



İSTANBUL TEKNİK ÜNİVERSİTESİ  
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ  
LOJİK DEVRELERİ LABORATUVARI  
DENEY RAPORU



DENEYİN ADI : **TASARIM ve UYGULAMA**  
**Seri Aritmetik ve Lojik Birim**

RAPORU HAZIRLAYAN : **BEYCAN KAHRAMAN**

**GRUP NO : Ç3**

Toplam 5( beş ) sayfadan oluşan bu raporu akademik dürüstlük kurallarının tümüne uygun davranarak hazırladım. Kismen de olsa açıkça belirtilen alıntılar dışında alıntı yapmadım.

İMZA

DENEY TARİHİ : **13.04.2005**

RAPOR TESLİM TARİHİ : **20.04.2005**

DENEYİ YAPTIRAN : **A. Çağatay TALAY**

ÖĞRETİM ELEMANI :

Bu kısım raporun değerlendirilmesi için kullanılacaktır.

## TASARIM ve UYGULAMA - Seri Aritmetik ve Lojik Birim

**I. Amaç:** Şu ana kadar öğrendiğimiz kuramsal ve uygulama bilgileriyle tasarımı tamamen bize bırakılmış bir devre tasarlamak. Hazırlayacağımız tasarımda öncelikle bir bit üzerinde 8 farklı işlem yapabilen bir ALU oluşturduk, daha sonra da bunu kullanarak 4 bitlik işlemleri gerçekledik.

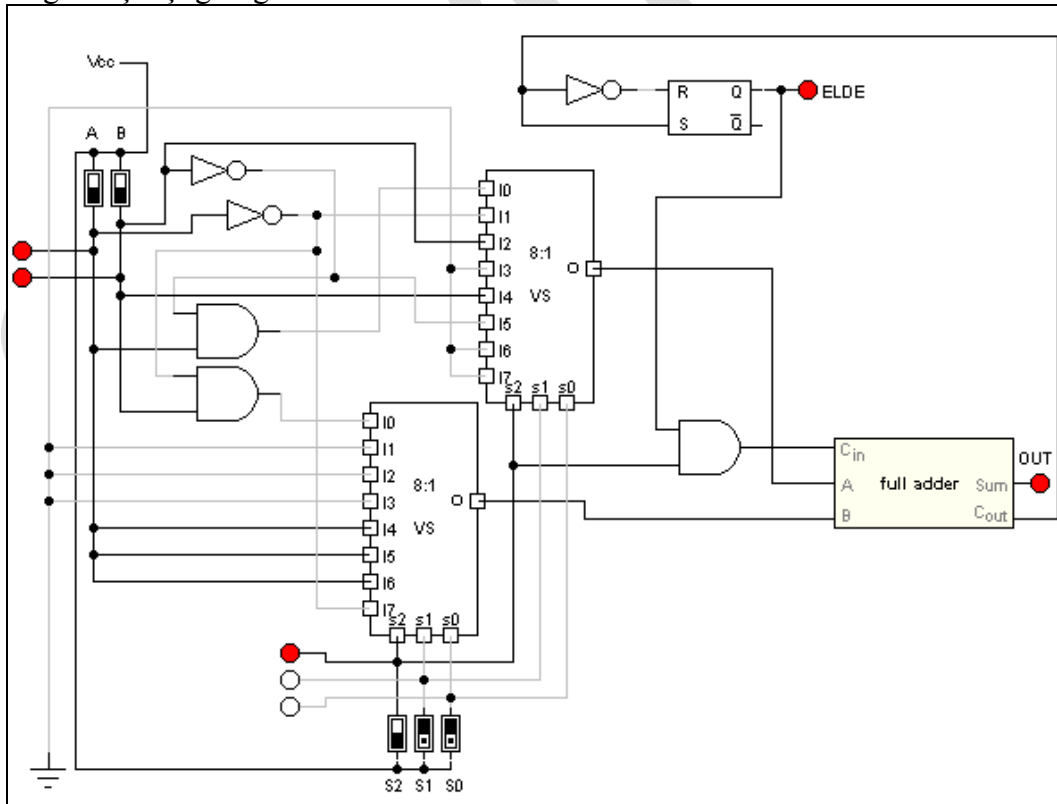
### II. Yapılan İşlemler:

#### 1. SORU:

Tasarladığımız devre aşağıdaki şekildeki devre gibiydi. Bu devreyi tasarlarken iki tane 8:1 veri seçici kullandık çünkü yapılacak olan işlemleri gerçeklerken 4 erli gruplara ayıramadık. Bunun sonucunda, devrenin her farklı işlem grubunda olması gereken değerleri ayrı ayrı toplama devresine vererek işlemimizi gerçekledik. Bu sırada Elde bitini Elde Flipflopunda tuttuk. Bunun sebebi, bir sonraki aşamada 4 bitlik veriyi toplarken her bittten sonra elde edilecek verinin bir sonraki işleme saklanmasıdır.

