

Mata kuliah

Pengantar system digital

Dosen pengampu: Arif Rifa'i Dwiyanto, ST., MTI

**TABEL KEBENARAN DAN MENGIMPLEMENTASI K-MAP
DAN DECODER & ENCODER**



Disusun oleh:

Khalila indana (202410715127)

F3A6

**LAPORAN PRAKTIKUM PENGANTAR SISTEM
DIGITAL
PROGRAM STUDI INFORMATIKA
UNIVERSITAS BHAYANGKARA JAKARTA RAYA
2025**

BAB I

PENDAHULUAN

1.1 Tujuan

- Mahasiswa bisa menyederhanakan ekspresi logika menggunakan (K-Map)
- Mahasiswa memahami apa itu konsep decoder dan encoder
- Mahasiswa mampu menerapkan hasil penyederhanaan pada Logisim

1.2 Alat dan Bahan

- Modul praktikum
- Laptop dengan aplikasi Logisim Evolution terinstal

1.3 Langkah-Langkah Kerja

A. Implementasi K-Map

AB\CD	00	01	11	10
00	0	0	0	0
01	1	1	1	0
11	1	1	1	0
10	0	0	0	0

- Langkah 1:** Tentukan Minterm
- Langkah 2:** penyederhanaan K-map
- Langkah 3:** membuat rangkaian simulasi pada logisim evolution
- Langkah 4:** membandingkan rangkaian sebelum dan sesudah disederhakan

Langkah 1:

Nilai 1 muncul pada posisi:

- $AB = 01, CD = 00, 01, 11$
- $AB = 11, CD = 00, 01, 11$

Minterm: $m(4), m(5), m(7), m(12), m(13), m(15)$

Fungsi logika awalnya yaitu:

$$Y = (A'BC'D') + (A'BC'D) + (A'BCD) + (ABC'D') + (ABC'D) + (ABCD)$$

[illegible]

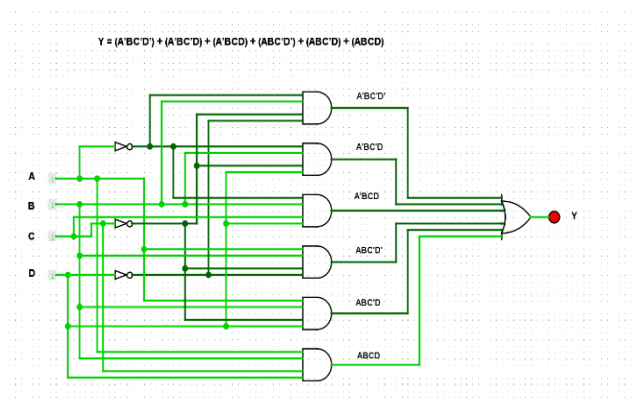
1	1	0	0	0	1	1	0	0	0	1	0	0	1
1	1	0	1	0	1	0	0	0	0	0	1	0	1
1	1	1	0	0	0	1	0	0	0	0	0	0	0
1	1	1	1	0	0	0	0	0	0	0	0	1	1

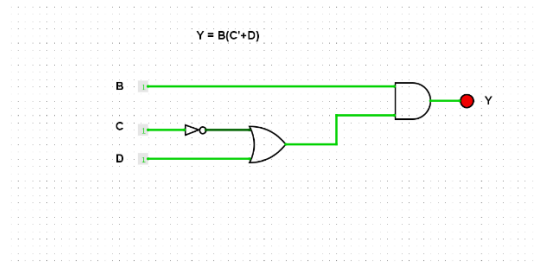
e. Langkah 2:

AB\CD	00	01	11	10
00	0	0	0	0
01	1	1	1	0
11	1	1	1	0
10	0	0	0	0

Maka hasil penyederhanaanya yaitu $Y = B(C' + D)$

f. Langkah 3:





g. Langkah 4:

Aspek	Rangkaian awal	Rangkaian setelah disederhanakan
Ekspresi logika	$Y = (A'BC'D') + (A'BC'D) + (A'BCD) + (ABC'D') + (ABC'D) + (ABCD)$	$Y = B(C' + D)$
Jumlah Gerbang AND	6 buah	1 buah
Jumlah Gerbang OR	1 buah	1 buah
Jumlah Gerbang NOT	3 buah	1 buah
Total Komponen	10 gerbang	3 gerbang
Output Simulasi	sama	sama

