

BUKU AJAR

MATA KULIAH ARSITEKTUR KOMPUTER

BAB 8

RANGKAIAN KOMBINASIONAL DAN RANGKAIAN SEKUENSIAL DALAM SISTEM DIGITAL

Berbasis Rencana Pembelajaran Semester (RPS)
Arsitektur Komputer IF221101 | 3 SKS | 2026

IDENTITAS BAB

Komponen	Keterangan
Mata Kuliah	Arsitektur Komputer
Kode	IF221101
Bobot	3 SKS
Bab	Bab 8
Judul	Rangkaian Kombinasional dan Rangkaian Sekuensial dalam Sistem Digital
Fokus Pembelajaran	Menganalisis blok rangkaian digital yang melakukan pemrosesan dan penyimpanan keadaan
Level Pembelajaran	Pemahaman konsep, analisis tabel keadaan, perancangan fungsi, dan penerapan sistem digital

CAPAIAN PEMBELAJARAN BAB

Setelah mempelajari bab ini, mahasiswa diharapkan mampu:

1. Membedakan karakteristik rangkaian kombinasional dan rangkaian sekuensial.
2. Menjelaskan fungsi multiplexer, demultiplexer, encoder, dan decoder.
3. Menganalisis operasi half adder dan full adder.
4. Menjelaskan konsep memori keadaan pada rangkaian sekuensial.
5. Mengidentifikasi prinsip dasar latch dan flip-flop.
6. Menganalisis tabel karakteristik flip-flop SR, JK, D, dan T.
7. Menjelaskan konsep register dan perannya dalam penyimpanan data.
8. Menjelaskan prinsip counter dan urutan keadaan.
9. Merancang solusi rangkaian sederhana berdasarkan kebutuhan masukan, keluaran, dan keadaan sistem.

PETA KONSEP

ALUR KONSEP BAB 8

Rangkaian Digital → Kombinasional → Adder / Multiplexer / Demultiplexer / Encoder / Decoder → Sekuensial → Clock → Latch → Flip-Flop → Register → Shift Register → Counter → Finite State Machine → Penerapan Sistem Komputer.

8.1 Pendahuluan

Pada bab sebelumnya, gerbang logika dibahas sebagai blok dasar pembentuk fungsi digital. Namun, sistem komputer tidak hanya membutuhkan operasi logis sederhana. Sistem juga memerlukan mekanisme pemilihan data, operasi aritmetika, penyandian, penyimpanan keadaan, serta perubahan keadaan yang teratur.

Rangkaian kombinasional membentuk keluaran berdasarkan kombinasi masukan saat ini. Rangkaian sekuensial memperluas konsep tersebut dengan melibatkan keadaan sebelumnya. Perbedaan inilah yang memungkinkan sistem digital memiliki kemampuan untuk menyimpan informasi dan menjalankan urutan operasi.

PERTANYAAN PEMANTIK

Bagaimana komputer memilih satu data dari banyak jalur? Bagaimana sebuah bit dapat disimpan? Bagaimana suatu sistem dapat menghitung 0, 1, 2, 3 secara berurutan?

8.2 Perbedaan Rangkaian Kombinasional dan Sekuensial

Aspek	Kombinasional	Sekuensial
Dasar keluaran	Masukan saat ini	Masukan saat ini dan keadaan sebelumnya
Elemen memori	Tidak menjadi komponen utama	Diperlukan untuk menyimpan keadaan
Clock	Tidak selalu diperlukan	Sering digunakan untuk sinkronisasi
Contoh	Adder, MUX, decoder	Flip-flop, register, counter

Secara konseptual, rangkaian kombinasional dapat dipandang sebagai fungsi $Y=f(X)$, sedangkan rangkaian sekuensial melibatkan keadaan Q sehingga keluaran atau keadaan berikutnya dapat bergantung pada X dan Q sebelumnya.

8.3 Rangkaian Kombinasional

Rangkaian kombinasional dibangun dari kombinasi gerbang logika tanpa mekanisme penyimpanan keadaan internal. Untuk setiap kombinasi masukan tertentu, rangkaian menghasilkan keluaran yang ditentukan oleh fungsi logikanya.

Contoh penting meliputi adder, subtractor, comparator, multiplexer, demultiplexer, encoder, dan decoder.

8.4 Half Adder dan Full Adder

8.4.1 Half Adder

Half adder menjumlahkan dua bit A dan B . Keluaran Sum menunjukkan hasil bit penjumlahan, sedangkan Carry menunjukkan pembawaan.

