

# MENGHITUNG SISA BAGI DARI BILANGAN BINER DENGAN BANYAK DIGIT TAK TERBATAS MENGGUNAKAN BAGAN FINITE STATE AUTOMATA (FSA)

RANGSANG PURNAMA<sup>[1]</sup>, RIFKI FAHRIL ZAINAL<sup>[2]</sup>

<sup>[1],[2]</sup>Teknik Informatika, Universitas Bhayangkara Surabaya

Jl A. Yani 114 Surabaya

e-mail: <sup>[1]</sup>rangsang.purnama@ubhara.ac.id, <sup>[2]</sup>rifki@ubhara.ac.id

## ABSTRACT

*Research on the use of Finite State Automata (FSA) charts to simplify the calculation process of finding the remainder between numerator A and denominator B, both in decimal form, has been conducted. However, the FSA charts produced from this research appear "complicated" because there are too many transition lines to move from one state to another. This research provides an alternative solution to describe the FSA chart of the same process can be made simpler. Here, number A must first be converted to a binary number, after which the remainder can be found with respect to number B. An FSA chart is created based on the value of number B. By exploring the previously created FSA chart, the remainder of A with respect to B can be found. Referring to the results of various trials that have been conducted, it can be concluded that this research produces very good results. The resulting FSA chart is very simple but can provide answers to find the remainder of the numerator A with respect to the denominator B accurately.*

**Keywords:** Finite State Automata (FSA), remainder, modulo

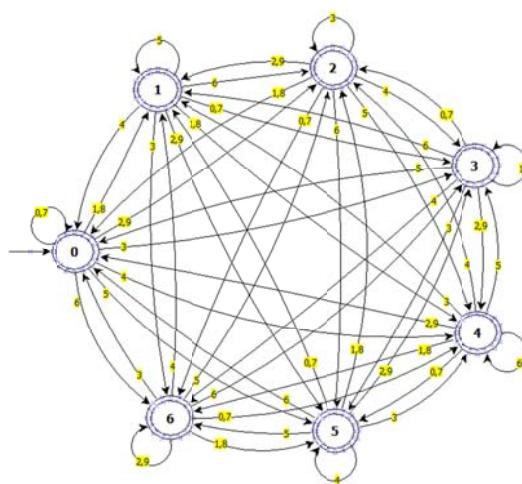
## ABSTRAK

*Penelitian mengenai penggunaan bagan Finite State Automata (FSA) untuk memudahkan proses perhitungan mencari sisa bagi antara pembilang A dengan penyebut B, keduanya dalam bentuk bilangan desimal, sudah pernah dilakukan, namun bagan FSA yang dihasilkan dari penelitian tersebut tampak "rumit" dikarenakan terlalu banyak garis transisi untuk berpindah dari satu state menuju ke state yang lain. Penelitian kali ini memberi solusi alternatif agar penggambaran bagan FSA dari proses yang sama bisa dibuat lebih sederhana. Di sini bilangan A harus diubah terlebih dahulu menjadi bilangan biner, setelah itu baru bisa dicari sisa baginya terhadap bilangan B. Sebuah bagan FSA dibuat berdasarkan nilai bilangan B. Melalui penelusuran bagan FSA yang sudah dibuat sebelumnya, bisa dicari hasil sisa bagi A terhadap B. Merujuk pada hasil berbagai ujicoba yang telah dilakukan, dapat disimpulkan bahwa penelitian ini memberikan hasil yang sangat baik. Bagan FSA yang dihasilkan sangat sederhana namun bisa memberi jawaban untuk pencarian sisa bagi dari pembilang A terhadap penyebut B secara akurat.*

**Kata kunci:** Finite State Automata (FSA), sisa bagi, modulo

## 1. PENDAHULUAN

Penelitian tentang penggunaan bagan Finite State Automata (FSA) untuk mencari sisa bagi dari dua bilangan dengan digit tak terbatas telah dilakukan di tahun 2024 [1]. Pada penelitian itu bilangan yang akan dicari sisa baginya (pembilang), dan bilangan pembaginya (penyebut), keduanya adalah bilangan desimal. Berikut ini adalah contoh bagan FSA sisa bagi 7 dari sembarang bilangan, sebagaimana yang dihasilkan dari penelitian tersebut:



Gambar 1. Mencari Sisa Bagi 7 Dari Sembarang Bilangan Desimal [1]

Gambar 1 memperlihatkan FSA dari proses mencari sisa bagi 7 yang diuraikan dalam penelitian [1]. Hasil bagan FSA pada gambar 1 memperlihatkan terdapat sangat banyak garis yang berpindah dari satu state ke state yang lain. Hal ini disebabkan karena bilangan yang terlibat dalam pencarian sisa bagi adalah bilangan desimal yang memiliki 10 (sepuluh) simbol yaitu 0, 1, 2, ... 9 sehingga diperlukan sejumlah garis untuk mengakomodasi seluruh simbol tersebut.

