



GEBZE TEKNİK ÜNİVERSİTESİ  
ELEKTRONİK MÜHENDİSLİĞİ

ELM235

LOJİK DEVRE TASARIM LABORATUVARI

LAB 0x1 Deney Raporu

Lojik Devreler ve Tasarım Laboratuvarı

Hazırlayanlar
1) 171024086 – Berat KIZILARMUT
2) 1801022037 – Ömer Emre POLAT

## 1. Giriş

Bu deney kapsamında, Boolean Cebri kullanarak, çeşitli lojik operasyonları inceleyip, aynı işlemi yapacak şekilde basitleştireceğiz. Bununla birlikte SOP ve POS gösterimleri kullanarak, verilen bir sinyal girişine göre istenen sinyal çıktısını verebilen bir devre tasarlayacağız.

## 2. Problemler

### 2.1. Problem I – Boolean Cebri

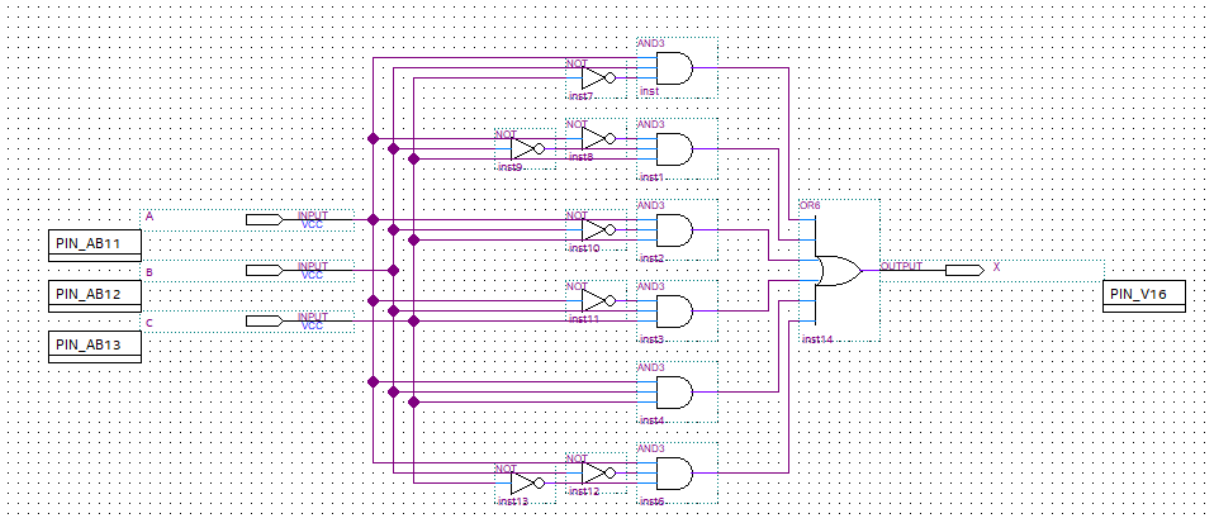
#### 2.1.1. Deneyin Yapılışı

##### A) Doğruluk Tablosu

A	B	C	X
0	0	0	0
1	0	0	1
0	1	0	0
0	0	1	1
1	1	0	1
1	0	1	1
0	1	1	1
1	1	1	1

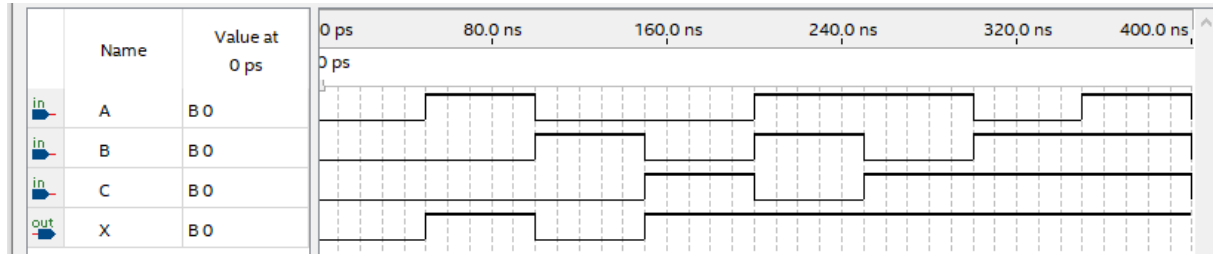
**Tablo1.** Doğruluk Tablosu

##### B) Devre Şeması ve Simülasyon



*Şekil 1 Devre Şeması*

Boolean Cebri ile basitleştirilmemiş devrenin Quartus üzerindeki şeması.



