

# PENGGUNAAN TABEL KEBENARAN LOGIKA DALAM MENDESAIN RANGKAIAN DIGITAL SERTA IMPLEMENTASINYA

Muhammad Ghufron Mahfudhi – NIM : 13508020

Program Studi Teknik Informatika  
Institut Teknologi Bandung  
Jl. Ganeca 10 Bandung  
e-mail: if18020@students.if.itb.ac.id

## ABSTRAK

Makalah ini membahas hubungan antara tabel kebenaran dengan proses pendesainan rangkaian digital. Dalam membuat tabel kebenaran, kita harus menerapkan konsep logika karena logika merupakan dasar dari semua penalaran. Logika merupakan hasil pertimbangan akal pikiran yang diutarakan lewat kata dan dinyatakan dalam bahasa. Tabel kebenaran adalah tabel matematika yang digunakan dalam logika untuk menentukan nilai kebenaran dari suatu ekspresi logika yang masing-masing nilai kombinasinya diambil dari variabel logika. Untuk mendapatkan persamaan logika dari tabel kebenaran, dapat menggunakan peta Karnaugh. Setelah mendapatkan persamaan tersebut, dapat digunakan gerbang-gerbang logika yang sesuai untuk mendesain suatu rangkaian digital yang merupakan sistem yang mempresentasikan sinyal sebagai nilai diskrit. Contoh dari rangkaian digital yang dibahas dalam makalah ini adalah rangkaian *BCD to seven segment* yang dapat mengkonversikan suatu nilai desimal terkode biner (BCD) ke pola segmen yang sesuai pada display *seven segment*. Jadi, dalam proses pendesainan rangkaian digital, tabel kebenaran berfungsi sebagai konseptor dalam membuat persamaan-persamaan logika yang didapatkan dari peta Karnaugh.

**Kata kunci:** logika, tabel kebenaran, rangkaian digital.

## 1. PENDAHULUAN

Logika merupakan dasar dari semua penalaran (reasoning). Untuk menyatukan beberapa logika, kita membutuhkan operator logika dan untuk membuktikan kebenaran dari logika, kita dapat menggunakan tabel kebenaran.

Tabel kebenaran menampilkan hubungan antara nilai kebenaran dari proposisi atomik. Dengan tabel kebenaran, suatu persamaan logika ataupun proposisi bisa dicari nilai kebenarannya.

Tabel kebenaran pasti mempunyai banyak aplikasi yang dapat diterapkan karena mempunyai fungsi tersebut. Salah satu dari aplikasi tersebut yaitu dengan menggunakan tabel kebenaran kita dapat mendesain suatu rangkaian logika. Dalam makalah ini akan dijelaskan bagaimana peran dan kegunaan tabel kebenaran dalam proses pendesainan suatu rangkaian logika.

## 2. DASAR TEORI

### 2.1 Logika

Logika berasal dari kata Yunani kuno *λόγος* (*logos*) yang berarti hasil pertimbangan akal pikiran yang diutarakan lewat kata dan dinyatakan dalam bahasa. Sebagai ilmu, logika disebut dengan *logike episteme* (Latin: *logica scientia*) atau ilmu logika (ilmu pengetahuan) yang mempelajari kecakapan untuk berpikir secara lurus, tepat, dan teratur.

Ilmu disini mengacu pada kemampuan rasional untuk mengetahui dan kecakapan mengacu pada kesanggupan akal budi untuk mewujudkan pengetahuan ke dalam tindakan. Kata logis yang dipergunakan tersebut bisa juga diartikan dengan masuk akal.