Penelitian ini mencoba menyederhanakan penggambaran bagan FSA untuk proses yang sama. Dalam penelitian ini bilangan yang akan dicari sisa baginya, yang disebut sebagai pembilang, adalah bilangan biner yang hanya memiliki 2 (dua) simbol yaitu 0 dan 1. Adapun untuk bilangan pembagi atau penyebut tetap bilangan desimal. Bilangan desimal penyebut ini akan muncul sebagai nama state pada bagan FSA.

## 2. METODE

### 2.1 BAGAN ALUR PENELITIAN

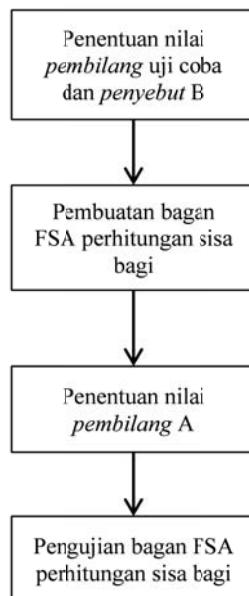
Untuk menghasilkan bagan FSA dari pencarian sisa bagi antara pembilang A dengan penyebut B, ada 2 (dua) kegiatan utama yang dilakukan yaitu Pembuatan Bagan FSA dan Pengujian Bagan FSA.

#### 1) Pembuatan Bagan FSA

Mula-mula ditentukan nilai untuk penyebut (sebut saja B). Berdasarkan nilai penyebut ini dibuatlah sejumlah state yang dipersiapkan sebagai state akhir. Langkah selanjutnya adalah menentukan bilangan biner yang akan dijadikan sebagai data pembilang untuk keperluan uji coba, jangan hanya satu namun beberapa data. Data uji coba ini akan digunakan untuk menyusun formasi state dan aliran proses. Lakukan penelusuran mulai digit awal data uji coba untuk menentukan aliran perpindahan proses dari state awal ke state berikutnya, sedemikian sehingga seluruh digit pada bilangan uji coba selesai ditelusuri.

#### 2) Pengujian Bagan FSA

Setelah bagan FSA untuk sisa bagi penyebut B selesai dibuat, selanjutnya bagan ini bisa diuji coba untuk sembarang bilangan pembilang A dengan cara menelusuri posisi state, dimulai dari state awal sampai dengan state akhir. Penelusuran itu sendiri dilakukan dengan menyusuri setiap digit dari bilangan pembilang A sejak digit pertama sampai digit terakhir.



Gambar 2. Bagan Alur Penelitian

## 2.2 KONSEP MENGHITUNG SISA BAGI BILANGAN

Konsep perhitungan mencari sisa bagi yang melibatkan 2 (dua) bilangan bulat positif telah dijelaskan pada penelitian di tahun 2024 [1]. Pada penelitian [1], prinsip utama dari proses mencari sisa bagi bilangan A terhadap bilangan B (ditulis dengan  $A \text{ mod } B$ ) adalah melakukan kegiatan-kegiatan berikut: (1) melakukan pembagian  $A/B$  dengan mengambil hasil berjenis bilangan bulat (tidak melibatkan hasil berupa bagian pecahan), (2) melakukan perkalian hasil pembagian tersebut dengan bilangan B, dan (3) menghitung selisih antara A dengan hasil perkalian tersebut. Bilangan sisa bagi adalah hasil selisih tersebut.

## 2.3 PENELUSURAN BAGAN FSA

Setelah bagan FSA terbentuk dan telah melalui tahap pengujian, selanjutnya bagan ini bisa digunakan untuk mencari sisa bagi dari sembarang bilangan biner dengan banyak digit tak terbatas. Perlu diingat bahwa sebuah bagan FSA yang telah dibentuk hanya diperuntukkan bagi sebuah nilai penyebut. Jika diinginkan untuk mencari sisa bagi dengan penyebut yang lain, maka prosesnya harus dimulai dari pembuatan bagan FSA baru. Penelusuran bagan FSA disajikan dalam Gambar 3.

