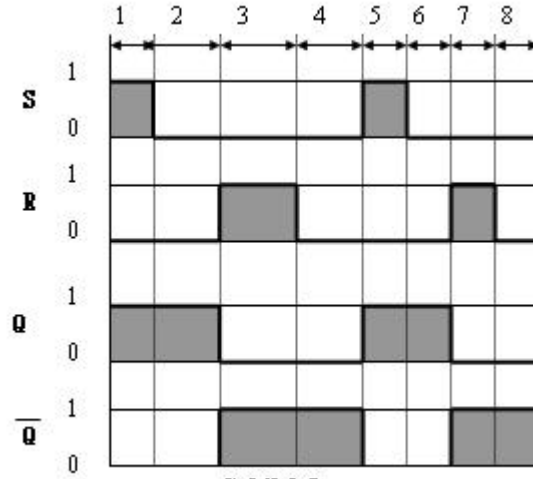
(ii) Reset girişi '0' iken Set girişi '1' yapılırsa Q çıkışı '1' olur. (Buna set by set denir). Bu durumda Q'nun değili '0' olur.

(iii) Set girişi '0' iken Reset girişi '1' yapıldığında Q='0' olur. (Buna reset by reset denir). Bu durumda Q'nun değili 1 olur.

(iv) Her iki girişin de '1' olduğu durumda belirsizlik vardır. Bu durumda çıkışlar herhangi bir değer alabilir. Kullanıcının bu durumu engelleyici bir çalışma yapması gerekir.

Flip-Flop'ta çıkış, giriş değerlerine ve bir önceki duruma bağlıdır. Şekil 3.5'te zaman çizelgesi verilmiştir. Burada yatay eksen zamanı ve soldaki eksen lojik seviyeyi göstermektedir.

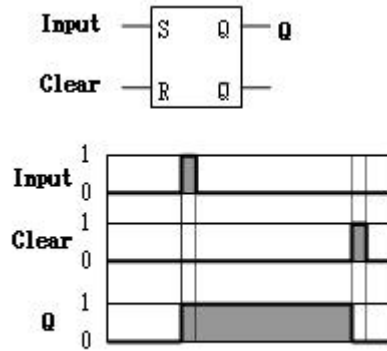
Giriş sinyal değeri ve bir önceki durumu dikkate alarak çıkışı belirlemek mümkündür.



Şekil 3.5

3.5.1.1. Kendi Kendine Tutma (Self-Holding) Fonksiyonu

Şekil 3.6’da RS-FF’nin zaman çizelgesi (time table) görülmektedir. Giriş (input) ‘1’ yapıldığında çıkış $Q=1$ olur. Daha sonra giriş (input) ‘0’ yapılırsa bile çıkışta ‘1’ saklı tutulur. Clear girişi ‘1’ yapıldığında $Q=0$ olur. Bu fonksiyona kendi kendini tutma (self holding) denir.



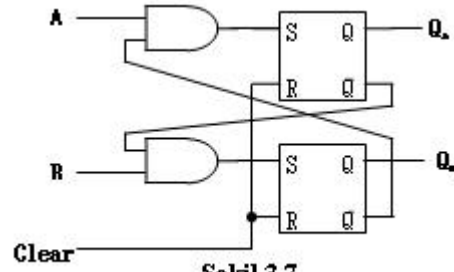
Şekil 3.6

3.5.1.2. Kilitleme (Locking) Fonksiyonu

VE kapısıyla RS-FF’nin diğer girişi kilitlenebilir.

Clear sinyali iki RS-FF’yi reset edebilir. $A=1$ olursa $Q_A=1$ olur ve bu değer hafızada tutulur. Q_A ’nın değili ise sıfırdır. Bu değer aşağıdaki VE kapı çıkışının sürekli ‘0’ olmasını sağlar. Böylece B girişi kilitlenmiş olur.

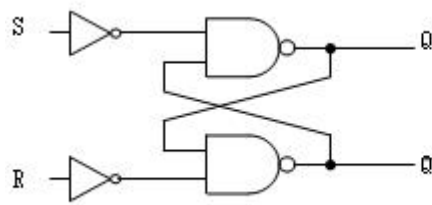
Resetten sonra eğer B girişi A’dan önce ‘1’ yapılırsa bu kez A girişi kilitlenmiş olur.



