

S.4.20. Sekiz bit (ikili) uzunluğundaki bir sözcük ($a_7a_6...a_0$) için çift eşlik bit'i üretmek istendiğinde 7 adet XOR geçidi yeterli olabilir. Bunun için iki değişik devre yapısı öneriniz ve bu yapıların adlarını belirtiniz.

Eğer bir XOR geçidinin yayılma gecikmesi 10 ns ise, önerdiğiniz devrelerden her birinin toplam yayılma gecikmesini hesaplayınız ve hız açısından hangi devrenin daha üstün olduğunu belirtiniz.

S.4.21. Bir işyerinde araçların park edebileceği alanda yan yana 4 yer bulunuyor. Park yerleri A, B, C ve D ile gösteriliyor. Park yerlerinin boş ya da dolu (0: boş ; 1: dolu) olduğu ise a, b, c ve d değişkenleri ile gösteriliyor.



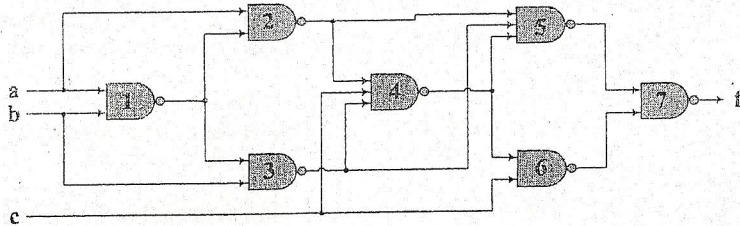
Yan yana en az 2 park yerinin boş olup olmadığını gösteren bir birleşimsel devre oluşturmanız isteniyor.



$y = 0$: yanyana 2 boş park yeri yok
 $y = 1$: yanyana en az 2 boş park yeri var

$y = f(a,b,c,d)$ işlevini bulup indirgeyin ve çarpımlar toplamı biçimindeki en küçük deyimi bulunuz.

S.4.22.



7 adet NAND geçidinden oluşan yukarıdaki birleşimsel devreyi çözümlüyüp $f(a,b,c)$ işlevini bulunuz. Bulduğunuz işlevi indirgeyip çarpımlar toplamı biçimindeki en küçük deyimi bulunuz.

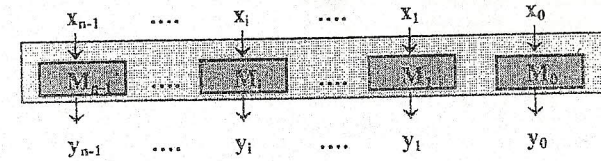
Not : Ara değer olarak, geçitlerin çıkışlarını y_1, y_2, \dots diye gösterebilirsiniz.

S.4.23. $x_3x_2x_1x_0$ bir BCD kod sözcüğü olsun.

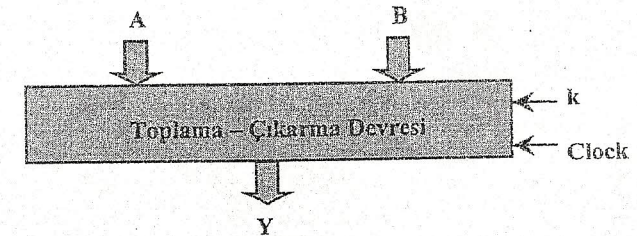
$y = f(x_3, x_2, x_1, x_0)$ işlevinin, $x_3x_2x_1x_0$ BCD kod sözcüğünün değeri asal olduğunda 1; asal olmadığında ise 0 değerini alması isteniyor. Önemli birleşimleri de dikkate alarak, y'nin en küçük çarpımlar toplamı biçimini bulunuz. Bu işlevi gerçekleştiren devreyi, en çok 5 adet "iki girişli NAND" geçidi ile oluşturunuz.

Not: Devrenin girişinde yalnız x_3, x_2, x_1 ve x_0 değerleri bulunuyor. Bu değerlerin tümüleri ise bulunmuyor.

