

Tugas Besar Minimisasi Logic
EL2008 - Pemecahan Masalah dengan C



Kelompok 9

Muhammad Daris Nurhakim (13220047)
Raihan Fadhil Yanuarsyah (13220049)
Aloysius Efrata Sumaryo (13220057)

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung

2022

Daftar isi

Metode dan Latar Belakang	2
Spesifikasi Software	2
Flowchart	2
DFD	3
Rencana Pengujian	3
Hasil Pengujian	4
Analisis	7
Kesimpulan	8
Pembagian Tugas	8
Referensi	8

A. Metode dan Latar Belakang

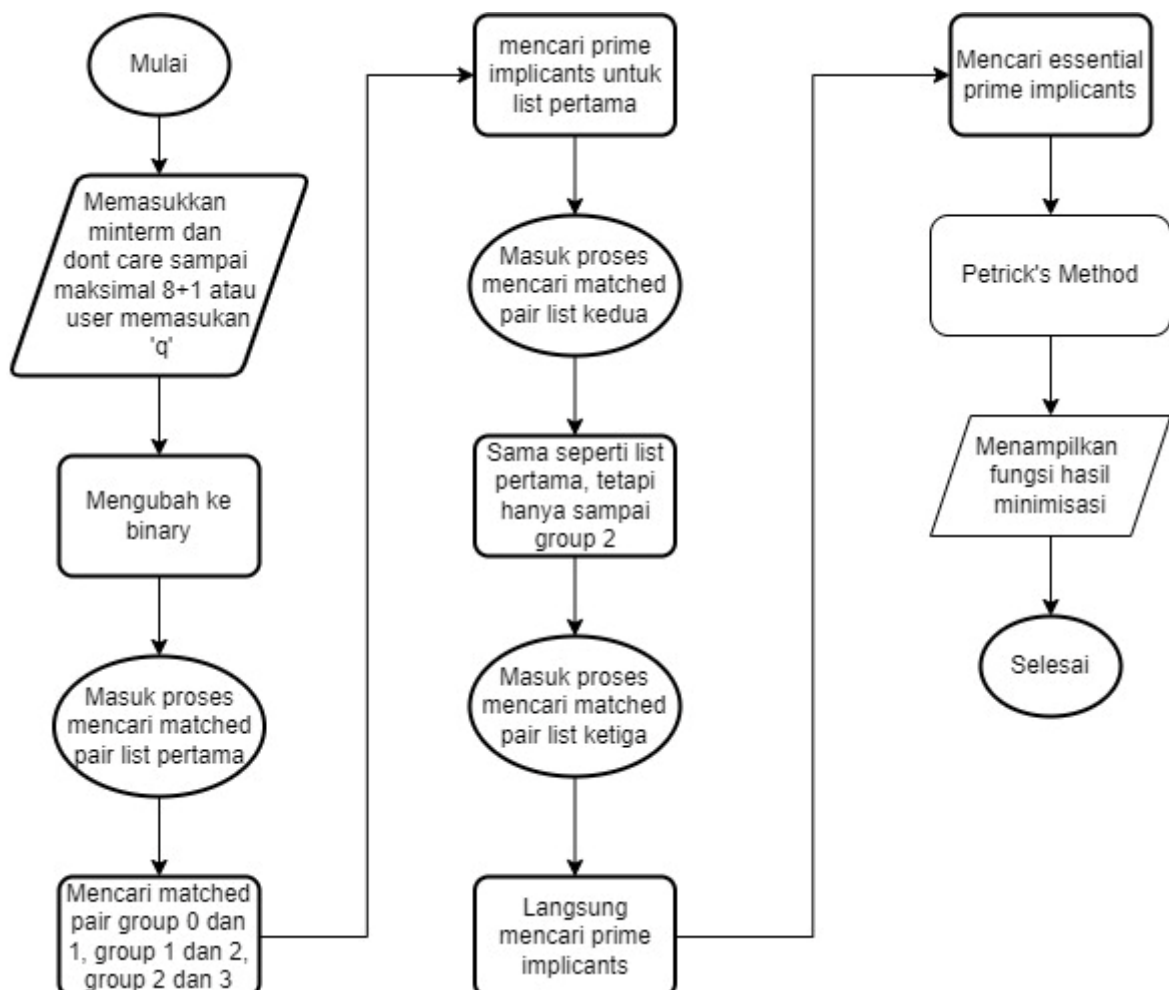
Fungsi logic yang kompleks dan rumit dapat diubah bentuknya menjadi lebih singkat. Hal itu disebut dengan logic minimization. Dengan mengubah fungsi logic menjadi lebih singkat, kita dapat mudah memahami fungsi tersebut, selain itu implementasi rangkaian logikanya akan lebih hemat biaya karena menggunakan gerbang logika yang lebih sedikit.