Şekil 3.7

3.5.3. RS Flip Flop'un Yapılışı

RS flip-flop farklı kapı uygulamaları ile yapılabilir. Şekil 3.8'te birçok durumda kullanılabilen RS flip-flop görülmektedir.



Şekil 3.8

Giriş		Çıkış
S	R	Q Q
0	0	Önceki durum
0	1	0 1
1	0	1 0
1	1	Belirsizlik

Önce VE DEĞİL kapı karakteristiğini hatırlayınız.

VE DEĞİL kapı özellikleri:

- 1) Girişlerden birisi '0' olduğu zaman çıkış sürekli 1'dir.
- 2) Girişlerden birisi '1' olduğu zaman çıkış, diğer girişin tersidir.

Şimdi bu özellikleri kullanarak RS flip flop'un doğruluk tablosunu kontrol edelim.

VE DEĞİL'in doğruluk tablosu

Giriş		Çıkış
A	B	X
U	0	1
0	1	1
1	0	→ 1
1	1	→ 0

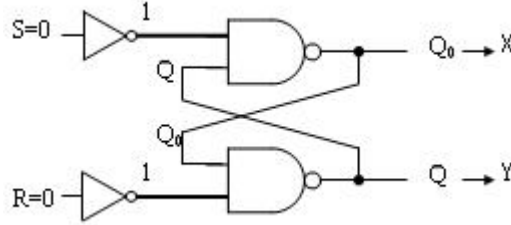
Sürekli '1'

Tersleme

Şekil 3.9

1) S=0, R=0

Q_0 çıkışı için X ve Q_0 'ın değili için Y ifadesini kullanalım. S=0, R=0 durumunda VE DEĞİL kapılarının birer girişleri '1' olacaktır. Bu nedenle kapı çıkışları diğer girişlerin tersi olacaktır.



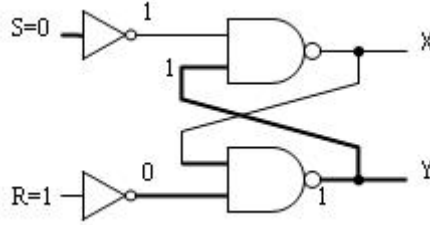
$$X = \overline{\overline{Q_0}} = Q_0, \quad Y = \overline{Q_0}$$

Sonuçta çıkışlar bir önceki durum ile aynı olacaktır.

2) S=0, R=1

R girişi '1' yapılırsa bağlı olduğu VE DEĞİL kapı girişi 0 ve çıkışı da '1' olur.

$$Y = 1, X = 0$$



UYGULAMA FAALİYETİ

İşlem basamaklarında belirtilen adımları izleyerek uygulama faaliyetini yapınız. Bu uygulama faaliyeti ile RS flip-flop ve kilitleme devre tasarımlarını yapabileceksiniz.

İşlem Basamakları	Öneriler
Ø R –S flip flop devresi doğruluk tablosunu çiziniz.	
Ø Teknik resim kurallarına göre devre şeması çiziniz.	
Ø 7400 ve 7404 entegrelerinin bağlantı şemasını çiziniz.	
Ø Devre elemanlarının bağlantısını yapınız.	Ø Bağlantı için zil teli kullanınız.
Ø Devreyi çalıştırıp sonuçları kontrol ediniz.	
Ø Sonuçları bir rapor hâlinde hazırlayınız.	Ø Sonuç raporunda devrenin kurulum aşamasını, çalışmasını ve devrede kullanılan malzemelerle ilgili bilgiler veriniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları doğru yanlış yöntemiyle cevaplayarak bu faaliyette kazandığınız bilgileri ölçünüz.

ÖLÇME SORULARI

1. VE DEĞİL kapısında girişlerden birisi '0' olduğu zaman çıkış sürekli 1'dir.
2. RS flip-flop'ta SET ve RESET girişleri vardır.
3. Flip-flop'lar, uygun şartlar sağlandığında çıkıştaki '1' yada '0' değerini saklı tutabilir.
4. $X = A + B$ ifadesi ÖZEL VEYA kapısının ifadesidir.
5. $X = A \cdot B$ ifadesi VE DEĞİL kapısının ifadesidir.
6. $A=A.A$ ifadesinin sadeleştirilmiş hali "1" dir.
7. $A+B = B+A$ ifadesi doğrudur.
8. ÖZEL VEYA kapısının açık formülü $A.B+B.A'$ dır.