A	B	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1

Sum = A XOR B dan Carry = A AND B.

8.4.2 Full Adder

Full adder menjumlahkan A, B, dan carry-in (Cin). Keluaran terdiri atas Sum dan carry-out (Cout). Blok ini dapat digunakan secara berantai untuk membentuk penjumlahan multi-bit.

A	B	Cin	Sum	Cout
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

ANALISIS

Untuk penjumlahan beberapa bit, carry dari posisi bit lebih rendah dapat menjadi carry-in bagi posisi berikutnya. Konsep ini merupakan dasar pembentukan penjumlahan multi-bit.

8.5 Multiplexer

Multiplexer atau MUX memilih satu dari beberapa masukan untuk diteruskan ke keluaran berdasarkan sinyal seleksi. Sebuah MUX 2-ke-1 memiliki dua input data, satu sinyal seleksi, dan satu output.

S	Output Y
0	I0
1	I1

MUX dapat dipandang sebagai pemilih jalur data. Pada sistem digital, konsep pemilihan ini digunakan untuk menentukan sumber data yang akan diteruskan ke bagian berikutnya.

8.5.1 Multiplexer 4-ke-1

MUX 4-ke-1 memiliki empat input data dan dua sinyal seleksi. Dua bit seleksi menghasilkan empat kombinasi untuk memilih I0, I1, I2, atau I3.

8.6 Demultiplexer

Demultiplexer atau DEMUX melakukan fungsi kebalikan secara konseptual: satu input data diarahkan ke salah satu dari beberapa output berdasarkan sinyal seleksi.

S	Jalur Aktif
0	Y0 menerima input
1	Y1 menerima input

DEMUX dapat digunakan untuk memahami distribusi data ke jalur keluaran yang berbeda.

8.7 Encoder

Encoder mengubah informasi dari sejumlah jalur input menjadi representasi kode yang lebih ringkas. Contoh sederhana adalah encoder 4-ke-2 yang secara ideal memetakan satu input aktif dari empat kemungkinan menjadi dua bit kode.

Input Aktif	Output
D0	00
D1	01
D2	10
D3	11

Dalam implementasi praktis, kondisi beberapa input aktif dapat memerlukan aturan tambahan, misalnya priority encoder.

8.8 Decoder

Decoder melakukan pemetaan dari kode input ke aktivasi keluaran tertentu. Decoder 2-ke-4 memiliki dua input dan empat output, dengan satu keluaran yang dipilih berdasarkan kombinasi input.

A	B	Output Aktif
0	0	Y0
0	1	Y1
1	0	Y2
1	1	Y3

Konsep decoder relevan untuk seleksi, pengalamatan, dan aktivasi jalur tertentu dalam sistem digital.

8.9 Comparator

Comparator membandingkan dua nilai biner. Untuk dua input sederhana, keluaran dapat menunjukkan kondisi $A=B$, $A>B$, atau $A<B$. Fungsi ini dibangun melalui kombinasi operasi logika pada bit-bit yang dibandingkan.

Comparator merupakan contoh bagaimana beberapa fungsi dasar dapat digabungkan menjadi unit pengambilan keputusan digital.

8.10 Pengantar Rangkaian Sekuensial

Rangkaian sekuensial memiliki konsep keadaan atau state. State menyimpan informasi tentang kondisi sebelumnya. Ketika terjadi perubahan masukan atau sinyal pengendali tertentu, rangkaian dapat berpindah ke keadaan baru.

IDE UTAMA

Jika rangkaian kombinasional menjawab “apa output untuk input sekarang?”, maka rangkaian sekuensial juga mempertimbangkan “apa yang terjadi sebelumnya?”.

8.11 Clock dan Sinkronisasi

Clock merupakan sinyal periodik yang digunakan sebagai acuan waktu pada banyak rangkaian sekuensial sinkron. Perubahan keadaan dapat dirancang terjadi pada tepi sinyal tertentu, misalnya rising edge atau falling edge.

Penggunaan clock membantu koordinasi perubahan keadaan antar bagian sistem. Detail karakteristik waktu dan implementasi fisik bergantung pada rancangan perangkat keras.

8.12 Latch

Latch adalah elemen penyimpan keadaan sederhana. Dalam konsep dasar, latch dapat mempertahankan nilai keluaran sampai kondisi pengendali tertentu mengizinkan perubahan.

Latch membantu memperkenalkan gagasan bahwa rangkaian digital dapat menyimpan informasi, bukan hanya menghitung keluaran dari input saat ini.

