

## MSI VE PLD ELEMANLARI

Şimdiye kadar anlatılanlar, herhangi bir lojik fonksiyonu dizayn etmek için kapı sayısını azaltan yöntemler şeklinde idi. Bu maliyeti daha az sayıda kapı devresi kullanarak düşürmek içindi. Fakat devre tasarımında entegre devreler (IC) kullanıldığı zaman bu tip bir yol izlemek her zaman geçerli olmaz.

Tüm devreler devre karmaşıklığına göre

- 1- Küçük ölçekte tümleşiklik (SSI)
- 2- Orta ölçekte tümleşiklik (MSI)
- 3- Büyük ölçekte tümleşiklik (LSI)
- 4- Çok büyük ölçekte tümleşiklik (VLSI)

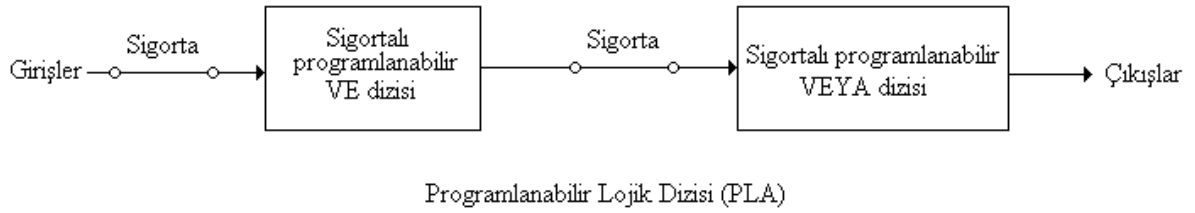
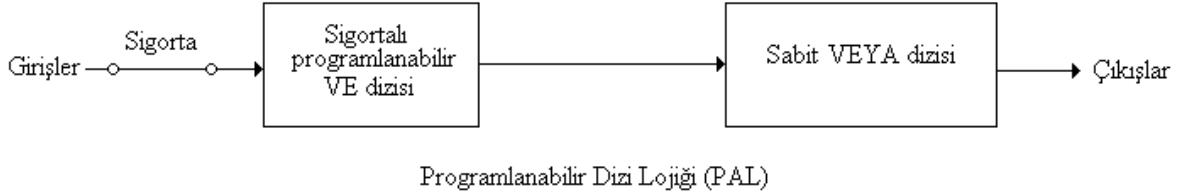
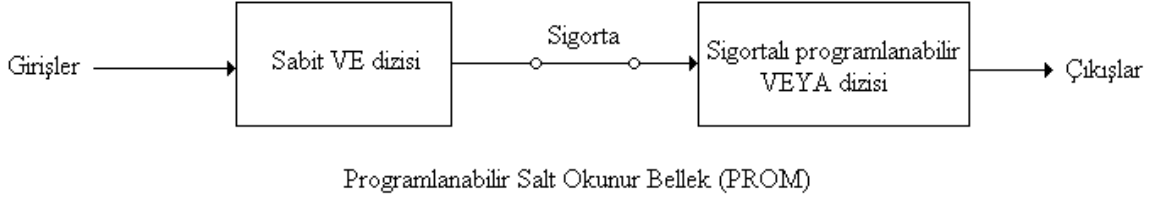
Olarak dörde ayrılmıştı. Çok sayıda birbirinden ayrı kapı içeren bir lojik devre bir birinden ayrı kapıları içeren SSI yapısı kullanılarak dizayn edilebilir. Kapı sayısının arttığı düşünülse bile ekonomik olarak tasarımcıya oldukça fazla avantaj sağlar.

Sayısal devrelerin tasarımında bazı kombinezonal devreler oldukça yaygın olarak kullanılırlar. Bu tip devreler tüm devre şeklinde mevcut olup MSI elemanları olarak sınıflandırılırlar. MSI elemanları sayısal sistemlerde yaygın olarak kullanılan ortak özellikteki bazı özel fonksiyonları gerçekleştirirler. Bu özel fonksiyonlar, toplayıcı, çıkarıcı, karşılaştırıcı, kodlayıcı (decoder), kod çözücü (encoder) ve veri seçicilerdir (multiplexer).

MSI devreleri SSI devrelerine göre maliyeti oldukça fazla aşağı çeker. Programlanabilen Lojik Aygıt (PLD) elemanları ise içinde yer alan lojik kapıların bir birlerine sigortalar yardımıyla bağlı olduğu bir tüm devredir. Aygıtın programlanması, birbirinden ayrılması gereken hatlara ilişkin sigortaların attırılması ile yapılır. Böylece özel bir yapı oluşur. Programlama sözcüğü, devrenin içyapısının belirlenmesine yönelik bir donanım işlevini yerine getirmektir. PLD'deki kapılar VE ve VEYA dizilerinden oluşmaktadır. Bu diziler bir araya gelerek çarpımların toplamını oluştururlar. PLD'deki tüm sigortalar başlangıçta attırılmamış durumdadırlar. Aygıtın programlanması ile gerekli fonksiyona ulaşmak için ilgili sigortalar attırılır.

