

## 5. RANGKAIAN KOMBINASIONAL

Pada dasarnya rangkaian logika (digital) yang dibentuk dari beberapa gabungan komponen elektronik yang terdiri dari bermacam-macam Gate dan rangkaian-rangkaian lainnya, sehingga membentuk rangkaian elektronika yang bersifatnya kompleks dan cukup rumit. Untuk mengatasi hal tersebut maka dipergunakanlah beberapa metode penyederhanaan rangkaian logika.

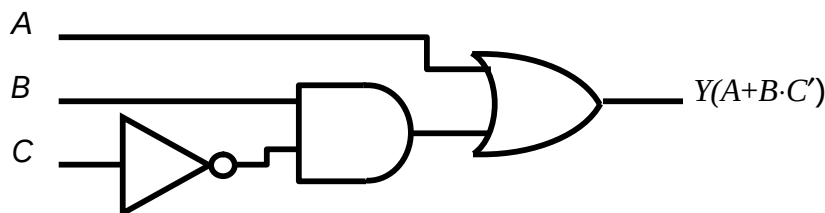
Dalam penyederhanaan rangkaian logika, dapat menggunakan beberapa cara, diantaranya :

1. Metode Karnaugh Map.
2. Metode Maksterm/Minterm.
3. Metode Aljabar Boole.

### 5.1 Aljabar Boole

Aljabar Boole mendefinisikan aturan-aturan untuk memanipulasi ekspresi simbol logika biner. Ekspresi Logika simbol biner terdiri dari variabel biner dan operator-operator seperti AND, OR dan NOT (contoh :  $A+B \times C'$ ). Nilai-nilai dari ekspresi Boolean dapat ditabulasikan dalam tabel kebenaran [*truth table*].

Contoh ekspresi beberapa operator yang telah menjadi suatu rangkaian :

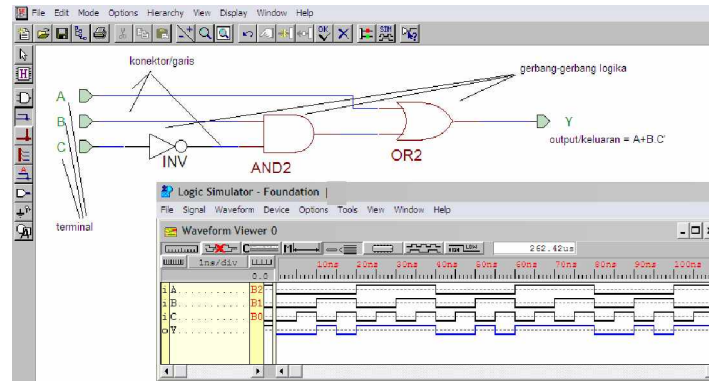


Gambar 28 : contoh ekspresi beberapa logika dasar

Tabel 3 : tabel kebenaran rangkaian gambar 28

| A | B | C | B.C | Y = (A+B.C') |
|---|---|---|-----|--------------|
| 0 | 0 | 0 | 0   | 0            |
| 0 | 0 | 1 | 0   | 0            |
| 0 | 1 | 0 | 1   | 1            |
| 0 | 1 | 1 | 0   | 0            |
| 1 | 0 | 0 | 0   | 1            |
| 1 | 0 | 1 | 0   | 1            |
| 1 | 1 | 0 | 1   | 1            |
| 1 | 1 | 1 | 0   | 1            |

Pada gambar 29 merupakan schematic dan Simulasi rangkaian gambar 28.



Gambar 29 : Schematic editor dan simulasi

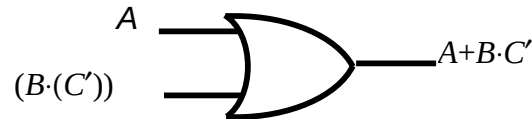
Ekspresi Boolean dapat dikonversikan ke rangkaian logika yang terbentuk dari gerbang-gerbang seperti AND, OR dan NOT.

