

Metode Quine McKluskey dan Program Bantu Komputer

Kuliah#6 TSK205 Sistem Digital

Eko Didik Widianto

Departemen Teknik Sistem Komputer, Universitas Diponegoro

Penyederhanaan Persamaan Logika

- ▶ Perancangan rangkaian logika minimal memerlukan teknik penyederhanaan persamaan logika
 - ▶ secara aljabar
 - ▶ peta Karnaugh
 - ▶ metode tabular Quine-McCluskey
- ▶ Metode tabular Quine-McCluskey lebih efisien digunakan di program komputer daripada peta Karnaugh

- ▶ Sebelumnya dibahas tentang optimasi rangkaian dengan **penyederhanaan ekspresi logika secara Aljabar, peta Karnaugh dan rangkaian multi-output** untuk rangkaian SOP maupun POS
- ▶ Dalam kuliah ini, akan dibahas tentang:
 - ▶ penyederhanaan fungsi logika menggunakan metode tabular Quine-McCluskey
 - ▶ program bantu komputer Bmin untuk melakukan sintesis rangkaian logika minimum
 - ▶ program bantu komputer simulator rangkaian Qucs untuk analisis rangkaian

- ▶ Setelah mempelajari bab ini, mahasiswa akan mampu:
 1. [C3] merancang rangkaian logika minimal dengan menggunakan algoritma/metode tabular Quine-McCluskey
 2. [C3] menggunakan perangkat lunak komputer Bmin untuk menyederhanakan rangkaian logika
 3. [C5] mengevaluasi hasil rancangan rangkaian logika menggunakan simulator Qucs

- ▶ 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 5: Metode Quine-McCluskey dan Program Bantu Komputer**)

- ▶ Materi:
 - ▶ 5.1 Metode Quine-McKluskey
 - ▶ 5.2 Program Bantu Komputer: Bmin, Qmls, Qucs
- ▶ Website:
 - ▶ <http://didik.blog.undip.ac.id/buku/sistem-digital/>



Bahasan

Metode Quine-McKluskey
Metode QM untuk SOP
Metode QM untuk POS Minimal

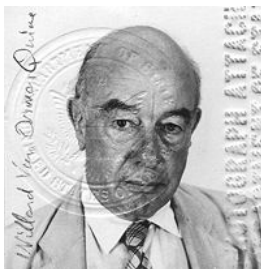
Program Bantu Komputer
Bmin: Visualisasi Penyederhanaan Boolean
Qmls: Quine-McCluskey Logic Simplifier
Qucs: Simulator Rangkaian

Ringkasan

Lisensi

Metode Quine-McKluskey (QM)

- ▶ Digunakan untuk menyederhanakan fungsi logika sehingga dihasilkan rangkaian logika minimal
 - ▶ Disebut juga **metode tabular**, karena menggunakan tabulasi
 - ▶ Dikembangkan oleh W.V. Quine dan Edward J. McCluskey
- Algoritma ini memberikan hasil yang deterministik untuk memastikan bahwa fungsi logika yang minimal telah tercapai
- ▶ Fungsinya seperti peta Karnaugh, namun lebih efisien untuk digunakan di program komputer
 - ▶ Untuk fungsi dengan lebih dari 4 variabel
 - ▶ Namun, jumlah variabel akan menaikkan waktu eksekusi (run-time) secara eksponensial



(Willard Quine, Wikipedia)

Algoritma Quine McKluskey:

1. Bangkitkan prime implicant
2. Susun tabel prime implicant
3. Sederhanakan tabel
 - 3.1 Buang prime implicant esensial. Note: nanti disertakan dalam fungsi akhirnya
 - 3.2 Menghapus row dominance (Maxterm/minterm terbanyak yang dicover oleh prime implicant)
 - 3.3 Memilih column dominance (prime implicant yang mengcover paling banyak Maxterm/minterm)
4. Selesaikan tabel

Tujuannya mencari **prime implicant esensial** (primer, sekunder, dst)

Metode Quine-McKluskey
Metode QM untuk SOP
Metode QM untuk POS Minimal

Program Bantu Komputer
Bmin: Visualisasi Penyederhanaan Boolean
Qmls: Quine-McCluskey Logic Simplifier
Qucs: Simulator Rangkaian

