

Penerapan Aljabar Boolean Untuk Peningkatan Efisiensi Penggunaan Listrik

Diyah Susan Nugrahani 13523080^{1,2}

Program Studi Teknik Informatika

Sekolah Teknik Elektro dan Informatika

Institut Teknologi Bandung, Jl. Ganesha 10 Bandung 40132, Indonesia

13523080@mahasiswa.itb.ac.id, diyahnugrahani04@gmail.com

Abstrak—Beberapa alat elektronik menggunakan sirkuit digital dalam sistemnya. Pemrosesan informasi dalam sirkuit digital dilakukan dalam bentuk biner dengan penerapan prinsip aljabar boolean. Sirkuit digital merepresentasikan operasi logika melalui ekspresi boolean. Penyederhanaan ekspresi boolean dapat dilakukan dengan metode peta Karnaugh. Penyederhanaan ekspresi boolean dapat berpengaruh pada peningkatan efisiensi karena berkurangnya kompleksitas pada sirkuit. Hal tersebut berdampak pada pengoptimalan penggunaan komponen dan pengurangan konsumsi listrik.

Kata kunci— Aljabar Boolean, Peningkatan Efisiensi, Peta Karnaugh, Sirkuit Digital

Abstract— Some electronic devices use digital circuits in their systems. Information processing in digital circuits is carried out in binary form with the application of boolean algebraic principles. Digital circuits represent logical operations through boolean expressions. Simplification of boolean expressions can be done with the Karnaugh map method. Simplification of boolean expressions can have an effect on increasing efficiency due to reduced complexity in circuits. This has an impact on optimizing the use of components and reducing electricity consumption.

Keywords— Boolean Algebra, Digital Circuits, Efficiency Improvement, Karnaugh Map

I. PENDAHULUAN

Listrik merupakan sumber energi yang esensial dan dibutuhkan seluruh manusia. Hampir semua barang yang ada di sekitar kita membutuhkan energi listrik untuk bekerja. Rumah juga membutuhkan listrik untuk dapat beroperasi dengan baik. Air yang digunakan untuk aktivitas rumah tangga, lampu untuk penerangan ruang, dan seluruh alat elektronik di rumah membutuhkan listrik. Penggunaan yang semakin meningkat membuat pengeluaran dana untuk biaya listrik menjadi meningkat juga. Selain masalah biaya, penggunaan listrik yang berlebihan merupakan sebuah tindakan pemborosan energi.

Pemborosan listrik memiliki berbagai dampak buruk bagi lingkungan, ekonomi, dan sosial. Sebagian besar listrik bersumber dari pembangkit listrik berbasis fosil sehingga penggunaan yang berlebihan akan

mengakibatkan emisi karbon yang tinggi dan mempeburuk pemanasan global. Dari segi ekonomi, penggunaan listrik berlebihan akan membuat tagihan listrik melonjak. Selain itu, sistem pembangkit, distribusi, dan transmisi listrik dirancang untuk bekerja pada kapasitas tertentu. Pemborosan listrik dapat membuat sistem bekerja lebih keras sehingga berakibat terhadap penurunan efisiensi sistem.

Dalam mengatasi persoalan-persoalan yang berkaitan dengan penghematan penggunaan listrik, salah hal yang dapat dilakukan adalah dengan menggunakan listrik secara efisien. Penggunaan listrik secara efisien dapat dilakukan dengan cara menggunakan alat-alat elektronik yang menerapkan efisiensi sistem di dalamnya. Efisiensi tersebut dapat dicapai dengan melakukan efisiensi sirkuit. Pengelolaan dan pemilihan sirkuit menentukan optimalitas energi listrik yang digunakan.

Aljabar boolean dapat digunakan untuk melakukan optimalisasi penggunaan listrik pada sirkuit. Aljabar boolean digunakan untuk menyederhanakan desain logika dalam sirkuit digital. Penyederhanaan ini dapat berakibat pada pengurangan jumlah gerbang logika yang digunakan. Hal tersebut berdampak pada penurunan konsumsi daya, waktu pemrosesan yang lebih cepat, dan pengurangan biaya pembuatan.