DEĞERLENDİRME

Soruların tamamını doğru olarak çözebildiyseniz bir sonraki faaliyete geçiniz. Cevaplarınız yanlış ya da eksik ise ilgili bölümü tekrar ediniz.

ÖĞRENME FAALİYETİ-4

AMAÇ

İhtiyaca uygun lojik entegre seçimini doğru olarak yapabileceksiniz.

ARAŞTIRMA

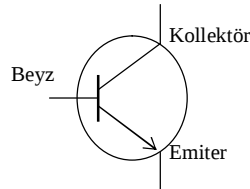
Ø Piyasada bulunan lojik entegreleri ve hangi devrelerde kullanıldığını araştırınız.

4. ENTEGRE (IC) KULLANIMI

Lojik devreler entegre kullanılarak kurulur. Bunun için öncelikle entegreler hakkında bilgi edinmek gerekir.

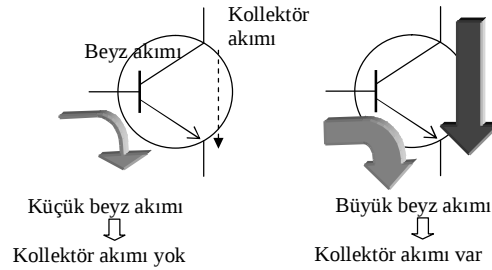
4.1. TTL ve CMOS Entegreler

Ø **Yapısı:** Entegrenin temelinde transistör vardır. Genel olarak iki çeşit transistör vardır. Bu nedenle entegreleri de iki bölümde inceleyebiliriz.



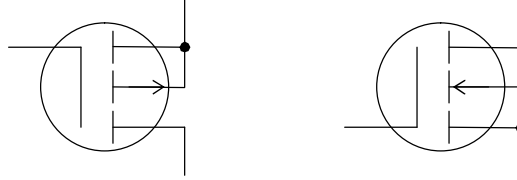
Şekil 4.1

Şekil 4.1’de TTL (Transistor Transistor Logic) IC (integrated circuit) bipolar transistor, sembolünde bacak isimleri görülmektedir.

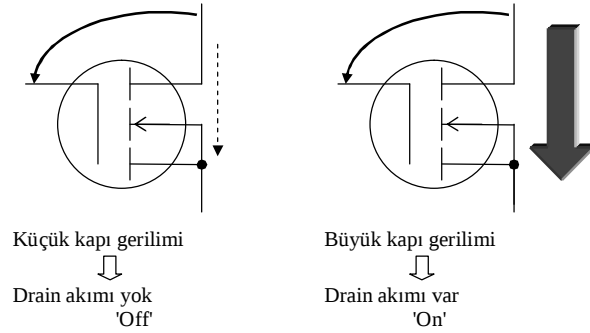


Şekil 4.2

Bipolar transistörde beyz akımı kollektör akımını kontrol eder. Eğer beyz akımı sıfır veya gerekli seviyeden daha düşük seviyedeysse kollektörden akım geçmez. Bu duruma kapalı (off) pozisyon denir. Fakat beyz akımı kollektör akımı için yeterli büyüklükte ise kollektörden akım geçer ve açık (on) durumunda olur (Şekil 4.2). Bu özelliğinden dolayı transistörler bir anahtar olarak kullanılabilir.



Şekil 7.3

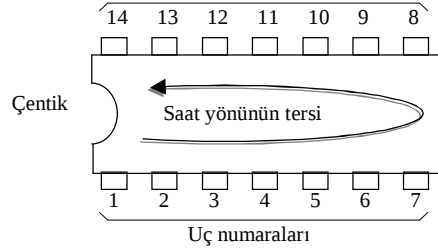


Şekil 4.4

Şekil 4.3'te **CMOS (Complementary Metal Oxide Semiconductor) IC**, diğer taraftan, **FET (Field Effect Transistor)** sembolleri görülmektedir. Bu tip transistörlerde kollektör akımı, V_{GS} gerilimi ile kontrol edilir. Eğer gerilim yeterli seviyeden daha düşük ise anahtar 'off' durumundadır ve drain akımı sıfırdır. Gerilim yeterli değerdeyse anahtar 'on' durumundadır ve drain akımı vardır. (Şekil 4.4)

MOS ismi “**Metal Oksit Semiconductor**” ifadelerinden gelmektedir. CMOS IC’lerde MOSFET kullanılır. P kanal ve N kanal olmak üzere iki çeşit MOSFET vardır. Bu yapının en büyük avantajı TTL IC’den daha küçük bir güç tüketiminin olmasıdır. FET, statik elektriğe karşı çok hassastır. Fakat günümüzde TTL’den daha çok CMOS IC’ler kullanılmaktadır.