Bu bölümde üç PLD ve sayısal sistemlere uygulanması anlatılacaktır. VE-VEYA dizilerinin yerleşimine göre üç farklı PLD vardır. Bunlar şekilde görüldüğü gibi sabit bir VE dizisi ve çıkışta yer alan VEYA kapılarına ilişkin programlanabilir sigortalara sahiptir. Bu tip PLD'ler programlanabilir salt okunur bellek (PROM) olarak adlandırılırlar. Programlanabilir dizi

lojiği (PAL) sigorta ile programlanabilir bir VE dizisi ve çıkışta sabit bir VEYA dizisinden oluşur. VE kapıları Boole fonksiyonundaki çarpım terimlerini oluşturmak üzere kullanılır. PLD'lerin daha esnek yapıda olanları programlanabilir lojik dizisi olarak bilinir. Hem VE hem de VEYA dizileri programlanabilir. Bu PLD elemanları daha sonra ayrıntılı olarak anlatılacaktır.



### İkili Toplayıcı ve Çıkarıcılar

Daha önce anlatılan tam toplayıcılar kullanılarak elde bitini de göz önüne alarak  $n$  bitlik iki sayının toplamı gerçekleştirilebilir. Bunun için aşağıdaki tabloyu incelemek gerekir.  $A_i$  toplanılan  $B_i$  toplanan  $C_i$  giriş eldesi  $C_{i+1}$  çıkış eldesi  $S_i$  ise toplam olmak üzere dört bitlik iki ikili sayının toplamı ( $A=1011$   $B=0011$   $S=1110$ )

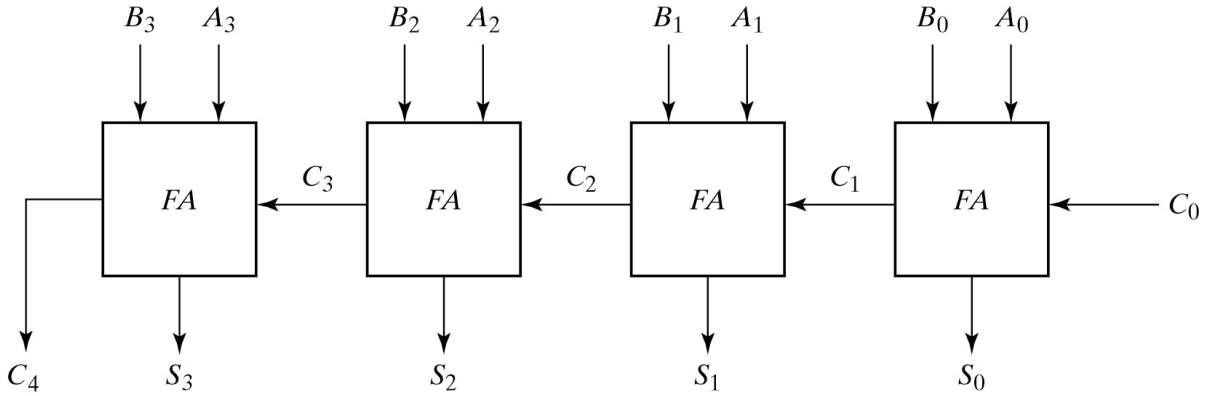
İndis $i$	3	2	1	0		Tam Toplayıcıdaki Karşılıkları
Giriş Eldesi	0	1	1	0	$C_i$	$z$
Toplanılan	1	0	1	1	$A_i$	$x$
Toplanan	0	0	1	1	$B_i$	$y$
Toplam	1	1	1	0	$S_i$	$S$
Çıkış Eldesi	0	0	1	1	$C_{i+1}$	$C$

Toplama işlemine en düşük anlamlı bittten başlanır. En düşük konumda yer alan  $C_0$  eldesi sıfır olmalıdır.  $C_{i+1}$  tam toplayıcının herhangi bir andaki çıkış eldesidir. Bu kendinden sonra yer alan daha yüksek anlamlı bitin toplamına giriş eldesi olarak gönderilir.

A ve B gibi n bitlik iki sayının toplamı seri ya da paralel olmak üzere iki şekilde yapılır. Seri toplama yönteminde sadece bir tam toplayıcı ve oluşan eldeyi tutmak üzere bir saklama aygıtı kullanılarak yapılır. Toplama işlemini seri olarak gerçekleştirmek için her bir zaman aralığında A ve B ikili sayılarının bir çift bitleri toplayıcıya giriş olarak verilir. Oluşan toplam sonucundaki elde biti ise bir sonraki girişe elde biti olarak aktarılır. Paralel toplama yönteminde ise n adet tam toplayıcı kullanılır. A ve B sayısının tüm bitleri aynı anda toplayıcı girişlerine verilir. Elde bitleri ise tam toplayıcıların çıkış elde bitleri bir sonraki daha yüksek anlamlı bitlerin toplamını yapan toplayıcıya giriş eldesi olarak aktarılır. Eldeler üretilir üretilmez çıkışlardaki toplam değeri ortaya çıkar.