Dalam makalah ini akan dianalisis terkait hubungan penerapan aljabar boolean dengan efisiensi sirkuit dalam penggunaan listrik. Selain itu, pada makalah ini juga dilakukan perbandingan biaya penghematan dana listrik pada keadaan sebelum dan setelah menerapkan pengefisienan sirkuit. Untuk akurasi yang lebih tinggi, penulis membuat sebuah program untuk mengimplementasikan permasalahan tersebut.

II. LANDASAN TEORI

2.1 Aljabar Boolean

Aljabar boolean adalah sebuah cabang matematika yang membahas tentang operasi logika pada nilai-nilai biner yang direpresentasikan sebagai 1 (*true*) dan 0 (*false*). Konsep ini pertama kali diperkenalkan oleh George Boole pada tahun 1854 dalam bukunya *An Investigation of the Laws of Thought*. Konsep aljabar boolean digunakan dalam

berbagai bidang, salah satunya menjadi dasar logika digital dan desain sirkuit elektronik.

A. Operasi-operasi dasar aljabar boolean

1. Operasi (+) atau OR
 - i. $1 + 1 = 0$
 - ii. $1 + 0 = 1$
 - iii. $0 + 0 = 0$
2. Opetasi (.) atau AND
 - i. $1 . 1 = 1$
 - ii. $1 . 0 = 0$
 - iii. $0 . 0 = 0$

B. Hukum-hukum dasar aljabar boolean.

1. Hukum Identitas
 - i. $a + 0 = a$
 - ii. $a . 0 = 0$
2. Hukum Idempoten
 - i. $a + a = a$
 - ii. $a . a = a$
3. Hukum Komplemen
 - i. $a + a' = 1$
 - ii. $a . a' = 0$
4. Hukum Involusi
 - i. $(a')' = a$
5. Hukum Dominasi
 - i. $a . 0 = 0$
 - ii. $a + 1 = 1$
6. Hukum Penyerapan
 - i. $a + ab = a$
7. Hukum Komutatif
 - i. $a (a + b) = a$
8. Hukum Asosiatif
 - i. $a + (b + c) = (a + b) + c$
 - ii. $a . (b . c) = (a . b) . c$
9. Hukum Distributif
 - i. $a . (b + c) = a . b + a . c$
 - ii. $a + (b . c) = (a + b) . (a + c)$
10. Hukum De Morgan
 - i. $(a + b)' = a'b'$
 - ii. $(ab)' = a' + b'$
11. Hukum 0/1
 - i. $0' = 1$
 - ii. $1' = 0$

2.2 Peta Karnaugh

Peta Karnaugh atau *K-map* adalah metode yang digunakan untuk menyederhanakan fungsi boolean. Metode ini ditemukan oleh Maurice Karnaugh pada tahun 1953. Peta Karnaugh terdiri dari kotak-kotak yang merepresentasikan sebuah *minterm*. *Minterm* adalah suku dalam ekspresi boolean yang mengandung literal lengkap dalam bentuk hasil kali. Kotak yang merupakan *minterm* diisi dengan "1" sedangkan sisanya diisi dengan "0".