8.13 Flip-Flop

Flip-flop merupakan elemen penyimpan satu bit yang banyak digunakan dalam rangkaian sekuensial sinkron. Jenis flip-flop yang umum dipelajari meliputi SR, JK, D, dan T.

8.13.1 SR Flip-Flop

SR flip-flop menggunakan masukan Set dan Reset. Secara konseptual, Set digunakan untuk mengarahkan keadaan ke 1, sedangkan Reset mengarahkan keadaan ke 0. Tabel karakteristik bergantung pada implementasi, dan terdapat kombinasi masukan tertentu yang dapat dianggap tidak valid pada bentuk dasar.

S	R	Q berikutnya
0	0	Tetap
0	1	0
1	0	1
1	1	Tidak valid pada model dasar

8.13.2 JK Flip-Flop

JK flip-flop dikembangkan untuk menyediakan perilaku toggle pada kombinasi J=1 dan K=1.

J	K	Q berikutnya
0	0	Tetap
0	1	0
1	0	1
1	1	Toggle

8.13.3 D Flip-Flop

D flip-flop menyimpan nilai input D pada momen pemicu clock yang sesuai. Secara konseptual, Q berikutnya mengikuti nilai D pada peristiwa clock.

D	Q berikutnya
0	0
1	1

8.13.4 T Flip-Flop

T flip-flop mempertahankan keadaan ketika T=0 dan mengubah keadaan menjadi kebalikannya ketika T=1 pada pemicu yang sesuai.

T	Q berikutnya
0	Tetap
1	Toggle

8.14 Register

Register adalah sekumpulan elemen penyimpanan yang digunakan untuk menyimpan sejumlah bit. Register 8-bit, misalnya, dapat menyimpan satu kata data sepanjang delapan bit.

Dalam organisasi komputer, register digunakan untuk menyimpan data sementara, alamat, instruksi, hasil antara, atau informasi kendali sesuai fungsi masing-masing.

8.15 Shift Register

Shift register merupakan register yang memungkinkan isi data bergeser dari satu posisi ke posisi lain berdasarkan mekanisme pengendalian. Pergerakan dapat digunakan untuk memahami operasi serial dan paralel.

Jenis Konseptual	Karakteristik
SISO	Serial Input, Serial Output
SIPO	Serial Input, Parallel Output
PISO	Parallel Input, Serial Output
PIPO	Parallel Input, Parallel Output

8.16 Counter

Counter adalah rangkaian sekuensial yang melewati urutan keadaan tertentu. Counter biner 2-bit dapat memiliki urutan $00 \rightarrow 01 \rightarrow 10 \rightarrow 11 \rightarrow 00$.

Counter digunakan sebagai konsep dasar untuk pencacahan, pengurutan keadaan, dan pembangkitan urutan dalam sistem digital.

Clock ke-	Keadaan 2-bit
0	00
1	01
2	10
3	11
4	00

8.17 Synchronous dan Asynchronous Counter

Pada counter sinkron, elemen penyimpan menggunakan acuan clock bersama sesuai rancangan. Pada counter asinkron, perubahan keadaan dapat merambat dari satu elemen ke elemen berikutnya. Perbedaan ini memengaruhi cara perubahan keadaan dianalisis.

Untuk tahap pengantar, mahasiswa perlu memahami urutan keadaan dan hubungan antara bit counter sebelum mempelajari detail waktu propagasi dan implementasi fisik.

8.18 Pengantar Finite State Machine

Finite State Machine (FSM) adalah model sistem dengan sejumlah keadaan terbatas dan aturan perpindahan antar keadaan. FSM membantu memodelkan sistem digital yang perilakunya bergantung pada state dan input.

Komponen FSM	Peran
State	Merepresentasikan kondisi sistem saat ini
Input	Informasi yang memengaruhi transisi
Transition	Aturan perpindahan antar state
Output	Respons sistem berdasarkan model yang digunakan

Model FSM memberikan jembatan antara teori rangkaian sekuensial dan desain pengendali digital.

8.19 Penerapan dalam Sistem Komputer

Komponen	Konsep Rangkaian
----------	------------------

ALU	Adder dan fungsi kombinasional
Jalur Data	Multiplexer untuk pemilihan sumber
Pengalamatan	Decoder untuk seleksi
Penyimpanan internal	Register dan elemen penyimpan
Pengendalian urutan	Counter dan state machine

Komponen sistem komputer dibangun melalui penggabungan blok-blok logika. Pemahaman terhadap blok tersebut membantu mahasiswa melihat hubungan antara gerbang dasar dan organisasi perangkat keras yang lebih besar.