Logika dimulai sejak Thales (624 SM - 548 SM), filsuf Yunani pertama yang meninggalkan segala dongeng, takhayul, dan cerita-cerita isapan jempol dan berpaling kepada akal budi untuk memecahkan rahasia alam semesta. Thales mengatakan bahwa air adalah *arkhe* yang berarti prinsip atau asas utama alam semesta. Saat itu Thales telah mengenalkan logika induktif. Aristoteles kemudian mengenalkan logika sebagai ilmu, yang kemudian disebut *logica scientia*. Aristoteles mengatakan bahwa Thales menarik kesimpulan bahwa air adalah *arkhe* alam semesta dengan alasan bahwa air adalah jiwa segala sesuatu. Dalam logika Thales, air adalah *arkhe* alam semesta, yang menurut Aristoteles disimpulkan dari: air adalah jiwa tumbuh-tumbuhan (karena tanpa air tumbuhan mati), air adalah jiwa hewan dan jiwa manusia, air jugalah uap, dan air jugalah es. Jadi, air adalah jiwa dari segala sesuatu, yang berarti, air adalah *arkhe* alam semesta.

Kegunaan logika sangatlah banyak. Sebagian dari kegunaan tersebut adalah sebagai berikut:

- Membantu setiap orang yang mempelajari logika untuk berpikir secara rasional, kritis, lurus, tetap, tertib, metodis dan koheren.
- Meningkatkan kemampuan berpikir secara abstrak, cermat, dan objektif.
- Menambah kecerdasan dan meningkatkan kemampuan berpikir secara tajam dan mandiri.
- Memaksa dan mendorong orang untuk berpikir sendiri dengan menggunakan asas-asas sistematis.
- Meningkatkan cinta akan kebenaran dan menghindari kesalahan-kesalahan berpikir, kekeliruan serta kesesatan.
- Mampu melakukan analisis terhadap suatu kejadian.
- Terhindar dari klenik.
- Apabila sudah mampu berpikir rasional, kritis, lurus, metodis, dan analitis sebagaimana tersebut pada butir pertama maka akan meningkatkan citra diri seseorang.

Dalam logika matematika, kita mengenal proposisi yang merupakan pernyataan yang mempunyai nilai benar atau salah. Contohnya yaitu dalam kalimat “Enam adalah bilangan genap” karena kalimat tersebut bernilai benar. Dalam logika digital, proposisi yang bernilai benar menunjukkan angka 1 sedangkan proposisi yang bernilai salah menunjukkan angka 0.

## 2.2 Tabel Kebenaran dan Operator Logika

Tabel kebenaran adalah tabel matematika yang digunakan dalam logika untuk menentukan nilai kebenaran dari suatu ekspresi logika yang masing-masing nilai kombinasinya diambil dari variabel logika. Tabel kebenaran dapat digunakan untuk mencari tahu apakah ekspresi proposisi tersebut bernilai benar untuk semua nilai input yang valid secara logis.

Satu atau lebih proposisi dapat dikombinasikan untuk menghasilkan proposisi baru. Operator yang digunakan untuk mengkombinasikan proposisi disebut operator logika. Proposisi baru yang diperoleh dari pengkombinasian tersebut dinamakan proposisi majemuk. Proposisi yang bukan merupakan kombinasi proposisi lain disebut proposisi atomik. Dengan kata lain, proposisi majemuk disusun dari proposisi-proposisi atomik.

Beberapa operator logika yang penting yaitu:

- Negasi ( $\text{NOT } p \sim p / \bar{p}$ )

Operator negasi menghasilkan lawan dari suatu proposisi.

**Tabel 1. Tabel Kebenaran Operator Negasi**

p	$\bar{p}$
1	0
0	1

- Konjungsi ( $p \text{ AND } q / p \wedge q / p \cdot q$ )

Tabel kebenaran dari operator konjungsi yaitu:

**Tabel 2. Tabel Kebenaran Operator Konjungsi**

p	q	p.q
1	1	1
1	0	0
0	1	0
0	0	0

- Disjungsi ( $p \text{ OR } q / p \vee q / p + q$ )

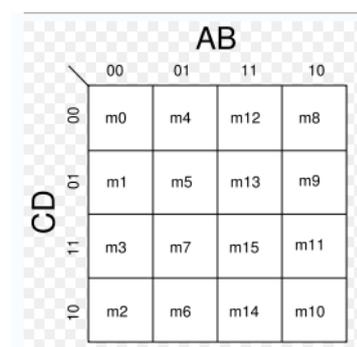
Tabel kebenaran dari operator disjungsi yaitu:

**Tabel 3. Tabel Kebenaran Operator Disjungsi**

p	q	p + q
1	1	1
1	0	1
0	1	1
0	0	0

## 2.3 Peta Karnaugh

Peta karnaugh adalah suatu metode yang digunakan untuk menyederhanakan suatu persamaan logika. Metode ini ditemukan oleh Edward W. Veitch pada tahun 1952 dan dikembangkan lebih lanjut oleh Maurice Karnaugh pada tahun 1953. Peta karnaugh adalah sebuah diagram/peta yang terbentuk dari kotak-kotak (berbentuk persegi) yang bersisian. Peta karnaugh dapat dibentuk dari fungsi logika yang dispedifikasikan dengan ekspresi logika maupun fungsi yang direpresentasikan dengan tabel kebenaran. Bentuk dari peta karnaugh dapat dilihat pada gambar 1.



**Gambar 1. Contoh Peta Karnaugh**

## 2.4 Gerbang Logika

Gerbang logika atau gerbang logik adalah suatu entitas dalam elektronika dan matematika Boolean yang mengubah satu atau beberapa masukan logik menjadi

sebuah sinyal keluaran logik. Gerbang logika terutama diimplementasikan secara elektronis menggunakan dioda atau transistor, akan tetapi dapat pula dibangun menggunakan susunan komponen-komponen yang memanfaatkan sifat-sifat elektromagnetik (*relay*), cairan, optik, dan bahkan mekanik.

**Tabel 4. Ringkasan Macam Gerbang Logika**

Nama	Fungsi	Lambang dalam rangkaian			Tabel kebenaran															
		IEC 60617-12	US-Norm	DIN 40700 (sebelum 1976)																
Gerbang-AND (AND)	$Y = A \wedge B$ $Y = A \cdot B$ $Y = AB$				<table border="1"><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	Y	0	0	0	0	1	0	1	0	0	1	1	1
A	B	Y																		
0	0	0																		
0	1	0																		
1	0	0																		
1	1	1																		
Gerbang-OR (OR)	$Y = A \vee B$ $Y = A + B$				<table border="1"><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>1</td></tr></table>	A	B	Y	0	0	0	0	1	1	1	0	1	1	1	1
A	B	Y																		
0	0	0																		
0	1	1																		
1	0	1																		
1	1	1																		
Gerbang-NOT (NOT, Gerbang-komplemen, Pembalik/inverter)	$Y = \bar{A}$ $Y = \neg A$				<table border="1"><tr><th>A</th><th>Y</th></tr><tr><td>0</td><td>1</td></tr><tr><td>1</td><td>0</td></tr></table>	A	Y	0	1	1	0									
A	Y																			
0	1																			
1	0																			
Gerbang-NAND (Not-AND)	$Y = \overline{A \wedge B}$ $Y = \overline{A \cdot B}$ $Y = \overline{AB}$				<table border="1"><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	Y	0	0	1	0	1	1	1	0	1	1	1	0
A	B	Y																		
0	0	1																		
0	1	1																		
1	0	1																		
1	1	0																		
Gerbang-NOR (Not-OR)	$Y = \overline{A \vee B}$ $Y = \overline{A + B}$				<table border="1"><tr><th>A</th><th>B</th><th>Y</th></tr><tr><td>0</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>1</td><td>1</td><td>0</td></tr></table>	A	B	Y	0	0	1	0	1	0	1	0	0	1	1	0
A	B	Y																		
0	0	1																		
0	1	0																		
1	0	0																		
1	1	0																		

## 2.5 Rangkaian Digital

Digital berasal dari kata *Digitus*, dalam bahasa Yunani berarti jari jemari. Apabila kita hitung jari jemari orang dewasa, maka berjumlah sepuluh (10). Nilai sepuluh tersebut terdiri dari 2 radix, yaitu 1 dan 0. Oleh karena itu, digital merupakan penggambaran dari suatu keadaan bilangan yang terdiri dari angka 0 dan 1 atau *off* dan *on* (bilangan biner). Semua sistem komputer menggunakan sistem digital sebagai basis datanya. Dapat disebut juga dengan istilah Bit (*Binary Digit*).