Langkah 1: tambahkan tanda kurung kurawal (parentheses) untuk memperjelas ekspresi operasi spt -  $A+(B \times (C'))$

Langkah 2 : Buatlah gerbang untuk operasi “last” (terakhir)

Gerbang keluaran (output) adalah nilai ekspresi rangkaian

Gerbang masukan (input) adalah ekspresi yang digabungkan oleh suatu operasi



Langkah 3 : Ulangi untuk sub-ekspresi dan teruskan sampai selesai

à Jumlah gerbang yang *paling sedikit* dibutuhkan untuk mengimplementasikan persamaan jumlah ekspresi dalam suatu ekspresi/persamaan.

- » Bagian *persamaan yang sama* yang paling mudah merupakan rangkaian yang lebih murah
- » Aljabar Boole menyediakan aturan-aturan untuk menyederhanakan suatu ekspresi atau persamaan logika tersebut.

Identitas dasar Aljabar Boolean :

- |                 |                      |
|-----------------|----------------------|
| 1. $X + 0 = X$  | 2. $X \times 1 = X$  |
| 3. $X + 1 = 1$  | 4. $X \times 0 = 0$  |
| 5. $X + X = X$  | 6. $X \times X = X$  |
| 7. $X + X' = 1$ | 8. $X \times X' = 0$ |

9.  $(X')' = X$
10.  $X + Y = Y + X$
11.  $X \times Y = Y \times X$
12.  $X + (Y + Z) = (X + Y) + Z$
13.  $X \times (Y \times Z) = (X \times Y) \times Z$
14.  $X(Y + Z) = X \times Y + X \times Z$
15.  $X + (Y \times Z) = (X + Y) \times (X + Z)$
16.  $(X + Y)' = X' \times Y'$
17.  $(X \times Y)' = X' + Y'$

**DeMorgan :**  
komutatif  
assosiatif  
distributif

Keterangan :

- Identitas mendefinisikan bagian-bagian intrinsik aljabar Boolean.
- Sangat berguna untuk penyederhanaan ekspresi Boolean expressions.
- Aturan nomor: 15-17 tidak punya counterpart dalam aljabar biasa
- Kolom paralel mengilustrasikan Prinsip dualitas (*duality principle*).

Membuktikan identitas menggunakan tabel kebenaran (*Truth Tables*) dapat dilihat pada tabel 4 dibawah ini.

Tabel 4 : Membuktikan identitas dengan tabel kebenaran

| X | Y | $(X+Y)' = X' \cdot Y'$ |               |
|---|---|------------------------|---------------|
|   |   | $(X + Y)'$             | $X' \cdot Y'$ |
| 0 | 0 | 1                      | 1             |
| 0 | 1 | 0                      | 0             |
| 1 | 0 | 0                      | 0             |
| 1 | 1 | 0                      | 0             |

| X | Y | Z | $X + (Y \cdot Z) = (X + Y) \cdot (X + Z)$ |                   |         |         |
|---|---|---|-------------------------------------------|-------------------|---------|---------|
|   |   |   | $Y \cdot Z$                               | $X + (Y \cdot Z)$ | $X + Y$ | $X + Z$ |
| 0 | 0 | 0 | 0                                         | 0                 | 0       | 0       |
| 0 | 0 | 1 | 0                                         | 0                 | 0       | 1       |
| 0 | 1 | 0 | 0                                         | 0                 | 1       | 0       |
| 0 | 1 | 1 | 1                                         | 1                 | 1       | 1       |
| 1 | 0 | 0 | 0                                         | 1                 | 1       | 1       |
| 1 | 0 | 1 | 0                                         | 1                 | 1       | 1       |
| 1 | 1 | 0 | 0                                         | 1                 | 1       | 1       |
| 1 | 1 | 1 | 1                                         | 1                 | 1       | 1       |

Dapat dilihat pembuktian persamaan logika menjadi lebih sederhana dengan menggunakan tabel kebenaran (truth tables) dengan memecah persamaan/ekspresi logika menjadi bagian yang lebih kecil