Ringkasan

Lisensi

Metode
Quine-McKluskey
Metode QM untuk SOP
Metode QM untuk POS
Minimal

Program Bantu
Komputer

Ringkasan

Lisensi

Buat Prime Implicant

Diinginkan rangkaian:

$$f(x_1, x_2, x_3, x_4) = \sum m(0, 2, 5, 6, 7, 8, 10, 12, 13, 14, 15)$$

Langkah 1: Bangkitkan Prime Implicant

Kolom 1			Kolom 2			Kolom 3			
0	0000	V	(0,2)	00-0	V	(0,2,8,10)	-0-0	$\bar{x}_2\bar{x}_4$	
2	0010	V	(0,8)	-000	V	(0,8,2,10)	-0-0		Duplikat
8	1000	V	(2,6)	0-10	V	(2,6,10,14)	--10	$x_3\bar{x}_4$	
5	0101	V	(2,10)	-010	V	(2,10,6,14)	--10		Duplikat
6	0110	V	(8,10)	10-0	V	(8,10,12,14)	1--0	$x_1\bar{x}_4$	
10	1010	V	(8,12)	1-00	V	(8,12,10,14)	1--0		Duplikat
12	1100	V	(5,7)	01-1	V	(5,7,13,15)	-1-1	x_2x_4	
7	0111	V	(5,13)	-101	V	(5,13,7,15)	-1-1		Duplikat
13	1101	V	(6,7)	011-	V	(6,7,14,15)	-11-	x_2x_3	
14	1110	V	(6,14)	-110	V	(6,14,7,15)	-11-		Duplikat
15	1111	V	(10,14)	1-10	V	(12,13,14,15)	11--	x_1x_2	
			(12,13)	110-	V	(12,14,13,15)	11--		Duplikat
			(12,14)	11-0	V				
			(7,15)	-111	V				
			(13,15)	11-1	V				
			(14,15)	111-	V				

► Baris duplikat dihapus

Susun Tabel Prime Implicant

Langkah 2: Susun Tabel Prime Implicant

- ▶ Disusun dari langkah 1, kolom 3

	$\bar{x}_2\bar{x}_4$	$x_3\bar{x}_4$	$x_1\bar{x}_4$	x_2x_4	x_2x_3	x_1x_2
	(0,2,8,10)	(2,6,10,14)	(8,10,12,14)	(5,7,13,15)	(6,7,14,15)	(12,13,14,15)
0	x					
2	x	x				
5				x		
6		x			x	
7				x	x	
8	x		x			
10	x	x	x			
12			x			x
13				x		x
14		x	x		x	x
15				x	x	x

Hapus Prime Implicant Esensial

Langkah 3a: Hapus Prime Implicant Esensial dari Tabel (Iterasi #1)

	x_2x_4 (0,2,8,10)	x_3x_4 (2,6,10,14)	x_1x_4 (8,10,12,14)	x_2x_4 (5,7,13,15)	x_2x_3 (6,7,14,15)	x_1x_2 (12,13,14,15)
0	X					
2	X	X				
5				X		
6		X			X	
7				X	X	
8	X		X			
10	X	X	X			
12			X			X
13				X		X
14		X	X		X	X
15				X	X	X

Column Removed

Column Removed

Distinguish row

Distinguish row

- ▶ **Prime implicant esensial:** $\bar{x}_2\bar{x}_4$ dan x_2x_4
 - ▶ dibuang untuk penyederhanaan lebih lanjut
 - ▶ **ditambahkan** di solusi akhir

Hapus Baris Dominan

Langkah 3b: Hapus Baris yang Mendominasi (Dominating Row)

	$x_3\bar{x}_4$ (2,6,10,14)	$x_1\bar{x}_4$ (8,10,12,14)	x_2x_3 (6,7,14,15)	x_1x_2 (12,13,14,15)
6	X		X	
12		X		X
14	X	X	X	X

Dominating row

- ▶ Baris ke-14 dihapus karena setiap term perkalian yang mengcover 6 atau 12 akan mengcover 14

Langkah 3c: Pilih Kolom

	$x_3\bar{x}_4$ (2,6,10,14)	$x_1\bar{x}_4$ (8,10,12,14)	x_2x_3 (6,7,14,15)	x_1x_2 (12,13,14,15)
6	X		X	
12		X		X

- ▶ prime implicant $x_3\bar{x}_4$ dan x_2x_3 saling mendominasi, bisa dipilih salah satu
- ▶ $x_1\bar{x}_4$ dan x_1x_2 saling mendominasi, bisa dipilih salah satu

Kemungkinan Solusi Pemilihan Prime Implicant

	$x_3\bar{x}_4$	$x_1\bar{x}_4$
	(2,6,10,14)	(8,10,12,14)
6	X	
12		X

Solusi 1

	$x_3\bar{x}_4$	x_1x_2
	(2,6,10,14)	(12,13,14,15)
6	X	
12		X

Solusi 2

	$x_1\bar{x}_4$	x_2x_3
	(8,10,12,14)	(6,7,14,15)
6		X
12	X	

Solusi 3

	x_2x_3	x_1x_2
	(6,7,14,15)	(12,13,14,15)
6	X	
12		X

Solusi 4

Hapus Prime Implicant Esensial Sekunder

Langkah 3a: Hapus Prime Implicant Esensial Sekunder (Iterasi #2)