## 3. HASIL DAN PEMBAHASAN

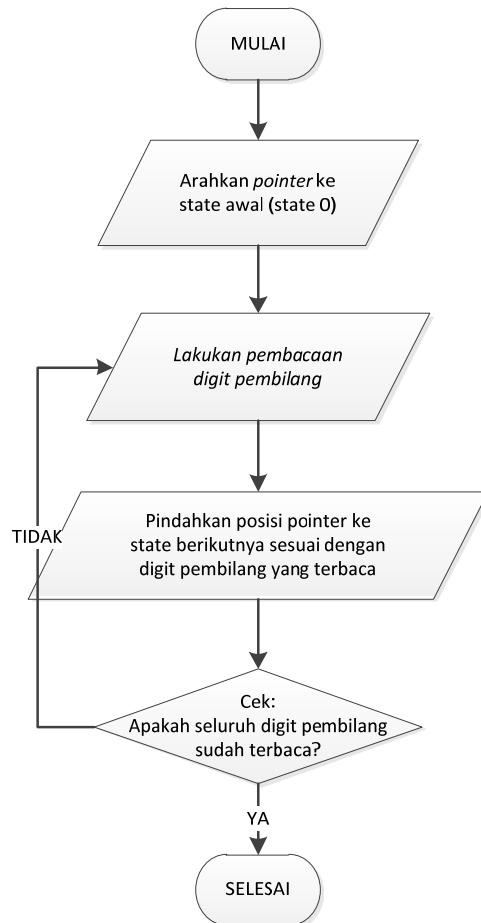
Pada penelitian ini dilakukan pencarian sisa bagi dengan penyebut berupa nilai 3, 5, dan 6. Untuk hasil bagi dari sembarang nilai penyebut N kemungkinannya adalah 0, 1, 2, ... ( $N-1$ ). Tabel 1 berikut ini menampilkan hasil sisa bagi dengan penyebut 3, 5, dan 6.

### 3.1 PENYUSUNAN DIAGRAM FSA SISA BAGI

Penelitian ini menggunakan penyebut dengan nilai 3 dan 5 sebagai studi kasus. Di sini akan dibuat bagan FSA untuk kedua nilai tersebut.

#### 1. Menyusun diagram FSA untuk hasil sisa bagi 3

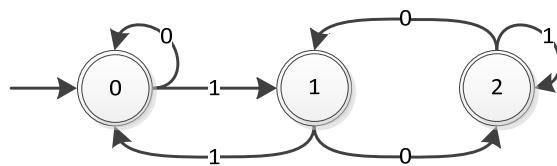
Kemungkinan hasil dari sisa bagi 3 adalah 0, 1, atau 2. Fakta ini diwujudkan dengan membuat 3 (tiga) state dalam bagan FSA dimana label dari setiap state mewakili nilai “0”, “1”, dan “2”.



Gambar 3. Penelusuran Bagan FSA Untuk Mencari Sisa Bagi

Tabel 1. Hasil perhitungan sisa bagi dengan penyebut 3, 5, dan 6

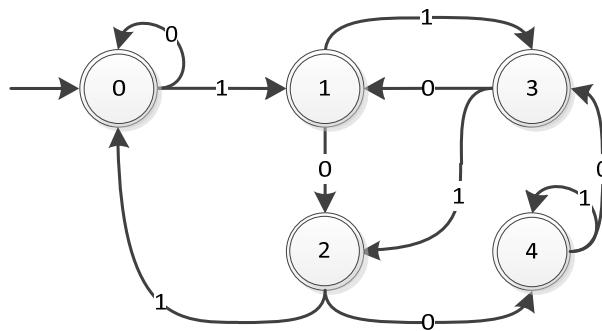
Bilangan desimal	Bilangan biner	Sisa bagi 3	Sisa bagi 5	Sisa bagi 6
0	0	0	0	0
1	1	1	1	1
2	10	2	2	2
3	11	0	3	3
4	100	1	4	4
5	101	2	0	5
6	110	0	1	0
7	111	1	2	1
8	1000	2	3	2
9	1001	0	4	3
10	1010	1	0	4
11	1011	2	1	5
12	1100	0	2	0
13	1101	1	3	1
14	1110	2	4	2
15	1111	0	0	3