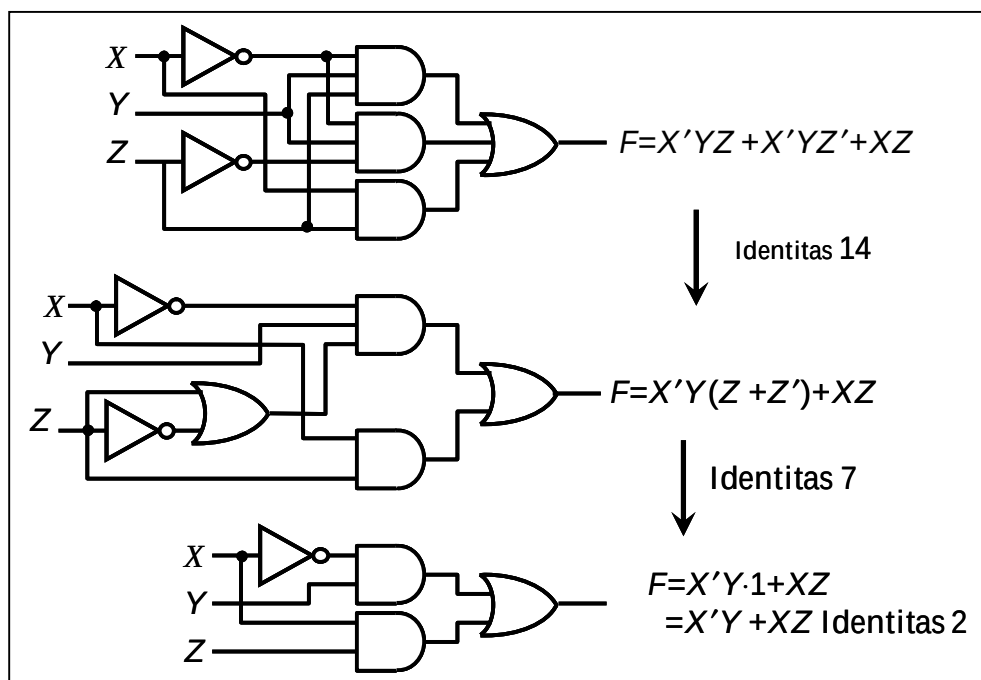
Hukum Morgan untuk n buah variabel :

🌈 Hukum DeMorgan's dapat diperluas menjadi 3 variabel dengan menggunakan aturan/hukum tersebut untuk 2 variabel.

$$\begin{aligned}
 (X + Y + Z)' &= (X + (Y + Z))' && \text{- Hukum asosiatif} \\
 &= X' \times (Y + Z)' && \text{- Hukum DeMorgan} \\
 &= X' \times (Y' \times Z') && \text{- Hukum DeMorgan} \\
 &= X' \times Y' \times Z' && \text{- Hukum asosiatif} \\
 (X \times Y \times Z)' &= (X \times (Y \times Z))' && \text{- Hukum asosiatif} \\
 &= X' + (Y \times Z)' && \text{- Hukum DeMorgan} \\
 &= X' + (Y' + Z') && \text{- Hukum DeMorgan} \\
 &= X' + Y' + Z' && \text{- Hukum asosiatif}
 \end{aligned}$$

🌈 Persamaan diatas dapat diperluas untuk n variabel :

$$\begin{aligned}
 \gg (X_1 + X_2 + \dots + X_n)' &= X'_1 \times X'_2 \times \dots \times X'_n \\
 \gg (X_1 \times X_2 \times \dots \times X_n)' &= X'_1 + X'_2 + \dots + X'_n
 \end{aligned}$$



Gambar 30 : Penyederhanaan ekspresi Boole

## PRINSIP DUALITAS :

✚ Persamaan Boole dual diperoleh dengan menggantikan semua gerbang-gerbang AND dan OR, dan semua nilai 0 dan 1.