Terdapat 2 solusi

	X_3X_4	X_1X_4	
	(2,6,10,14)	(8,10,12,14)	
6	X		Distinguish row
12		X	Distinguish row
	Prime implicant esensial	Prime implicant esensial	
	X_2X_3	X_1X_4	
	(6,7,14,15)	(8,10,12,14)	
6	X		Distinguish row
12		X	Distinguish row
	Prime implicant esensial	Prime implicant esensial	
	X_3X_4	X_1X_3	
	(2,6,10,14)	(12,13,14,15)	
6	X		Distinguish row
12		X	Distinguish row
	Prime implicant esensial	Prime implicant esensial	
	X_2X_3	X_1X_3	
	(6,7,14,15)	(12,13,14,15)	
6	X		Distinguish row
12		X	Distinguish row
	Prime implicant esensial	Prime implicant esensial	

- ▶ **Prime implicant esensial sekunder:** $x_3\bar{x}_4$ dan $x_1\bar{x}_4$ atau x_2x_3 dan x_1x_2
 - ▶ dibuang untuk penyederhanaan lebih lanjut
 - ▶ **ditambahkan** di solusi akhir

Langkah 4: Solusi Akhir

- ▶ Tidak ada lagi baris yang perlu disederhanakan
- ▶ Solusi minimum akan berisi **prime implicant esensial primer** dan **sekunder**

$$f_{min} = \underbrace{\bar{X}_2\bar{X}_4 + X_2X_4}_{PIE \text{ primer}} + \underbrace{\left\{ \begin{array}{l} X_3\bar{X}_4 + X_1\bar{X}_4 \\ X_3\bar{X}_4 + X_1X_2 \\ X_2X_3 + X_1\bar{X}_4 \\ X_2X_3 + X_1X_2 \end{array} \right\}}_{PIE \text{ sekunder}}$$

Metode Quine-McKluskey

Metode QM untuk SOP

Metode QM untuk POS Minimal

Program Bantu Komputer

Bmin: Visualisasi Penyederhanaan Boolean

Qmls: Quine-McCluskey Logic Simplifier

Qucs: Simulator Rangkaian

Ringkasan

Lisensi

Metode QM untuk POS Minimal

- ▶ Diinginkan rangkaian POS dari

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

- ▶ Langkah 1: membangkitkan prime implicant

Kolom 1			Kolom 2			Kolom 3		
1	0001	V	(1,3)	00-1	V	(1,3,9,11)	-0-1	$x_2 + \bar{x}_4$
4	0100	$x_1 + \bar{x}_2 + x_3 + x_4$	(1,9)	-001	V	(1,9,3,11)	-0-1	Duplikat
3	0011	V	(3,11)	-011	V			
9	1001	V	(9,11)	10-1	V			
11	1011	V						

Susun Tabel Prime Implicant

Langkah 2: Susun Tabel Prime Implicant

	$x_2 + \bar{x}_4$	$x_1 + \bar{x}_2 + x_3 + x_4$
	(1,3,9,11)	(4)
1	X	
4		X
3	X	
9	X	
11	X	

Hapus Prime Implicant Esensial

Langkah 3a: Hapus Prime Implicant Esensial dari Tabel (Iterasi #1)

	$x_2 + \bar{x}_4$ (1,3,9,11)	$x_1 + \bar{x}_2 + x_3 + x_4$ (4)	
1	X		Distinguish row
4		X	Distinguish row
3	X		Distinguish row
9	X		Distinguish row
11	X		Distinguish row
	Column Removed		

Prime implicant: $x_2 + \bar{x}_4$ dan $x_1 + \bar{x}_2 + x_3 + x_4$

Solusi akhir

$$f_{min} = (x_2 + \bar{x}_4)(x_1 + \bar{x}_2 + x_3 + x_4)$$

Program Bantu Komputer

- ▶ Program bantu komputer (CAD: *Computer-Aided Design*) dapat dimanfaatkan untuk desain dan simulasi rangkaian logika
 - ▶ program desain: untuk mensintesis rangkaian logika dari suatu fungsi logika
 - ▶ menghasilkan rangkaian logika minimal
 - ▶ **Bmin** - Visualizer of Boolean Minimization (<http://bukka.eu/bmin/0.5.0>). Program GUI Qt untuk minimalisasi fungsi logika dengan K-Map dan tabular Quine-McKluskey
 - ▶ **Qmls** - Quine-McCluskey Logic Simplifier (<http://sourceforge.net/projects/qmls/>). Program CLI untuk minimalisasi fungsi logika dengan tabular Quine-McCluskey
 - ▶ program simulasi/analisis rangkain logika
 - ▶ **Qucs** - Quite Universal Circuit Simulator (<http://qucs.sourceforge.net/>). Program GUI Qt untuk desain dan simulasi rangkaian elektronik, termasuk rangkaian digital