		yz			
		00	01	11	10
x	0	$x'y'z'$	$x'y'z$	$x'yz$	$x'yz'$
	1	$xy'z'$	$xy'z$	xyz	xyz'

		yz			
		00	01	11	10
x	0	0	0	1	1
	1	1	0	1	1

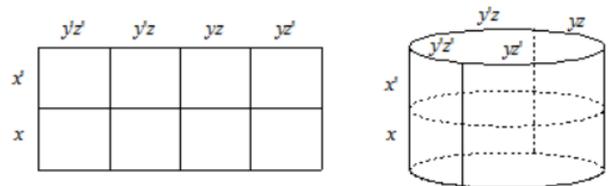
$xz' + y$

Gambar 1 Peta Karnaugh

Penggunaan Peta Karnaugh untuk penyederhanaan fungsi boolean dilakukan dengan cara menggabungkan kotak-kotak yang bernilai 1 dan saling beririsan. Kotak-kotak tersebut dapat membentuk pasangan (dua), kuad (empat), atau oktet (delapan). Selain itu, Peta Karnaugh juga dapat disederhanakan dengan penggulungan. Caranya dengan mengaitkan bagian sisi kiri dan sisi kanan kemudian digulung.

yz		00	01	11	10
wx	00	0	0	0	0
	01	0	0	0	0
	11	1	1	0	0
	10	1	1	0	0

Gambar 2 Contoh Pengisian Peta Karnaugh



Gambar 3 Penggulungan Peta Karnaugh

(Sumber: [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-\(2024\)-bagian2.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-(2024)-bagian2.pdf))

Hasil penyederhanaan dengan Peta Karnaugh tidak selalu menghasilkan persamaan boolean yang unik. Terdapat kemungkinan suatu persamaan boolean memiliki banyak variasi bentuk Peta Karnaugh. Meskipun demikian, jumlah literal dan jumlah suku untuk semua pola adalah sama.

2.3 Sirkuit Digital

Sirkuit digital adalah rangkaian elektronik yang digunakan dala digunakan untuk pemrosesan sinyal biner (0 dan 1), berbeda dengan sirkuit analog yang digunakan untuk dalam pemrosesan sinyal kontinu. Sirkuit digital merupak bentuk implementasi dari operasi logika sehingga hukum-hukum aljabar boolean juga berlaku pada penerapan sirkuit digital.

Komponen dasar sirkuit digital adalah inverter. Jenis inverter berbasis OECT pertama yang ditemukan adalah inverter berbasis ladder. Berdasarkan bias pada OECT,

resistive ladder digunakan untuk mengubah inverter ke kondisi *on* atau *off*. Seiring perkembangan zaman, gerbang logika yang lebih kompleks seperti NAND dan NOR telah dikembangkan dengan memanfaatkan inverter berbasis resistive ladder. Terdapat beberapa jenis gerbang logika yang menjadi dasar dari penggunaan gerbang logika pada sirkuit digital, antara lain :

1. Gerbang AND



Gambar 4 Gerbang Logika AND

(Sumber: [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-\(2024\)-bagian1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-(2024)-bagian1.pdf))

Input		Output
x	y	
0	0	0
0	1	0
1	0	0
1	1	1

2. Gerbang OR

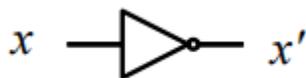


Gambar 5 Gerbang Logika OR

(Sumber: [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-\(2024\)-bagian1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-(2024)-bagian1.pdf))

Input		Output
x	y	
0	0	0
0	1	1
1	0	1
1	1	1

3. Gerbang NOT



Gambar 6 Gerbang Logika NOT

(Sumber: [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-\(2024\)-bagian1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-(2024)-bagian1.pdf))

Input	Output
x	
0	1

1	0
---	---

4. Gerbang XOR

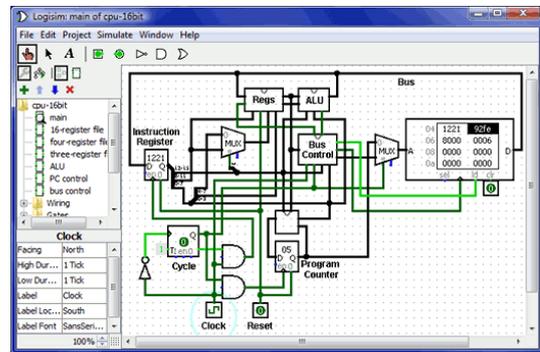


Gambar 7 Gerbang Logika XOR

(Sumber: [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-\(2024\)-bagian1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-(2024)-bagian1.pdf))

Input		Output
x	y	
0	0	0
0	1	1
1	0	1
1	1	0

Gerbang logika tersebut dapat disusun sedemikian rupa untuk membentuk sirkuit digital. Penggabungan dari beberapa jenis gerbang digital akan menghasilkan sirkuit digital yang kompleks. Beberapa dari sirkuit kompleks tersebut dapat disimplifikasi agar menghasilkan sirkuit yang lebih efisien.