Salah satu metode dalam melakukan logic minimization adalah metode Quine McCluskey. Metode Quine McCluskey atau lebih sering dikenal dengan metode tabular pada dasarnya adalah mencari prime implicant dengan cara mencari mengelompokkan matched pairs, yaitu dua minterm dengan perbedaan diantaranya hanya 1 bit. Setelah mencari prime implicant dilakukan pencarian essential prime implicant. Disebut metode tabular karena pengerjaannya menggunakan tabel.

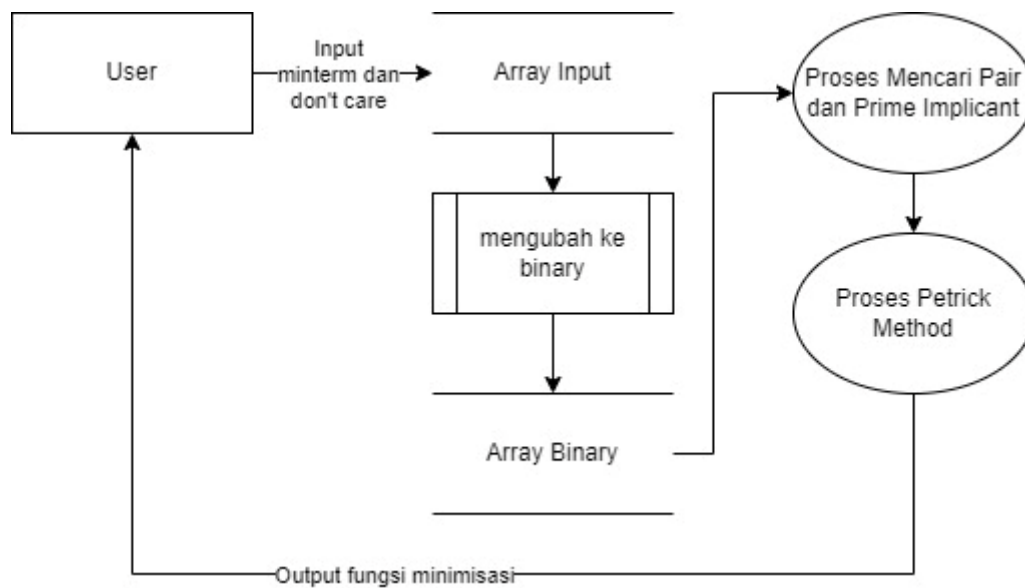
B. Spesifikasi Software

Program akan menerima input berupa minterm dan don't care dengan maksimal masukan sebanyak 9 masukan. Minterm yang dimasukkan akan diubah menjadi binary terlebih dahulu. Kemudian, akan dilakukan metode tabular yaitu mencari pair. Keluaran program berupa fungsi yang sudah ter-minimisasi.

C. Flowchart



D. DFD



E. Rencana Pengujian

1. Input random tanpa don't care

```

Masukkan minterm pertama (q untuk selesai): 1
Masukkan minterm berikutnya (q untuk selesai): 3
Masukkan minterm berikutnya (q untuk selesai): 6
Masukkan minterm berikutnya (q untuk selesai): q
  
```

```

Masukkan don't care term pertama (q untuk selesai): q
  
```

$$F = AB(\sim C) + (\sim A)C$$

2. Input random dengan don't care

```

Masukkan minterm pertama (q untuk selesai): 1
Masukkan minterm berikutnya (q untuk selesai): 2
Masukkan minterm berikutnya (q untuk selesai): 7
Masukkan minterm berikutnya (q untuk selesai): q
  
```

```

Masukkan don't care term pertama (q untuk selesai): 3
Masukkan don't care term berikutnya (q untuk selesai): 5
Masukkan don't care term berikutnya (q untuk selesai): q
  