Bahasan

Metode Quine
McKluskey dan
Program Bantu
Komputer

@2017,Eko Didik
Widianto

Metode Quine-McKluskey
Metode QM untuk SOP
Metode QM untuk POS Minimal

Program Bantu Komputer
Bmin: Visualisasi Penyederhanaan Boolean
Qmls: Quine-McCluskey Logic Simplifier
Qucs: Simulator Rangkaian

Ringkasan

Lisensi

Metode
Quine-McKluskey

Program Bantu
Komputer

Bmin: Visualisasi
Penyederhanaan Boolean

Qmls: Quine-McCluskey
Logic Simplifier

Qucs: Simulator Rangkaian

Ringkasan

Lisensi

Bmin: Visualizer of Boolean Minimization

- ▶ Program GUI untuk meminimalkan fungsi logika
 - ▶ Dibuat oleh Jakub Zelenka. Versi terbaru 0.5.0
 - ▶ Pustaka grafis: Qt
 - ▶ Masukan: Maxterm dan minterm, don't care
 - ▶ Metode minimalisasi: peta Karnaugh dan Quine-McCluskey
 - ▶ Representasi fungsi minimal: SOP dan POS
 - ▶ Batasan:
 - ▶ Peta Karnaugh untuk fungsi sampai 6 variabel
 - ▶ Quine-McCluskey untuk fungsi sampai 10 variabel
 - ▶ Nama variabel harus satu buah karakter ASCII
 - ▶ Alamat website: <http://bukka.eu/bmin/0.5.0>

Menu Utama Bmin

Metode Quine
McKluskey dan
Program Bantu
Komputer

@2017,Eko Didik
Widianto

Metode
Quine-McKluskey

Program Bantu
Komputer

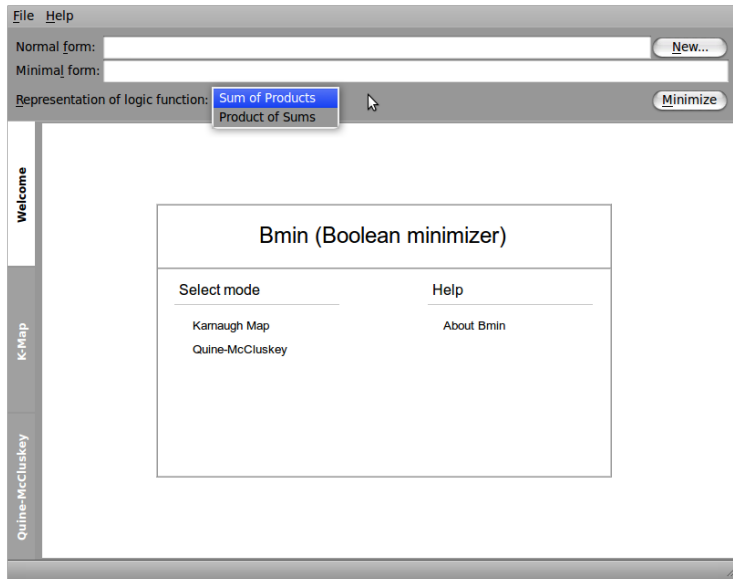
Bmin: Visualisasi
Penyederhanaan Boolean

Omls: Quine-McKluskey
Logic Simplifier

Oucs: Simulator Rangkaian

Ringkasan

Lisensi



Masakan Bmin

$$\text{Fungsi } f(x_1, x_2, x_3, x_4) = \sum m(2, 3, 7, 9, 11, 13) + \sum d(1, 10, 15)$$

Index	d	c	b	a
0	0	0	0	0
1	0	0	0	1
2	0	0	1	0
3	0	0	1	1
4	0	1	0	0
5	0	1	0	1
6	0	1	1	0
7	0	1	1	1
8	1	0	0	0

- ▶ Jumlah variabel masukan maksimal 10
- ▶ Nama variabel: karakter ascii
- ▶ Nilai variabel: 0, 1, X (don't care)

Hasil Desain Bmin dengan K-map

- ▶ Masukan: $f(d, c, b, a) = \text{sum } m(2, 3, 7, 8, 11, 13) + \text{sum } d(0, 10, 15)$
- ▶ **Fungsi sederhana:** $f(d, c, b, a) = ab + a'c' + acd$ (output)

The screenshot shows the Quine-McCluskey software interface. At the top, the 'Normal form' is $f(d,c,b,a) = \text{sum } m(2,3,7,8,11,13) + \text{sum } d(0,10,15)$ and the 'Minimal form' is $f(d,c,b,a) = ab + a'c' + acd$. The 'Representation of logic function' is set to 'Sum of Products'. The main area displays a K-map with the following values:

dc \ ba	00	01	11	10
00	X	0	1	1
01	0	0	1	0
11	0	1	X	0
10	1	0	1	X

The K-map shows prime implicants: a green box covering (00,0), (10,0), (00,1), and (10,1); a red box covering (00,1), (01,1), (10,1), and (11,1); and a purple box covering (11,0) and (11,1). The 'Minterms' list on the right includes 0-7, with 'Show covers' checked. The simplified terms 'ab', 'a'c'', and 'acd' are listed at the bottom.