Rangkaian digital adalah sistem yang mempresentasikan sinyal sebagai nilai diskrit. Dalam sebuah sirkuit digital, sinyal direpresentasikan dengan satu dari dua macam kondisi yaitu 1 (*high, active, true,*) dan 0 (*low, nonactive, false*) atau jika direpresentasikan dalam tegangan 1 dapat berarti tegangan maksimum (umumnya 5 V atau 3 V) dan 0 berarti tegangan minimum (umumnya 0 v, tapi ada pula yang 2,5 V).

Contoh dari rangkaian digital yang dibahas dalam makalah ini yaitu rangkaian *BCD to seven segment* yang merupakan rangkaian logika kombinasional. Rangkaian ini digunakan untuk mengkonversikan suatu nilai desimal terkode biner (BCD) ke pola segmen yang sesuai pada display *seven segment*.

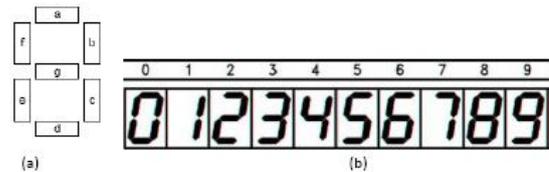
BCD atau *binary-coded decimal* adalah suatu cara penulisan angka desimal yang pada setiap digitnya

direpresentasikan dengan bilangan biner. BCD biasanya direpresentasikan oleh 4 bit bilangan biner yang mempresentasikan angka 0-9. Contoh dari penulisan BCD yaitu:

$$15 = 0001\ 0101$$

$$347 = 0011\ 0100\ 0111$$

*Seven segment* adalah alat tampilan elektronik yang bisa menampilkan penomoran decimal. *Seven segment* biasanya diidentifikasi dalam industri menggunakan huruf a-g seperti pada gambar berikut ini:



**Gambar 2. Konversi Penomoran Seven Segment dan Pola Displaynya**

Untuk mengimplementasikan rangkaian ini pada *seven segment*, kita memerlukan FPGA atau *Field Programmable Gate Array* yang merupakan IC yang terdiri dari banyak transistor.

Penggunaan tabel kebenaran dalam mendesain rangkaian *BCD to seven segment* dapat kita lihat dalam metode penelitian berikut ini.

## 3. METODE PENELITIAN

### 3.1 Komponen dan Alat yang Digunakan

Komponen dan alat yang digunakan antara lain: komputer dengan aplikasi Quartus II 6.0, board FPGA tipe UP2, catu daya, kabel, dan konektor tambahan serta kabel downloader.

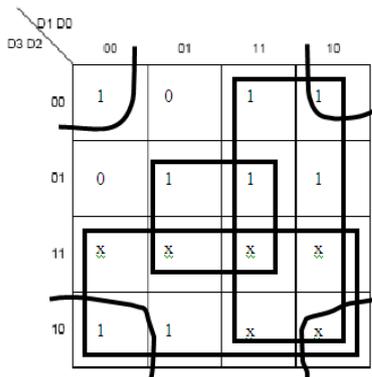
### 3.2 Langkah Penelitian

Langkah pertama yang dilakukan untuk membuat rangkaian *BCD to seven segment* yaitu membuat tabel kebenaran seperti di bawah ini.