```

$$F = (\sim A)B + C$$

F. Hasil Pengujian

1. Input random tanpa don't care

$$F = AB(\sim C) + (\sim A)C$$

$F(A, B, C) = \sum m(1, 3, 6)$

1

NEW FUNCTION

2. Finding Prime Implicants

Group	Decimal	Binary
0	1	001 ✓
1	3	011 ✓
1	6	110 ✖

3. Finding Prime Implicants

Group	Decimal	Binary
0	1, 3	0-1 ✖

The Prime Implicants Are $ABC', A'C$

4

5. Prime Implicants Chart

Minterms	1	3	6	Cost
ABC'			X	5
$A'C$	X	X		4

6. Finding Unique Minterms

Minterms	1	3	6	Cost
ABC'			X	5
$A'C$	X	X		4

7 The Essential Implicants Are ABC' , $A'C$

8 Possible Function Minimizations
 $F = ABC' + A'C$

Terbukti bahwa hasil pada program sudah benar dengan dibuktikan melalui perhitungan step by step.

2. Input random dengan don't care

$$F = (\sim A)B + C$$

1 $F(A, B, C) = \sum m(1, 2, 7) + \sum d(3, 5)$

NEW FUNCTION

2. Finding Prime Implicants		
Group	Decimal	Binary
0	1	001 ✓
0	2	010 ✓
1	3	011 ✓
1	5	101 ✓
2	7	111 ✓

3. Finding Prime Implicants		
Group	Decimal	Binary
0	1, 5	-01 ✓
0	1, 3	0-1 ✓
0	2, 3	01- ✖
1	3, 7	-11 ✓
1	5, 7	1-1 ✓

4. Finding Prime Implicants		
Group	Decimal	Binary
0	1, 3, 5, 7	--1 ✖

5

The Prime Implicants Are
A'B, C

6. Prime Implicants Chart				
Minterms	1	2	7	Cost
A'B		X		4
C	X		X	2

7. Finding Unique Minterms				
Minterms	1	2	7	Cost
A'B		X		4
C	X		X	2

8

The Essential Implicants Are

A'B, C

9

Possible Function Minimizations

$F = A'B + C$

Terbukti bahwa hasil pada program sudah benar dengan dibuktikan melalui perhitungan step by step.

G. Analisis

1. Input random tanpa don't care

Mengurutkan minterm secara menaik sesuai nilainya dan nilai grup sesuai dengan jumlah suku yang ada dalam representasi binernya. Bandingkan minterm yang ada pada grup yang berurutan. Jika hanya terdapat perbedaan 1 bit, maka ambil kedua minterm tersebut dan ubah dengan simbol pada bagian bit yang berbeda. Ulangi langkah sebelumnya dengan suku-suku yang baru hingga kita mendapatkan semua nilai prime implicants. Buat tabel prime implicant. Cari essential prime implicant dengan cara mengamati tabel prime implicant, jika minterm hanya dicakup oleh satu prime implicant maka itu dapat disebut sebagai essential prime implicant. Kurangi tabel dengan menghapus setiap baris yang sesuai dengan essential prime implicant. Ulangi

langkah tersebut untuk mengurangi tabel prime implicant. Proses akan berhenti jika semua persyaratan minterm dari fungsi Boolean sudah terpenuhi.

2. Input random dengan don't care

Perbedaan terdapat don't care dan tidak ada adalah don't care akan mempengaruhi solusi akhir dari minterm. Hal ini terjadi karena saat pembentukan tabel prime implicant, angka yang termasuk dalam don't care tidak menjadi bagian dari solusi akhir.

H. Kesimpulan

Setelah dilakukan perancangan program, maka dapat disimpulkan bahwa:

1. Terdapat beberapa algoritma penyelesaian logic minimization seperti metode Tabular dan K-Map.
2. Tabular memiliki kompleksitas yang lebih rendah dibanding K-Map.
3. Tabular lebih efisien digunakan dalam algoritma komputer.

I. Pembagian Tugas

Tugas besar ini dikerjakan secara bersama-sama dimulai dari flowchart, DFD, code, laporan, dan file presentasi.

J. Referensi

1. https://www.tutorialspoint.com/digital_circuits/digital_circuits_quine_mccluskey_tabular_method.htm
2. https://www.lions.odu.edu/~rljones/310/homework/Chap3/EET310_Ch3_G_QM_Dont_Care.pdf
3. <https://www.codeproject.com/Articles/37031/Karnaugh-Map-Minimizer-Three-Variables>
4. <https://geekyboy.github.io/Quine-McCluskey-Solver/>