

The background of the entire page is a close-up photograph of a printed circuit board (PCB). It shows various electronic components such as integrated circuits, capacitors, and resistors, with their labels (e.g., C102, R57, U10) visible. The image is slightly blurred, creating a sense of depth. There are also some semi-transparent orange rectangular overlays on the left side of the image.

ORGANISASI & ARSITEKTUR KOMPUTER

Romindo, Rini Nuraini, Rita Komalasari, Fajar Sari Kurniawan,
Zul Rachmat, Wahyuddin S, Zen Munawar, Johni S Pasaribu,
Novianti Indah Putri, Niko Akbar, Listra Firgia, Amna

ORGANISASI DAN ARSITEKTUR KOMPUTER

**Romindo
Rini Nuraini
Rita Komalasari
Fajar Sari Kurniawan
Zul Rachmat
Wahyuddin S
Zen Munawar
Johni S Pasaribu
Novianti Indah Putri
Niko Akbar
Listra Firgia
Amna**



PT GLOBAL EKSEKUTIF TEKNOLOGI

ORGANISASI DAN ARSITEKTUR KOMPUTER

Penulis :

Romindo
Rini Nuraini
Rita Komalasari
Fajar Sari Kurniawan
Zul Rachmat
Wahyuddin S
Zen Munawar
Johni S Pasaribu
Novianti Indah Putri
Niko Akbar
Listra Furgia
Amna

ISBN : 978-623-198-191-2

Editor : Ari Yanto. M.Pd.

Penyunting : Tri Putri Wahyuni, S.Pd

Desain Sampul dan Tata Letak : Atyka Trianisa, S.Pd

Penerbit : PT GLOBAL EKSEKUTIF TEKNOLOGI

Anggota IKAPI No. 033/SBA/2022

Redaksi :

Jl. Pasir Sebelah No. 30 RT 002 RW 001
Kelurahan Pasie Nan Tigo Kecamatan Koto Tengah
Padang Sumatera Barat

Website : www.globaleksekutifteknologi.co.id

Email : globaleksekutifteknologi@gmail.com

Cetakan pertama, April 2023

Hak cipta dilindungi undang-undang
Dilarang memperbanyak karya tulis ini dalam bentuk
dan dengan cara apapun tanpa izin tertulis dari penerbit.

KATA PENGANTAR

Segala Puji dan syukur atas kehadiran Allah SWT dalam segala kesempatan. Sholawat beriring salam dan doa kita sampaikan kepada Nabi Muhammad SAW. Alhamdulillah atas Rahmat dan Karunia-Nya penulis telah menyelesaikan Buku Organisasi Dan Arsitektur Komputer ini.

Buku Ini Membahas Jenis-jenis operasi computer, Perkembangan evolusi computer, metode dan teknologi transfer data antar komponen di dalam computer, Internal memory, Eksternal memory, Cara kerja peralatan input dan peralatan output, Dukungan sistem operasi dalam kinerja computer, Dasar perhitungan aritmatika dalam *computer*, *Instruction Addressing* dan format, CPU, RISC/Reduce instruction set, Proses paralel dan superscalar dalam operasional computer.

Proses penulisan buku ini berhasil diselesaikan atas kerjasama tim penulis. Demi kualitas yang lebih baik dan kepuasan para pembaca, saran dan masukan yang membangun dari pembaca sangat kami harapkan.

Penulis ucapkan terima kasih kepada semua pihak yang telah mendukung dalam penyelesaian buku ini. Terutama pihak yang telah membantu terbitnya buku ini dan telah mempercayakan mendorong, dan menginisiasi terbitnya buku ini. Semoga buku ini dapat bermanfaat bagi masyarakat Indonesia.