### İkili Paralel Toplayıcı

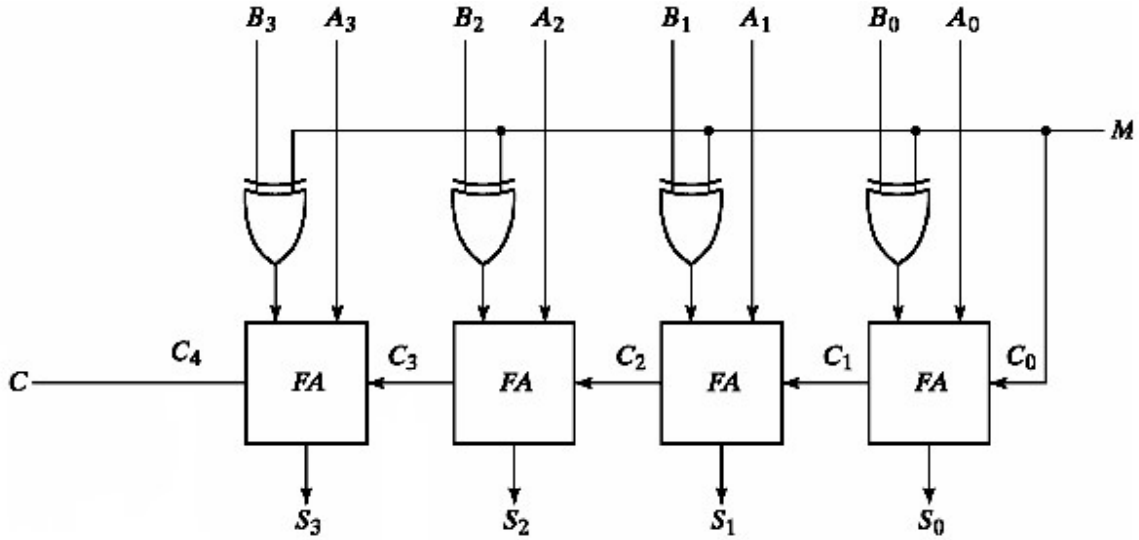
İkili paralel toplayıcı bir birine zincir şeklinde bağlanmış tam toplayıcılardan oluşur. her bir tam toplayıcının çıkış eldesi diğer bir tam toplayıcının giriş eldesine bağlanmıştır. Burada FA ile belirtilen elemanlar birer tam toplayıcı olup entegre tiptedir.



9 girişli bir devrenin tasarımında  $2^9=512$  içerikli bir doğruluk tablosu gerekecektir. Bu nedenle MSI tipindeki tüm devreler ile bilinen fonksiyonların peş peşe eklenmesi ile doğruluk tablosu yapmadan fonksiyon gerçekleştirilebilir.

### İkili Toplayıcı-Çıkarıcı

Bilindiği üzere sayısal bilgisayarlar çıkarma işlemi yapmaz. Bunun için toplama işlemi kullanılır. A-B işlemini yapmak için B'nin 2'ye tümleyeni alınıp A ile toplanması ile yapılmaktadır. 2'ye tümleyen işi B'nin 1'e tümleyeni alınıp en düşük anlamlı bite 1 eklenmesi ile elde edilir.



Yukarıdaki devrede M seçme biti 1 ise  $B \oplus 1 = B'$  olduğu için devre  $A+B+1$  işlemini gerçekleştirir. M girişi aynı zamanda  $C_0$  elde girişine de bağlıdır. Dolayısıyla devre, A ile B'nin 2'ye tümleyeninin toplamını gerçekleştirir.  $M=0$  ise  $B \oplus 0 = B$  ve  $C_0$  eldesi de 0 olduğu için toplama işi yapılır.

### Eldenin Yayılması

Toplama işleminin paralel olarak yapılabilmesi için eldenin tüm paralel toplayıcıların girişlerinde aynı anda mevcut olması gerekmektedir.  $C_0$  eldesi hesaplanmadan  $C_4$  eldesinin ve diğerlerinin kapı girişlerinde değerleri olsa bile sırayla değerlerine ulaşırlar ve tüm elde değerleri bulunduktan sonra çıkışlarda toplam değeri görülebilir. Bu yayılma gecikmesi kullanılan tümdevrelere bağlı olarak değişir. Yayılma gecikmesi kapı kademe sayısı ve kapının yayılma gecikmesi ile çarpımına eşittir.  $C_0$ 'dan  $C_4$ 'e kadar elde biti  $2 \times 4$  tane VEYA kapısından geçerek gelir. Eldenin yayılma gecikmesi ikili sayıların paralel toplanmasındaki hızı etkileyen en büyük faktörlerden biridir. Eldenin yayılmasını hızlandırmak için daha hızlı elemanlar kullanılabilir fakat fiziksel devrelerin hızları sınırlıdır. Hızı artırmak için diğer bir çözüm ise devre karmaşıklığını arttırarak elde yayılma süresi en aza indirgenebilir. Bu yöntem *elde öngörü*

prensibidir. Aşağıdaki devreyi göz önüne alalım. Bu devre bir tam toplayıcı devresidir. Görüldüğü üzere ara katlarda  $P_i$  ve  $G_i$  olmak üzere iki yeni değişken tanımlanmıştır.

