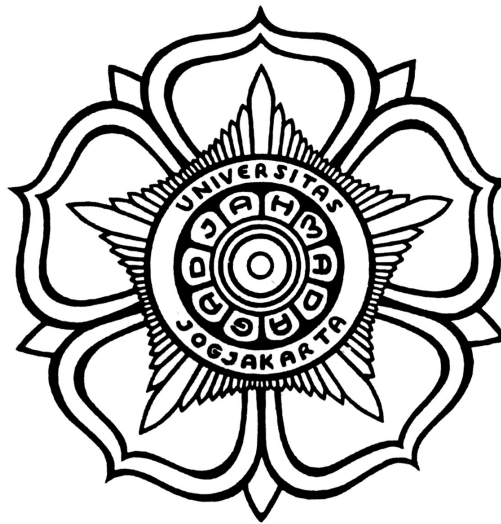


MODUL 02

PRAKTIKUM SISTEM DIGITAL

TNF 2178 - 1 SKS



Disusun oleh:

Prof. Ir. Sunarno, M.Eng., Ph.D.

dan

Tim Asisten Praktikum Sistem Digital

**LABORATORIUM SENSOR DAN SISTEM TELEKONTROL
DEPARTEMEN TEKNIK NUKLIR DAN TEKNIK FISIKA
FAKULTAS TEKNIK
UNIVERSITAS GADJAH MADA
2018**

Chapter 1

PETA KARNAUGH

1.1 Tujuan

- a. Mahasiswa mampu mengubah suatu fungsi aljabar menjadi bentuk yang paling sederhana dan menerapkan ke dalam tabel kebenaran kemudian mengimplementasikannya ke rangkaian kombinasional.
- b. Mahasiswa mampu merancang rangkaian kombinasional dari analisis tabel kebenaran.
- c. Mahasiswa mampu mendesain rangkaian kombinasional dengan menggunakan IC seminimal mungkin.

1.2 Materi

- a. Identifikasi masalah ke dalam tabel kebenaran
- b. Penyederhanaan fungsi logika dengan K-Map
- c. Penyelesaian logika dari tabel kebenaran dengan menggunakan metode SOP dan POS dan implementasi pada rancangan rangkaian logikanya.

1.3 Teori

- a. **Identifikasi masalah ke dalam tabel kebenaran**

Ada kalanya suatu kasus logika disajikan dalam bentuk suatu fungsi logika atau suatu diagram gerbang-gerbang logika yang belum tersaji secara efisien. Oleh sebab itu penting bagi perancang rangkaian logika untuk mengerti bagaimana merancang suatu rangkaian logika dari setiap masalah yang dihadapi.

 - (a) **Apa yang dilakukan jika kasus yang disajikan adalah suatu fungsi logika?**

Pada kasus ini kita harus meneliti dahulu apakah fungsi logika yang disajikan tersebut sudah dalam bentuk yang paling sederhana/efisien ataukah masih dalam bentuk yang apa adanya (belum paling sederhana).

(b) **Apa yang dilakukan jika suatu fungsi sudah dalam bentuk yang paling sederhana?**

Jika kita mempunyai semua gerbang dapat memenuhi semua gerbang logika yang ada pada fungsi tersebut, segeralah merancangnya. Tetapi jika kita ingin mengubah menjadi satu macam tipe gerbang saja seperti NAND atau NOR kita harus mengubah fungsi tersebut menjadi bentuk seperti berikut:

Untuk membuat rangkaian hanya dari gerbang **NAND** :

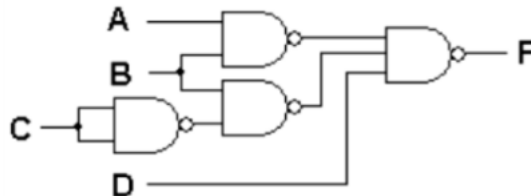
$$\text{Fungsi : } F = B(A + \bar{C}) + D$$

ubahlah fungsi tersebut menjadi bentuk SOP (*Sum of Product*), sehingga menjadi: $F = AB + B\bar{C} + D$