Gambar 8 Sirkuit Digital

(Sumber: <https://sourceforge.net/projects/circuit/>)

III. ANALISIS DAN PEMBAHASAN

Setiap alat elektronik memiliki siste, yang telah didesain sedemikian rupa agar dapat menghasilkan performa terbaik. Salah satu aspek penting dalam pengoptimalan performa berkaitan dengan efisiensi penggunaan komponen di dalam sistem tersebut. Dalam konteks sirkuit digital, efisiensi dapat dicapai dengan melakukan pengurangan *gate* dan *literal* yang ada pada suatu ekspresi logika. *Gate* merupakan istilah lain untuk penggunaan gerbang logika. *Literal* merupakan banyaknya variable yang muncul dalam ekspresi logika tersebut. Semakin sedikit *gate* dan *literal* yang digunakan, maka akan semakin efisien suatu sirkuit logika.

Salah satu cara untuk mengurangi *gate* dan *literal* adalah dengan menyederhanakan bentuk ekspresi logika. Caranya dapat dilakukan dengan pemanfaatan hukum-hukum

aljabar boolean. Ekspresi yang memiliki pola berulang atau terlalu boros akan diganti ke bentuk yang lebih sederhana.

Selain menggunakan aljabar boolean, penyederhanaan ekspresi boolean juga dapat dilakukan dengan menggunakan peta Karnaugh. Peta Karnaugh akan menyederhanakan bentuk ekspresi boolean dengan cara mengelompokkan kotak-kotak bernilai 1 yang saling beririsan. Tingkat prioritas pengelompokkan dimulai dari yang tertinggi yaitu oktet dan diakhiri dengan pasangan.

Setelah dilakukan penyederhanaan bentuk ekspresi boolean akan terbentuk fungsi minimasi yang memiliki *gate* dan *literal* lebih sedikit dibandingkan sebelumnya. Jumlah *gate* dan *literal* tersebut dapat dibandingkan untuk menganalisis seberapa berpengaruh proses penyederhanaan tersebut terhadap *efficiency reduction*.

$$EI(\%) = \left(\frac{\text{Original count} - \text{Simplified count}}{\text{Original count}} \right)$$

ER (Efficiency Improvement), menunjukkan seberapa besar peningkatan efisiensi setelah dilakukan penyederhanaan sirkuit digital.

$$GR(\%) = \left(\frac{\text{Original gate} - \text{Simplified gate}}{\text{Original gate}} \right)$$

GR (Gate Reduction), menunjukkan seberapa besar pengurangan *gate* yang dapat dilakukan dengan penyederhanaan sirkuit digital.

$$LR(\%) = \left(\frac{\text{Original literal} - \text{Simplified literal}}{\text{Original literal}} \right)$$

LR (Literal Reduction), menunjukkan seberapa besar pengurangan *literal* yang dapat dilakukan dengan penyederhanaan sirkuit digital.