$$P_i = A_i \oplus B_i$$

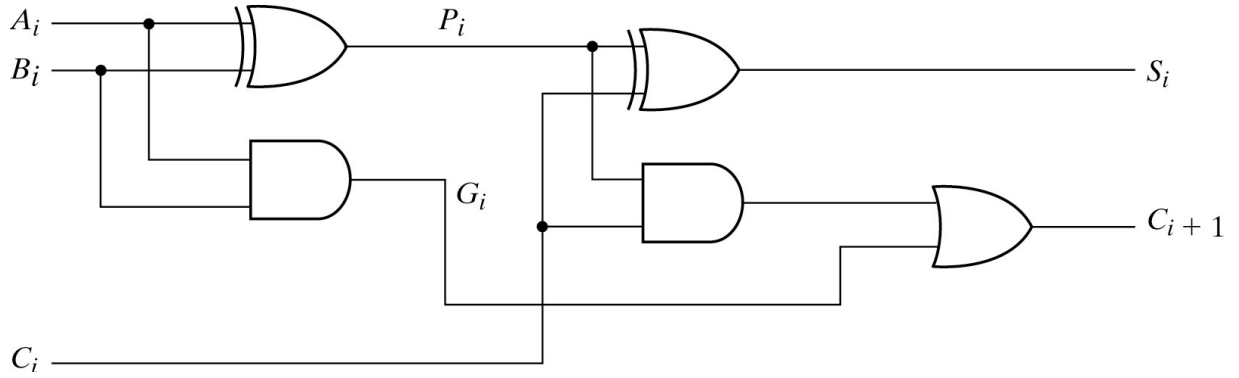
$$G_i = A_i B_i$$

Toplam ve elde çıkışları ise

$$S_i = P_i \oplus C_i$$

$$C_{i+1} = G_i + P_i C_i$$

$G_i$  elde üretici olarak adlandırılır.  $A_i$  ve  $B_i$ 'den her ikisi de aynı anda 1 ise çıkış elde biti vardır.  $P_i$  ise elde yayılması olarak adlandırılır çünkü, hem  $C_i$  hem de  $C_{i+1}$  yayılması ile ilişkili bir terimdir.

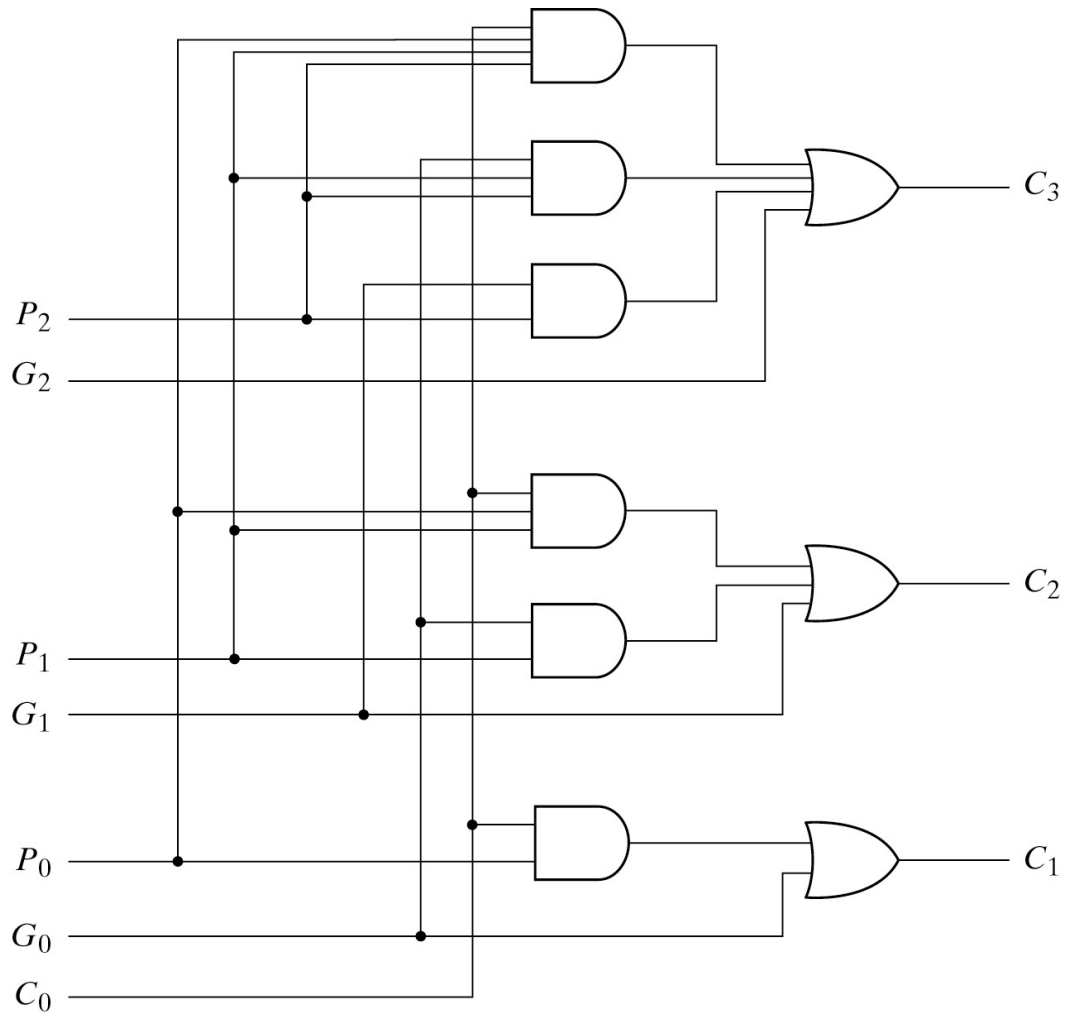


Çıkış elde biti için her kademedeki Boole fonksiyonları yazılacak olursa ve  $C_i$  önceki denklemlerde yerine yazılacak olursa;