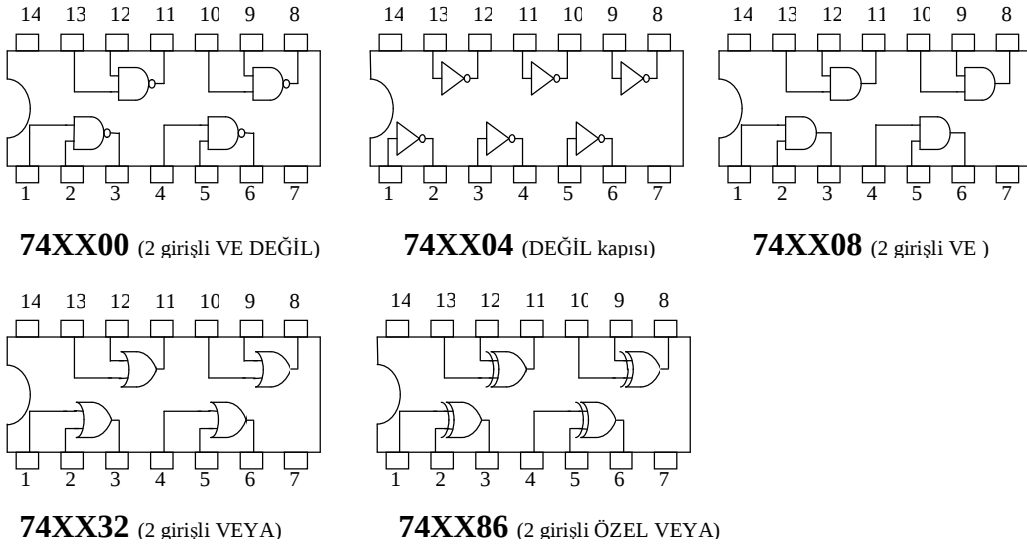
Ø Temel Entegreler



Şekil 4.5

Şekil 4.5'te entegrenin üst görünüşü verilmiştir. Her entegrede bir çentik yada buna benzer bir işaret vardır. Bu işaret şekilde olduğu gibi solda tutulduğunda saat yönünün tersi yönde uçlar sırayla numaralandırılır.

Şekil 4.6'da temel entegrelerin iç yapıları ve uç düzenlemesi görülmektedir.



Şekil 4.6

Her entegrenin (IC) üzerinde bir numara vardır ve IC numarası üç bölümden oluşur.

İlk iki rakam entegrenin grup adını göstermektedir. (Bu bazen serisi olarak da adlandırılır) 74 serisinin çok kullanılan IC çeşitleri şekil 4.6'da gösterilmiştir.

Ortada görünen alfabetik karakterler ile ilk iki rakam, bu IC'nin ailesi olarak adlandırılır. TTL IC'ler nispeten yavaşırlar ve daha fazla güç tüketirler. 74LS (Low power Shottky) ve 74ALS (Advanced Low power Shottky) ise yeni ve daha iyi özelliklere sahiptir.

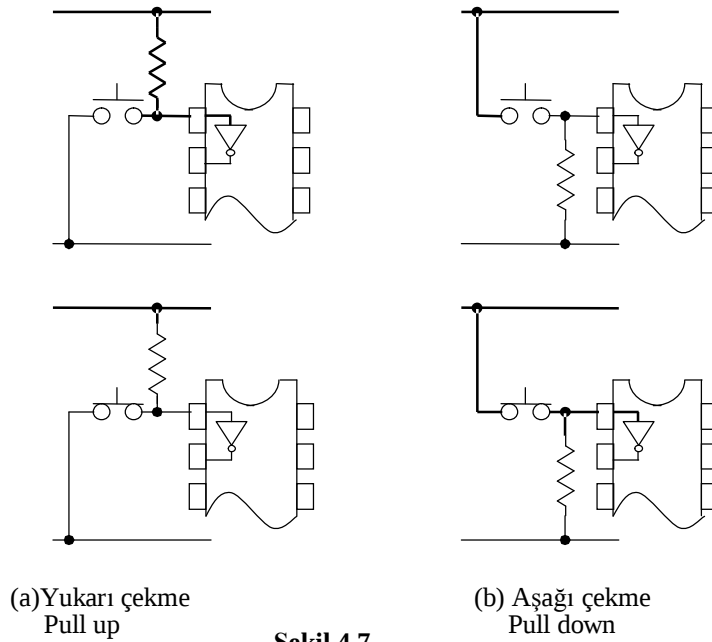
Fakat günümüzde 74HCXX (**H**igh speed **C**MOS) ailesi IC'ler geliştirilmiştir. Son iki rakam ise uçların yerleşimini göstermektedir. Katalog kullanarak uç düzenlemesi belirlenebilir.