B. Encoder dan Decoder

1) Merancang 8-to-3 Encoder

Langkah 1: Buat Tabel Kebenaran untuk Encoder 8-to-3

Langkah 2: mengimplementasikan rangkaian dengan Logisim

Langkah 1:

Bilangan	A	B	C
----------	---	---	---

Oktal	2^2	2^1	2^0
0	0	0	0
1	0	0	1
2	0	1	0
3	0	1	1
4	1	0	0
5	1	0	1
6	1	1	0
7	1	1	1

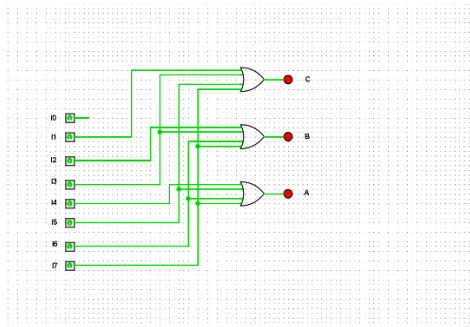
Input								Output		
I ₀	I ₁	I ₂	I ₃	I ₄	I ₅	I ₆	I ₇	A	B	C
1	0	0	0	0	0	0	0	0	0	0
0	1	0	0	0	0	0	0	0	0	1
0	0	1	0	0	0	0	0	0	1	0
0	0	0	1	0	0	0	0	0	1	1
0	0	0	0	1	0	0	0	1	0	0
0	0	0	0	0	1	0	0	1	0	1
0	0	0	0	0	0	1	0	1	1	0
0	0	0	0	0	0	0	1	1	1	1

$$A = I_4 + I_5 + I_6 + I_7$$

$$B = I_2 + I_3 + I_6 + I_7$$

$$C = I_1 + I_3 + I_5 + I_7$$

Langkah 2:



2) Merancang 3-to-8 Decoder

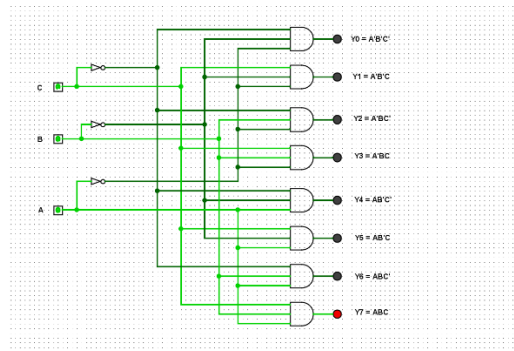
Langkah 1: Buat Tabel Kebenaran untuk Decoder 3-to-8

Langkah 2: mengimplementasikan table menggunakan logisim

Langkah 1:

Input			Output							
A	B	C	Y ₀	Y ₁	Y ₂	Y ₃	Y ₄	Y ₅	Y ₆	Y ₇
0	0	0	1	0	0	0	0	0	0	0
0	0	1	0	1	0	0	0	0	0	0
0	1	0	0	0	1	0	0	0	0	0
0	1	1	0	0	0	1	0	0	0	0
1	0	0	0	0	0	0	1	0	0	0
1	0	1	0	0	0	0	0	1	0	0
1	1	0	0	0	0	0	0	0	1	0
1	1	1	0	0	0	0	0	0	0	1

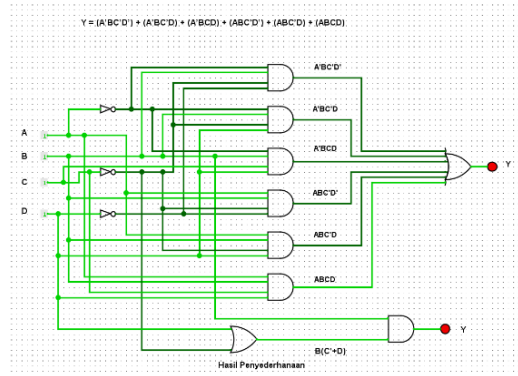
Langkah 2:



BAB II

HASIL SIMULASI DAN PEMBAHASAN

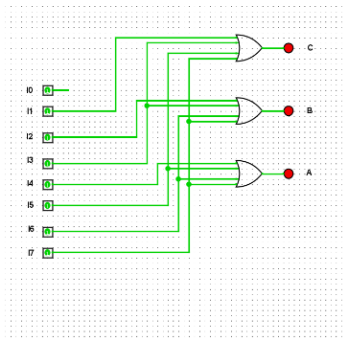
a) K-Map



Diperoleh fungsi logika awal yang disusun dalam bentuk Sum Of Product (SOP) yang panjang, karena setiap minterm yang menghasilkan output 1 direpresentasikan oleh kombinasi gerbang AND yang berbeda. Fungsi logika yang diimplementasikan adalah $Y = (A'BC'D') + (A'BC'D) + (A'BCD) + (ABC'D') + (ABC'D) + (ABCD)$. Rangkaian awal tersebut terlihat cukup rumit, memiliki banyak koneksi dan jenis gerbang logika.