Gambar 4. Bagan State FSA Untuk Sisa Bagi 3

2. Menyusun diagram FSA untuk hasil sisa bagi 5

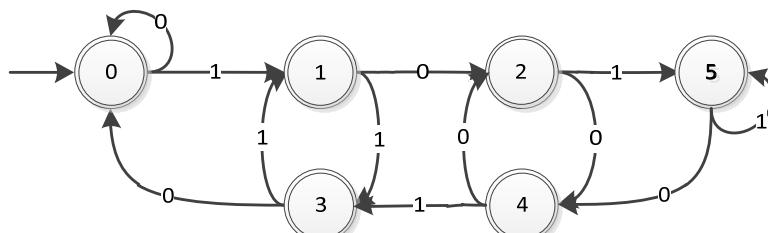
Kemungkinan hasil dari sisa bagi 5 adalah 0, 1, 2, 3, atau 4. Fakta ini diwujudkan dengan membuat 5 (lima) state dalam bagan FSA dimana label dari setiap state mewakili nilai “0”, “1”, “2”, “3”, dan “4”.



Gambar 5. Bagan State FSA Untuk Sisa Bagi 5

3. Menyusun diagram FSA untuk hasil sisa bagi 6

Kemungkinan hasil dari sisa bagi 6 adalah 0, 1, 2, 3, 4, atau 5. Fakta ini diwujudkan dengan membuat 6 (enam) state dalam bagan FSA dimana label dari setiap state mewakili nilai “0”, “1”, “2”, “3”, “4”, dan “5”.



Gambar 6. Bagan State FSA Untuk Sisa Bagi 6

### 3.2 PROSES PENCARIAN SISA BAGI

Pada tahap ini akan dilakukan pencarian sisa bagi dari sembarang bilangan N. Untuk memudahkan penjelasan, di sini digunakan bilangan 65 dan 870 sebagai pembilang. Kedua bilangan itu berjenis bilangan desimal sebagai sistem bilangan yang umum digunakan.

1. Konversi desimal ke biner

Sebelum dicari sisa baginya, kedua bilangan 65 dan 870 perlu diubah terlebih dahulu menjadi bilangan biner. Proses konversi ini bisa dilakukan dengan melakukan proses pembagian 2 dari masing-masing bilangan tersebut.

Konversi 65 menjadi biner:

$$\begin{array}{r} 2 \overline{)65} \quad 1 \\ \quad 32 \\ 2 \overline{)32} \quad 0 \\ \quad 16 \\ 2 \overline{)16} \quad 0 \\ \quad 8 \\ 2 \overline{)8} \quad 0 \\ \quad 4 \\ 2 \overline{)4} \quad 0 \\ \quad 2 \\ 2 \overline{)2} \quad 0 \\ \quad 1 \end{array}$$

Hasil akhir:  $65_d \rightarrow 1000001_b$

Konversi 870 menjadi biner:

$$\begin{array}{r} 2 \overline{)870} \quad 0 \\ \quad 435 \\ 2 \overline{)435} \quad 1 \\ \quad 217 \\ 2 \overline{)217} \quad 1 \\ \quad 108 \\ 2 \overline{)108} \quad 0 \\ \quad 54 \\ 2 \overline{)54} \quad 0 \\ \quad 27 \\ 2 \overline{)27} \quad 1 \\ \quad 13 \\ 2 \overline{)13} \quad 1 \\ \quad 6 \\ 2 \overline{)6} \quad 0 \\ \quad 3 \\ 2 \overline{)3} \quad 1 \\ \quad 1 \end{array}$$

Hasil akhir:  $870_d \rightarrow 1101100110_b$

## 2. Mencari sisa bagi 3

Proses mencari sisa bagi 3 dari bilangan 65 ( $1000001_b$ ) dan 870 ( $1101100110_b$ ) ditunjukkan pada tabel berikut:

Tabel 2. Proses Menelusuri Bagan FSA Mencari Sisa Bagi 3

Bilangan	State awal	Digit transisi	State berikutnya	State akhir
65 ( $1000001_b$ )	“0”	1	“1”	-
		0	“2”	-
		0	“1”	-
		0	“2”	-
		0	“1”	-
		0	“2”	-
		1	“2”	“2”
870 ( $1101100110_b$ )	“0”	1	“1”	-
		1	“0”	-
		0	“0”	-
		1	“1”	-
		1	“0”	-
		0	“0”	-
		0	“0”	-
		1	“1”	-
		1	“0”	-
		0	“0”	“0”

3. Mencari sisa bagi 5

Proses mencari sisa bagi 5 dari bilangan 65 ( $1000001_b$ ) dan 870 ( $1101100110_b$ ) ditunjukkan pada tabel berikut:

Tabel 3. Proses Menelusuri Bagan FSA Mencari Sisa Bagi 5

Bilangan	State awal	Digit transisi	State berikutnya	State akhir
65 ( $1000001_b$ )	“0”	1	“1”	-
		0	“2”	-
		0	“4”	-
		0	“3”	-
		0	“1”	-
		0	“2”	-
		1	“0”	“0”
870 ( $1101100110_b$ )	“0”	1	“1”	-
		1	“3”	-
		0	“1”	-
		1	“3”	-
		1	“2”	-
		0	“4”	-
		0	“3”	-
		1	“2”	-
		1	“0”	-
		0	“0”	“0”

4. Mencari sisa bagi 6

Proses mencari sisa bagi 6 dari bilangan 65 ( $1000001_b$ ) dan 870 ( $1101100110_b$ ) ditunjukkan pada tabel berikut:

Tabel 4. Proses Menelusuri Bagan FSA Mencari Sisa Bagi 6

Bilangan	State awal	Digit transisi	State berikutnya	State akhir
65 ( $1000001_b$ )	“0”	1	“1”	-
		0	“2”	-
		0	“4”	-
		0	“2”	-
		0	“4”	-
		0	“2”	-
		1	“5”	“5”
870 ( $1101100110_b$ )	“0”	1	“1”	-
		1	“3”	-
		0	“0”	-
		1	“1”	-
		1	“3”	-
		0	“0”	-
		0	“0”	-
		1	“1”	-
		1	“3”	-
		0	“0”	“0”

Sebagai bahan perbandingan mengenai ketepatan hasil akhir, gambar berikut ini adalah *screenshot* dari perhitungan yang sama menggunakan aplikasi MS Excel:

	B14	f(x)	
	A	B	C
2			
3	Sisa bagi 3 dari 65 (65 mod 3):	2	
4	Sisa bagi 3 dari 870 (870 mod 3):	0	
5			
6	Sisa bagi 5 dari 65 (65 mod 5):	0	
7	Sisa bagi 5 dari 870 (870 mod 5):	0	
8			
9	Sisa bagi 6 dari 65 (65 mod 6):	5	
10	Sisa bagi 6 dari 870 (870 mod 6):	0	
11			

Gambar 7. Tangkapan Layar Hasil Perhitungan Menggunakan Aplikasi MS Excel

#### 4. SIMPULAN

Berdasarkan hasil ujicoba yang telah dilakukan, selain terbukti bahwa hasil perhitungan mendapatkan hasil yang benar yang didukung dengan penggunaan aplikasi MS Excel sebagai pembanding hasil perhitungan, penelitian ini juga memperlihatkan bahwa bagan FSA untuk hasil perhitungan sisa bagi yang dihasilkan lebih sederhana dibandingkan dengan penelitian terdahulu. Oleh karena bilangan pembilang yang akan dicari sisa baginya berjenis biner, kelemahan dari penelitian ini adalah proses konversi dari bilangan desimal yang besar menjadi biner akan menghabiskan waktu yang cukup lama. Oleh karena itu disarankan agar penelitian selanjutnya bisa mengatasi masalah proses konversi ini agar bisa pembuatan bagan state FSA menghitung sisa bagi ini bisa lebih sederhana.

#### REFERENSI

- [1] Purnama, Rangsang; Hidayat, M. Mahaputra. 2023. Implementation of Finite State Automata (FSA) Chart in the Process of Calculation of The Reminder of Two Numbers with an Infinite Many Digits. Jurnal Restikom: Riset Teknik Informatika dan Komputer, vol. 6 no. 3, Desember 2024.
- [2] Emi, Titiani, Fakihotun, Putri, Sukmawati Anggraeni, & Gata, Windu . 2020. Penerapan Konsep Finite State Automata Pada Aplikasi Simulasi Vending Machine Jamu Tradisional. Jurnal Informatika, vol. 7 no. 2 September 2020.
- [3] Lajanto, Dan. 2023. Konsep dan Teorema Mengenai Modulo. [https://www.danlajanto.com/2013/12/konsep-dan-teorema-mengenai-modulo\\_5330.html](https://www.danlajanto.com/2013/12/konsep-dan-teorema-mengenai-modulo_5330.html). Diakses 28 November 2023.
- [4] Stallings, William. 2010. Computer Organization and Architecture. Designing for Performance. Eighth Edition.Pearson Education, Inc., Upper Saddle River, New Jersey.
- [5] Wikipedia. 2023. Automata theory. [https://en.wikipedia.org/wiki/Automata\\_theory](https://en.wikipedia.org/wiki/Automata_theory). Diakses 28 November 2023.