» Contoh : dual dari  $A+(B \times C') + 0$  adalah  $A \times (B + C') \times 1$

✚ Prinsip dualitas menyatakan bahwa jika  $E1$  dan  $E2$  adalah persamaan Boolean maka:

$$E1 = E2 \Leftrightarrow \text{dual}(E1) = \text{dual}(E2)$$

dimana  $\text{dual}(E)$  adalah dual dari  $E$ .

$$\text{Contoh : } A+(B \times C') + 0 = (B' \times C) + D \Leftrightarrow A \times (B + C') \times 1 = (B' + C) \times D$$

Konsekuensinya, pasangan identitas (1,2), (3,4), (5,6), (7,8), (10,11), (12,13), (14,15) dan (16,17)

## 5.2 Teorema Konsensus :

Teorema.  $XY + X'Z + YZ = XY + X'Z$

Pembuktian :  $XY + X'Z + YZ = XY + X'Z + YZ(X + X')$  2,7  
 $= XY + X'Z + XYZ + X'YZ$  14  
 $= XY + XYZ + X'Z + X'YZ$  10  
 $= XY(1 + Z) + X'Z(1 + Y)$  2,14  
 $= XY + X'Z$  3,2

Contoh.  $(A + B)(A' + C) = AA' + AC + A'B + BC$   
 $= AC + A'B + BC$   
 $= AC + A'B$

Dual.  $(X + Y)(X' + Z)(Y + Z) = (X + Y)(X' + Z)$

Menggunakan komplemen suatu fungsi :

Metode 1. Menggunakan teorema DeMorgan.

$$\begin{aligned}(X(Y'Z' + YZ))' &= X' + (Y'Z' + YZ)' \\ &= X' + (Y'Z')'(YZ)' \\ &= X' + (Y + Z)(Y' + Z')\end{aligned}$$

Metode 2. Komplemen literal dan menggunakan dualitas

$$\begin{aligned}(X(Y'Z' + YZ))' &= \text{dual}(X'(YZ + Y'Z')) \\ &= X' + (Y + Z)(Y' + Z')\end{aligned}$$

Bentuk hasil yang didapat :

Hasilnya adalah dalam bentuk satu dari dua standar persamaan Boolean.

$$\langle \text{Hasil ekspresi boolean} \rangle = \langle \text{term} \rangle + \langle \text{term} \rangle \dots + \langle \text{term} \rangle$$

Contoh à  $X'Y'Z + X'Z + XY + XYZ$

- minterm* terdiri dari semua variabel baik.

Contoh : Pada persamaan diatas  $X'Y'Z$  adalah minterm, tetapi  $X'Z$  bukan minterm.

- Jumlah ekspresi *minterm* adalah jumlah ekspresi dimana disetiap term berisi minterm.

Contoh :  $X'Y'Z + X'YZ + XYZ' + XYZ$  adalah jumlah ekspresi minterm yang ekuivalen dengan ekspresi diatas.

### 5.3 Peta Karnaugh

Peta karnaugh (peta-K) adalah satu satu cara untuk menyederhanakan rangkaian secara grafis atau diagram berdasarkan teknik pengenalan pola.

Peta-K berisi semua kemungkinan kombinsai dari sistem logika. Kombinasi ini di rangkain ke dalam bentuk tabel. Peta yang paling sederhana terdiri dari dua buah masukan, untuk lebih jelasnya perhatikan beberapa diagram atau gambar dibawah ini.

|     |   |     |   |  |
|-----|---|-----|---|--|
|     |   | $B$ |   |  |
|     |   | 0   | 1 |  |
| $A$ | 0 | 0   | 1 |  |
|     | 1 | 1   | 0 |  |

$F=AB'+A'B$

|     |   |     |   |  |
|-----|---|-----|---|--|
|     |   | $B$ |   |  |
|     |   | 0   | 1 |  |
| $A$ | 0 | 0   | 1 |  |
|     | 1 | 1   | 1 |  |

$F=AB+A'B+AB'$  à 2 variabel peta  
 $F=A+B$

|   |   |    |    |    |    |                                        |
|---|---|----|----|----|----|----------------------------------------|
|   |   | BC |    |    |    |                                        |
|   |   | 00 | 01 | 11 | 10 |                                        |
| A | 0 | 0  | 1  | 0  | 1  | $F=AB'C'+B'C+ABC+BC'$<br>$F=A+B'C+BC'$ |
|   | 1 | 1  | 1  | 1  | 1  |                                        |