**Tabel 5. Tabel Kebenaran BCD to Seven Segment**

No.	D3	D2	D1	D0	A	B	C	D	E	F	G
0	0	0	0	0	1	1	1	1	1	1	0
1	0	0	0	1	0	1	1	0	0	0	0
2	0	0	1	0	1	1	0	1	1	0	1
3	0	0	1	1	1	1	1	1	0	0	1
4	0	1	0	0	1	1	1	0	0	1	1
5	0	1	0	1	1	0	1	1	0	1	1
6	0	1	1	0	1	0	1	1	1	1	1
7	0	1	1	1	1	1	1	0	0	0	0
8	1	0	0	0	1	1	1	1	1	1	1
9	1	0	0	1	1	1	1	1	0	1	1
10	1	0	1	0	X	X	X	X	X	X	X
11	1	0	1	1	X	X	X	X	X	X	X
12	1	1	0	0	X	X	X	X	X	X	X
13	1	1	0	1	X	X	X	X	X	X	X
14	1	1	1	0	X	X	X	X	X	X	X
15	1	1	1	1	X	X	X	X	X	X	X

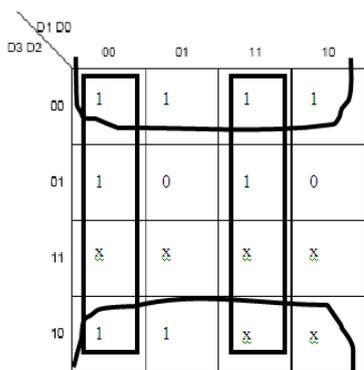
Dari tabel kebenaran di atas, kita bisa membuat peta Karnaugh yang akan menghasilkan suatu persamaan logika seperti di bawah ini.



**Gambar 3. Peta Karnaugh Segment A**

Persamaan segment A:

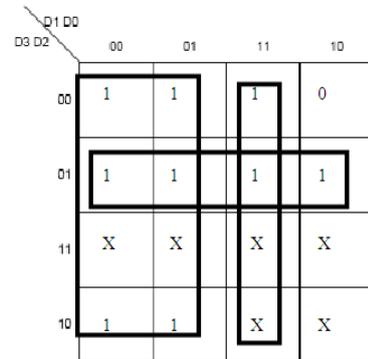
$$A = D3 + D1 + \overline{D2D0} + D2D0 \quad (1)$$



**Gambar 4. Peta Karnaugh Segment B**

Persamaan segment B:

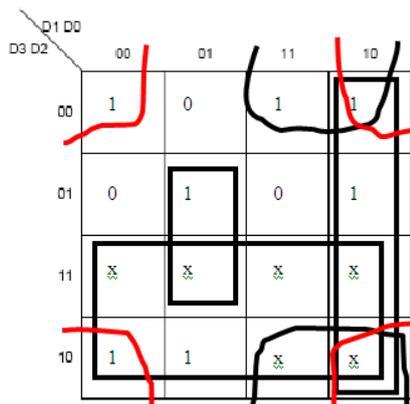
$$B = \overline{D2} + D1D0 + \overline{D1D0} \quad (2)$$



**Gambar 5. Peta Karnaugh Segment C**

Persamaan segment C:

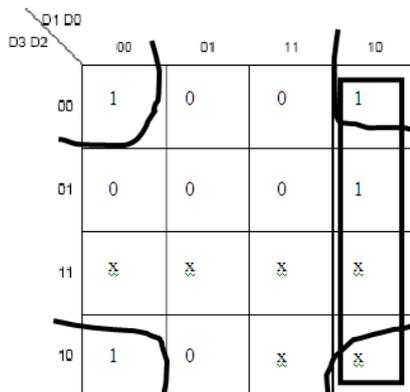
$$C = \overline{D1} + D1D0 + D2 \quad (3)$$