1. $A + B + E$	5. $A \text{ O } B$
2. $A + B' + E$	6. $A'$
3. $A + E$	7. $B$
4. $A' + E$	8. $0$

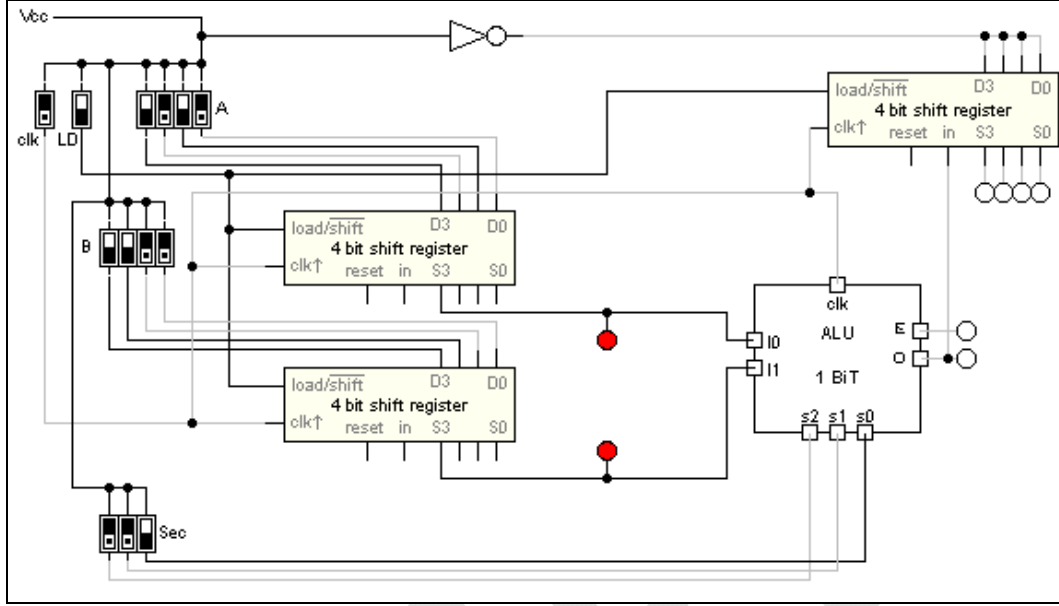
Yukarıdaki gibi grupladığımız devrenin şemasına bakarak yaptığımız deneyde istenen sonuca ulaşabildik ve devrenin doğru çalıştığını gördük. Yine de bu ilk soru için elde flipflopunu girişi yerine ayrı bir giriş ucu verilebilirdi. Deneyde her ikisi için de tasarımın sağlıklı çalıştığını gördük.



Bir bitlik tutuculu ALU devresi

İkinci deneyde yukarıda oluşturduğumuz devreyi kullanarak 4 bitlik sayılara istenen işlemi uygulamayı gerçekledik. Deneydeki Elde flipflobunu D tipi

flipflop ile deđiřtirdik, böylece eldenin sadece zaman tetiklemelerinde çalıştığını gözlemledik. Deneyde oluşturduğumuz devreden bazı işlemlerde olumlu yanıt alabilmemize rağmen, devrenin bazılarında çalışmadığını gördük. Zamanımızın yetmemesi sebebiyle deneyi tamamlayamadık.



4 Bitlik Veride 1 Bitlik ALU yardımıyla seri işlemlerin gerçekleştirilmesi

## 2. SORU:

İstenen çarpma devresinin doğruluk tablosunu çizmeye kalkarsak

		$a_1$	$a_2$	$a_3$	$a_4$
$x$	$b_1$	$a_1b_1$	$a_2b_1$	$a_3b_1$	$a_4b_1$
	$b_2$	$a_1b_2$	$a_2b_2$	$a_3b_2$	$a_4b_2$
	$b_3$	$a_1b_3$	$a_2b_3$	$a_3b_3$	$a_4b_3$
	$b_4$	$a_1b_4$	$a_2b_4$	$a_3b_4$	$a_4b_4$

SONUÇ

Elde edeceğimiz sonucun 3. bitinde 4 terim işleme sokulduğundan devremiz bu işlemi gerçekleyemez. Bunu gerçekleyen devrenin baştan gerçekleştirilmesi gereklidir.