Şekil 2 Giriş ve çıkış sinyalleri

Bütün olasıklar da sinyal girişi ve o girişlere göre elde ettiğimiz çıkışlar.

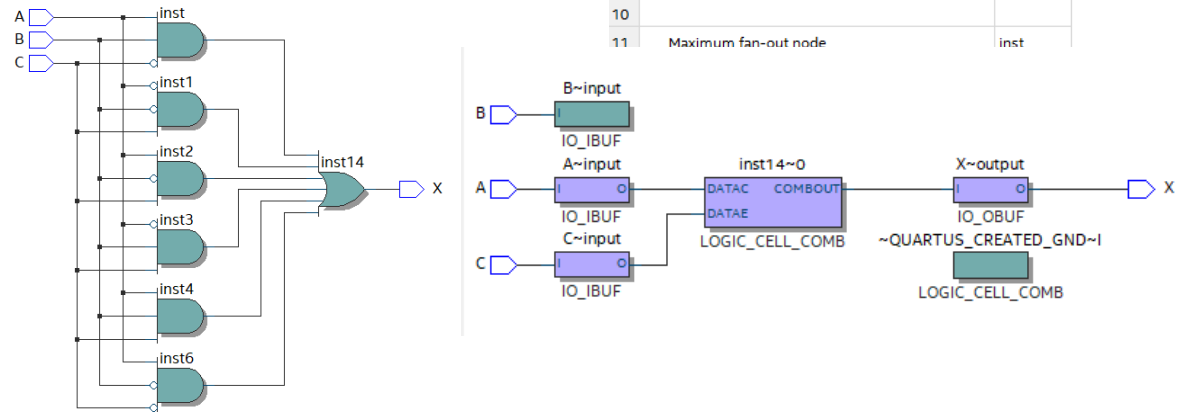
### C) Simülasyon ve Teorik Sonuçların Karşılaştırılması

Simülasyonda elde ettiğimiz sonuç teorik olarak hesapladığımız doğruluk tablosu ile uyumaktadır. Lojik devre beklenildiği gibi davranmıştır.

### D) Analiz ve Sentez Sonuçları

Basitleştirilmemiş devrede oldukça fazla sayıda lojik eleman bulunmasına rağmen kullandığımız kaynaklar beklenilenden daha az miktardadır.

	Resource	Usage
1	Estimate of Logic utilization (ALMs needed)	1
2		
3	Combinational ALUT usage for logic	1
1	-- 7 input functions	0
2	-- 6 input functions	0
3	-- 5 input functions	0
4	-- 4 input functions	0
5	-- <=3 input functions	1
4		
5	Dedicated logic registers	0
6		
7	I/O pins	4
8		
9	Total DSP Blocks	0
10		
11	Maximum fan-out node	inst



Şekil 3 RTL ve Eşleşme Ardı Devre Şemaları

Quartus Prime kurulan devreyi mantıksal olarak inceleyip, daha basit düzeyde kurduğu için, B girişinin çıkışa etki etmediğini tespit edip devreden çıkarmıştır.

### 2.1.2. Sonuçların Yorumu

Kurulan devre program tarafından daha karışık olmasına rağmen olabildiğince basitleştirilmiştir. Bu basitleştirilme sonucunda B girişinin mantıksal işlemlerin sonucuna bir etkisi olmadığı tespit etmiştir ve lojik devreden çıkarılmıştır. Bu işlemler sonucunda kurduğumuz lojik devre basitleştirilerek hafızada daha az yer kaplamıştır.