TTL IC'lerde besleme gerilimi V_{CC} ($=5[V]$) ve GND ($=0[V]$) olarak gösterilir. Fakat CMOS IC için V_{DD} ($=5[V]$) ve V_{SS} ($=0[V]$) kullanılır.

4.2. Entegre Girişleri

Giriş anahtarı

Şekil 4.7 giriş buton bağlantı yöntemlerini göstermektedir.



Şekil 4.7

Şekil 4.7(a)'daki bağlantı yukarı çekme '**pull up**' olarak adlandırılır. Buton basılı değilken giriş ucunun gerilimi $5[V]$ 'tur. Şekil 4.7(b) ise aşağı çekme '**pull down**' olarak adlandırılır. Buton basılı değilken giriş ucu gerilimi $0[V]$ 'tur.

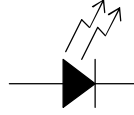
CMOS IC'ler için her iki bağlantıyı da kullanabiliriz. Fakat TTL IC'ler için $I_{IL(max)}$ değeri dikkate alındığında pull up yönteminin kullanılması daha uygun olur. Çünkü pull down için daha fazla akıma ihtiyaç olacaktır. Genellikle pull up direnci $2,2[k\Omega]$ ile $33[k\Omega]$ arasındadır.

Ø Bağlantısı olmayan giriş uçları

TTL IC'lerde boş bırakılan girişlerde '1' varmış gibi işlem yapılır. IC'de kullanılmayan kapılar için herhangi bir bağlantıya da ihtiyaç yoktur. Ancak CMOS IC'ler için kullanılmayan uçlara '0' veya '1' verilmelidir. Boş bırakılması yanlış çalışmalara sebep olabilir.

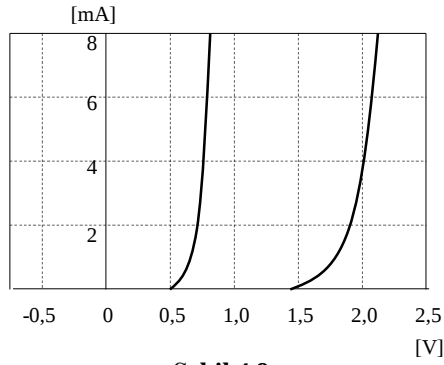
4.3. Led (Light Emitting Diode)

LED (Light Emitting Diode) geniş kullanım alanı olan bir diyottur. Genellikle sinyal durumunu gösteren bir çıkış elemanı olarak kullanılır. Şekil 4.8'de led için devre sembolü görülmektedir.

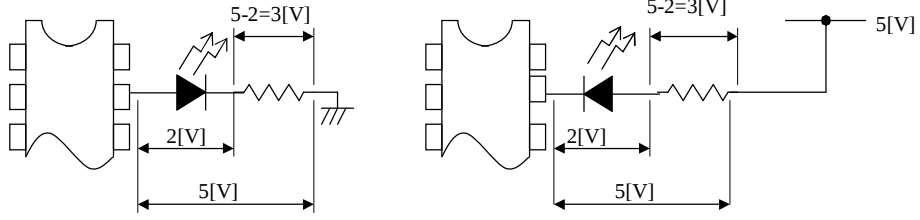


Şekil 4.8

Genel olarak LED'ler küçük (çapı 3-5mm) boyutlu imal edilir ve 5 ile 20[mA] arasında akım çeker. Gerilimi 2[V] civarındadır. Şekil 4.9'da örnek bir LED'in karakteristik eğrisi görülmektedir.



Şekil 4.9

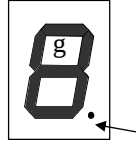


Şekil 4.10

Şekil 4.10’da ledin IC uçlarına iki farklı şekilde bağlantısı görülmektedir.