S.4.24. Bir ikili sayının (X) değerinin 1 eksiltilmesi için, sayının ikilerinin sağdan sola doğru tarandığı bir algoritma yazınız. X ikili sayısının değerini 1 eksilterek $Y = X - 1$ değerini bulan bir devreyi modüler olarak tasarlamamız isteniyor. Bunun için devrenin bir modülünün (i. Modül) giriş ve çıkışlarını belirleyip tanımlayınız. Modülün çıkış işlevlerini bulunuz. İlk (M_0) ve son (M_{n-1}) basamaklardaki giriş/çıkış özelliklerini belirtiniz.



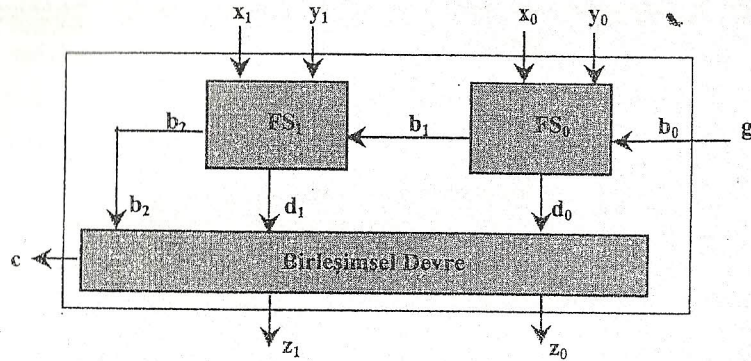
S.4.25.



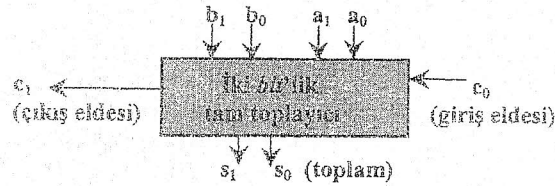
n bit'lik A ve B ikili sayılarının, $k = 0$ olduğunda toplamını, $k = 1$ olduğunda ise farkını bulan bir devreyi modüler yaklaşımla tasarlamamız isteniyor.

$$k = 0 \Rightarrow Y = A + B$$

$$k = 1 \Rightarrow Y = A - B$$

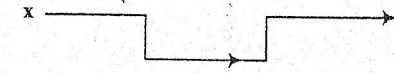
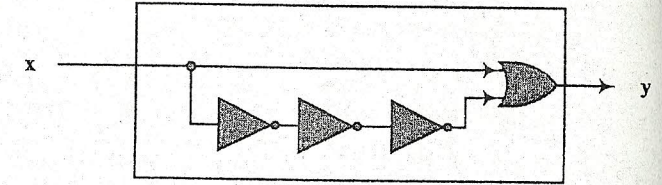


S.4.17. İki bir'lik tam toplayıcı devreyi bir bütün olarak tasarlamamız isteniyor.



İki bir'lik tam toplayıcı devrenin çıkış işlevlerini, doğruluk çizelgesi, harita ve benzerini oluşturmadan, mantıksal bir düşünceye dayandırarak ve bu düşünceyi açıkça belirterek (toplamın 1 olması için gereklidir gibi) doğrudan yazınız. Çıkış işlevleri istenilen biçimde yazılabilir.

S.4.18. Gerekli açıklamaları yaparak aşağıdaki devrenin zaman çizeneğini tamamlayınız.



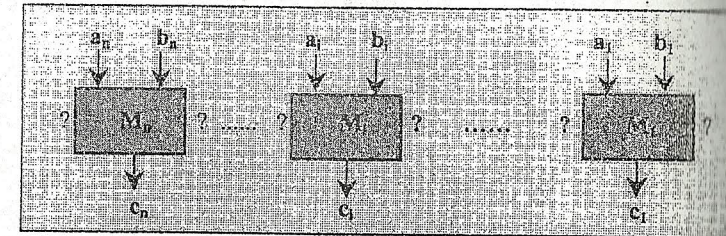
y ?