Padang, April 2023

Penulis

DAFTAR ISI

KATA PENGANTAR	i
DAFTAR ISI	ii
DAFTAR GAMBAR	vi
DAFTARTABEL	ix
BAB 1 JENIS-JENIS OPERASI KOMPUTER	1
1.1 Pendahuluan.....	1
1.2 Peranan dan Kategori Sistem Operasi	10
1.3 Jenis-Jenis Sistem Operasi Komputer.....	11
DAFTAR PUSTAKA.....	20
BAB 2 EVOLUSI PERKEMBANGAN KOMPUTER	21
2.1 Pengertian Komputer	21
2.2 Awal Penemuan Komputer	21
2.3 Cikal Bakal Komputer Digital	22
2.4 Lahirnya Bahasa Pemrograman.....	23
2.5 Personal Computer (PC)	24
2.6 Komputer era 2000-an	25
2.7 <i>Internet of Things</i> (IoT)	25
2.8 Sejarah dan Evolusi Perangkat Komputasi	25
2.9 Sejarah Perkembangan Komputer Secara Umum	34
2.10 Ciri-Ciri Komputer Generasi Awal Hingga Kelima.....	35
DAFTAR PUSTAKA.....	46
BAB 3 METODE DAN TEKNOLOGI TRANSFER	
DATA ANTAR KOMPONEN DI DALAM KOMPUTER.....	47
3.1 Struktur Interkoneksi.....	47
3.2 Interkoneksi Bus	49
DAFTAR PUSTAKA.....	59
BAB 4 MEMORI INTERNAL	61
4.1 Pendahuluan.....	61
4.2 Pengertian Memori Internal.....	62
4.3 Jenis-jenis Memori Internal	63
4.3.1 <i>Read Only Memory</i> (ROM)	64

4.3.2 <i>Random Access Memory</i> (RAM).....	67
4.3.3 <i>Cache Memory</i>	72
4.4 Modul Memori Internal.....	74
DAFTAR PUSTAKA.....	78
BAB 5 EKSTERNAL MEMORY	79
5.1 Pendahuluan.....	79
5.2 Pengertian Memori Eksternal	80
5.3 Jenis - Jenis Memori Eksternal.....	82
5.4 Karakteristik Memori Eksternal	88
DAFTAR PUSTAKA.....	90
BAB 6 CARA KERJA PERALATAN INPUT DAN PERALATAN OUTPUT	91
6.1 Pendahuluan.....	91
6.2 Komponen Utama Sistem Komputer	92
6.2.1 <i>Hardware</i> (Perangkat Keras)	92
6.2.2 <i>Software</i> (Perangkat Lunak)	93
6.2.3 <i>Brainware</i> (Pengguna)	93
6.3 Peralatan Input Komputer	94
6.3 Peralatan Output Komputer	98
DAFTAR PUSTAKA.....	103
BAB 7 DUKUNGAN SISTEM OPERASI DALAM KINERJA KOMPUTER	105
7.1 Pendahuluan.....	105
7.2 Sistem Operasi MS Windows	106
7.2.1 Kinerja Sistem Operasi	107
7.2.2 Evaluasi Kinerja.....	109
7.2.3 Faktor Kinerja dan Atribut Sistem.....	110
7.3 Dukungan Kinerja Pada SO Windows.....	110
7.3.1 Teknik Kinerja Windows XP	111
7.3.2 Peningkatan Kinerja Pada Windows Vista	111
7.3.3 Peningkatan Kinerja Pada Windows 7	113
DAFTAR PUSTAKA.....	116
BAB 8 DASAR ARITMATIKA KOMPUTER.....	119