4.4. 7 Segment LED ve Sürücüsü

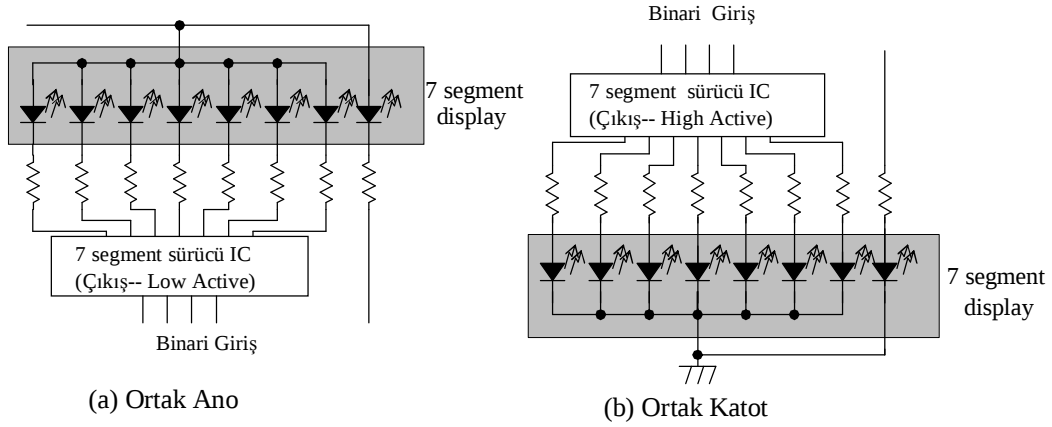
7 segment display de denir. 7 segment LED (Şekil 4.11) desimal numaraları görüntülemek için birleşik bir paket hâline getirilmiş ledlerden oluşur. Desimal nokta için bir led daha vardır.



Şekil 4.11

Yeterli parlaklık için akım diğer normal LED’lerde olduğu gibi 5[mA] ile 20[mA] arasındadır. LED gerilimi yaklaşık 2[V] uygulanmalıdır.

Şekil 4.12’de görüldüğü gibi 7 segment LED’lerin iki çeşidi vardır.



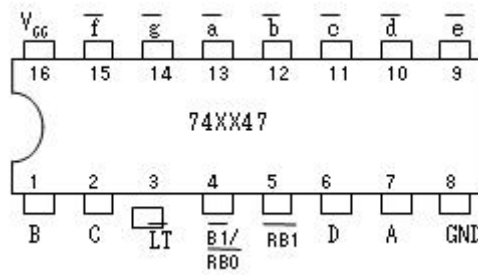
Şekil 4.12

Ortak anotlu 7 segment LED’lerde yeterli parlaklık için sürücü çıkışı sıfır (düşük) olmalıdır. Buna **active - low** (veya **low-active**) denir. Sinyal sıfır (düşük) olduğunda akım geçişi olur ve led ışık verir. Diğer yandan ortak katotlu tipler için çıkış uçları **active - high** (veya **high-active**) olmalı. Sinyal ‘1’ (yüksek) olduğu zaman mantık doğrudur. **Low - active** bazen IC’lerde ve mantık devrelerinde de kullanılır.

7 segment LED’in ortak anotlu veya ortak katotlu olmasına göre iki farklı IC sürücüsü kullanılabilir. IC sürücünün azalan akımı veya kaynak akımı normal IC değerinden daha fazladır (Şekil 4.13). Böylece LED’leri sürmek için yeterli akım verebilir.

Ø 74XX47

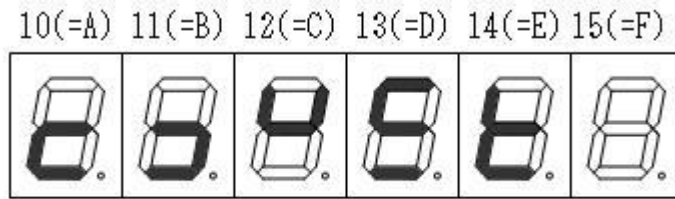
74XX47 TTL’dir. Ortak anotlu 7 segment LED’lerde kullanılır. Şekil 4.13’te 74XX47’nin uç bağlantı isimleri görülmektedir.