S.4.19. İki ikili sayının en büyüğünü bulan devreyi iteratif devre yaklaşımıyla tasarlamamız isteniyor.

$$A = a_n \dots a_i \dots a_1$$

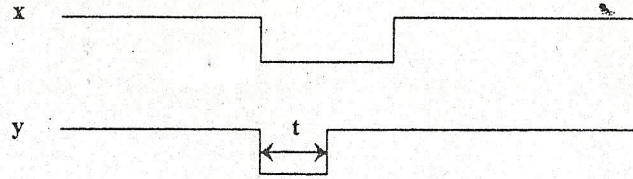
$$B = b_n \dots b_i \dots b_1$$

$$C = \max(A, B) = c_n \dots c_i \dots c_1$$



Önce A ve B'yi soldan sağa tarayarak C'yi bulan bir algoritma yazınız. Daha sonra devrenin bir modülünü (M_i) tasarlayınız. Bunun için veri giriş ve çıkışlarına (a_i , b_i , c_i) ek olarak en az sayıda iteratif denetim giriş/çıkışı kullanınız. Bu giriş/çıkışları tanımlayınız. Modülün çıkış işlevlerini en küçük çarpımlar toplamı biçiminde bulunuz. İlk ve son (1. ve n.) modüllerin giriş/çıkış özelliklerini belirtiniz.

Y.4.18.



$$t = 3\tau$$

τ = Bir NOT geçidinin gecikme süresi

Y.4.19. Algoritma

$k = 0; i = n;$

while $i > 0$ do;

if $(k = 0)$ and $(a_i > b_i)$ then $k = 1;$

if $(k = 0)$ and $(a_i < b_i)$ then $k = 2;$

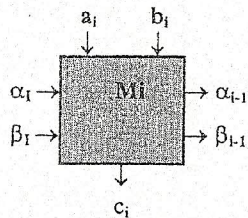
if $(k = 0)$ or $(k = 1)$ then $c_i = a_i$

else $c_i = b_i;$

$i = i - 1;$

end;

Algoritmadan da görüldüğü gibi 3 değerli bir denetim değişkenine (k) ihtiyaç vardır. Bunu için de iki tane ikili değişken kullanmak gerekir. Bu değişkenler α ve β olsun.



Denetim imlerinin anlamları:

$\alpha_i = 0, \beta_i = 0 \Rightarrow i.$ basmağın solunda A ve B'nin tüm ikiliği eşit.

$\alpha_i = 1, \beta_i = 0 \Rightarrow A > B$

$\alpha_i = 0, \beta_i = 1 \Rightarrow A < B$

$\alpha_i = 1, \beta_i = 1 \Rightarrow$ önemsiz birleşim

Devrenin çıkış işlevleri:

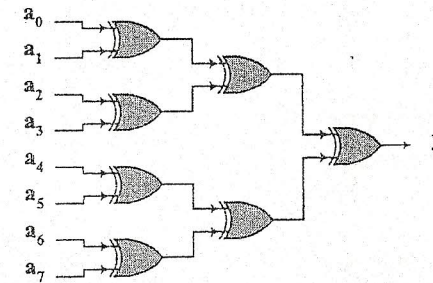
$$c_i = \beta_i' a_i + \alpha_i' b_i$$

$$\alpha_{i-1} = \alpha_i + \beta_i' a_i b_i'$$

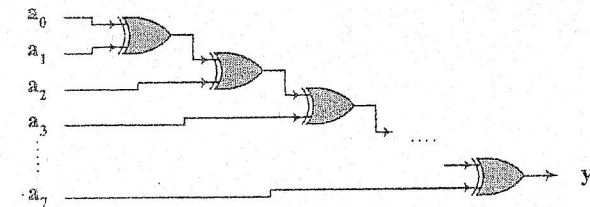
$$\beta_{i-1} = \beta_i + \alpha_i' a_i' b_i$$

Y4.20.

a) Ağaç Devre Yapısı



b) İteratif (ötelemeli) Devre Yapısı



Ağaç devre yapısında yayılma gecikmesi = $3\tau = 30$ ns

İteratif devre yapısında yayılma gecikmesi = $7\tau = 70$ ns

Yayılma gecikmesi açısından, ağaç devre yapısı daha üstündür.