8.1 Pendahuluan.....	119
8.2 ALU (<i>Arithmetic Logic Unit</i>)	122
8.3 Representasi Integer.....	122
8.3.1 Bilangan bulat Biner tak bertanda (<i>Unsigned Integer</i>)	123
8.3.2 Bilangan bulat Biner bertanda (<i>Signed Integer</i>)	125
8.3.3 Penjumlahan dan Pengurangan.....	131
8.3.4 Perkalian.....	135
8.3.5 Pembagian.....	140
8.4 Representasi Floating Point.....	143
DAFTAR PUSTAKA.....	147
BAB 9 INSTRUCTION ADDRESSING DAN FORMAT	149
9.1 Pendahuluan.....	149
9.2 Addressing Modes.....	150
9.2.1 Klasifikasi Instruksi.....	151
9.2.2 Operasi Instruksi Tiga Alamat	152
9.3 Intruction Format.....	154
9.3.1 Format Instruksi 8086	154
9.3.2 Alokasi Bit.....	157
9.3.3 Instruksi Panjang Variabel.....	158
DAFTAR PUSTAKA.....	159
BAB 10 CPU	161
10.1 Apa Itu CPU?	161
10.2 Kegunaan CPU pada komputer.....	161
10.3 Hal yang diketahui tentang Microprocessor	162
10.4 Hukum dalam Peningkatan Performa sebuah Sistem... 163	
10.5 Cara Mengukur Performa Komputer	163
10.6 Rangkaian Instruksi.....	164
10.7 Elemen instruksi mesin.....	164
10.8 Perwujudan Instruksi.....	165
10.9 Tipe Instruksi	166
10.10 Rancangan rangkaian instruksi	168
10.11 Jenis Operan	169

10.12 Angka	170
10.13 Karakter.....	170
10.14 Jenis – Jenis Operasi pada CPU	171
10.15 Transfer data	176
DAFTAR PUSTAKA.....	177
BAB 11 REDUCE INSTRUCTION SET	
COMPUTER (RISC)	179
11.1 Pendahuluan.....	179
11.2 Evolusi RISC	180
11.2.1 Kelebihan dan Kekurangan RISC.....	182
11.2.2 Konsep RISC	182
11.3 RISC vs CISC.....	184
11.3.1 Definisi CISC.....	185
11.3.2 Evolusi CISC	186
DAFTAR PUSTAKA.....	189
BAB 12 PROSES PARALEL DAN SUPERSCALAR	
DALAM OPERASIONAL KOMPUTER	191
12.1 Proses Paralel.....	191
12.2 Teknik Pemrosesan Paralel	193
12.3 Kepentingan Proses Paralel	194
12.4 Superscalar	195
12.5 Proses Paralel dan Superscalar.....	199
12.6 Proses Paralel dan Superscalar dalam Operasional Komputer	200
12.7 Proses Paralel dan Superscalar dalam Peningkatan Kinerja Komputer	201
DAFTAR PUSTAKA.....	204
BIODATA PENULIS	

DAFTAR GAMBAR

Gambar 1.1 : Komponen-komponen Sistem Komputer.....	2
Gambar 1.2 : Sistem Komputer.....	3
Gambar 1.3 : Sistem Operasi Windows.....	15
Gambar 2.1 : Merupakan Bentuk Dari Sempoa.....	26
Gambar 2.2 : Merupakan Bentuk Dari Napier's Bone	27
Gambar 2.3 : Merupakan Bentuk Dari Pascaline.....	27
Gambar 2.4 : Tampilan Dari Stepped Reckoner Atau Leibnitz Wheel.....	29
Gambar 2.5 : Bentuk Dari Difference Engine.....	30
Gambar 2.6 : Tampilan Dari Analytical Engine.....	31
Gambar 2.7 : Tampilan Dari Mesin Tabulasi.....	32
Gambar 2.8 : Tampilan Dari Differential Analyzer	33
Gambar 2.9 : Tampilan Dari Mark I.....	34
Gambar 2.10 : Tampilan Dari Colossus.....	36
Gambar 2.11 : Tampilan Dari Komputer Generasi Kedua.....	38
Gambar 2.12 : Komputer Generasi Ketiga	40
Gambar 2.13 : Komputer Generasi Keempat	43
Gambar 2.14 : Tampilan Dari Komputer Generasi Kelima.....	45
Gambar 3.1 : Modul Komputer	48
Gambar 3.2 : Skema Interkoneksi Bus.....	51
Gambar 3.3 : Synchronous bus	55
Gambar 3.4 : Asynchronous bus	56
Gambar 3.5 : Protokol <i>handshaking</i> asinkron terdiri dari tujuh langkah untuk membaca kata dari memori dan menerimanya dalam perangkat I/O.....	57
Gambar 4.1 : Jenis Memori Internal	63

