

Peta Karnaugh & Rangkaian Multi-Keluaran (Bagian 2)

Kuliah#5 TKC205 Sistem Digital

Eko Didik Widianto

Departemen Teknik Sistem Komputer, Universitas Diponegoro

- ▶ Sebelumnya dibahas tentang:
 - ▶ penyederhanaan fungsi logika menggunakan peta Karnaugh melalui *Grouping* minterm untuk rangkaian SOP , baik fungsi 2-variabel sampai 6-variabel
 - ▶ Terminologi dalam K-map, yaitu implicant, prime implicant (esensial, non-esensial), cover dan cost beserta contoh penggunaan istilah-istilah tersebut
- ▶ Rangkaian SOP optimal diperoleh dengan **penyederhanaan ekspresi logika menggunakan K-map**

- ▶ Dibahas proses sintesis rangkaian logika minimal menggunakan peta Karnaugh untuk menyederhanakan persamaan fungsi logika
 - ▶ Peta Karnaugh juga digunakan untuk merancang rangkaian multikeluaran minimal
- ▶ Pokok Bahasan:
 - ▶ minimisasi POS (pengelompokan Maxterm)
 - ▶ implementasi rangkaian logika POS optimal dengan OR-AND dan/atau NOR-NOR
 - ▶ rangkaian multi-keluaran

- ▶ Setelah mempelajari bab ini, mahasiswa akan mampu:
 1. [C2] memahami prinsip-prinsip penyederhanaan fungsi logika menggunakan peta Karnaugh;
 2. [C3] menggunakan *Don't care* dalam peta Karnaugh;
 3. [C4] merancang dan menganalisis rangkaian logika POS minimal (OR-AND atau NOR-NOR) menggunakan peta Karnaugh;
 4. [C4] merancang dan menganalisis rangkaian logika minimal SOP atau POS dengan menggabungkan beberapa fungsi dalam satu rangkaian multi-keluaran;
- ▶ Link
 - ▶ Website: <http://didik.blog.undip.ac.id/2017/03/06/tkc205-sistem-digital-2016-genap/>
 - ▶ Email: didik@live.undip.ac.id

Eko Didik Widianto, Sistem Digital:
Analisis, Desain dan Implementasi, Edisi
Pertama, Graha Ilmu, 2014 (**Bab 4: Peta
Karnaugh dan Rangkaian
Multikeluaran**)

- ▶ Materi:
 - ▶ 4.1.8 Rangkaian POS Minimal
 - ▶ 4.1.9 Fungsi Tidak Lengkap
 - ▶ 4.2 Rangkaian Multikeluaran
- ▶ Website:
 - ▶ [http://didik.blog.undip.ac.id/
buku/sistem-digital/](http://didik.blog.undip.ac.id/buku/sistem-digital/)



Bahasan

Peta Karnaugh
Rangkaian POS Optimal
Fungsi Tidak Lengkap

Rangkaian Multi-Keluaran

Ringkasan

Lisensi

- ▶ Rangkaian optimal
 - ▶ *Cost* rangkaian sekecil mungkin: jumlah gerbang (dan transistor), jumlah jalur
 - ▶ Fungsional terpenuhi
 - ▶ *Constraint* terpenuhi: delay, *fanout (driving)*, area
- ▶ Rangkaian optimal biasanya minimal
- ▶ Rangkaian optimal bisa diperoleh dengan teknik:
 1. Penyederhanaan fungsi logika
 - ▶ Menggunakan prinsip-prinsip Aljabar Boolean
 - ▶ Menggunakan Karnaugh Map
 2. Penggunaan gerbang secara bersama untuk beberapa fungsi sekaligus, membentuk **rangkaian multi-keluaran**

Peta Karnaugh Rangkaian POS Optimal Fungsi Tidak Lengkap

Rangkaian Multi-Keluaran

Ringkasan

Lisensi

Minimisasi Ekspresi POS

- ▶ Menggunakan prinsip dualitas
- ▶ K-map dapat langsung dibentuk baik dari ekspresi $\sum m$ maupun $\prod M$
 - ▶ Grouping Maxterm yang bernilai 0 sebesar mungkin
 - ▶ Bentuk persamaan POS dari himpunan Maxterm minimum
 - ▶ Prinsip prime implicant esensial berlaku? berlaku, dengan pengertian implicant adalah Maxterm atau group Maxterm