Şekil 4.13

Uçlar	Kullanım amacı	Sinyal seviyesi	Açıklama
A-D	Binari girişler	active-high	Sinyal giriş uçlarıdır.
$\overline{\text{a-g}}$	7 seg. LED için çıkışlar	active-low	7 adet LED'in bağlanacağı çıkışlar.
$\overline{\text{LT}}$	Lamba testi	active-low	BI high iken bu uç seviyesi low ise tüm çıkışlar low olur. Bu nedenle normal kullanımda high veya açık olmalı (bağlantı olmamalı).
$\overline{\text{RB I}}$	Ripple blanking input	active-low	Binari girişi sıfır olduğu zaman etkilidir. Bu giriş low ise tüm çıkışlar yüksek olur. (LED'ler yanmaz)
$\overline{\text{BI/RB0}}$	Blanking input veya Ripple blanking output.	active-low	Bu giriş düşük olduğunda tüm çıkışlar yüksek olur. (LED'ler yanmaz) Bu uç aynı zamanda RB0 için çıkış olarak da kullanılır. RBI düşük ve tüm binari girişler sıfır olduğunda bu uç da low olur.

Bu IC'lerle display üzerinde 10'dan sonraki heksadesimal karakterler de görüntülenebilir. Şekil 4.14 bu çıkışları göstermektedir.



Şekil 4.14

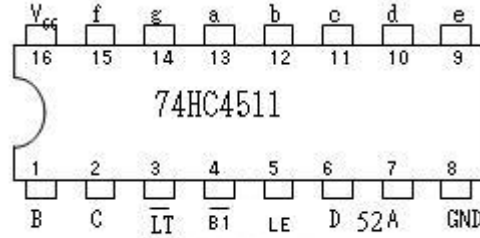
Her bir çıkış ucunun azalan akımı 24[mA]'den fazladır. Bu nedenle 330[Ω] direnç kullanabiliriz.

Gerçekte her bir LED'den geçen akım aşağıdaki gibi hesaplanabilir.

$$I = \frac{5-2}{330} \approx 9.1 \text{ [mA]}$$

Ø 74HC4511

74HC4511 CMOS'dur ve ortak katotlu 7 segment LED için kullanılır. (Şekil 4.15)



Şekil 4 .15

74HC4511 uçlarının fonksiyonları aşağıdaki gibidir.

Uçlar	Kullanım amacı	Sinyal seviyesi	Açıklama
A-D	Binari giriş	active-high	Sinyal giriş uçlarıdır.
a-g	7 seg. LED için çıkış	active-high	7 adet LED'in bağlanacağı çıkışlar.
$\overline{LT}$	Lamba test	active-low	Bu seviye düşükse bütün çıkışlar yüksek olurlar.
$\overline{BI}$	Blanking input	active-low	Bu giriş low ve LT high olduğunda tüm çıkışlar düşük olurlar (LED yanmaz)
$\overline{LE}$	latch enable		Bu giriş normalde düşük seviyede olmalıdır. High olursa IC bütün çıkışlarını değişim süresince saklı tutar.

Her çıkış ucunun kaynak akımı 40 [mA]'dır. Bu nedenle 330[Ω]'luk direnç kullanılır.

UYGULAMA FAALİYETİ

İşlem basamaklarında belirtilen adımları izleyerek uygulama faaliyetini yapınız. Bu uygulama faaliyeti ile çeşitli entegrelerle devre tasarımı yapabileceksiniz.

İşlem Basamakları	Öneriler
Ø Herhangi bir lojik entegreyi deneybord üzerine takınız.	Ø Entegrenin çalışmasını, karakteristik özelliklerini inceleyiniz.
Ø Lojik entegrenin Vcc ve GND bağlantılarını yapınız.	Ø Bağlantı için zil teli kullanınız.
Ø Lojik entegre çıkışına uygun bir direnç ile led bağlantısını yapınız.	
Ø Entegre girişinde bağlantı yok iken led durumunu kontrol ediniz.	
Ø Entegre girişinde GND bağlantısı ile led durumunu kontrol ediniz.	
Ø Bir önceki işlem basamağındaki bağlantıyı kaldırıp led kontrolü yapınız.	
Ø Entegre girişine Vcc bağlayıp led kontrolü yapınız.	
Ø Bağlantıyı kaldırıp led kontrolü yapınız.	
Ø Sonuçları bir rapor hâlinde hazırlayınız.	Ø Sonuç raporunda devrenin kurulum aşamasını, çalışmasını ve devrede kullanılan malzemelerle ilgili bilgiler veriniz.

ÖLÇME VE DEĞERLENDİRME

Aşağıdaki soruları doğru yanlış yöntemiyle cevaplayarak bu faaliyette kazandığınız bilgileri ölçünüz.