**Gambar 6. Peta Karnaugh Segment D**

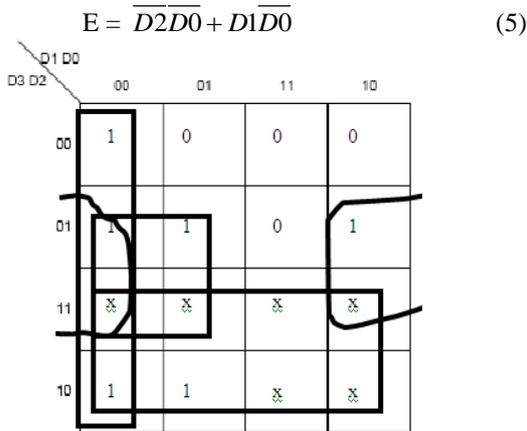
Persamaan segment D:

$$D = D3 + \overline{D2D0} + D1D0 + \overline{D2D1} + D2\overline{D1D0} \quad (4)$$



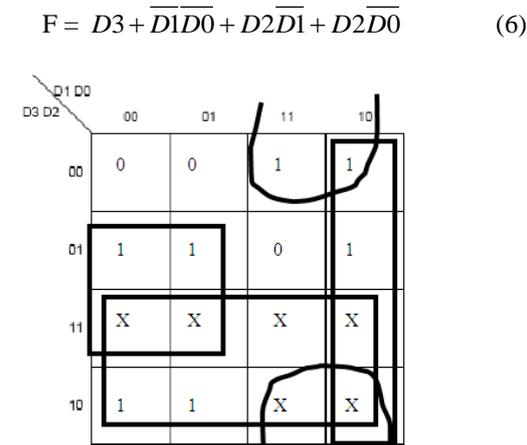
**Gambar 7. Peta Karnaugh Segment E**

Persamaan segment E:



Gambar 8. Peta Karnaugh Segment F

Persamaan segment F:

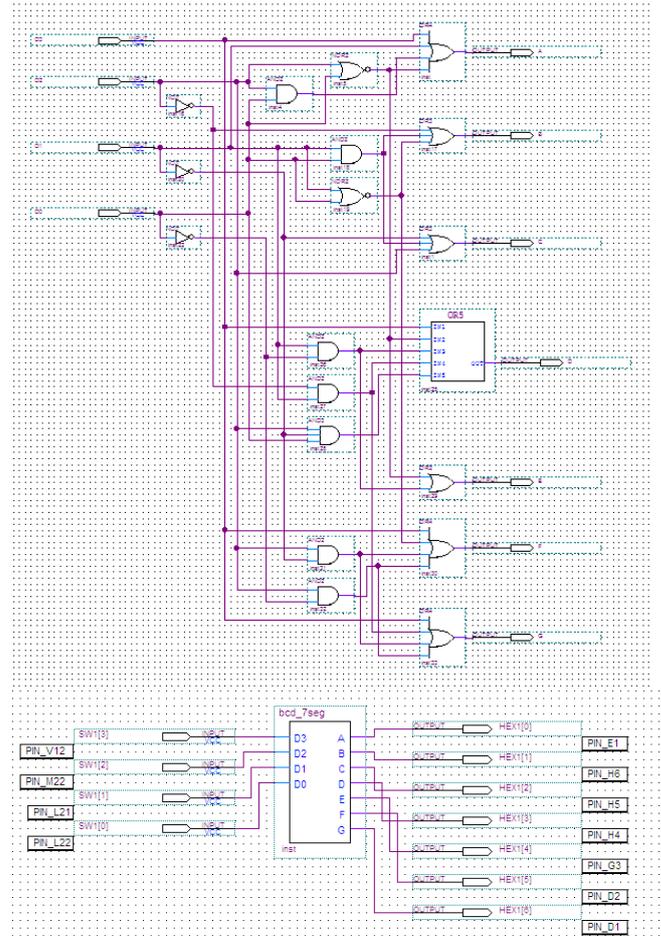


Gambar 9. Peta Karnaugh Segment G

Persamaan segment G:

$$G = D3 + \overline{D2D1} + D2\overline{D0} + D2\overline{D1} \quad (7)$$

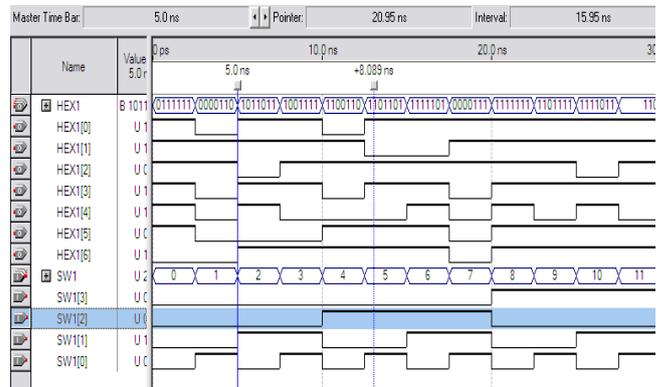
Dari persamaan-persamaan (1) sampai (7), kita bisa menyusun rangkaian digital seperti berikut:



Gambar 10. Rangkaian BCD to Seven Segment dan Rangkaian Testing-nya

### 3.3 Hasil Penelitian dan Pembahasan

Hasil simulasi dari rangkaian tersebut yaitu:



Gambar 11. Simulasi Rangkaian BCD to Seven Segment

Dari gambar 10 tersebut, kita dapat melihat bahwa hasil simulasi yang diperoleh telah sesuai dengan tabel kebenaran pada tabel 5. Misalnya, saat masukan SW3,

SW2, SW1, dan SW0 secara berturut-turut menunjukkan 1, 1, 0, 0 maka A, B, C, D, E, F, dan G secara berturut-turut menunjukkan logika 1, 1, 1, 1, 0, 0, 1.

Setelah diimplementasikan pada FPGA, *seven segment* yang dihasilkan yaitu:



**Gambar 12. Contoh Implementasi Angka 1, 7, dan 9**

Dari gambar tersebut, dapat kita ketahui bahwa hasil yang diperoleh telah menunjukkan hasil yang sesuai dengan apa yang ada dalam tabel 5.

#### **4. KESIMPULAN**

Dalam mendesain rangkaian logika digital, kita memerlukan persamaan-persamaan logika yang dapat menyusun rangkaian tersebut. Langkah-langkah yang bisa diterapkan dalam menentukan persamaan tersebut dan mendesain rangkaian logika digital yaitu membuat rancangan hasil yang kita harapkan dengan menggunakan tabel kebenaran, menyusun dan menyederhanakan persamaan logika dengan menggunakan bantuan peta karnaugh, menyusun rangkaian sesuai dengan persamaan yang diperoleh, dan melakukan simulasi atau diimplementasikan pada FPGA. Oleh karena itu, dapat kita ketahui bahwa peran dari tabel kebenaran dalam proses pendesainan rangkaian logika digital adalah sebagai media konseptor awal untuk menyusun persamaan logika yang akan dibuat.

#### **REFERENSI**

- [1] Rinaldi Munir, "Diktat Kuliah IF2091, Struktur Diskrit", Program Studi Teknik Informatika, STEI, ITB, 2008.
- [2] Mervin T. Hutabarat, dkk, "Buku Petunjuk Praktikum Sistem Digital", STEI, ITB, 2009.
- [3] Wikipedia,  
<http://id.wikipedia.org/wiki/Logika#>  
(17 Desember 2009, 09.00)
- [4] Wikipedia,  
[http://en.wikipedia.org/wiki/Truth\\_table](http://en.wikipedia.org/wiki/Truth_table)  
(17 Desember 2009, 09.30)
- [5] Wikipedia,  
[http://id.wikipedia.org/wiki/Gerbang\\_logika](http://id.wikipedia.org/wiki/Gerbang_logika)  
(17 Desember 2009, 19.30)
- [6] Wikipedia,  
[http://en.wikipedia.org/wiki/Digital\\_electronics](http://en.wikipedia.org/wiki/Digital_electronics)  
(18 Desember 2009, 19.00)

- [7] Wikipedia,  
<http://en.wikipedia.org/wiki/Digital>  
(18 Desember 2008, 19.15)

[http://en.wikipedia.org/wiki/Truth table](http://en.wikipedia.org/wiki/Truth_table)