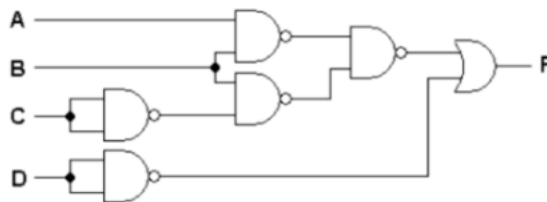
double-bar fungsi tersebut menjadi: $F = \overline{\overline{AB} + \overline{B\bar{C}} + \overline{D}}$

operasikan *bar* yang terbawah dari *double-bar*, sehingga menjadi: $F = \overline{\overline{AB} \cdot \overline{B\bar{C}} \cdot \overline{D}}$

rangkaiannya: jika hanya dibuat dari gerbang NAND



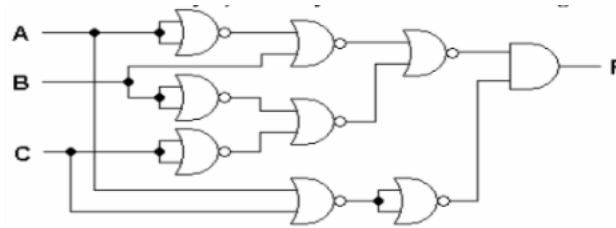
saja dengan 2 input saja dan gerbang OR: Untuk membuat rangkaian



hanya dengan gerbang **NOR**:

$$\text{Fungsi : } F = (\bar{A} + B)(\bar{B} + \bar{C})(A + C)$$

Karena sudah dalam bentuk POS, maka langsung *double-bar* fungsi tersebut, sehingga menjadi : $F = \overline{(\bar{A} + B)(\bar{B} + \bar{C})(A + C)}$ kemudian, operasikan *bar* yang terbawah dari *double-bar*, sehingga menjadi: $F = \overline{(\bar{A} + B)} + \overline{(\bar{B} + \bar{C})} + \overline{(A + C)}$



rangkaian kombinasionalnya jika hanya memakai NOR dengan 2 input:

- (c) **Apa yang dilakukan jika fungsi belum dalam bentuk yang paling sederhana?**

Sederhanakan dahulu fungsi tersebut dengan metode penyederhanaan fungsi aljabar boolean atau dengan menggunakan K-Map (akan dibahas pada sub bab penyederhanaan fungsi logika dengan K-Map). Sesudah itu rancanglah rangkaian kombinasionalnya. Sama seperti proses pembuatan yang sudah dijelaskan di atas.

- (d) **Apa yang akan dilakukan jika kasus disajikan dalam bentuk tabel kebenaran?**

Dengan adanya tabel kebenaran kita dapat berbuat lebih banyak diantaranya: Membentuk fungsi logika secara efisien (dengan cara penyederhanaan dengan K-Map). Membuat ekspresi fungsi logika ke arah SOP atau POS (akan dibahas pada sub bab penyelesaian logika dari tabel kebenaran dengan menggunakan metode SOP dan POS dan implementasi pada rancangan rangkaian logikanya).

- (e) **Apa yang akan dilakukan jika kasus disajikan dalam bentuk diagram gerbang logika?**

Jika dengan diagram gerbang kita dapat analisis apakah diagram gerbang tersebut sudah benar-benar efisien. Oleh karena itu kita cek dahulu apakah gerbang tersebut sudah cukup sederhana. Caranya dengan menelusuri dahulu diagram gerbang tersebut untuk mendapatkan fungsi logikanya. Jika ekspresi fungsi logikanya sudah sederhana segeralah merancang rangkaian kombinasionalnya. Jika belum sederhana lakukan proses penyederhanaan. Dapat dengan penyederhanaan aljabar boolean atau dengan K-Map.