$$Y.4.21. \quad y = f(a,b,c,d) = a'b' + b'c' + c'd'$$

$$Y.4.22. \quad y_1 = (ab)'$$

$$y_2 = a' + ab = a' + b$$

$$y_3 = b' + ab = a + b'$$

$$y_4 = (cy_2y_3)' = c' + y_2' + y_3'$$

$$y_5 = (y_2y_3y_4)'$$

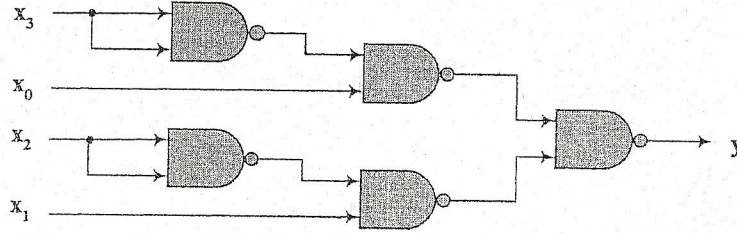
$$y_6 = (cy_4)'$$

$$f = y_7 = (y_5y_6)' = y_5' + y_6' = y_2y_3y_4 + cy_4 = y_2y_3(c' + y_2' + y_3') + c(c' + y_2' + y_3')$$

$$f = y_2y_3c' + cy_2' + cy_3' = (a'+b)(a+b')c' + c(a'+b)' + c(a+b)'$$

$$f = abc' + a'b'c' + ab'c + a'bc$$

Y.4.23. $y = x_3'x_0 + x_2'x_1$

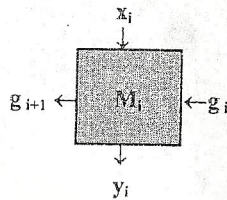


Y.4.24. Algoritma:

```

dur = 0; i = -1;
while dur = 0 do;
    i = i + 1;
    if (xi = 1) then xi = xi';
                    dur = 1;
    if (i = n - 1) then dur = 1;
end;
```

Devrenin bir modülünün işlevsel çizeneği:



Eğer i. basamağın sağındaki basamaklarda 1 yoksa : $g_i = 0$

Eğer i. basamağın sağındaki basamaklarda 1 varsa : $g_i = 1$

Eğer i. basamak ve sağındaki basamaklarda 1 yoksa : $g_{i+1} = 0$

Eğer i. basamak ve sağındaki basamaklarda 1 yoksa : $g_{i+1} = 1$

Çıkış işlevleri:

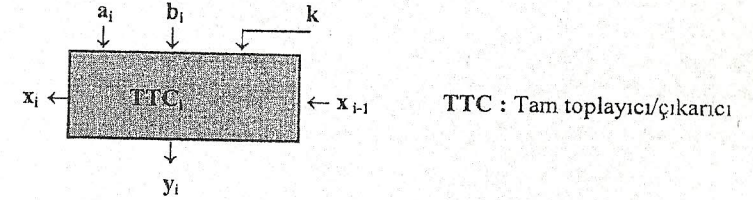
$$y_i = x_i g_i + x_i' g_i'$$

$$g_{i+1} = x_i + g_i$$

İlk basamakta $g_0 = 0$ olmalıdır.

Son basamakta eğer $g_n = 0$ olursa: $X = 0$ 'dır ve $Y = 2^n - 1$ hesaplanmıştır.

Y.4.25.



TTC : Tam toplayıcı/çıkarıcı

En küçük çıkış işlevleri:

$$y_i = x_{i-1}a_i'b_i' + x_{i-1}'a_i'b_i + x_{i-1}'a_i'b_i + x_{i-1}a_i'b_i = x_{i-1} \oplus a_i \oplus b_i$$

$$x_i = x_{i-1}b_i + kx_{i-1}a_i' + ka_i'b_i + k'x_{i-1}a_i + k'a_i'b_i$$

Y.4.26. $b_0 = x_3'x_2'$

$$b_1 = x_3x_2' + x_3'x_2$$

$$b_2 = x_3x_2$$

$$a_0 = x_1'x_0'$$

$$a_1 = x_1x_0' + x_1'x_0$$

$$a_2 = x_1x_0$$

$$z_0 = a_0b_0$$

$$z_1 = a_0b_1 + a_1b_0$$

$$z_2 = a_0b_2 + a_1b_1 + a_2b_0$$

$$z_3 = a_1b_2 + a_2b_1$$

$$z_4 = a_2b_2$$