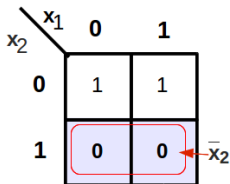
Representasi K-map POS

x_1	x_2	$f(x_1, x_2)$	Maxterm
0	0	M_0	$x_1 + x_2$
0	1	M_1	$x_1 + \bar{x}_2$
1	0	M_2	$\bar{x}_1 + x_2$
1	1	M_3	$\bar{x}_1 + \bar{x}_2$

$x_1 \backslash x_2$	0	1
0	M_0	M_2
1	M_1	M_3

Contoh K-map POS

- ▶ Nyatakan fungsi sederhana dari POS $f(x_1, x_2) = \prod M(1, 3)$



- ▶ Menghasilkan $f(x_1, x_2) = \prod M(1, 3) = \bar{x}_2$
- ▶ Bukti:

$$\begin{aligned}f(x_1, x_2) &= (x_1 + \bar{x}_2)(\bar{x}_1 + \bar{x}_2) \\ &= \bar{x}_2\end{aligned}$$

POS Minimal dari $\sum m$ atau $\prod M$

x_1x_2	00	01	11	10
x_3 0	1	1	0	0
1	1	0	0	1

Red boxes highlight the 0s in the top-right and middle-right cells. Red arrows point to these boxes with labels \bar{x}_1+x_3 and $\bar{x}_2+\bar{x}_3$.

x_1x_2	00	01	11	10
x_3 0	1	1	1	0
1	0	1	1	0

Red boxes highlight the 0s in the top-right and bottom-right cells. Red arrows point to these boxes with labels \bar{x}_1+x_2 and $x_2+\bar{x}_3$.

Diberikan:

$$f(x_1, x_2, x_3) = \sum m(0, 1, 2, 5)$$

$$\begin{aligned} f &= \sum m(0, 1, 2, 5) \\ &= (\bar{x}_1 + x_3)(\bar{x}_2 + \bar{x}_3); \text{ POS} \\ &= \bar{x}_1\bar{x}_3 + \bar{x}_2x_3; \text{ SOP} \\ &= \prod M(3, 4, 6, 7) \end{aligned}$$

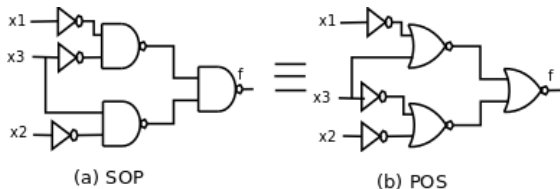
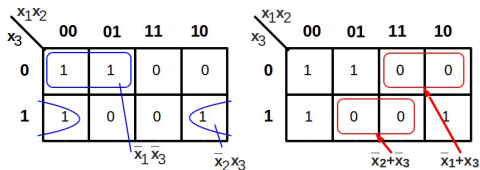
Diberikan:

$$f(x_1, x_2, x_3) = \prod M(1, 4, 5)$$

$$\begin{aligned} f &= \prod M(1, 4, 5) \\ &= (\bar{x}_1 + x_2)(x_2 + \bar{x}_3); \text{ POS} \\ &= x_2 + \bar{x}_1\bar{x}_3; \text{ SOP} \\ &= \sum m(0, 2, 3, 6, 7) \end{aligned}$$

Desain Rangkaian SOP dan POS

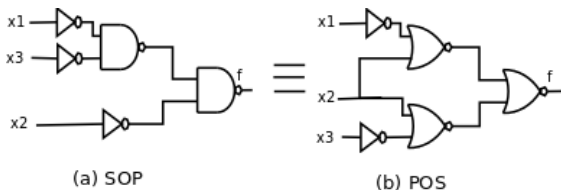
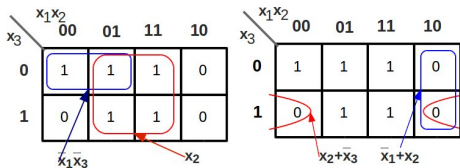
- ▶ Diketahui fungsi SOP $f(x_1, x_2, x_3) = \sum m(0, 1, 2, 5)$.
Desain rangkaian NAND-NAND dan NOR-NOR



- ▶ Cost?