b. Penyederhanaan fungsi logika dengan K-Map

Salah satu metode penyederhanaan fungsi logika untuk maksimal 4 variabel dapat dilakukan dengan metode K-Map (Karnaugh Map). Sebab jika lebih dari 4 variabel kita menggunakan metode Quine Mc Cluskey. Adapun contoh penyederhanaan fungsi logika dengan menggunakan K-Map adalah sebagai berikut:

$$F = (\bar{A} + \bar{B} + C)(\bar{A} + \bar{C})(\bar{B} + \bar{C})$$

Karena bentuk ekspresi fungsi diatas adalah POS, maka kita tempatkan 0 pada K-Map. Sehingga K-Map akan tampak seperti berikut: hasil

CD \ AB	00	01	11	10
00				
01			0	0
11	0	0	0	0
10			0	0

penyederhanaan K-Map adalah:

$$F = (\bar{A} + \bar{B})(\bar{B} + \bar{C})(\bar{A} + \bar{C})$$

c. **Penyelesaian logika dari tabel kebenaran dengan menggunakan metode SOP dan POS dan implementasi pada rancangan rangkaian logikanya.**

Jika diberikan suatu tabel kebenaran dari suatu kasus maka kita dapat menggunakan metode SOP atau POS untuk merancang suatu rangkaian kombinasionalnya.

Seperti yang telah dijelaskan diatas. Untuk menentukan suatu rancangan biasanya kita menghendaki suatu rancangan yang paling efisien. Dengan adanya tabel kebenaran kita dapat menentukan mana diantara metode yang paling efisien untuk diimplementasikan. Untuk menentukan metode mana yang paling efisien, kita lihat bagian *output* pada tabel kebenaran tersebut. Jika jumlah *output* yang mempunyai nilai 1 lebih sedikit dari jumlah *output* yang mempunyai nilai 0, maka kita dapat menentukan bahwa metode SOP yang lebih efisien. Jika jumlah *output* yang mempunyai nilai 0 lebih sedikit dari jumlah *output* yang mempunyai nilai 1, maka kita dapat menentukan metode POS yang lebih efisien. Cara menentukan rangkaian kombinasional dari tabel kebenaran berikut ini: Karena *output* dengan nilai 0 lebih banyak maka kita gunakan metode

A	B	C	OUTPUT
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	1
1	0	1	1
1	1	0	1
1	1	1	1

POS. Sehingga K-Map akan berbentuk: Ekspresi fungsi logikanya dari

A \ BC				
	00	01	11	10
0	0	0	0	
1				

hasil K-Map tersebut adalah:

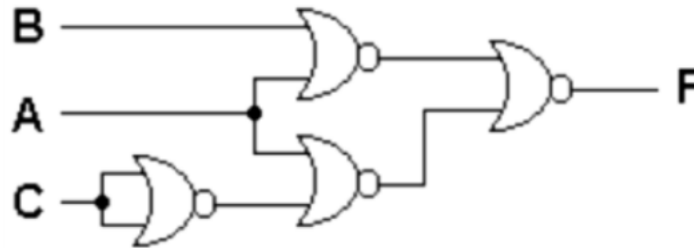
$$F = (A + B)(A + \bar{C})$$

Dari fungsi logika tersebut kita dapat merancang rangkaian kombinasionalnya dari gerbang NOR saja dengan cara memberi *double bar* kemudian *bar* terbawah dioperasikan sehingga:

$$F = \overline{\overline{(A + B)(A + \bar{C})}}$$

$$F = \overline{\overline{(A + B)} + \overline{(A + \bar{C})}}$$

Dan rangkaian kombinasionalnya: Kadangkala suatu hasil dari tabel dis-



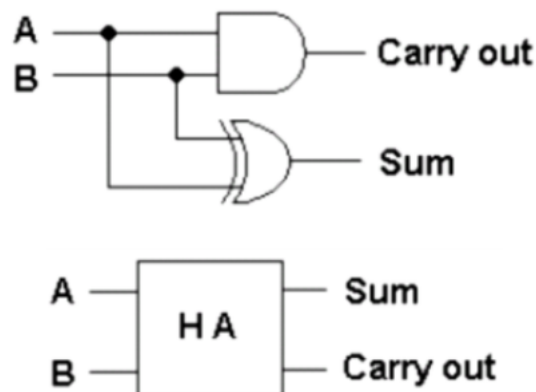
ajikan dalam bentuk fungsi. Dan kita akan mengenal simbol “S” melambangkan operasi SOP sehingga yang ditampilkan adalah *output* yang mempunyai nilai 1 dan simbol “P” melambangkan operasi POS sehingga yang ditampilkan adalah *output* yang mempunyai nilai 0.