Untuk memberikan gambaran terkait performa yang dihasilkan dengan pemanfaatan konsep ini penulis akan menjelaskan salah satu contoh kasus terkait efisiensi sirkuit.

```

Enter a Boolean expression: A & B | A & ~B
Simplified Expression:
A
  
```

Truth Table		
A	B	Output
0	0	0
0	1	0
1	0	1
1	1	1

```

Karnaugh Map:
  0  1
  ---
  1 | 1 | 1 |
  ---
  
```

Literal Reduction: 50.00%
Gate Reduction: 100.00%

Gambar 9 Contoh Output Kasus 1 (sumber : dokumentasi pribadi penulis)

Berdasarkan hasil penyederhanaan ekspresi boolean pada gambar 9, diperoleh data *gate* awal berjumlah 4 dan *gate* akhir dikurangi hingga tersisa 0. Dengan

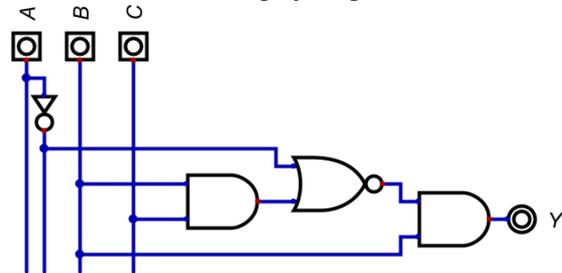
menggunakan rumus Rumus 2, diperoleh persentase *gate reduction* sebesar 100%.

$$GR(\%) = \left(\frac{4 - 0}{4} \right) = 100\%$$

Untuk *literal*, pada ekspresi awal terdapat dua *literal*, sedangkan setelah penyederhanaan menjadi tersisa 1. Dengan menggunakan rumus Rumus 3, diperoleh *literal reduction* sebesar 50%. Dengan menggabungkan kedua nilai tersebut, maka diperoleh rata-rata *efficiency improvement* sebesar 75%.

$$EI(\%) = \left(\frac{2 - 1}{2} \right) = 50\%$$

Penerapan aljabar boolean dalam melakukan penyederhanaan sirkuit untuk meningkatkan efisiensi penggunaan listrik terbukti menghasilkan nilai yang cukup signifikan. Pemanfaatan konsep ini dapat diperluas lagi untuk bidang-bidang lainnya. Salah satunya adalah untuk pembangunan gedung. Representasikan *input* A, B, dan C sebagai sumber listrik yang dapat diatur nyala arusnya. Y merepresentasikan ruangan di sebuah gedung yang membutuhkan pasokan listrik. Meskipun memiliki tiga sumber daya, namun sebenarnya untuk mengalirkan listrik hingga sampai ke Y tidak harus menggunakan tiga sumber. Penggunaan aljabar boolean untuk menyederhanakan sirkuit tersebut akan memperjelas perbedaan efisiensi.



Gambar 10 Sirkuit Kasus 1 (sumber : dokumentasi pribadi penulis)

Sirkuit tersebut apabila diubah ke bentuk ekspresi logika adalah $B \& (\sim(\sim A \mid (B \& C)))$. Masukkan inputan tersebut ke program yang telah dibuat sehingga diperoleh :

```

Enter a Boolean expression: B & (~(~A | (B & C)))
Simplified Expression:
A & B & ~C
  
```

Truth Table			
B	A	C	Output
0	0	0	0
0	0	1	0
0	1	0	0
0	1	1	0
1	0	0	0
1	0	1	0
1	1	0	1
1	1	1	0

```

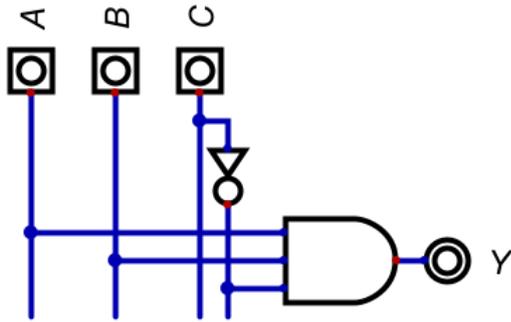
Karnaugh Map:
  00 01 11 10
  ---
  1 | 0 | 0 | 1 | 0 |
  ---
  
```

Literal Reduction: 0.00%
Gate Reduction: 40.00%

Gambar 11 Contoh Output Kasus 2 (sumber : dokumentasi pribadi penulis)

Berdasarkan hasil yang diperoleh, dapat dibuat desain sirkuit yang lebih efisien dengan menggunakan ekspresi yang telah disederhanakan menjadi $A \& B \& \sim C$. Penerapan konsep ini dapat menghasilkan *gate reduction*

sebesar 40% atau setara dengan meningkatkan efisiensi rata-rata sebesar 20%. Hal tersebut dapat menghemat penggunaan daya listrik pada suatu gedung sekaligus menurunkan anggaran untuk biaya listrik.