## 2.2. Problem II

### 2.2.1. Teorik Araştırma

Boolean Cebrinde mantıksal basitleştirme işlemleri

### 2.2.2. Deneyin Yapılışı

#### A) Devrenin Sadeleştirilmesi

Problem I’de ki devre Boolean Cebri yardımı ile bu işlemler ile sadeleştirildi;

$$A B \bar{C} + \bar{A} \bar{B} C + A B C + A \bar{B} C + A B \bar{C} + A B C$$

$$B(A\bar{C} + \bar{A}C + AC) + \bar{B}(A\bar{C} + \bar{A}C + AC)$$

$$(B + \bar{B})(A\bar{C} + \bar{A}C + AC)$$

$$1(A\bar{C} + \bar{A}C + AC + AC)$$

$$A(\bar{C} + C) + C(\bar{A} + A)$$

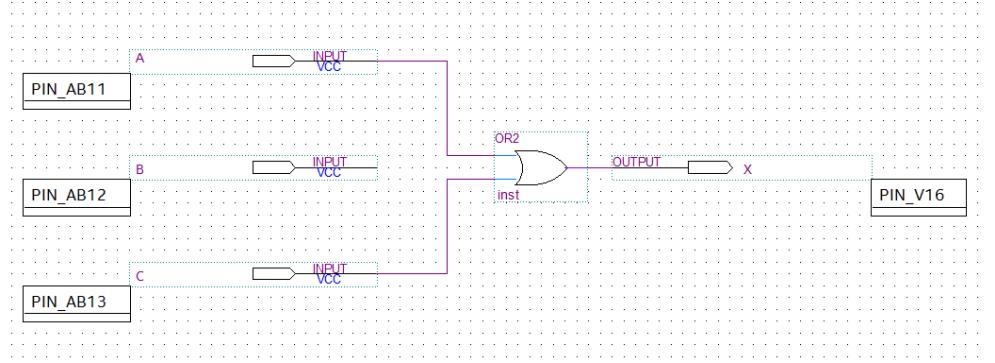
$$A1 + C1 = A + C$$

#### B) Sadeleştirilmiş Denklemin Doğruluk Tablosu

A	C	X
1	1	1
1	0	1
0	1	1
0	0	0

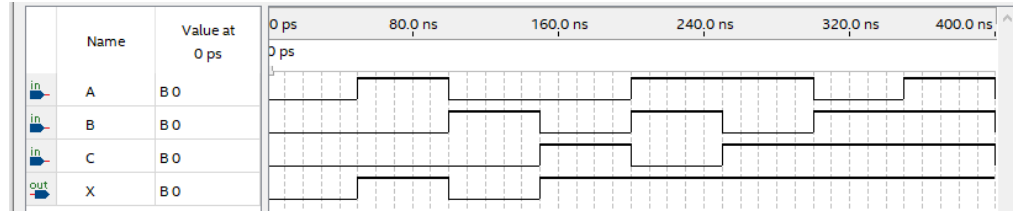
**Tablo2.** Doğruluk Tablosu

### C) Devre Şeması ve Simülasyon



Şekil 4 Basitleştirilmiş devre şeması

Boolean Cebri ile basitleştirilmemiş devrenin Quartus üzerindeki şeması. Basitleştirme işlemi sonucu B girişinin lojik devrenin sonucuna etki etmediği tespit edilmiştir.



Şekil 5 Giriş ve Çıkış Sinyalleri

Devrenin basitleştirilmesine rağmen devrenin girişlere verdiği tepki aynı kalmıştır, aynı çıktılar elde edilmiştir.

### D) Simülasyon ve Teorik Sonuçların Karşılaştırılması

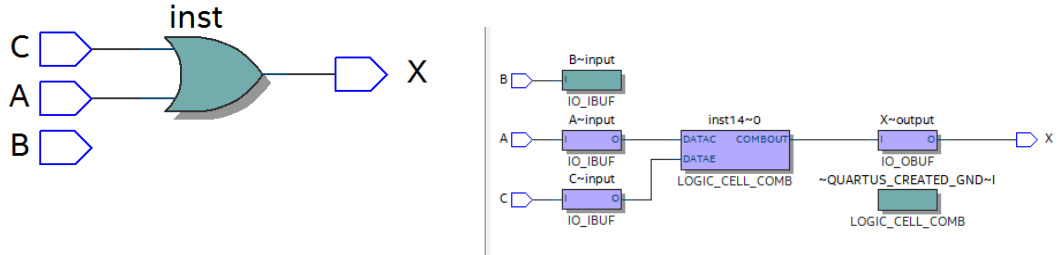
Elde edilen doğruluk tablosu ile giriş ve çıkış sinyalleri aynı sonuçları vermektedir. Devre olması gerektiği gibi bir problem bulunmadan işlev görmektedir.