Gambar 4.2 : ROM BIOS	64
Gambar 4.3 : Chip PROM Texas Instrumen TBP18SA030N	65
Gambar 4.4 : Chip EPROM Intel C1702A.....	66
Gambar 4.5 : Chip EEPROM STMicro M24C02	67
Gambar 4.6 : Jenis RAM.....	68
Gambar 4.7 : Letak Memori <i>Cache</i>	73
Gambar 5.1 : Floppy Disk.....	83
Gambar 5.2 : Flashdisk	83
Gambar 5.3 : <i>Harddisk</i>.....	84
Gambar 5.4 : Solid-State Drive	85
Gambar 5.5 : Compact Disk.....	86
Gambar 5.6 : Digital Versatile Disk	87
Gambar 5.7 : Memory Card.....	87
Gambar 5.8 : Cloud Storage.....	88
Gambar 6.1 : <i>Keyboard</i> (Papan Ketik).....	94
Gambar 6.2 : <i>Keyboard</i> (Papan Ketik).....	95
Gambar 6.3 : <i>Scanner</i> (Pemindai).....	95
Gambar 6.4 : <i>Webcam</i> (Kamera Web)	96
Gambar 6.5 : <i>Webcam</i> (Tuas Kendali)	96
Gambar 6.6 : <i>Touchpad</i> (Bantalan Sentuh)	97
Gambar 6.7 : <i>Flashdisk</i> (Diska Lepas USB)	97
Gambar 6.8 : <i>CD-ROM</i> (Cakram Optik Digital)	98
Gambar 6.9 : Monitor	99
Gambar 6.10 : Printer.....	99
Gambar 6.11 : Plotter	100
Gambar 6.12 : Speaker	100
Gambar 6.13 : Projector.....	101
Gambar 6.14 : Headphones	101
Gambar 7.1 : Penjadwalan Thread Pada Windows XP	113
Gambar 7.2 : Penjadwalan Thread Pada Windows Vista	113
Gambar 8.1 : Organisasi komputer menunjukkan lima	

komponen klasik: input, output, memori, datapath dan control.....	120
Gambar 8.2 : Input dan output dari ALU	122
Gambar 8.3 Jenis-jenis aritmetika komputer	123
Gambar 8.4 : Penjumlahan dalam Komplemen Dua	132
Gambar 8.5 : Diagram Blok Hardware untuk Penjumlahan.....	133
Gambar 8.6 : Diagram Blok Hardware untuk Pengurangan.....	134
Gambar 8.7 : Pengurangan dalam Komplemen Dua	135
Gambar 8.8 : Diagram Alir Algoritma Booth untuk Perkalian dalam Komplemen Dua	137
Gambar 8.9 : Algoritma Booth untuk Pembagian Tidak Bertanda (+/+, -/-).....	141
Gambar 8.10 : Format floating point presisi tunggal 32 bit	143
Gambar 9.1 : Operasi stack push dan pop	153
Gambar 9.2 : Contoh Format Instruksi 8086.....	155
Gambar 9.3 : Format Instruksi PDP-10.....	158
Gambar 11.1 : Arsitektur RISC.....	182
Gambar 12.1 : Multiprocessing.....	195
Gambar 12.2 : Prosesor superscalar secara umum.....	199
Gambar 12.3 : Aliran informasi dalam prosesor superscalar.....	200
Gambar 12.3 : Pentium-4.....	202
Gambar 12.4 : Operasi komputer paralel dengan 4 CPU.....	202

DAFTAR TABEL

Tabel 4.1 : Perbedaan Tipe Memori	71
Tabel 4.2 : Perbandingan SIMM dan DIMM.....	77
Tabel 7.1 : Faktor Kinerja dan Atribut Sistem	110
Tabel 9.1 : S-Bit Bersama Dengan W-Bit	156
Tabel 10.1 : instruksi Alamat ke-tiga.....	170
Tabel 10.2 : Instruksi Alamat ke dua.....	170
Tabel 11.1 : Konsep <i>Reduce Instruction Set</i> <i>Computer</i> (RISC).....	185