d. **Half Adder**

Adalah suatu operasi penjumlahan dua bit biner tanpa menyertakan *carry-in* nya. *Half adder* ini dapat dibuat tabel kebenarannya sebagai berikut: Dari tabel kebenaran tersebut kita dapat merancang rangkaian

A	B	SUM	Carry Out
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

kombinasionalnya menjadi: Jika kita buat diagram menurut rangkaian kombinasional diatas *half adder* tersebut menjadi:

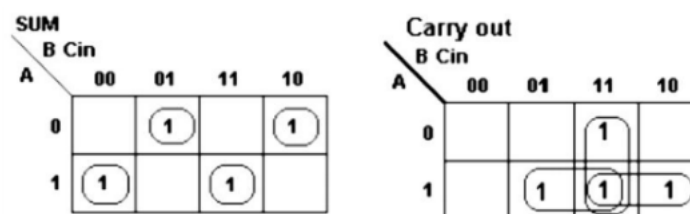


e. **Full Adder**

Adalah operasi penjumlahan dua bit biner dengan menyertakan *carry-in* nya. Tabel kebenaran untuk *full adder* ini adalah sebagai berikut: Den-

A	B	Carry in	SUM	Carry Out
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

gan K-Map kita bisa merancang rangkaian full addernya sebagai berikut: Rangkaian kombinasionalnya: atau dapat juga rangkaian full adder terse-

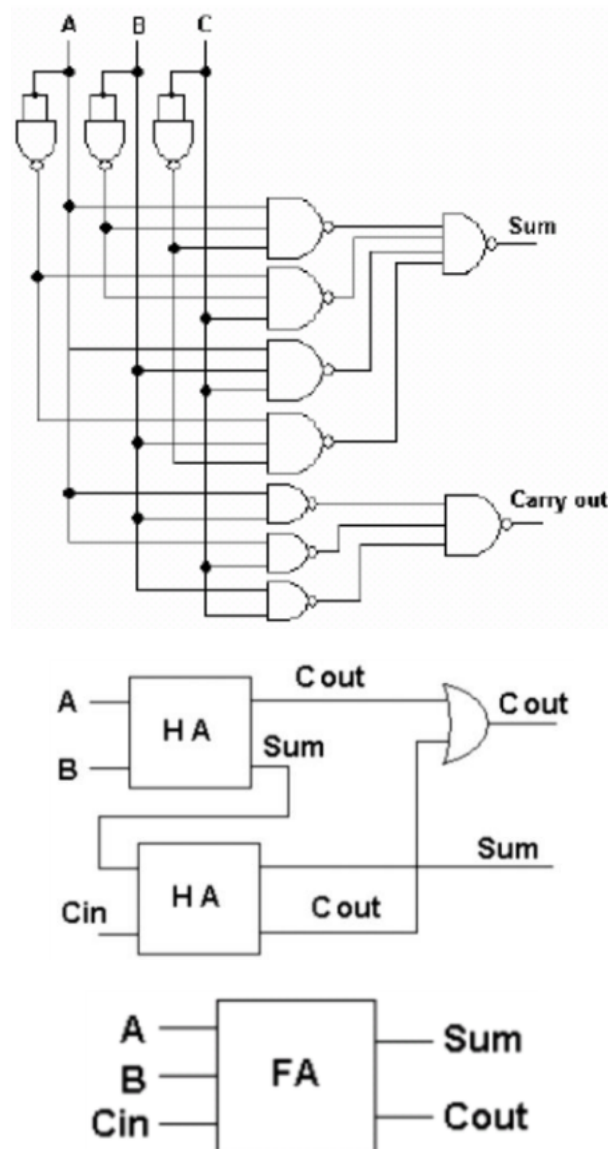


C = Carry in

$$\text{Sum} = \bar{A}\bar{B}C + \bar{A}B\bar{C} + A\bar{B}\bar{C} + \bar{A}BC$$

$$\text{Carry out} = AB + AC + BC$$

but dibuat dari 2 buah rangkaian *half adder*. Sehingga bentuknya menjadi seperti berikut: Rangkaian tersebut dapat dibuat diagram logikanya menjadi:



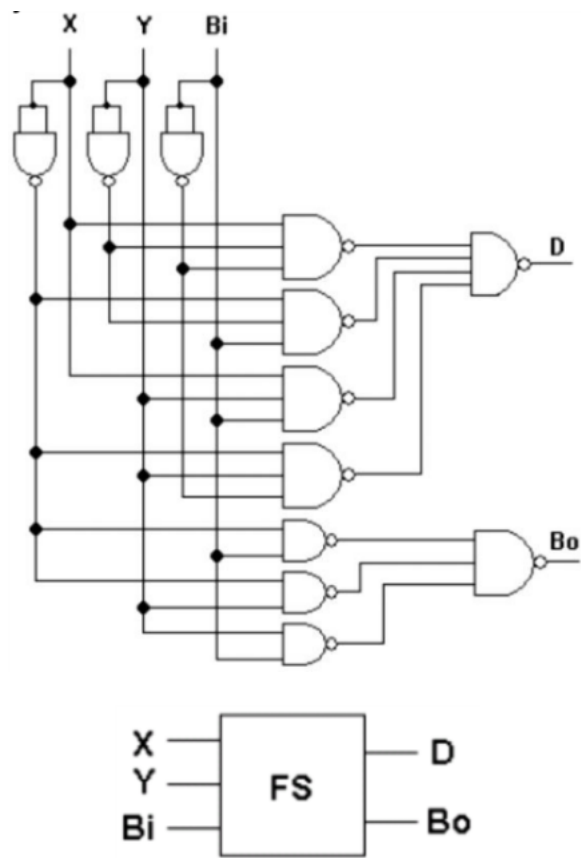
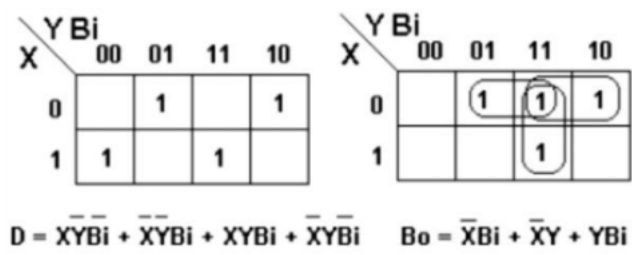
f. ***Full subtractor***

Rangkaian logika lainnya yang dapat dikelompokkan sebagai unit rangkaian aritmatika adalah *full subtractor*. *Full subtractor* ini merupakan operasi pengurangan dua bit biner yang mengikutsertakan *borrow-in* nya di dalam operasi pengurangannya. Tabel kebenaran dari *full subtractor* ini adalah sebagai berikut: Dengan menggunakan K-Map untuk mencari fungsi logika yang paling sederhana. Rangkaian logikanya adalah: Rangkaian tersebut dapat dibuat diagram logikanya sebagai berikut:

1.4 Alat dan Bahan

- Buku Praktikum

X	Y	Bin	D	Bout
0	0	0	0	0
0	0	1	1	1
0	1	0	1	1
0	1	1	0	1
1	0	0	1	0
1	0	1	0	0
1	1	0	0	0
1	1	1	1	1



- b. Power Project Board
- c. IC AND 74xx08

- d. IC OR 74xx32
- e. IC NOT 74xx04
- f. IC NAND 74xx00
- g. IC NOR 74xx02
- h. IC XOR 74xx86
- i. Lampu LED
- j. Resistor 330 Ω

1.5 Metodologi Percobaan

- a. Peserta terlebih dahulu berdoa agar praktikum berjalan lancar.
- b. Kemudian membaca dan mempelajari materi praktikum dan pada awal pertemuan instruktur menerangkan teori dan cara kerja percobaan.