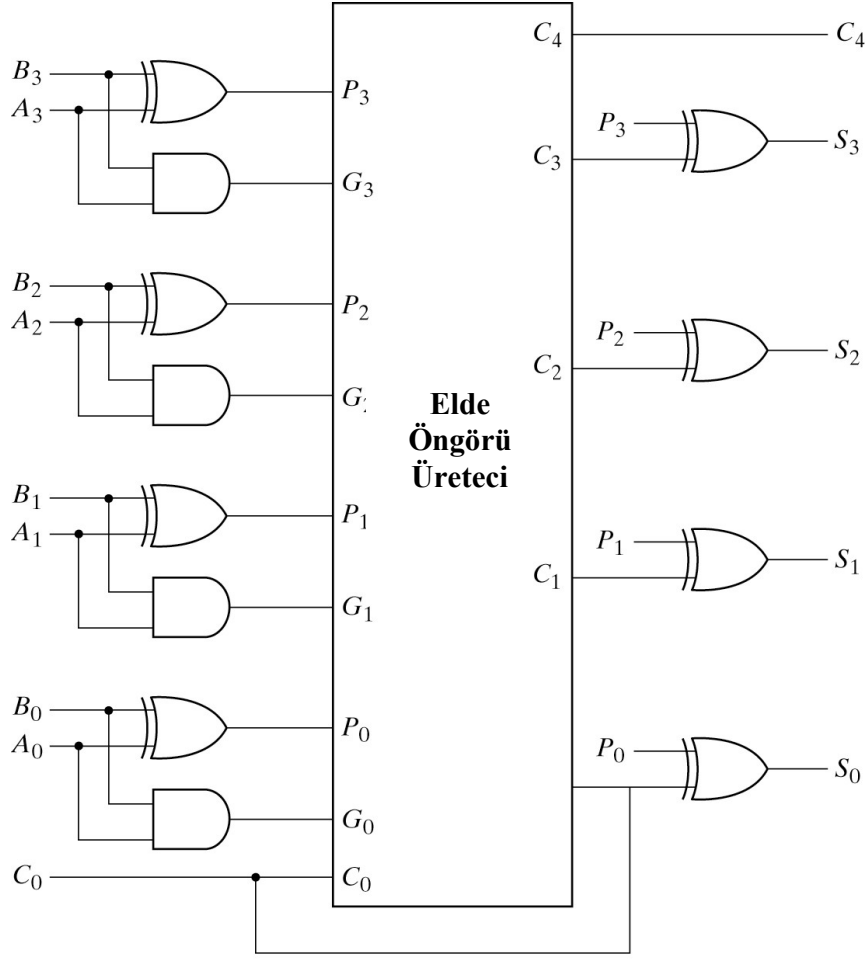
$$C_1 = G_0 + P_0 C_0$$

$$C_2 = G_1 + P_1 C_1 = G_1 + P_1 (G_0 + P_0 C_0) = G_1 + P_1 G_0 + P_1 P_0 C_0$$

$$C_3 = G_2 + P_2 C_2 = G_2 + P_2 (G_1 + P_1 G_0 + P_1 P_0 C_0) = G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0$$



Bu Boole fonksiyonlarını gerçekleştiren lojik devre yukarıda görülmektedir. Bu şekilde tüm eldeler bir birilerinin değerlerini beklemeden oluşturulabilir.  $C_3$ ,  $C_1$  ve  $C_2$  ile aynı zamanda oluşur.



Elde öngörülü 4 bitlik bir paralel toplayıcının yapısı yukarıda görülmektedir. Burada tüm kapıların gecikme süreleri aynı zamanda olmaktadır. Dolayısıyla tüm toplam çıkışları aynı gecikme sürelerine sahip olduklarından dolayı aynı anda üretilirler.  $C_4$  eldesi ise denklemlerde gösterilmemiştir fakat yerine koyma yöntemi ile elde edilebilir.

$$C_4 = G_3 + P_3 C_3 = G_3 + P_3 (G_2 + P_2 G_1 + P_2 P_1 G_0 + P_2 P_1 P_0 C_0) = G_3 + P_3 G_2 + P_3 P_2 G_1 + P_3 P_2 P_1 G_0 + P_3 P_2 P_1 P_0 C_0$$

### Onlu Toplayıcı

Bilgisayarlar ve hesap makineleri ikili kodlanmış şekilde onlu gösterimdeki sayıları kullanarak aritmetik işlemleri yürütürler. Onlu toplayıcılar her onlu haneyi kodlamak için 8 giriş ve elde biti için bir giriş ve çıkış için ise 4 bit veri ve bir bit de elde biti için olmak üzere 9 girişli 5 çıkışlı olurlar.