## E) Analiz ve Sentez Sonuçları

Analiz ve Sentez sonuçları deneyin bir önceki adımına göre farklılık göstermemiştir. Bunun sebebi A adımında yapılan sadeleştirme işlemlerini Quartus Prime otomatik olarak yapılmasıdır.

Bu işlemler bir önceki adımda da program tarafından yapıldığı için iki devrenin analiz sonuçları aynıdır.

	Resource	Usage
1	Estimate of Logic utilization (ALMs needed)	1
2		
3	▼ Combinational ALUT usage for logic	1
1	-- 7 input functions	0
2	-- 6 input functions	0
3	-- 5 input functions	0
4	-- 4 input functions	0
5	-- <=3 input functions	1
4		
5	Dedicated logic registers	0
6		
7	I/O pins	4
8		
9	Total DSP Blocks	0
10		
11	Maximum fan-out node	inst
12	Maximum fan-out	1
13	Total fan-out	7
14	Average fan-out	0.78



Şekil 6 RTL ve Eşleşme Ardi Devre Şemaları

### 2.2.3. Sonuçların Yorumu

Bir önceki adımdaki karışık devre Boolean Cebri yardımı ile basitleştirilmiştir. Basitleştirilmiş devre ile programın kendisinin yaptığı basitleştirme hareketleri sonuç olarak aynı devreyi vermiştir. Yani doğru bir basitleştirme süreci yapılmıştır. Analiz ve sentez sonuçlarında da görüldüğü gibi birbirinden oldukça farklı gözükse de bu iki lojik devre aslında mantıksal olarak aynı işlemleri yapmaktadır.

### 2.3. Problem III

#### 2.3.1. Teorik Araştırma

SOP (Sum of Products) ve POS (Product of Sums) yazımı ve basitleştirilmesi.

#### 2.3.2. Deneyin Yapılışı

##### A) Doğruluk Tablosu

G	T	U	E	X	Y
1	1	1	1	0	0
0	1	1	1	0	1
1	0	1	1	0	0
0	0	1	1	0	1
1	1	0	1	0	1
0	1	0	1	0	1
1	0	0	1	0	1
0	0	0	1	0	1
1	1	1	0	1	0
0	1	1	0	1	1
1	0	1	0	1	0
0	0	1	0	1	1
1	1	0	0	0	1
0	1	0	0	0	1
1	0	0	0	1	1
0	0	0	0	1	1

Tablo3.Doğruluk Tablosu

##### B) SOP ve POS denklemleri

$$\text{SOP } X = GTU\bar{E} + \bar{G}TUE + G\bar{T}U\bar{E} + \bar{G}\bar{T}U\bar{E} + G\bar{T}\bar{U}\bar{E} + \bar{G}\bar{T}\bar{U}\bar{E}$$

$$G(TU\bar{E} + \bar{T}U\bar{E} + \bar{T}\bar{U}\bar{E}) + \bar{G}(TUE + \bar{T}U\bar{E} + \bar{T}\bar{U}\bar{E})$$