Konversi ke POS

- ▶ Masukan: $f(d, c, b, a) = \text{prod } m(1, 4, 5, 6, 9, 12, 14) * \text{prod } d(0, 10, 15)$
- ▶ **Fungsi sederhana:** $f(d, c, b, a) = (b + d)(a + c')(a' + b + c)$ (output)

The screenshot shows the Quine-McCluskey software interface. At the top, the 'Normal form' is $f(d,c,b,a) = \text{prod } m(1,4,5,6,9,12,14) * \text{prod } d(0,10,15)$. The 'Minimal form' is $f(d,c,b,a) = (b + d)(a + c')(a' + b + c)$. The 'Representation of logic function' is set to 'Product of Sums'. The main area displays a K-Map for variables ba (columns) and dc (rows). The K-Map is a 4x4 grid with the following values:

	00	01	11	10
00	X	0	1	1
01	0	1	1	0
11	0	1	X	0
10	1	0	1	X

The K-Map shows several groups: a red box around (00,01), (01,01), (00,11), (01,11); a green box around (01,00), (01,01), (11,00), (11,01); a purple box around (00,01), (10,01); and a blue box around (00,10), (10,10). The 'Maxterms' list on the right shows the following terms:

- 0 $a + b + c + d$
- 1 $a' + b + c + d$
- 2 $a + b' + c + d$
- 3 $a' + b' + c + d$
- 4 $a + b + c' + d$
- 5 $a' + b + c' + d$
- 6 $a + b' + c' + d$
- 7 $a' + b' + c' + d$

The 'Show covers' section is checked and displays the simplified expression: $b + d$, $a + c'$, and $a' + b + c$.

Hasil Desain Bmin dengan QM

- ▶ Masukan: $f(d, c, b, a) = \text{sum } m(2, 3, 7, 8, 11, 13) + \text{sum } d(0, 10, 15)$
- ▶ Fungsi sederhana: $f(d, c, b, a) = ab + a'c' + acd$ (output)

File Help

Normal form: $f(d,c,b,a) = \text{sum } m(2,3,7,8,11,13) + \text{sum } d(0,10,15)$ Edit...

Minimal form: $f(d,c,b,a) = ab + a'c' + acd$

Representation of logic function: Sum of Products Minimize

Finding Prime Implicants

Number of 1s	Size 1 primes		Size 2 primes		Size 4 primes	
	Minterm	0-cube	Minterm	1-cube	Minterm	2-cube
0	m0	0000	m(0,2) m(0,8)	00-0 -000	m(0,2,8,10)	-0-0*
1	m2 m8	0010 1000	m(2,3) m(2,10) m(8,10)	001- -010 10-0	m(2,3,10,11)	-01-*
2	m3 m10	0011 1010	m(3,7) m(3,11) m(10,11)	0-11 -011 101-	m(3,7,11,15)	--11*
3	m7 m11 m13	0111 1011 1101	m(7,15) m(11,15) m(13,15)	-111 1-11 11-1*		
4	m15	1111				

Prime Implicants Table

	2	3	7	8	11	13
m(0,2,8,10)	X			X		
m(2,3,10,11)	X	X			X	
m(3,7,11,15)		X	X		X	
m(13,15)						X

QM: Konversi ke POS

- ▶ Masukan: $f(d, c, b, a) = \text{prod } m(1, 4, 5, 6, 9, 12, 14) * \text{prod } d(0, 10, 15)$
- ▶ Fungsi sederhana: $f(d, c, b, a) = (b + d)(a + c')(a' + b + c)$ (output)

File Help

Normal form: $f(d,c,b,a) = \text{prod } m(1,4,5,6,9,12,14) * \text{prod } d(0,10,15)$ Edit...

Minimal form: $f(d,c,b,a) = (b + d)(a + c')(a' + b + c)$

Representation of logic function: Product of Sums Minimize

Finding Prime Implicants

Number of 0s	Size 1 primes		Size 2 primes		Size 4 primes	
	Maxterm	0-cube	Maxterm	1-cube	Maxterm	2-cube
0	M15	1111	M(14,15)	111-*		
1	M14	1110	M(6,14) M(10,14) M(12,14)	-110 1-10* 11-0	M(4,6,12,14)	-1-0*
2	M5 M6 M9 M10 M12	0101 0110 1001 1010 1100	M(1,5) M(1,9) M(4,5) M(4,6) M(4,12)	0-01 -001* 010- 01-0 -100	M(0,1,4,5)	0-0-*
3	M1 M4	0001 0100	M(0,1) M(0,4)	000- 0-00		
4	M0	0000				

Prime Implicants Table

	1	4	5	6	9	12	14
M(0,1,4,5)	X	X	X				
M(1,9)	X			X			
M(4,6,12,14)	X	X	X	X	X	X	
M(10,14)						X	

Welcome

K-Map

Quine-McCluskey

Persamaan 10 Variabel

- Nyatakan persamaan minimal fungsi $f(j, i, h, g, f, e, d, c, b, a) = \sum(1, 73, 75, 77, 79, 203, 205, 207, 329, 331, 335, 463, 1023) + d(201, 333, 457, 459, 461)$ menggunakan bmin

File Help

Normal form: $f(j,i,h,g,f,e,d,c,b,a) = \sum m(1,73,75,77,79,203,205,207,329,331,335,463,1023) + \sum d(201,333,457,459,461)$ Edit...