BAB 1

JENIS-JENIS OPERASI KOMPUTER

Oleh Romindo

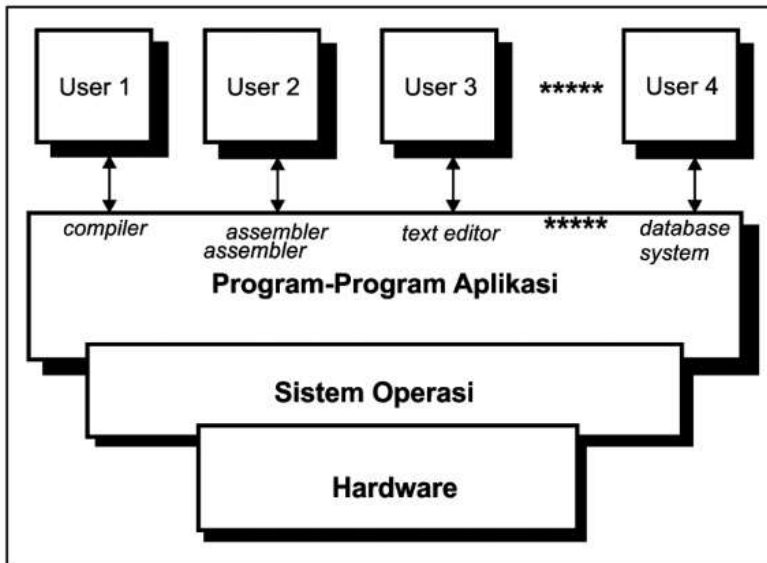
1.1 Pendahuluan

Sistem operasi memainkan peran yang sangat penting dalam sistem komputer. Sistem operasi bertindak sebagai perantara antara perangkat keras dan program aplikasi, memungkinkan program untuk berinteraksi dengan hardware dengan cara yang efisien dan terkoordinasi.

Selain itu, sistem operasi juga mengatur dan mengelola sumber daya komputer, seperti memori dan CPU, serta mengatur akses ke perangkat I/O seperti printer, keyboard, dan mouse. Sistem operasi juga menyediakan lingkungan yang aman dan stabil untuk program yang berjalan pada sistem (Rizal, *et al.*, 2023).

Program aplikasi sendiri merupakan perangkat lunak yang dirancang untuk melakukan tugas-tugas tertentu, seperti pengolahan kata, pengolahan gambar, atau permainan. Compiler dan basis data juga merupakan program aplikasi yang sangat penting.

Dalam suatu sistem komputer, pengguna memanfaatkan program aplikasi untuk menyelesaikan masalah atau tugas tertentu. Oleh karena itu, perangkat keras, sistem operasi, dan program aplikasi bekerja sama untuk memungkinkan pengguna memanfaatkan sumber daya komputer secara efektif dan efisien.



Gambar 1.1 : Komponen-komponen Sistem Komputer
(Rizal, et al., 2023)