Memilih Desain? SOP atau POS

- ▶ Desain rangkaian sederhana untuk $f(x_1, x_2, x_3) = \prod M(1, 4, 5)$



- ▶ Cost? Mana yang dipilih?

Ketentuan Rangkaian POS

- ▶ POS minimum berisi semua implicant utama esensial
- ▶ Langkah menemukan rangkaian dengan cost minimum:
 1. Mencari semua implicant utama dari fungsi f
 2. Mencari himpunan implicant utama esensial
 3. Jika himpunan tersebut telah meng-cover semua Maxterm bernilai 0, maka set ini adalah cover dari f yang diinginkan. Jika terdapat Maxterm bernilai 0 yang belum ter-cover, maka perlu dipilih implicant utama non-esensial yang harus ditambahkan ke dalam fungsi agar fungsi valid, namun tetap minimum.
Penentuan implicant utama non-esensial dapat dilakukan secara heuristik, yaitu mencoba semua kemungkinan untuk mendapatkan cover dengan biaya rangkaian minimal

POS 4-Variabel Minimal

$x_3x_4 \backslash x_1x_2$	00	01	11	10
00	0	0	0	1
01	0	0	1	1
11	1	0	0	1
10	1	0	0	1

$$(x_1, x_2, x_3, x_4) = \sum m(2, 3, 8, 9, 10, 11, 13, 14, 15) \\ = \prod M(0, 1, 4, 5, 6, 7, 12, 15)$$

- ▶ Prime implicant: $x_1 + x_3$, $\bar{x}_2 + \bar{x}_3$, $\bar{x}_2 + x_4$ dan $x_1 + \bar{x}_2$
- ▶ Esensial: $x_1 + x_3$, $\bar{x}_2 + \bar{x}_3$, dan $\bar{x}_2 + x_4$
- ▶ non-esensial: $x_1 + \bar{x}_2$ (biru)
- ▶ $f_{min} = (x_1 + x_3)(\bar{x}_2 + \bar{x}_3)(\bar{x}_2 + x_4)$

x_3x_4 \ x_1x_2	00	01	11	10
00	0	1	0	1
01	0	1	1	1
11	1	1	0	1
10	1	0	0	1

- ▶ Persamaan SOP dan POS
- ▶ Cari semua prime implicant dari f
- ▶ Cari set prime implicant esensial
- ▶ Cari cover dengan cost terendah dari semua kombinasi prime implicant non-esensial

Peta Karnaugh

Rangkaian POS Optimal

Fungsi Tidak Lengkap

Rangkaian Multi-Keluaran

Ringkasan

Lisensi

Peta Karnaugh
Rangkaian POS Optimal
Fungsi Tidak Lengkap

Rangkaian
Multi-Keluaran

Ringkasan

Lisensi

Fungsi Tidak Lengkap

- ▶ Dalam sistem digital, sering terjadi beberapa kondisi input yang tidak akan pernah terjadi
- ▶ Kombinasi input seperti itu disebut kondisi **don't care**
- ▶ Dalam desain rangkaian, kondisi don't care dapat diabaikan (keluaran untuk kondisi tersebut dapat diberikan 0 atau 1 di tabel kebenaran)
- ▶ Fungsi yang mengandung kondisi don't care disebut fungsi yang dispesifikasikan tidak lengkap (***incompletely specified***)