Doğruluk tablosu oluşturarak toplayıcı yapmak için  $2^9 = 512$  içerikli bir doğruluk tablosu gerekecektir. Bu doğruluk tablosundaki etkisiz koşullarda göz önüne alındığında tablonun büyük

bir kısmı bu etkisiz koşullarla dolacaktır. Bunu bilgisayarla yaptığımızda ise büyük ihtimalle düzensiz kapı desenleri ile karşılaşılacaktır. Buna alternatif bir yol olarak sadece kullanılan kombinasyonların bulunduğu ikili tam toplama devreleri kullanılabilir. Çıkış ise onlu kodların geçerli olduğu ikili kodlar şeklinde düzenlenmelidir.

### BCD Toplayıcı

Onlu sayıların 9 'u geçemeyeceği düşünülürse ve elde biti de 1 olursa toplam  $9+9+1=19$  olacaktır. Toplayıcı toplama sonucunu ikili formda oluşturacak ve 0-19 arasında bir sayı üretecektir. Bu ikili sayılar tabloda K, Z<sub>8</sub>, Z<sub>4</sub>, Z<sub>2</sub> ve Z<sub>1</sub> olarak gösterilmiştir. K eldeyi, Z'nin alt indisleri ise 4 bit karşılık düşen 8, 4, 2, 1 ağırlıklarını göstermektedir.

K	İkili Toplam				C	BCD Toplam				Onlu
	Z <sub>8</sub>	Z <sub>4</sub>	Z <sub>2</sub>	Z <sub>1</sub>		S <sub>8</sub>	S <sub>4</sub>	S <sub>2</sub>	S <sub>1</sub>	
0	0	0	0	0	0	0	0	0	0	0
0	0	0	0	1	0	0	0	0	1	1
0	0	0	1	0	0	0	0	1	0	2
0	0	0	1	1	0	0	0	1	1	3
0	0	1	0	0	0	0	1	0	0	4
0	0	1	0	1	0	0	1	0	1	5
0	0	1	1	0	0	0	1	1	0	6
0	0	1	1	1	0	0	1	1	1	7
0	1	0	0	0	0	1	0	0	0	8
0	1	0	0	1	0	1	0	0	1	9
0	1	0	1	0	1	0	0	0	0	10
0	1	0	1	1	1	0	0	0	1	11
0	1	1	0	0	1	0	0	1	0	12
0	1	1	0	1	1	0	0	1	1	13
0	1	1	1	0	1	0	1	0	0	14
0	1	1	1	1	1	0	1	0	1	15
1	0	0	0	0	1	0	1	1	0	16
1	0	0	0	1	1	0	1	1	1	17
1	0	0	1	0	1	1	0	0	0	18
1	0	0	1	1	1	1	0	0	1	19

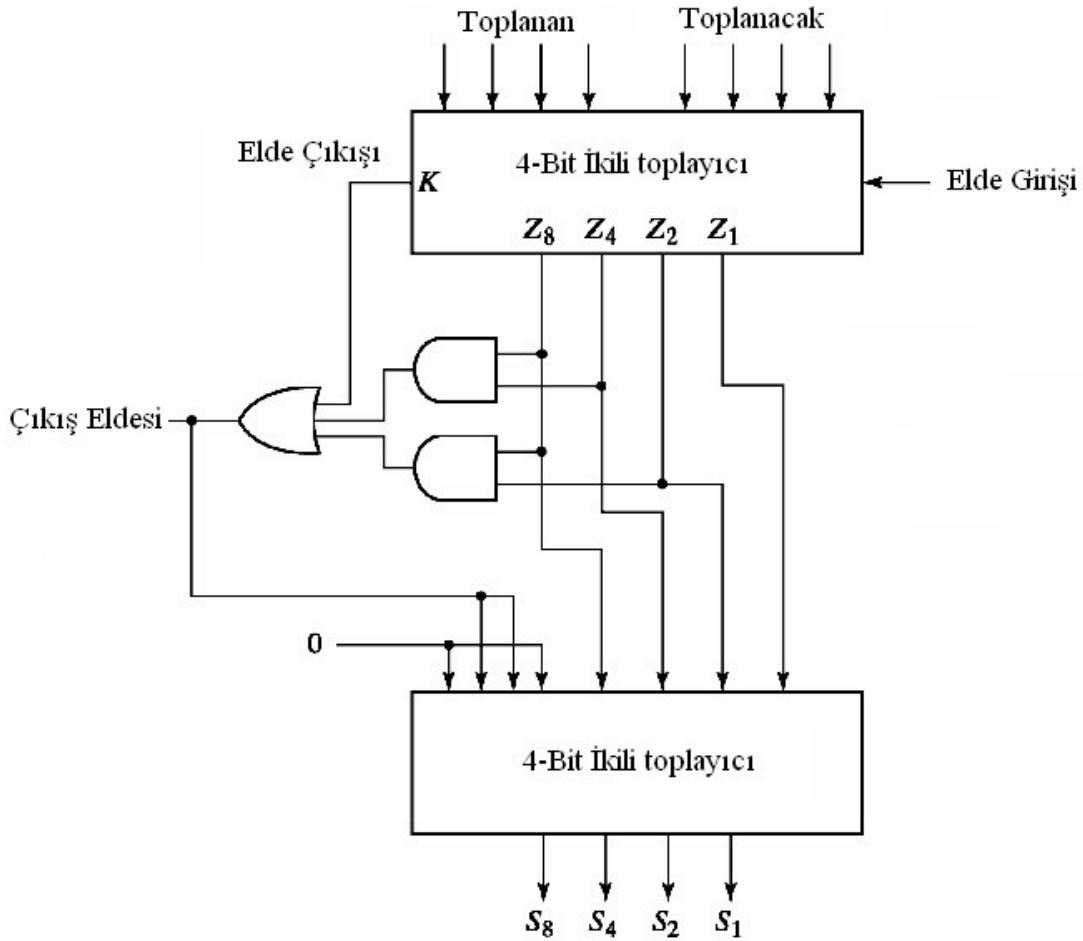
Tabloya dikkat edilecek olursa, ikili toplam sonucunun 9(01001) değerinden sonra değiştiği görülmektedir. BCD'de 9'dan büyük sayılar için herhangi bir gösterim karşılığı yoktur. Bu durumda C ile belirlenen bir çıkış eldesi oluşacaktır. Görüldüğü gibi 9'dan büyük değerler için BCD koduna 6 (0110) eklenmiştir. O halde iki sayı toplandıktan sonra çıkış eldesi olarak belirtilen C'nin bu toplama eklenmesi gerekmektedir. Bir başka deyişle bir düzeltme işlemi gerekmektedir. Toplam işlemi onlu 9 modülüne göre yapıldığı düşünülürse işlemin anlaşılması