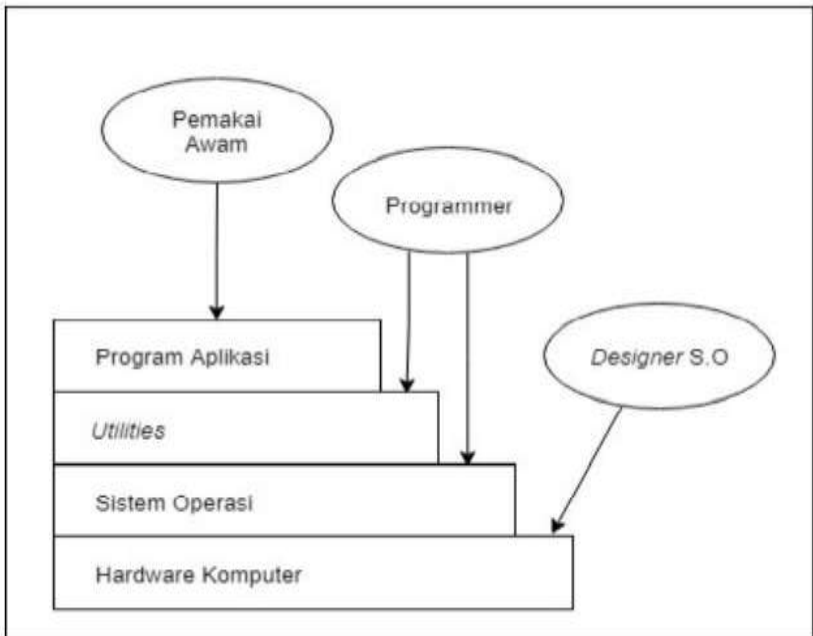
Sistem operasi bertindak sebagai antarmuka antara pengguna dan hardware komputer. Pengguna tidak perlu mengetahui secara detail tentang hardware komputer, karena sistem operasi menyediakan lingkungan yang familiar dan mudah digunakan untuk interaksi dengan perangkat keras.

Aplikasi dikembangkan dengan menggunakan satu atau beberapa bahasa pemrograman dan alat-alat pemrograman lainnya. Beberapa bahasa pemrograman populer adalah C++, Java, Python, dan JavaScript. Selain itu, terdapat berbagai alat pemrograman seperti editor kode, debugger, dan pengelola versi, yang membantu para pengembang membangun aplikasi dengan lebih mudah.

Sistem operasi sendiri dikembangkan oleh para desainer sistem operasi yang memiliki pengetahuan yang mendalam tentang cara kerja hardware komputer. Mereka menggunakan

bahasa pemrograman dan alat pemrograman untuk membuat sistem operasi yang efisien dan dapat mengelola sumber daya komputer dengan baik.

Sedangkan, utility adalah program aplikasi yang digunakan untuk melakukan tugas-tugas tertentu yang berhubungan dengan pengelolaan sistem, seperti pengarsipan, backup, atau pemantauan performa. Utility digunakan oleh para administrator sistem atau pengguna yang memiliki kebutuhan khusus dalam mengelola sistem operasi atau perangkat keras.



Gambar 1.2 : Sistem Komputer
(Rizal, et al., 2023)

Menurut Tingkatan dan Pandangan Pengertian system operasi ditinjau dari 3 sudut pandang yang berbeda :

1. Sudut Pandang Pengguna

Desain sistem operasi harus mempertimbangkan sejumlah faktor dalam memberikan keseimbangan antara kemudahan penggunaan dan kinerja sistem. Memang, kemudahan penggunaan penting bagi sistem operasi karena dapat mempercepat produktivitas pengguna dalam menggunakan sistem operasi dan menjalankan program aplikasi. Sebaliknya, kinerja dan utilitas sumber daya juga sangat penting untuk menjalankan tugas-tugas tertentu pada sistem komputer dengan cepat dan efisien.

Dalam lingkungan multiuser, sistem operasi harus dirancang untuk memaksimalkan penggunaan sumber daya komputer. Sistem operasi harus dapat mengatur dan mengelola sumber daya secara efisien dan adil antara pengguna yang berbeda agar sumber daya dapat dimanfaatkan secara maksimal. Dalam lingkungan multiuser, kemudahan penggunaan masih penting, tetapi harus diimbangi dengan kebutuhan untuk memastikan bahwa sumber daya komputer digunakan secara optimal.

2. Sudut Pandang Sistem

Sistem operasi bertanggung jawab untuk mengatur dan mengelola sumber daya komputer secara efisien, termasuk mengalokasikan sumber daya, menangani konflik permintaan sumber daya, dan memprioritaskan tugas-tugas yang berjalan pada sistem komputer. Sistem operasi juga berfungsi untuk mengatur eksekusi program aplikasi dan operasi Input/Output (I/O) pada perangkat keras.