$$(G + \bar{G})(TU\bar{E} + \bar{T}U\bar{E} + \bar{T}\bar{U}\bar{E})$$

$$1(TU\bar{E} + \bar{T}U\bar{E} + \bar{T}\bar{U}\bar{E})$$

$$\bar{E}(TU + \bar{T}U + \bar{T}\bar{U} + \bar{T}\bar{U})$$

$$\bar{E}(U(T + \bar{T}) + \bar{T}(U + \bar{U}))$$

$$\bar{E}(U1 + \bar{T}1)$$

$$X = \bar{E}(U + \bar{T})$$

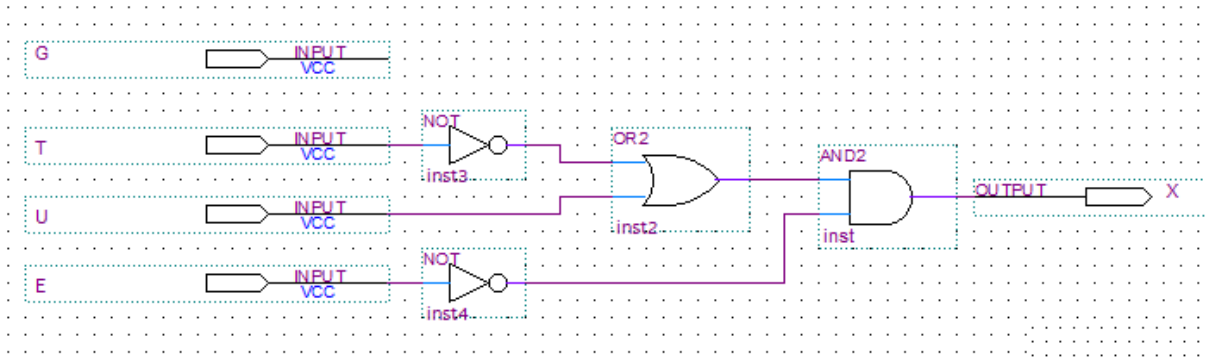
$$POS Y = (\bar{G} + \bar{T} + \bar{U} + \bar{E})(\bar{G} + T + \bar{U} + \bar{E})(\bar{G} + \bar{T} + \bar{U} + E)(\bar{G} + T + \bar{U} + E)$$

$$((\bar{G} + \bar{U}) + \bar{T} + \bar{E})(\bar{G} + \bar{U}) + T + \bar{E})(\bar{G} + \bar{U}) + \bar{T} + E)((\bar{G} + \bar{U}) + T + E))$$

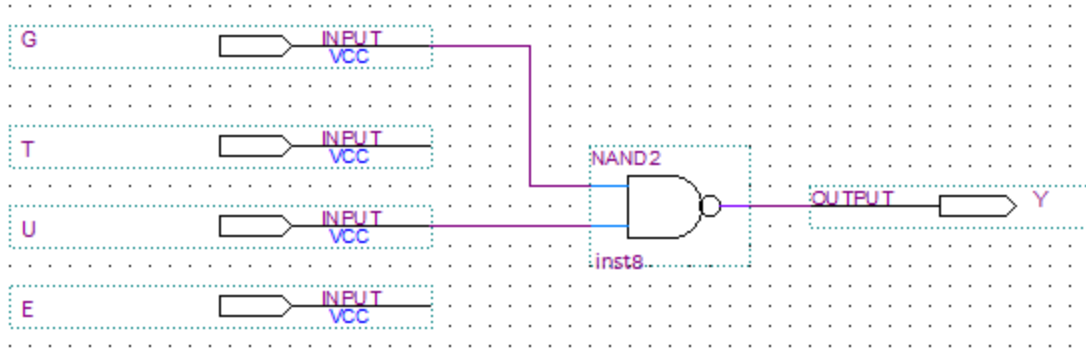
$(\bar{G} + \bar{U})$  ve  $(T + E)$  nin her türlü kombinasyonu olduğu için