daha kolaylaşabilir. Eğer Çıkış eldesi C, 1 olursa bu durumda bir düzeltme olacağı anlaşıldığı için ikinci 4 bitlik toplayıcının girişlerine daha önce elde edilen toplam ile beraber düzeltme sayısı 0CC0 eklenmelidir (C yazan yerlere 1 ya da 0 gelecek) 0 gelirse zaten toplam etkilenmeyecek ve herhangi bir düzeltmeye gerek olmadığı anlaşılabacaktır. O halde çıkış eldesi C'ye ne zaman gerek duyulacağı araştırılmalıdır. Tablo incelenecek olursa K, Z<sub>8</sub>, Z<sub>4</sub>, Z<sub>2</sub> ve Z<sub>1</sub>'e bağlı olarak C'nin değiştiği görülür. O halde elde edilecek Boole fonksiyonu aşağıdaki gibi olmalıdır. (Bu düzeltme devresinin nasıl elde edilir?)

$$C = K + Z_8 Z_4 + Z_8 Z_2$$

Aşağıda bir BCD toplayıcının blok diyagramı görülmektedir.



### Genlik Karşılaştırıcısı

İki sayının karşılaştırılması bu iki sayıdan hangisinin büyük, küçük ya da bir birlerine eşit olup olmadıklarının araştırılmasıdır. Bir genlik karşılaştırıcısı A ve B gibi iki sayıyı karşılaştıran kombinezonal bir devredir. Burada, 4 bitlik iki sayının karşılaştırılması için bir algoritma türetme yöntemi incelenecektir. Algoritma, izlendiğinde bir problemin çözümünü veren sonlu adımlar grubuyla belirlenen bir yoldur.

A ve B gibi 4 bitlik iki ikili sayısını göz önüne alalım. Sayıların katsayıları aşağıdaki gibi azalan şekilde olsun.

$$A=A_3A_2A_1A_0$$

$$B=B_3B_2B_1B_0$$

Bu hanelerden aynı indeks numarasına sahip olanların değerleri eşit ise sayılar birbirine eşittir. Her bitin eşitlik ilişkisi bir eşdeğer fonksiyonla aşağıdaki gibi ifade edilebilir.

$$x_i=A_iB_i+A_i'B_i' \quad i=0,1,2,3$$

Burada i indeksli bit değerlerinin her ikisi de bir birine eşitse  $x_i$  1 olacaktır. Bu durumda bu işlem tüm giriş bitlerine uygulanacak ve elde edilen tüm  $x_i$  değerleri VE işlemine tabi tutulacaktır. Eğer VE kapısının çıkışı 1 ise  $A=B$  demektir.

$$(A=B)=x_3x_2x_1x_0$$

En düşük anlamlı bittten başlanacak olursa diğer üç hanenin eşit olduğunu düşünelim

$x_3x_2x_1=1$  olduğu durum eğer  $A_0B_0'=1$  ise  $A_0=1$  ve  $B_0=0$  demektir ve  $A>B$  'dir anlamına gelir.  $A_0B_0=1$  ise  $A_0=0$  ve  $B_0=1$  olup  $A<B$  'dir anlamına gelir.

$$A>B \quad x_3x_2x_1A_0B_0'$$

$$A<B \quad x_3x_2x_1A_0'B_0$$

$x_0$  değerlerinin eşitliği bir önceki adımda kontrol edildi.  $x_3$  ve  $x_2$ 'nin eşit olduğu düşünülürse