ÖLÇME SORULARI

1. () Transistörde kapı gerilimi GATE ucundaki gerilimi ifade etmektedir.
2. () Günümüzde TTL'den daha çok CMOS entegreler kullanılmaktadır.
3. () 7400 entegresi VEYA kapı entegresidir.
4. () 7404 entegresi VE DEĞİL kapı entegresidir.
5. () 7447 TTL'dir. Ortak anotlu 7 segment LED'lerde kullanılır.
6. () 74HC4511 CMOS'dur ve ortak katotlu 7 segment LED için kullanılır.
7. () Ortak katotlu entegreler için çıkış uçları **active - high** olmalıdır.
8. () Ortak anotlu 7 segment LED'lerde yeterli parlaklık için sürücü çıkışı sıfır olmalıdır.

DEĞERLENDİRME

Soruların tamamını doğru olarak çözebildiyseniz bir sonraki faaliyete geçiniz. Cevaplarınız yanlış ya da eksik ise ilgili bölümü tekrar ediniz.

MODÜL DEĞERLENDİRME

PERFORMANS TESTİ (YETERLİK ÖLÇME)

Uygulama faaliyetinde yaptığınız işlemlere göre aşağıdaki tabloyu doldurarak kendinizi değerlendiriniz.

Modülün Adı		Öğrencinin	Adı : Soyadı: Sınıf : Nu :
AÇIKLAMA: Aşağıda listelenen işlem basamaklarındaki davranışları öğrencide gözlemlediyseniz Evet sütununa, gözlemleyemediyseniz Hayır kısmına X işareti yazınız.			
GÖZLENECEK DAVRANIŞLAR		Evet	Hayır
1. Deneybordunu doğru bir şekilde kullanabildiniz mi?			
2. VCC ve GND bağlantılarını öğrenebildiniz mi?			
3. Lojik kapıların hangi ifadeleri gösterdiğini öğrenebildiniz mi?			
4. Yarım toplayıcı devresinin yapısını anlayabildiniz mi?			
5. Tam toplayıcı devresinin yapısını anlayabildiniz mi?			
6. RS flip-flop devresini doğru olarak çalıştırabildiniz mi?			
7. Lojik devreye göre entegre seçimini yapabildiniz mi?			
8. Devre elemanının gücünü hesaplayabildiniz mi?			
9. Sadeleştirme işlemlerinde venn şemasını kullanabildiniz mi?			
10. Sadeleştirme işlemlerinde karnaugh haritasını kullanabildiniz mi?			
11. Boolean matematiğini kullanarak sadeleştirme yapabildiniz mi?			
12. İş güvenliğine uygun olarak hareket edebildiniz mi?			
13. Malzemeleri kullandıktan sonra yerine iade edebildiniz mi?			

DEĞERLENDİRME

Hayır cevaplarınız var ise ilgili uygulama faaliyetini tekrar ediniz. Cevaplarınızın tümü evet ise bir sonraki modüle geçebilirsiniz.

CEVAP ANAHTARLARI

ÖĞRENME FAALİYETİ-1 CEVAP ANAHTARI

1	C
2	D
3	A
4	B
5	B
6	D
7	A
8	C

ÖĞRENME FAALİYETİ-2 CEVAP ANAHTARI

1	B
2	B
3	A
4	A
5	YANLIŞ
6	YANLIŞ
7	DOĞRU
8	DOĞRU

ÖĞRENME FAALİYETİ-3 CEVAP ANAHTARI

1	DOĞRU
2	DOĞRU
3	DOĞRU
4	YANLIŞ
5	YANLIŞ
6	YANLIŞ
7	DOĞRU
8	YANLIŞ

ÖĞRENME FAALİYETİ-4 CEVAP ANAHTARI

1	DOĞRU
2	DOĞRU
3	YANLIŞ
4	YANLIŞ
5	DOĞRU
6	DOĞRU
7	DOĞRU
8	DOĞRU

KAYNAKÇA

- Ø AKTAŞ İsmail, Dijital Elektronik Ders Notları, İzmir 2005.
- Ø ISHIDA Yasuhiro. Turgay İŞBİLEN, Selim GÜLÇEN, Sedat ELBİR, **Temel Endüstri Uygulamaları** İzmir, 2002
- Ø T.YAMAUCHI, OSMAN KÖSE, SELİM GÜLÇEN, **Devre Analizi**, Konya 2003.