Minimal form: $f(j,i,h,g,f,e,d,c,b,a) = ab'c'd'e'f'g'h'i'j' + ade'f'g'j' + abcdefghij$

Representation of logic function: Sum of Products Minimize

	m		m		m	
	m203	0011001011	m(79,335)	0-01001111	m(203,207,459,463)	0-11001-11
	m205	0011001101	m(203,207)	0011001-11	m(205,207,461,463)	0-110011-1
	m331	0101001011	m(203,459)	0-11001011	m(331,335,459,463)	01-1001-11
	m333	0101001101	m(205,207)	00110011-1	m(333,335,461,463)	01-10011-1
	m457	0111001001	m(205,461)	0-11001101	m(457,459,461,463)	0111001--1
			m(331,335)	0101001-11		
			m(331,459)	01-10010111		
			m(333,335)	01010011-1		
			m(333,461)	01-10011011		
			m(457,459)	01110010-1		
			m(457,461)	0111001-01		
6	m207	0011001111	m(207,463)	0-11001111		
	m335	0101001111	m(335,463)	01-10011111		
	m459	0111001011	m(459,463)	0111001-11		
	m461	0111001101	m(461,463)	01110011-1		
7	m463	0111001111				
8						
9						
10	m1023	1111111111*				

Prime Implicants Table

	1	73	75	77	79	203	205	207	329	331	335	463	1023
m1	X												
m(73,75,77,79,201,203,205,207,329,331,333,335,457,459,461,463)	X	X	X	X	X	X	X	X	X	X	X	X	
m1023													X

Quine-McKluskey

Bahasan

Metode Quine
McKluskey dan
Program Bantu
Komputer

@2017,Eko Didik
Widianto

Metode Quine-McKluskey
Metode QM untuk SOP
Metode QM untuk POS Minimal

Program Bantu Komputer

Bmin: Visualisasi Penyederhanaan Boolean
Qmls: Quine-McCluskey Logic Simplifier
Qucs: Simulator Rangkaian

Metode
Quine-McKluskey

Program Bantu
Komputer

Bmin: Visualisasi
Penyederhanaan Boolean

Qmls: Quine-McCluskey
Logic Simplifier

Qucs: Simulator Rangkaian

Ringkasan

Lisensi

Ringkasan

Lisensi

Qmls: Quine-McCluskey Logic Simplifier

- ▶ Program CLI (command line interface) untuk meminimalkan fungsi logika
 - ▶ Dibuat oleh Dannel Albert <dalbert@capitol-college.edu>. Versi terbaru 0.2
 - ▶ CLI, masukan diberikan dari command line atau file teks
 - ▶ Masukan: minterm, don't care
 - ▶ Nama variabel: string sebarang, bisa dengan indeks (misalnya: x4, a1 dan seterusnya)
 - ▶ Metode minimalisasi: Quine-McCluskey
 - ▶ Representasi fungsi minimal: SOP
 - ▶ Alamat website:
<http://sourceforge.net/projects/qmls/>

Masukan Fungsi dari File Teks

```
didik@didik-laptop:/media/DATA/Kuliah-UNDIP/TSK205-SistemDigital/Files/qmls-0.25$ cat problem01.txt
f(x1,x2,x3,x4) = m(2,3,7,8,11,13) + d(0,10,15)
didik@didik-laptop:/media/DATA/Kuliah-UNDIP/TSK205-SistemDigital/Files/qmls-0.25$ ./qmls/qmls < problem01.txt > solution01.txt
didik@didik-laptop:/media/DATA/Kuliah-UNDIP/TSK205-SistemDigital/Files/qmls-0.25$ tail -n14 solution01.txt
[PI]
13,15 (2) PI#1
0,2,8,10 (2,8) PI#2
2,3,10,11 (1,8) PI#3
3,7,11,15 (4,8) PI#4

*   1       2   3   7   8   11  13
*   1       X       X       X
*   2       X       X       X
*   3       X   X       X
*   4       X   X   X       X

f(x1,x2,x3,x4) = x3x4 + x2'x4' + x1x2x4
didik@didik-laptop:/media/DATA/Kuliah-UNDIP/TSK205-SistemDigital/Files/qmls-0.25$
```