Gambar 12 Sirkuit Kasus 2 (sumber : dokumentasi pribadi penulis)

IV. IMPLEMENTASI PROGRAM

Program diimplementasikan dengan menggunakan bahasa python. Setelah *input* dimasukkan, *input* akan disimplifikasi bentuk ekspresi booleannya menggunakan hukum-hukum aljabar boolean. Pada implementasi ini digunakan *library* sympy untuk penyederhanaan. Contoh keluaran hasil simplifikasi terdapat pada gambar 11.

```
def simplify_booleam(expression):
    try:
        parsed_expr = sympy.sympify(expression, evaluate=False)
        simplified_expr = sympy.simplify_logic(parsed_expr)
        return simplified_expr
    except Exception as e:
        return f"Error: {e}"
```

Gambar 13 Penyederhanaan Ekspresi Boolean (sumber : dokumentasi pribadi penulis)

Selain dengan metode aljabar boolean secara langsung, penyederhanaan ekspresi boolean dapat dilakukan dengan pembuatan peta Karnaugh. Dalam pembuatan peta Karnaugh diperlukan table kebenaran. Table kebenaran tersebut menggambarkan seluruh kombinasi nilai kebenaran dari proposisi yang terlibat. Contoh keluaran dari tabel kebenaran terdapat pada gambar 11.

```
def generate_truth_table(expression):
    try:
        parsed_expr = sympy.sympify(expression, evaluate=False)
        variables = list(parsed_expr.free_symbols)
        rows = [dict(zip(variables, values)) for values in np.ndindex(2, * len(variables))]

        #tabel kebenaran
        print("\nTruth Table")
        print("-" * 40)
        print("| ".join([str(var) for var in variables] + ["Output"]))
        print("-" * 40)
        for row in rows:
            result = int(bool(parsed_expr.subs(row)))
            print("| ".join([str(int(row.get(var, 0))) for var in variables] + [str(result)]))

        return rows, variables, parsed_expr
    except Exception as e:
        print(f"Error: {e}")
```

Gambar 14 Program Tabel Kebenaran (sumber : dokumentasi pribadi penulis)

Hasil data dari tabel kebenaran akan diolah untuk pembuatan peta Karnaugh. Hasil *output* dari tabel kebenaran akan ditempatkan pada grid di peta Karnaugh yang bersesuaian. Dalam implementasi ini, jumlah