Program pengendali atau kernel merupakan inti dari sistem operasi, yang menjalankan tugas-tugas penting seperti mengelola sumber daya, menangani interupsi perangkat keras, menjalankan proses, dan memfasilitasi komunikasi antara program aplikasi dan perangkat keras. Kernel

biasanya berjalan setiap saat dan bertanggung jawab untuk memastikan bahwa sistem operasi berjalan dengan lancar dan efisien. Karena kernel merupakan bagian terpenting dari sistem operasi, ia seringkali ditulis dalam bahasa mesin atau bahasa pemrograman tingkat rendah lainnya agar dapat berjalan dengan kecepatan tinggi dan memanfaatkan sumber daya secara efisien.

3. Sudut Pandang Tujuan

salah satu fungsi utama dari sistem operasi adalah untuk memberikan kemudahan dan kenyamanan dalam penggunaan komputer. Dengan adanya sistem operasi, pengguna dapat menjalankan program aplikasi dan menyelesaikan masalah penggunaan dengan lebih mudah dan efisien. Sistem operasi memberikan antarmuka yang user-friendly atau ramah pengguna yang memudahkan pengguna dalam mengakses sumber daya dan fungsionalitas komputer. Selain itu, sistem operasi juga menyediakan berbagai layanan dan utilitas yang membantu pengguna dalam mengatur, memantau, dan mengoptimalkan kinerja sistem komputer.

Empat Komponen Manajemen Utama pada Sistem Operasi :

1. Manajemen Proses

proses adalah sebuah program yang sedang dieksekusi dan membutuhkan beberapa sumber daya untuk menyelesaikan tugasnya. Sumber daya tersebut dapat berupa CPU time, memori, berkas-berkas, dan perangkat-perangkat masukan/keluaran (I/O).

Sistem operasi bertanggung jawab untuk mengalokasikan sumber daya-sumber daya tersebut ketika proses tersebut diciptakan atau sedang dijalankan. Sistem operasi harus memastikan bahwa sumber daya yang digunakan oleh proses

tidak tumpang tindih dan tidak mengalami konflik, sehingga proses dapat berjalan dengan aman dan efisien.

Ketika proses tersebut berhenti dijalankan, baik karena telah selesai atau karena terjadi kesalahan, sistem operasi akan mengembalikan semua sumber daya yang digunakan oleh proses tersebut ke sistem operasi, sehingga sumber daya tersebut bisa digunakan kembali oleh proses-proses lainnya. Proses ini dikenal sebagai dealokasi atau penghapusan sumber daya. sistem operasi bertanggung jawab atas manajemen proses, termasuk di antaranya:

- a. Membuat dan menghapus proses pengguna dan sistem proses: Sistem operasi bertanggung jawab untuk membuat proses baru ketika sebuah program atau aplikasi dijalankan, serta menghapus proses yang telah selesai atau dihentikan. Proses sistem juga dikelola oleh sistem operasi untuk menjalankan fungsi-fungsi sistem seperti manajemen memori, manajemen file, dan lain-lain.
- b. Menunda atau melanjutkan proses: Sistem operasi dapat menunda atau menghentikan sementara suatu proses, misalnya ketika ada proses lain yang membutuhkan sumber daya yang sama. Proses dapat dilanjutkan kembali setelah sumber daya tersebut tersedia lagi.
- c. Menyediakan mekanisme untuk proses sinkronisasi: Sistem operasi harus memastikan bahwa beberapa proses yang berjalan bersamaan tidak mengalami konflik dalam menggunakan sumber daya yang sama. Untuk itu, sistem operasi menyediakan mekanisme untuk sinkronisasi antar proses, seperti semafor, monitor, dan lain-lain.
- d. Menyediakan mekanisme untuk proses komunikasi: Proses dalam sistem operasi seringkali harus saling berkomunikasi, misalnya untuk bertukar data atau

melakukan koordinasi tugas. Sistem operasi menyediakan mekanisme untuk komunikasi antar proses, seperti shared memory, message passing, dan lain-lain.