8.20 Studi Kasus 1: Pemilih Data 4-ke-1

Sebuah unit membutuhkan satu keluaran data dari empat sumber: I0, I1, I2, dan I3. Dua bit seleksi S1 dan S0 digunakan untuk menentukan sumber aktif.

S1S0	Sumber Keluaran
00	I0
01	I1
10	I2
11	I3

Kasus ini dapat dianalisis sebagai fungsi pemilihan data dan menjadi dasar untuk memahami penggunaan multiplexer dalam jalur data.

8.21 Studi Kasus 2: Counter Parkir Sederhana

Sebuah sistem menghitung jumlah kendaraan secara sederhana menggunakan counter. Ketika kendaraan masuk, state counter bertambah. Ketika kondisi keluar dianalisis, sistem dapat menerapkan aturan perubahan state yang berbeda sesuai rancangan.

ANALISIS

Mahasiswa diminta menentukan jumlah bit yang diperlukan untuk merepresentasikan kapasitas tertentu dan menyusun urutan state yang sesuai.

8.22 Studi Kasus 3: Register Data 4-bit

Sebuah register 4-bit menyimpan data 1011. Setelah proses load, empat bit tersebut dipertahankan sebagai keadaan register sampai ada mekanisme perubahan yang sesuai.

Kasus ini menunjukkan perbedaan utama antara data sesaat pada rangkaian kombinasional dan informasi yang dipertahankan oleh elemen penyimpan.

8.23 Aktivitas Pembelajaran Mahasiswa

Aktivitas 1: Analisis Adder

Buat tabel kebenaran full adder dan identifikasi kondisi yang menghasilkan carry-out.

Aktivitas 2: MUX

Rancang tabel pemilihan untuk MUX 4-ke-1.

Aktivitas 3: Encoder dan Decoder

Bandingkan proses pengkodean dan dekode menggunakan contoh 4-ke-2 dan 2-ke-4.

Aktivitas 4: Flip-Flop

Bandingkan perubahan state SR, JK, D, dan T berdasarkan tabel karakteristik.

Aktivitas 5: Register

Buat simulasi konseptual register 4-bit dengan operasi load dan hold.

Aktivitas 6: Counter

Susun urutan keadaan counter modulo-5 dan tentukan jumlah bit minimum yang diperlukan.

Aktivitas 7: FSM

Rancang diagram state sederhana untuk sistem dengan minimal tiga keadaan.

8.24 Rangkuman

- Rangkaian kombinasional menghasilkan keluaran berdasarkan masukan saat ini.
- Rangkaian sekuensial melibatkan masukan dan keadaan sebelumnya.
- Adder digunakan sebagai blok dasar operasi penjumlahan biner.
- MUX memilih satu jalur data dari beberapa masukan, sedangkan DEMUX mendistribusikan satu masukan ke jalur keluaran yang dipilih.
- Encoder dan decoder melakukan transformasi antara representasi jalur dan kode.
- Clock memberikan acuan sinkronisasi pada banyak rangkaian sekuensial.
- Flip-flop menyimpan satu bit keadaan.
- Register menyimpan sekumpulan bit.
- Shift register mendukung perpindahan data antar posisi penyimpanan.
- Counter menghasilkan urutan keadaan.
- FSM memodelkan perilaku sistem berdasarkan state, input, dan transisi.
- Blok kombinasional dan sekuensial menjadi bagian penting dalam organisasi sistem komputer.

8.25 Glosarium Bab

Istilah	Pengertian
Adder	Rangkaian kombinasional untuk operasi penjumlahan biner.

Clock	Sinyal acuan waktu untuk sinkronisasi perubahan keadaan.
Counter	Rangkaian sekuensial yang menghasilkan urutan state.
Decoder	Rangkaian yang mengaktifkan keluaran berdasarkan kode input.
Demultiplexer	Rangkaian yang mengarahkan satu input ke keluaran yang dipilih.
Encoder	Rangkaian yang mengubah jalur input menjadi kode.
Finite State Machine	Model sistem dengan sejumlah state dan aturan transisi.
Flip-Flop	Elemen penyimpan satu bit keadaan.
Latch	Elemen penyimpan keadaan yang dikendalikan oleh kondisi tertentu.
Multiplexer	Rangkaian pemilih satu sumber dari beberapa input.
Register	Sekumpulan elemen penyimpan untuk menyimpan beberapa bit.
Rangkaian Kombinasional	Rangkaian yang outputnya ditentukan oleh input saat ini.
Rangkaian Sekuensial	Rangkaian yang melibatkan input dan state sebelumnya.
Shift Register	Register yang mendukung perpindahan data antar posisi.
State	Keadaan internal sistem pada suatu waktu.