variabel dibatasi sebanyak empat variabel. Contoh keluaran dari peta Karnaugh terdapat pada gambar 11. Dengan menggunakan informasi dari peta Karnaugh, dapat diperoleh bentuk yang lebih sederhana dari ekspresi boolean yang dimasukkan. Angka 1 pada peta Karnaugh menandakan keberadaan variabel yang terkait.

```
def generate_karnaugh_map_terminal(rows, variables, parsed_expr):
    try:
        if len(variables) not in [2, 3, 4]: #versi yang terbatas untuk maksimal 4 variabel
            print("Karnaugh maps are only supported for 2, 3, or 4 variables.")
            return

        # membuat grid
        if len(variables) == 2:
            grid = np.zeros((2, 2), dtype=int)
            row_labels = ["0", "1"]
            col_labels = ["0", "1"]
        elif len(variables) == 3:
            grid = np.zeros((2, 4), dtype=int)
            row_labels = ["0", "1"]
            col_labels = ["00", "01", "11", "10"]
        elif len(variables) == 4:
            grid = np.zeros((4, 4), dtype=int)
            row_labels = ["00", "01", "11", "10"]
            col_labels = ["00", "01", "11", "10"]

        for row in rows:
            inputs = [int(row[var]) for var in variables]
            result = int(bool(parsed_expr.subs(row)))

            if len(variables) == 2:
                grid[inputs[0], inputs[1]] = result
            elif len(variables) == 3:
                grid[inputs[0], 2 * inputs[1] + inputs[2]] = result
            elif len(variables) == 4:
                grid[2 * inputs[0] + inputs[1], 2 * inputs[2] + inputs[3]] = result

        # peta karnaugh
        print("\nKarnaugh Map:")
        print(" " + " ".join(col_labels))
        print(" " + "-" * (len(col_labels) * 4 + 1))
        for i, row_label in enumerate(row_labels):
            row_output = " | ".join(str(grid[i, j]) for j in range(grid.shape[1]))
            print(f"{row_label} | {row_output} |")
            print(" " + "-" * (len(col_labels) * 4 + 1))

    except Exception as e:
        print(f"Error generating Karnaugh Map: {e}")
```

Gambar 15 Program Peta Karnaugh (sumber : dokumentasi pribadi penulis)

Penyederhanaan ekspresi boolean memberikan dampak terhadap efisiensi performa sirkuit listrik. Hal tersebut terjadi karena gerbang logika dan *literal* yang telah disederhanakan berkurang sehingga konsumsi listriknya menjadi berkurang juga. Konsep ini juga dapat diterapkan dalam pembuatan sistem atau alat elektronik untuk meningkatkan performa dan lebih hemat daya. Contoh dari perhitungan efisiensi terdapat pada gambar 11.

```
def calculate_efficiency(original_expr, simplified_expr):
    try:
        # literals
        original_literals = sum(1 for _ in sympy.sympify(original_expr).atoms(sympy.Symbol))
        simplified_literals = sum(1 for _ in sympy.sympify(simplified_expr).atoms(sympy.Symbol))

        # gates
        original_gates = count_gates(original_expr)
        simplified_gates = count_gates(simplified_expr)

        # persentase
        literal_reduction = ((original_literals - simplified_literals) / original_literals) * 100 if original_literals > 0 else 0
        gate_reduction = ((original_gates - simplified_gates) / original_gates) * 100 if original_gates > 0 else 0

        return literal_reduction, gate_reduction, original_literals, simplified_literals, original_gates, simplified_gates
    except Exception as e:
        return f"Error calculating efficiency: {e}", 0, 0, 0, 0, 0
```

Gambar 16 Program Menghitung Efisiensi (sumber : dokumentasi pribadi penulis)

V. KESIMPULAN

Beberapa alat elektronik memanfaatkan sirkuit digital dalam pembuatannya. Sirkuit digital adalah sistem elektronik yang memproses informasi dalam bentuk biner. Desain sirkuit tersebut didasarkan pada prinsip-prinsip aljabar boolean. Aljabar boolean dalam hal ini merepresentasikan operasi logika yang dilakukan oleh

gerbang logika dalam bentuk ekspresi boolean.

Penyederhanaan ekspresi boolean akan menghasilkan sirkuit yang lebih efisien. Penyederhanaan tersebut dapat dilakukan salah satunya dengan pemanfaatan peta Karnaugh. Cara kerjanya adalah dengan mengidentifikasi pola pada tabel kebenaran yang kemudian digabungkan dan menghasilkan ekspresi boolean yang lebih sederhana. Ekspresi boolean yang lebih sederhana akan berpengaruh terhadap efisiensi dan peningkatan performa pada sistem. Peningkatan efisiensi bergantung kepada kompleksitas ekspresi boolean yang dipakai. Konsep ini dapat dimanfaatkan juga pada desain jalur listrik pada suatu gedung dengan mencari kombinasi paling optimal dalam pemakaian komponen sehingga dapat mengurangi penggunaan listrik.