1.6 Percobaan

- a. Matikan *power project board*
- b. Masukkan IC yang akan diamati ke tempat yang sesuai, diperbolehkan memasang beberapa IC sekaligus. Kemudian, hubungkan kaki Vcc IC ke catu daya +5 Vdc dengan menggunakan kabel/*jumper*, demikian juga kaki GND dihubungkan ke *ground* pada *project board*.
- c. Hubungkan *output* gerbang dengan LED, di antaranya diberi R 330 Ω . Pengamatan kaki *output* ditujukan pada kaki yang terhubung ke saklar.
- d. Nyalakan *power project board*. Isi tabel dengan angka 1 jika LED menyala dan angka 0 untuk sebaliknya.
- e. Rangkaian yang akan dibuat mengacu pada tabel kebenaran berikut.

Peta Karnaugh (K-Map)

1. Membuat 3 (tiga) rangkaian dan notasi/rumusan yang paling sederhana untuk tabel kebenaran berikut ini

INPUT				OUTPUT		
D	C	B	A	Y1	Y2	Y3
0	0	0	0	0	1	1
0	0	0	1	1	1	0
0	0	1	0	0	1	1
0	0	1	1	1	1	1
0	1	0	0	1	1	0
0	1	0	1	0	0	1
0	1	1	0	1	1	0
0	1	1	1	1	0	1
1	0	0	0	0	1	1
1	0	0	1	1	0	0
1	0	1	0	0	0	1
1	0	1	1	1	0	1
1	1	0	0	1	1	1
1	1	0	1	1	1	1
1	1	1	0	1	0	1
1	1	1	1	1	1	1

2. Membuat 3 (tiga) rangkaian dan notasi/rumusan yang paling sederhana untuk tabel kebenaran berikut ini (*output diisi oleh asisten*)

INPUT				OUTPUT		
D	C	B	A	Y4	Y5	Y6
0	0	0	0			
0	0	0	1			
0	0	1	0			
0	0	1	1			
0	1	0	0			
0	1	0	1			
0	1	1	0			
0	1	1	1			
1	0	0	0			
1	0	0	1			
1	0	1	0			
1	0	1	1			
1	1	0	0			
1	1	0	1			
1	1	1	0			
1	1	1	1			

HASIL RANCANGAN

Tabel Kebenaran 1

➤ K-Map

➤ Rangkaian

➤ K-Map

➤ Rangkaian

DAFTAR PUSTAKA

David Bucchlah, Wayne McLahan, “Applied Electronic Instrumentation And Measurement”, MacMilian Publishing Company, 1992.

Eggebrecht, Lewis C., Interfacing to The IBM PC, Howard W. Sams & Co., Indianapolis, 1987.

Hall, Douglas V., Microprocessor and Interfacing : Programming and Hardware, McGraw-Hill Book Company, Singapore, 1987.

Hodges D. , Jacson, Nasution S. “Analisa dan Desain Rangkaian Terpadu Digital”, Erlangga, Jakarta, 1987.

Ian Robertson Sinclair, Suryawan, “Panduan Belajar Elektronik Digital”, ElexMedia Komputindo, Jakarta, 1993.

K.F. Ibrahim, “Teknik Digital”, Andi Offset , Jakarta, 1996.

Sendra, Smith, Keneth C., “ Rangkaian Mikroelektronika”, Penerbit Erlangga, Jakarta, 1989.

Singh, Avtar & Walter A. Triebel, The 8088 Microprocessor : Programming, Interfacing, Software, Hardware and Applications, Prentice-Hall International Inc., New Jersey, 1987.

Sofyan H. Nasution, “Analisa dan Desain Rangkaian Terpadu Digital”, Penerbit Erlangga, Jakarta, 1987.

Sutrisno, “Rangkaian Digital dan Rancangan Logika”, Erlangga, Jakarta, 1990.

Tokheim. R., “Elektronika Digital”, Edisi Kedua, Erlangga, Jakarta, 1995.

Wijaya Widjanarka N., “Teknik Digital”, Erlangga, Jakarta, 2006.