8.26 Soal Latihan

A. Soal Pemahaman

- Jelaskan perbedaan utama rangkaian kombinasional dan rangkaian sekuensial.
- Apa fungsi Sum dan Carry pada half adder?
- Mengapa full adder membutuhkan carry-in?
- Jelaskan fungsi multiplexer.
- Jelaskan perbedaan multiplexer dan demultiplexer.
- Bandingkan encoder dan decoder.
- Mengapa clock penting pada rangkaian sekuensial sinkron?
- Jelaskan fungsi D flip-flop.
- Apa peran register dalam sistem komputer?
- Jelaskan konsep state pada finite state machine.

B. Soal Perhitungan dan Analisis

- Buat tabel kebenaran full adder.
- Rancang tabel pemilihan MUX 8-ke-1 dan tentukan jumlah bit seleksi yang diperlukan.
- Tentukan jumlah bit minimum untuk counter yang memiliki 10 keadaan.
- Tuliskan urutan state counter modulo-6.
- Bandingkan keluaran JK flip-flop ketika $J=1$ dan $K=1$ untuk dua keadaan awal yang berbeda.

25. Jika register 8-bit menyimpan 11001010, jelaskan perubahan data ketika dilakukan operasi shift konseptual satu posisi ke kiri dengan asumsi input baru 0.

C. Soal HOTS dan Studi Kasus

26. Rancang sistem pemilihan data dengan enam sumber dan jelaskan pendekatan penggunaan MUX yang sesuai.
27. Analisis perbedaan kebutuhan rangkaian kombinasional dan sekuensial pada sistem kontrol lampu lalu lintas.
28. Buat model FSM sederhana untuk pintu otomatis dengan keadaan tertutup, membuka, terbuka, dan menutup.
29. Rancang counter untuk urutan state tidak berurutan, misalnya $00 \rightarrow 10 \rightarrow 11 \rightarrow 01 \rightarrow 00$, dan jelaskan transisinya.
30. Analisis bagaimana register, multiplexer, dan counter dapat bekerja bersama dalam suatu sistem pemrosesan sederhana.

8.27 Proyek Mini

Judul proyek: Simulator Rangkaian Kombinasional dan Sekuensial.

Mahasiswa membuat aplikasi, simulasi, atau rancangan yang memuat:

- Minimal satu rangkaian kombinasional, misalnya MUX, decoder, atau full adder.
- Minimal satu rangkaian sekuensial, misalnya register atau counter.
- Tabel kebenaran atau tabel karakteristik yang relevan.
- Visualisasi urutan state atau perubahan register.
- Minimal lima skenario pengujian.
- Analisis perbedaan antara keluaran langsung dan perubahan state.
- Dokumentasi hubungan rancangan dengan komponen sistem komputer.

OUTPUT YANG DIHARAPKAN

Dokumentasi desain, tabel fungsi, tabel state, hasil pengujian, analisis, dan kesimpulan mengenai peran rangkaian kombinasional serta sekuensial.

8.28 Daftar Pustaka

- Mano, M. M., & Ciletti, M. D. Digital Design. Pearson.
- Wakerly, J. F. Digital Design: Principles and Practices. Pearson.
- Harris, D. M., & Harris, S. L. Digital Design and Computer Architecture. Morgan Kaufmann.
- Heuring, V. P., & Murdocca, M. J. Principles of Computer Architecture.
- Englander, I., & Wong, W. The Architecture of Computer Hardware, Systems Software, and Networking. John Wiley & Sons.
- Dokumen Rencana Pembelajaran Semester Mata Kuliah Arsitektur Komputer IF221101, Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Pembangunan Nasional Veteran Jawa Timur.

CATATAN PENYUSUNAN BAB

Bab ini melanjutkan pembahasan gerbang logika dengan mengembangkan blok rangkaian yang lebih kompleks. Materi bergerak dari rangkaian kombinasional untuk pemrosesan data menuju rangkaian sekuensial yang memiliki state dan kemampuan penyimpanan. Urutan tersebut menjadi dasar untuk memahami register, jalur data, pengendalian, dan organisasi komputer pada pembahasan berikutnya.