$$Y = (\bar{G} + \bar{U})$$

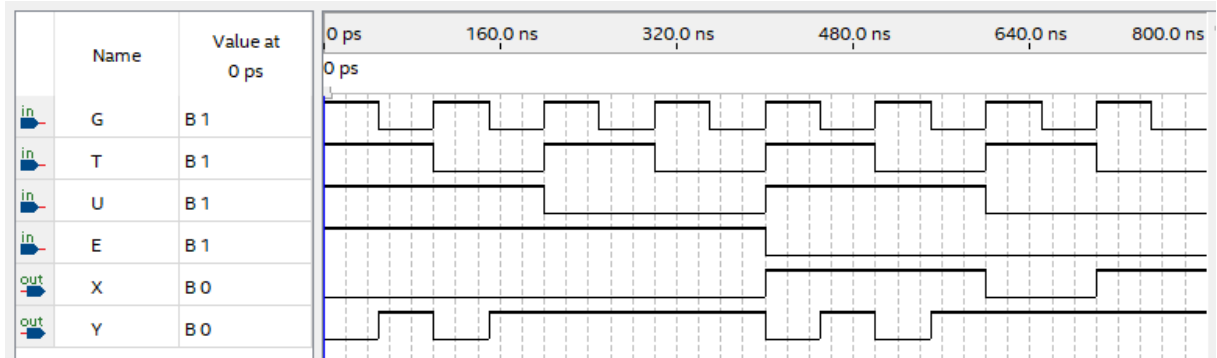
### C) Devre Şeması ve Simülasyon



Şekil 7 X Çıkışlı Devre



Şekil 8 Y Çıkışlı Devre

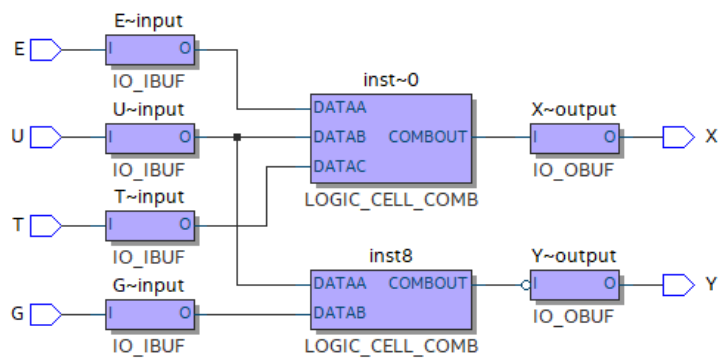
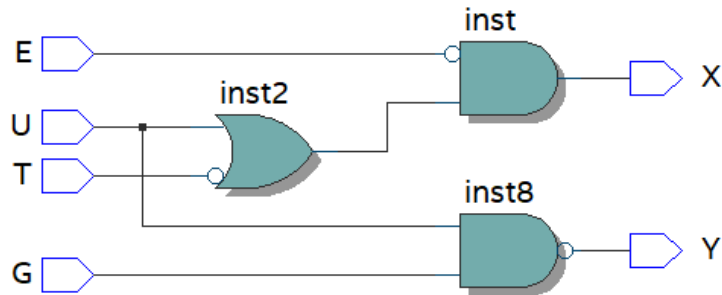


Şekil 9 X ve Y devrelerinin girişlere davranışları



## D) Analiz ve Sentez

	Resource	Usage
1	Estimate of Logic utilization (ALMs needed)	1
2		
3	▼ Combinational ALUT usage for logic	2
1	-- 7 input functions	0
2	-- 6 input functions	0
3	-- 5 input functions	0
4	-- 4 input functions	0
5	-- <=3 input functions	2
4		
5	Dedicated logic registers	0
6		
7	I/O pins	6
8		
9	Total DSP Blocks	0
10		
11	Maximum fan-out node	U~input
12	Maximum fan-out	2
13	Total fan-out	13



### 3.3.3. Sonuçların Yorumu

Bu problemde verilen bir giriş sinyaline karşılık istenilen çıkış sinyalini vere SOP ve POS şeklinde cebirsel ifadeler oluşturduk ve o cebirsel ifadeyi basitleştirerek birer lojik devreye çevirdik.

Hangi devreyi hangi dönüşüm ile çevireceğimize ise çıkış sinyallerinin karakteristiklerine göre karar verildi. Çıkışında ağırlıklı olarak high bulunan sinyal POS formunda, çıkışında ağırlıklı olarak low bulunan sinyal SOP formunda yazıldı.

### 3. Sonuçlar ve Genel Yorumlar

Bu deneyde Boolean Cebrinin genel teoremlerini cebri ifadeleri basitleştirmede kullanmayı öğrendik. Bu bilgilerin üzerine istenen davranışı gösteren SOP (Sum Of Products) veya POS (Product Of Sums) şeklinde boolean cebri ifadeler oluşturmayı ve bu cebri ifadeleri öğrendiğimiz teoremler yardımı ile basitleştirmeyi öğrendik. Son olarak da bu basitleşmiş cebri ifadelerin devre denklemlerini oluşturup sinyaller ile denenmesini öğrendik.

### 4. Kaynaklar

4.1.1. Harris Digital Design and Computer Architecture ARM Edition

4.1.2. Information on SOP and POS representation of circuits

<https://www.electricaltechnology.org/2018/05/sum-of-product-sop-product-of-sum-pos.html>