- ▶ Dari fungsi $f(x_1, x_2, x_3, x_4) = m(2, 3, 7, 8, 11, 13) + d(0, 10, 15)$
- ▶ Menghasilkan fungsi sederhana yang sama:
$$f(x_1, x_2, x_3, x_4) = x_3x_4 + x_2'x_4' + x_1x_2x_4$$

Metode
Quine-McKluskey

Program Bantu
Komputer

Bmin: Visualisasi
Penyederhanaan Boolean

Omls: Quine-McKluskey
Logic Simplifier

Oucs: Simulator Rangkaian

Ringkasan

Lisensi

Fungsi dengan 10 Variabel (Qmls)

```
didik@didik-laptop:/media/DATA/Kuliah-UNDIP/TSK205-SistemDigital/Files/qmls-0.2$ cat problem02.txt
f(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10) = m(1023,73,75,77,79,203,205,207,329,331,335,463,1) + d(201,333,457,459,461)
didik@didik-laptop:/media/DATA/Kuliah-UNDIP/TSK205-SistemDigital/Files/qmls-0.2$ ./qmls/qmls < problem02.txt > solution02.txt
didik@didik-laptop:/media/DATA/Kuliah-UNDIP/TSK205-SistemDigital/Files/qmls-0.2$ tail -n14 solution02.txt
----
[PI]
1 PI#1
1023 PI#2
73,75,77,79,201,203,205,207,329,331,333,335,457,459,461,463 (2,4,128,256) PI#3

      1023  73  75  77  79  203  205  207  329  331  335  463  1
*      1
*      2      X
*      3      X  X  X  X  X  X  X  X  X  X  X

f(x1,x2,x3,x4,x5,x6,x7,x8,x9,x10) = x1x2x3x4x5x6x7x8x9x10 + x1'x4x5'x6'x7x10 + x1'x2'x3'x4'x5'x6'x7'x8'x9'x10
didik@didik-laptop:/media/DATA/Kuliah-UNDIP/TSK205-SistemDigital/Files/qmls-0.2$
```

► Fungsi: $f(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}) = m(1023, 73, 75, 77, 79, 203, 205, 207, 329, 331, 335, 463, 1) + d(201, 333, 457, 459, 461)$

► Fungsi sederhana:
 $f(x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, x_9, x_{10}) = x_1x_2x_3x_4x_5x_6x_7x_8x_9x_{10} + x_1'x_4x_5'x_6'x_7x_{10} + x_1'x_2'x_3'x_4'x_5'x_6'x_7'x_8'x_9'x_{10}$

Metode
Quine-McKluskey

Program Bantu
Komputer

Bmin: Visualisasi
Penyederhanaan Boolean

Omls: Quine-McKluskey
Logic Simplifier

Qucs: Simulator Rangkaian

Ringkasan

Lisensi

Bahasan

Metode Quine
McKluskey dan
Program Bantu
Komputer

@2017,Eko Didik
Widianto

Metode Quine-McKluskey
Metode QM untuk SOP
Metode QM untuk POS Minimal

Program Bantu Komputer

Bmin: Visualisasi Penyederhanaan Boolean
Qmls: Quine-McCluskey Logic Simplifier
Qucs: Simulator Rangkaian

Ringkasan

Lisensi

Metode
Quine-McKluskey

Program Bantu
Komputer

Bmin: Visualisasi
Penyederhanaan Boolean
Qmls: Quine-McCluskey
Logic Simplifier

Qucs: Simulator Rangkaian

Ringkasan

Lisensi

▶ Qucs (Quite Universal Circuit Simulator)

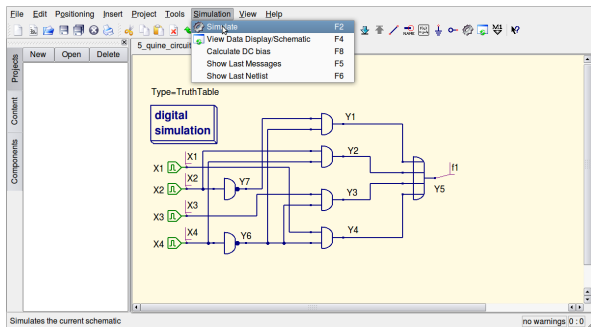
- ▶ untuk mensimulasikan rangkaian elektronika secara umum, termasuk rangkaian digital (logika)
- ▶ dikembangkan oleh Michael Margraf dan berbasis pustaka grafis Qt
- ▶ dapat dijalankan di Linux atau Windows
- ▶ masukan program adalah berupa skematik rangkaian logika yang tersusun atas gerbang logika dan interkoneksinya
- ▶ Alamat website: <http://qucs.sourceforge.net/>