Contoh Kondisi Don't Care

- ▶ Diinginkan sistem untuk mendeteksi suhu ekstrem di bawah $10^{\circ}C$ dan di atas $80^{\circ}C$. Deteksi suhu menggunakan dua buah sensor suhu, yang masing-masing dapat menghasilkan nilai 1 jika suhu $> 10^{\circ}C$ dan jika suhu $> 80^{\circ}C$. Jika suhu di bawah $10^{\circ}C$ dan di atas $80^{\circ}C$, maka sebuah lampu akan menyala. Nyatakan deskripsi sistem tersebut dalam tabel kebenaran
- ▶ **Solusi.** Jika x_1 menyatakan suhu $> 10^{\circ}C$ dan x_2 suhu $> 80^{\circ}C$, maka

x_1	x_2	f	<i>keterangan</i>
0	0	1	suhu $< 10^{\circ}C$
0	1	d	tidak pernah terjadi
1	0	0	$10^{\circ}C < \text{suhu} < 80^{\circ}C$
1	1	1	suhu $> 80^{\circ}C$

Contoh Don't Care

- Di K-Map, masukan don't care **bisa diberi nilai 0 atau 1** sedemikian sehingga diperoleh fungsi yang optimal

x_1	x_2	x_3	f
0	0	0	1
0	0	1	1
0	1	0	d
0	1	1	d
1	0	0	1
1	0	1	1
1	1	0	0
1	1	1	0

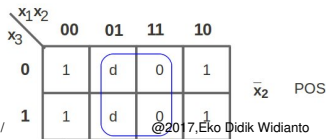
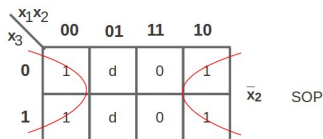
- Asumsi fungsi 3 variabel. Kombinasi masukan $\{x_1 x_2 x_3\} = 010 | 011$ tidak pernah terjadi, selebihnya

$$f(x_1, x_2, x_3) = \sum m(1, 4, 5, 6)$$

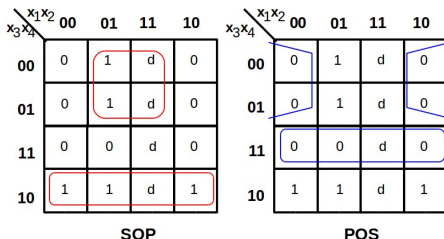
$$f(x_1, x_2, x_3) = \sum m(1, 4, 5, 6) + d(2, 3);$$

atau

$$f = \prod M(0, 7) \cdot D(2, 3)$$

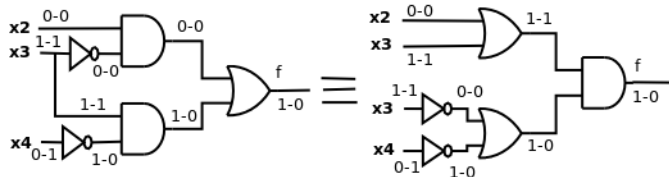


Contoh Don't Care 4 variabel



- ▶ SOP: $f(x_1, x_2, x_3, x_4) = \sum m(2, 4, 5, 6, 10) + D(12, 13, 14, 15)$
- ▶ POS: $f(x_1, x_2, x_3, x_4) = \prod M(0, 1, 3, 7, 8, 9, 11) \cdot D(12, 13, 14, 15)$
- ▶ SOP: $f_{min} = x_2\bar{x}_3 + x_3\bar{x}_4$, POS: $f_{min} = (x_2 + x_3)(\bar{x}_3 + \bar{x}_4)$
- ▶ Jika don't care tidak disertakan: misalnya menganggap nilainya selalu 0
 - ▶ SOP: $f = \bar{x}_1x_2\bar{x}_3 + \bar{x}_1x_3\bar{x}_4 + \bar{x}_2x_3\bar{x}_4$
 - ▶ POS: $f = (x_2 + x_3)(\bar{x}_3 + \bar{x}_4)(\bar{x}_1 + \bar{x}_2)$
 - ▶ **Cost mungkin lebih tinggi**

- $f_{min} = x_2\bar{x}_3 + x_3\bar{x}_4$ dan $f_{min} = (x_2 + x_3)(\bar{x}_3 + \bar{x}_4)$