proses penyederhanaan dengan menggunakan peta Karnaugh. Dengan penyederhanaan ini, diperoleh bentuk fungsi logika yang lebih sederhana tetapi tidak mengubah hasil outputnya. Setelah disederhanakan, jumlah rangkaian menjadi lebih mudah. Fungsi logika setelah disederhanakan adalah $Y = B(C' + D)$. Dilihat dari gambar rangkaian setelah disederhanakan, output Y tetap sama, yaitu bernilai 1 (menyala) pada kondisi yang sama seperti rangkaian awal. Yang beda jumlah dan susunan gerbang logika.

b) Encoder



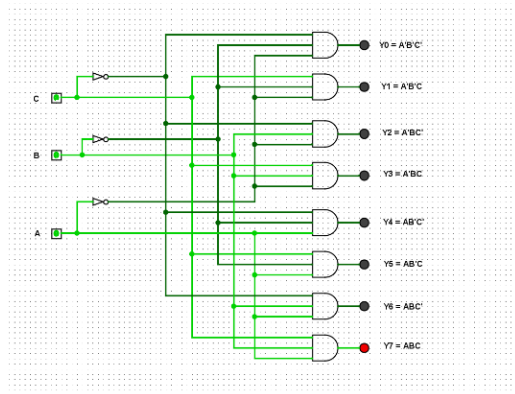
dirancang rangkaian encoder 8-to-3 menggunakan aplikasi Logisim. Encoder adalah jenis rangkaian logika kombinasi yang berperan untuk mengubah beberapa input menjadi kombinasi output biner yang lebih sedikit.

- Setiap input terhubung ke gerbang OR sesuai dengan persamaan logika yang terlihat dalam gambar.
- Tiga output dihasilkan dari kombinasi logika OR yang mewakili kode biner dari input yang aktif.
- Hasil pengujian menunjukkan bahwa ketika salah satu input diberi nilai 1, output menghasilkan kombinasi biner yang sesuai dengan posisi input tersebut.

Pada tabel kebenaran, persamaan logika untuk masing output:

- Output A akan bernilai 1 jika salah satu input dari I_4 hingga I_7 aktif
- Output B akan bernilai 1 jika salah satu input dari I_2, I_3, I_6, I_7 aktif
- Output C akan bernilai 1 jika salah satu input dari I_1, I_3, I_5, I_7 aktif

c) Decoder



decoder adalah kebalikan dari encoder. Rangkaian decoder 3-to-8 memiliki 3 input yaitu A,B dan C yang masing-masing terhubung ke gerbang NOT untuk menghasill sinyal komplemen (A' , B' , C'). 8 output yaitu Y_0 - Y_7 , diperoleh melalui gerbang AND yang mengkombinasikan sinyal input dan sinyal komplemennya sesuai persamaan logika. Setiap output dihubungkan ke LED

Contoh berdasarkan hasil simulasi:

- Jika input = 111 ($A=1$, $B=1$, $C=1$) maka output yang aktif adalah Y_7
- Jika input = 100 ($A=1$, $B=0$, $C=0$) maka output yang aktif adalah Y_4
- Jika input = 000 ($A=0$, $B=0$, $C=0$) maka output yang aktif adalah Y_0

Hal ini sesuai dengan tabel kebenaran yang telah dibuat.

BAB III

KESIMPULAN

Berdasarkan praktikum yang telah dilakukan,. dapat disimpulkan bahwa peta Karnaugh sangat efektif dalam menyederhanakan fungsi logika. Dari proses penyederhanaan diperoleh fungsi $Y = B(C' + D)$ yang menghasilkan pengurangan jumlah gerbang logika dari 10 komponen menjadi hanya 3 komponen. selain itu, desain encoder 8-to-3 dan decoder 3-to-8 menunjukkan penggunaan rangkaian kombinasi. Encoder berfungsi mengubah sinyal input yang hanya satu menjadi kode biner 3 bit, sedangkan decoder melakukan proses sebaliknya. Percobaan ini dilakukan menggunakan aplikasi Logisim.