$x_3x_2=1$  bu durumda 1 indeksli hanelere bakmak gerekecektir.  $A_1B_1'=1$  ise  $A>B$   $A_1B_1=1$  ise  $A<B$  'dir anlamına gelir.

$$A>B \quad x_3x_2A_1B_1'$$

$$A<B \quad x_3x_2A_1'B_1$$

$x_0$  ve  $x_1$  değerlerinin eşitliği bir önceki adımlarda kontrol edildi.  $x_3$  'ün eşit olduğu düşünülürse

$x_3=1$  bu durumda 2 indeksli hanelere bakmak gerekecektir.  $A_2B_2'=1$  ise  $A>B$   $A_2'B_2=1$  ise  $A<B$ 'dir anlamına gelir.

$$A>B \quad x_3A_2B_2'$$

$$A<B \quad x_3A_2'B_2$$

$x_3$  dışındaki diğer tüm eşitlikler kontrol edildiğine göre

$$A>B \quad A_3B_3'$$

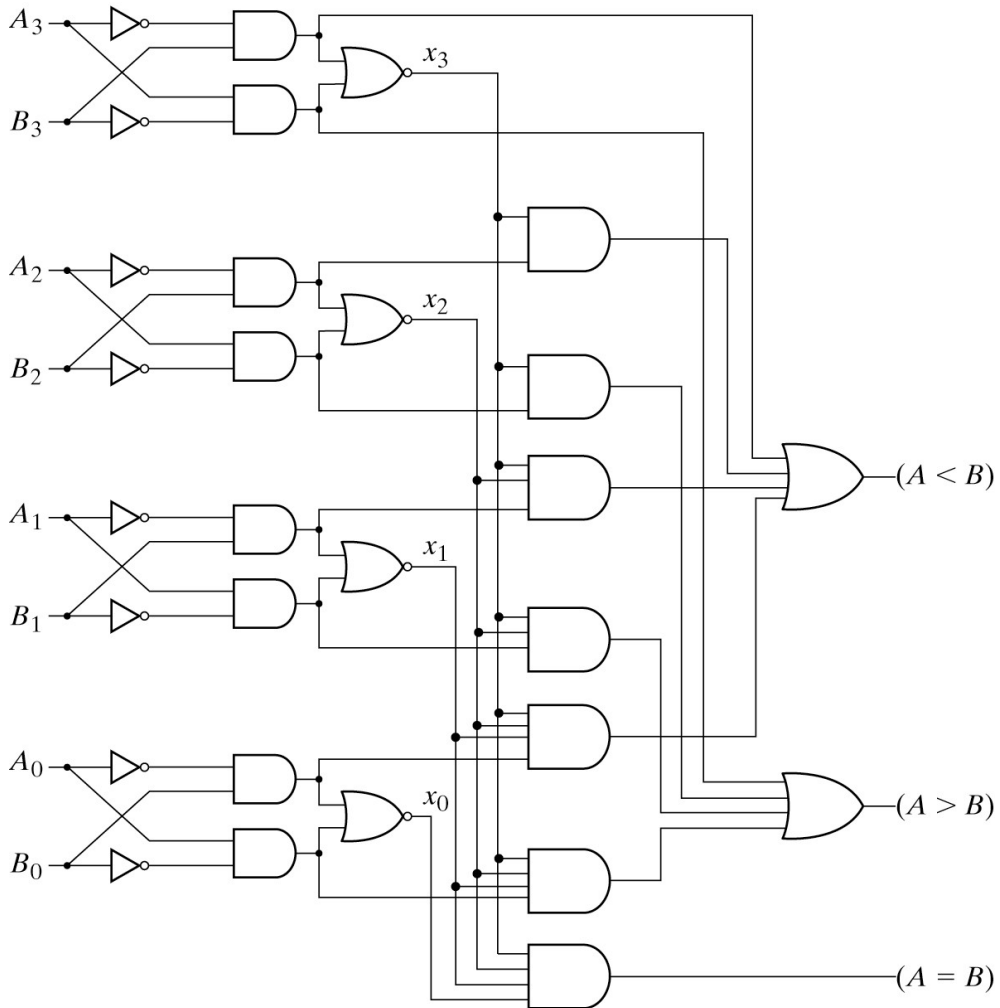
$$A<B \quad A_3'B_3$$

Tüm şartların aynı anda sağlanması için elde edilen tüm Boole fonksiyonları toplanmalıdır (VEYA).

$$(A>B)=x_3x_2x_1A_0B_0'+x_3x_2A_1B_1'+x_3A_2B_2'+A_3B_3'$$

$$(A<B)=x_3x_2x_1A_0'B_0+x_3x_2A_1'B_1+x_3A_2'B_2+A_3'B_3$$

Hangi durum sağlanıyorsa aşağıdaki kombinezonal devrede ilgili çıkış 1 olur.



Görüldüğü üzere devre belirli bir desene sahiptir. Giriş bit sayısının arttığı durumda Boole fonksiyonlarına bakıldığında bu desen rahatlıkla görülmektedir. Bu devrenin girişine BCD formundaki sayılar da uygulanabilir.