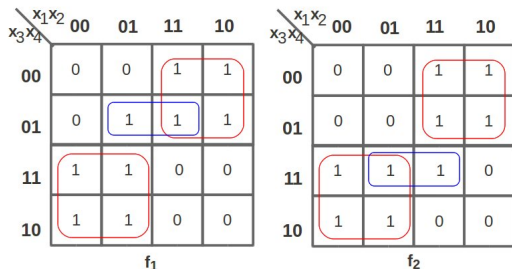


Rangkaian dengan Banyak Keluaran

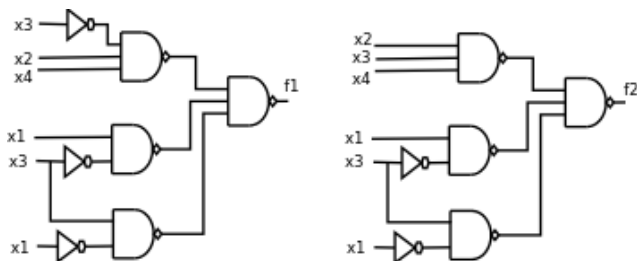
- ▶ Sebelumnya dibahas fungsi dengan keluaran tunggal berikut dengan implementasi rangkaiannya
- ▶ Dalam prakteknya, beberapa fungsi tunggal tersebut merupakan bagian dari rangkaian logika yang lebih besar
- ▶ Rangkaian-rangkaian dari fungsi tersebut mungkin dapat dikombinasikan ke dalam **rangkaian tunggal dengan cost lebih rendah** dengan keluaran lebih dari satu
 - ▶ Pemakaian bersama blok gerbang oleh beberapa rangkaian fungsi tunggal

Contoh Rangkaian Multi-Keluaran

- ▶ $f_1(x_1, x_2, x_3, x_4) = \sum m(2, 3, 5, 6, 8, 13) + d(7, 9, 11, 12)$
- ▶ $f_2(x_1, x_2, x_3, x_4) = \prod M(0, 1, 4, 5, 10, 11, 14) \cdot D(2, 3)$



Rangkaian Terpisah

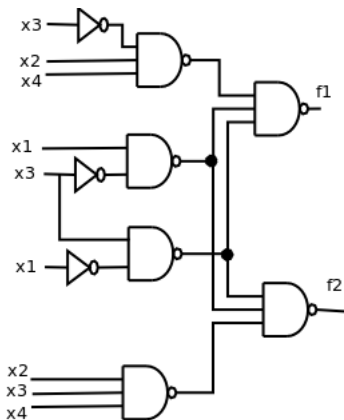


- ▶ $f_1 = x_1\overline{x_3} + \overline{x_1}x_3 + x_2\overline{x_3}x_4$, Cost=4 gerbang + 10 input(=14)
- ▶ $f_2 = x_1\overline{x_3} + \overline{x_1}x_3 + x_2x_3x_4$, Cost=4 gerbang + 10 input (=14)
- ▶ Cost total jika kedua fungsi diimplementasikan terpisah: 8 gerbang + 20 input (=28)
- ▶ Jika gerbang NOT diperhitungkan?

Contoh Rangkaian Multi-Keluaran

- ▶ Mengkombinasikan (prime) implicant yang sama dari dua/lebih fungsi mungkin bisa mengurangi cost

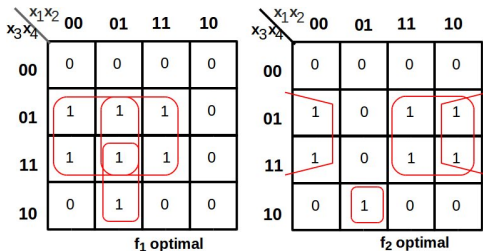
- ▶ Rangkaian multi-keluaran: $\begin{Bmatrix} f_1 \\ f_2 \end{Bmatrix} = x_1\bar{x}_3 + \bar{x}_1x_3 + \begin{Bmatrix} x_2\bar{x}_3x_4 \\ x_2x_3x_4 \end{Bmatrix}$