VI. SARAN

Dalam pembuatan makalah ini, penulis menemukan beberapa hal yang dapat dijadikan pembelajaran untuk penulisan dikemudian hari dan untuk pembaca. Penulis menyarankan untuk lebih mengeksplor lagi penerapan berkaitan dengan aljabar boolean karena penulis melihat potensi pengembangan yang cukup besar dalam bidang ini. Dikarenakan *domain* dari persoalan aljabar boolean banyak berkaitan dengan bidang elektronika maka penulis menyarankan untuk melakukan riset lebih dalam terhadap bidang tersebut agar penulisan dapat dilakukan dengan lebih mudah dan hasilnya dapat maksimal. Berkaitan dengan implementasi, penulis menyarankan untuk implementasi diterapkan secara lebih menarik dan variatif.

VII. LAMPIRAN

Tautan *repository* :

<https://github.com/DiyahSusan/Implementasi-Aljabar-Boolean-Untuk-Peningkatan-Efisiensi-Sistem>

Tautan video : <https://shorturl.at/sChXZ>

VIII. UCAPAN TERIMA KASIH

Pertama-tama penulis ingin mengucapkan rasa syukur kepada Tuhan Yang Maha Esa sehingga penulis dapat menyelesaikan makalah ini dengan tepat waktu dan sesuai dengan yang diharapkan. Penulis juga ingin mengucapkan terima kasih kepada dosen matakuliah Matematika Diskrit karena telah memberikan banyak ilmu baru dan membuat penulis mengeksplor banyak hal baru di semester ini. Penulis ingin memberikan ucapan terima kasih untuk orang-orang yang karya tulisannya telah menginspirasi penulis dalam menyelesaikan makalah ini. Penulis juga ingin memberikan ucapan terima kasih kepada seluruh pembaca yang telah meluangkan waktu untuk membaca makalah ini. Penulis berharap makalah ini dapat bermanfaat bagi pembaca secara khusus dan masyarakat secara umum.

REFERENSI

- [1] Mano, M. Morris, "Digital Logic and Computer Design," *Pearson Education*, 2017.
- [2] Munir, Rinaldi (2024). Bahan kuliah IF1220 Aljabar Boolean Bagian 1. Institut Teknologi Bandung. [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-\(2024\)-bagian1.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-(2024)-bagian1.pdf)
- [3] Munir, Rinaldi (2024). Bahan kuliah IF1220 Aljabar Boolean Bagian 2. Institut Teknologi Bandung. [https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-\(2024\)-bagian2.pdf](https://informatika.stei.itb.ac.id/~rinaldi.munir/Matdis/2024-2025/13-Aljabar-Boolean-(2024)-bagian2.pdf)
- [4] Reem B. Rashid, Xudong Ji, Jonathan Rivnay (2021). Organic Electrochemical Transistors in Bioelectronic Circuits, Biosensors and Bioelectronics Volume 190. Diakses pada 8 Januari 2024 dari ScienceDirect <https://doi.org/10.1016/j.bios.2021.113461>
- [5] Roth, Charles H. Jr., "Fundamentals of Logic Design," *Cengage Learning*, 2013.
- [6] S. Merkli, A. Domahidi, J. Jerez and R. S. Smith, "Globally Optimal AC Power System Upgrade Planning under Operational Policy Constraints," 2018 European Control Conference (ECC), Limassol, Cyprus, 2018, pp. 131-136, doi: 10.23919/ECC.2018.8550334.

PERNYATAAN

Dengan ini saya menyatakan bahwa makalah yang saya tulis ini adalah tulisan saya sendiri, bukan saduran, atau terjemahan dari makalah orang lain, dan bukan plagiasi.

Bandung, 8 Januari 2025



Diyah Susan Nugrahani 13523080