à 3 variabel peta

Peta - K :

|    |    |    |    |    |    |  |                      |                   |
|----|----|----|----|----|----|--|----------------------|-------------------|
|    |    | 00 | 01 | 11 | 10 |  |                      |                   |
| AB | 00 | 0  | 0  | 0  | 1  |  | $F=A'BC'+A'CD'+ABC$  |                   |
|    | 01 | 1  | 1  | 0  | 1  |  | $+AB'C'D'+ABC'+AB'C$ |                   |
|    | 11 | 1  | 1  | 1  | 1  |  | $F=BC'+CD'+AC+AD'$   | à 3 variabel peta |
|    | 10 | 1  | 0  | 1  | 1  |  |                      |                   |

|    |    | CD |    |    |    |                                                                                                                                                 |
|----|----|----|----|----|----|-------------------------------------------------------------------------------------------------------------------------------------------------|
|    |    | 00 | 01 | 11 | 10 |                                                                                                                                                 |
| AB | 00 | 0  | 0  | 0  | 1  | $F' = A'B'C' + B'C'D + A'CD$<br>Jika digunakan komplemen dari ekspresi ini,<br>maka didapat bentuk hasil sbb:<br>$F = (A+B+C)(B+C+D')(A+C'+D')$ |
|    | 01 | 1  | 1  | 0  | 1  |                                                                                                                                                 |
|    | 11 | 1  | 1  | 1  | 1  |                                                                                                                                                 |
|    | 10 | 1  | 0  | 1  | 1  |                                                                                                                                                 |

Kondisi Don't Care :

- Suatu kondisi dimana nilai fungsi tidak berfungsi untuk beberapa keadaan kombinasi variabel. Kombinasi ini tidak mungkin diterapkan untuk beberapa keadaan atau nilai fungsi tidak berpengaruh jika kombinasi tersebut terjadi.
- Dalam keadaan ini disebut suatu keadaan dimana suatu fungsi tidak dispesifikasikan secara penuh dan terdapat beberapa fungsi logika (yang dispesifikasikan lengkap) yang dapat digunakan dalam perancangan. Sehingga dapat dipilih fungsi yang lebih sederhana.

Penyederhanaan Map menggunakan "Don't Care" :

|    |    | CD |    |    |    |                      |
|----|----|----|----|----|----|----------------------|
|    |    | 00 | 01 | 11 | 10 |                      |
| AB | 00 | 0  | 1  | 0  | 0  | $F = A'C'D + B + AC$ |
|    | 01 | X  | X  | X  | 1  |                      |
|    | 11 | 1  | 1  | 1  | X  |                      |
|    | 10 | X  | 0  | 1  | 1  |                      |

à Alternatif lainnya :

|    |    | CD |    |    |    |                                |
|----|----|----|----|----|----|--------------------------------|
|    |    | 00 | 01 | 11 | 10 |                                |
| AB | 00 | 0  | 1  | 0  | 0  | $F = A'B'C'D + ABC' + BC + AC$ |
|    | 01 | X  | X  | X  | 1  |                                |
|    | 11 | 1  | 1  | 1  | X  |                                |
|    | 10 | X  | 0  | 1  | 1  |                                |

### Contoh soal :

Sederhanakanlah persamaan berikut dengan menggunakan metode penyederhanaan yang paling sederhana dan simulasikan rangkaian yang belum disederhanakan dengan yang telah disederhanakan menggunakan simulator Xilinx.

$$1. X = A\bar{B}\bar{C}D + \bar{A}B\bar{C}D + A\bar{B}C\bar{D} + \bar{A}BC\bar{D} + ABC\bar{D} + \bar{A}\bar{B}\bar{C}D$$

$$2. Z = \bar{A}\bar{B}\bar{C} + \bar{A}\bar{B}\bar{C}D + \bar{A}\bar{B}CD + ABD + A\bar{B}\bar{C}D + ABC\bar{D} + \bar{A}\bar{B}\bar{C}D$$