- ▶ Cost=6 gerbang + 16 input (=22), jika tanpa NOT
- ▶ Dengan NOT: biaya total = 28

Contoh Rangkaian Multi-Keluaran

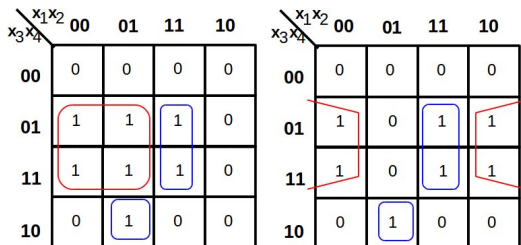
- Di contoh sebelumnya, terdapat prime implicant yang bersama. Kalau tidak ada yang bersama?



- $f_1 = \bar{x}_1x_4 + x_2x_4 + \bar{x}_1x_2x_3$, Cost=4 gerbang + 10 input(=14)
- $f_2 = x_1x_4 + \bar{x}_2x_4 + \bar{x}_1x_2x_3\bar{x}_4$, Cost=4 gerbang + 11 input (=15)
- Tidak ada gerbang prime implicant yang dapat dishared, sehingga cost total dari kombinasi 2 rangkaian adalah 8 gerbang + 21 input (=29)

Contoh Rangkaian Multi-Keluaran

- ▶ Tapi ada alternatif realisasi lainnya: menggunakan **implicant bersama** antara 2 fungsi



Gabungan $f_1 f_2$ optimal

- ▶ $f_1 = x_1x_2x_4 + \bar{x}_1x_2x_3\bar{x}_4 + \bar{x}_1x_4$
- ▶ $f_2 = x_1x_2x_4 + \bar{x}_1x_2x_3\bar{x}_4 + \bar{x}_2x_4$
- ▶ Rangkaian multikeluaran:
$$\left\{ \begin{matrix} f_1 \\ f_2 \end{matrix} \right\} = x_1x_2x_4 + \bar{x}_1x_2x_3\bar{x}_4 + \left\{ \begin{matrix} \bar{x}_1x_4 \\ \bar{x}_2x_4 \end{matrix} \right\}$$
- ▶ Cost gabungan total= 6 gerbang + 17 input (=23)

- ▶ Yang telah kita pelajari hari ini:
 - ▶ Penyederhanaan fungsi logika menggunakan peta Karnaugh melalui Grouping Maxterm untuk fungsi POS, baik fungsi 2-variabel sampai 6-variabel
 - ▶ Fungsi tidak lengkap dengan masukan don't care
 - ▶ Rangkaian multi-keluaran untuk mengoptimalkan penggunaan gerbang
- ▶ Yang akan kita pelajari di pertemuan berikutnya adalah penyederhanaan fungsi logika menggunakan Quine-McKluskey untuk memperoleh rangkaian yang optimal
 - ▶ Pelajari: <http://didik.blog.undip.ac.id/2017/03/06/tkc205-sistem-digital-2016-genap/>

Creative Common Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0)

- ▶ Anda bebas:
 - ▶ untuk **Membagikan** — untuk menyalin, mendistribusikan, dan menyebarkan karya, dan
 - ▶ untuk **Remix** — untuk mengadaptasikan karya
- ▶ Di bawah persyaratan berikut:
 - ▶ **Atribusi** — Anda harus memberikan atribusi karya sesuai dengan cara-cara yang diminta oleh pembuat karya tersebut atau pihak yang mengeluarkan lisensi. Atribusi yang dimaksud adalah mencantumkan alamat URL di bawah sebagai sumber.
 - ▶ **Pembagian Serupa** — Jika Anda mengubah, menambah, atau membuat karya lain menggunakan karya ini, Anda hanya boleh menyebarkan karya tersebut hanya dengan lisensi yang sama, serupa, atau kompatibel.
- ▶ Lihat: **Creative Commons Attribution-ShareAlike 3.0 Unported License**
- ▶ Alamat URL: <http://didik.blog.undip.ac.id/buku/sistem-digital/>