## III. Kısa Sınav Soruları:

1. Seri aritmetik lojik birimle paralel aritmetik lojik birim arasındaki temel farklılıkları açıklayınız.
2. “Çıkarma” ve “iki arttırma” işlemlerini gerçekleştiren 2-bitlik ALB tasarlayınız.

## Cevaplar:

1. Seri aritmetik lojik birim paralel aritmetik lojik birime göre oldukça yavaş çalışır. Paralel aritmetik lojik birimde istenen işlem bir çırpıda hesaplanırken, seri aritmetik lojik birimde ise n bitlik veri için n ayrı işlem

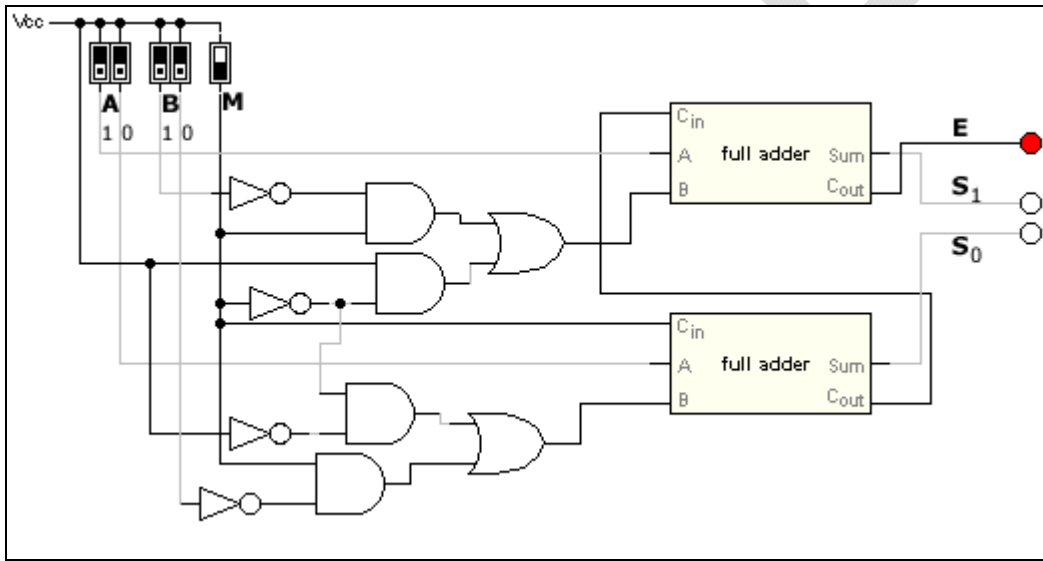
yapılır ve bu n işlem süresi boyunca beklenir. Paralel aritmetik lojik birimin çıkışları veri olarak kullanılabilirken, seri aritmetik lojik birim çıkışlarının giriş ve çıkışlarında ötelemeli saklayıcıların kullanılması zorunludur.

2.

Çıkarma/ 2 artırma

A\B	00	01	11	10
00	00 / 10	11 / 10	01 / 10	10 / 10
01	01 / 11	00 / 11	10 / 11	11 / 11
11	11 / 01	10 / 01	00 / 01	01 / 01
10	10 / 00	01 / 00	11 / 00	00 / 00

Devremizde normal bir iki bitlik toplama çıkarma yapabilen devre kullanabiliriz. Seçme girişi "1" iken çıkarma işlevini yerine getirecek olan devre seçme girişi "0" iken 10 değeriyle toplama yapmalıdır.



**IV. Yorum ve Görüşler:** Bu hafta, tasarımı tamamen bize bırakılan deneyi, önceki bilgilerimizi de kullanarak gerçekledik. Şu ana kadar yaptığımız tüm deneylerdeki bilgilerimizi pekiştiren ve gerçekten ne kadarını öğrendiğimizi anlamamızı sağlayan bir deney oldu. Deneyde yapılan bağlantı sayısının fazla olması, deneyi bitirebilmemizi önledi. Hem kablo sayısının azalması hem de yapılabilecek hata oranının artması deneyi bitirebilmemiz için ayarlanmış sürenin yetmemesine sebep oldu.