Simulasi Rangkaian dengan Qucs

Metode Quine
McKluskey dan
Program Bantu
Komputer

@2017, Eko Didik
Widianto

- ▶ Skematik rangkaian $f_{min} = \bar{X}_2\bar{X}_4 + X_2X_4 + X_3\bar{X}_4 + X_1\bar{X}_4$
- ▶ Simulasi digital menggunakan tabel kebenaran



Metode
Quine-McKluskey

Program Bantu
Komputer

Bmin: Visualisasi
Penyederhanaan Boolean
Logic Simplifier

Qucs: Simulator Rangkaian

Ringkasan

Lisensi

Hasil Simulasi dengan Qucs

Metode Quine
McKluskey dan
Program Bantu
Komputer

@2017,Eko Didik
Widianto

Metode
Quine-McKluskey

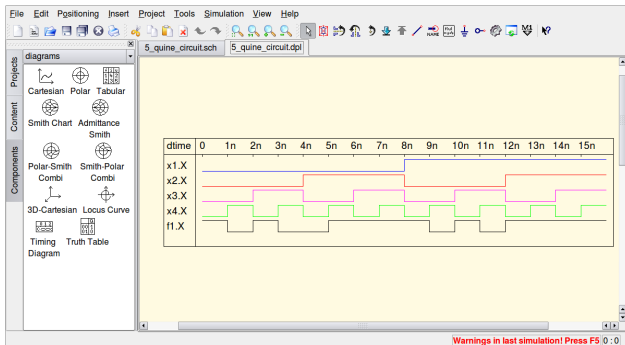
Program Bantu
Komputer

Bmin: Visualisasi
Penyederhanaan Boolean
Qmls: Quine-McKluskey
Logic Simplifier

Qucs: Simulator Rangkaian

Ringkasan

Lisensi



► Verifikasi rangkaian logika fungsi

$$f(x_1, x_2, x_3, x_4) = \sum m(2, 3, 7, 9, 11, 13) + \sum d(0, 10, 15)$$

Metode
Quine-McKluskey

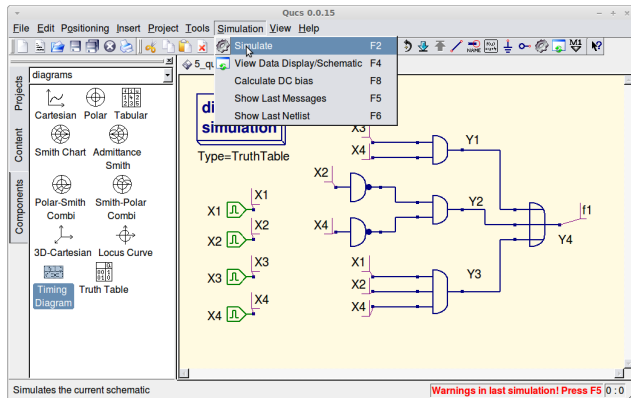
Program Bantu
Komputer

Bmin: Visualisasi
Penyederhanaan Boolean
Omls: Quine-McKluskey
Logic Simplifier

Qucs: Simulator Rangkaian

Ringkasan

Lisensi



Contoh: Hasil Simulasi

Metode Quine
McKluskey dan
Program Bantu
Komputer

@2017,Eko Didik
Widianto

Metode
Quine-McKluskey

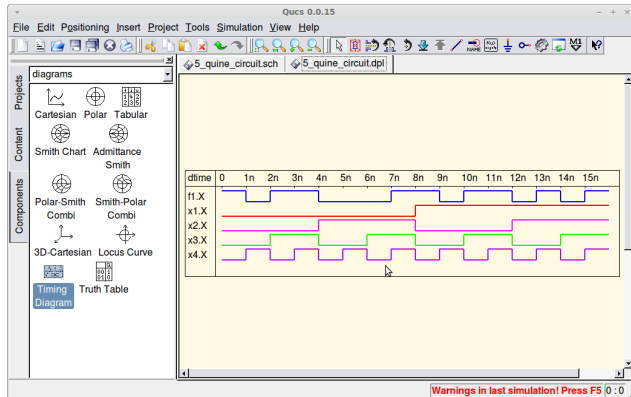
Program Bantu
Komputer

Bmin: Visualisasi
Penyederhanaan Boolean
Omls: Quine-McKluskey
Logic Simplifier

Qucs: Simulator Rangkaian

Ringkasan

Lisensi



- ▶ Yang telah kita pelajari hari ini:
 - ▶ Penyederhanaan fungsi logika menggunakan metode tabular Quine-McKluskey untuk aplikasi komputer
 - ▶ Program bantu komputer untuk melakukan sintesis rangkaian logika minimum dan analisis rangkaian, yaitu Bmin, Qmls dan Qucs
- ▶ Latihan:
 - ▶ Lihat Tugas#3
- ▶ Yang akan kita pelajari di pertemuan berikutnya adalah teknologi CMOS dan chip terintegrasi TTL (*transistor-transistor logic*) untuk mengimplementasikan gerbang logika
 - ▶ 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/>