- e. Menyediakan mekanisme untuk penanganan deadlock: Deadlock terjadi ketika beberapa proses saling menunggu satu sama lain untuk melepaskan sumber daya yang mereka butuhkan. Sistem operasi harus dapat mendeteksi deadlock dan melakukan tindakan untuk mengatasinya, seperti melepaskan sumber daya yang salah satu proses butuhkan atau menghentikan salah satu proses yang terlibat dalam *deadlock*.

2. Manajemen Memori

Memori utama atau yang juga dikenal sebagai RAM (*Random Access Memory*) adalah tempat penyimpanan sementara data dan program yang sedang diakses oleh CPU (*Central Processing Unit*) dan perangkat input/output. Memori utama juga terdiri dari sejumlah sel atau lokasi memori yang masing-masing memiliki alamat unik untuk mengakses data dan instruksi yang disimpan di dalamnya.

Memori utama memiliki kecepatan akses yang sangat tinggi dan ukurannya biasanya lebih kecil dibandingkan dengan hard disk. Karena bersifat volatile, data yang tersimpan di dalam memori utama akan hilang saat komputer dimatikan atau kehilangan daya. Oleh karena itu, memori utama hanya digunakan sebagai tempat penyimpanan sementara, sedangkan hard disk atau media penyimpanan lainnya digunakan untuk menyimpan data secara permanen. Sistem operasi juga bertanggung jawab atas manajemen memori yang meliputi:

- a. Memetakan program ke dalam memori: Sistem operasi menempatkan program ke dalam memori saat program tersebut dijalankan dan mengalokasikan memori sesuai dengan kebutuhan program.

- b. Memastikan tidak ada tumpang tindih: Sistem operasi memastikan bahwa program tidak menimpa atau menumpuk pada area memori yang sama. Hal ini mencegah kesalahan yang mungkin terjadi jika program mengakses area memori yang tidak sah.
- c. Mengalokasikan dan membebaskan memori: Sistem operasi mengalokasikan dan membebaskan memori ketika program membutuhkan memori baru atau tidak lagi memerlukan memori yang sudah dialokasikan.
- d. Memastikan keamanan memori: Sistem operasi memastikan bahwa program tidak mengakses memori yang tidak sah atau memori yang sudah dialokasikan oleh program lain. Hal ini mencegah program dari mengakses data rahasia atau merusak data yang lain.
- e. Paging atau segmentasi memori: Sistem operasi menggunakan teknik paging atau segmentasi untuk mengatur bagaimana program mengakses memori, yang memungkinkan program untuk memanfaatkan memori secara lebih efisien dan menghindari tumpang tindih.

3. Manajemen Sistem Berkas

Sistem operasi mengelola berkas dengan menggunakan media penyimpanan massa seperti disk, tape, flash drive, dan lain sebagainya. Sistem operasi mengatur cara penyimpanan dan pengambilan data dari media tersebut, dan memberikan akses ke berkas-berkas tersebut kepada aplikasi dan pengguna. Sistem operasi juga bertanggung jawab atas manajemen hak akses ke berkas, seperti siapa yang dapat membaca, menulis, atau mengeksekusi berkas. Selain itu, sistem operasi juga menyediakan utilitas untuk melakukan manajemen berkas, seperti membuat, menghapus, dan mengganti nama berkas dan direktori. Sistem operasi bertanggung jawab atas manajemen berkas, antara lain:

- a. Membuat dan menghapus berkas - Sistem operasi menyediakan perintah untuk membuat dan menghapus berkas dari media penyimpanan, serta menentukan aksesibilitas dan hak akses pada berkas tersebut.
 - b. Membuat dan menghapus direktori - Sistem operasi juga menyediakan perintah untuk membuat dan menghapus direktori pada media penyimpanan, serta menentukan hak akses pada direktori tersebut.
 - c. Mendukung manipulasi berkas dan direktori - Sistem operasi menyediakan perintah untuk mengubah nama, memindahkan, menyalin, dan memodifikasi berkas dan direktori.
 - d. Memetakan berkas ke secondary-storage - Sistem operasi bertanggung jawab dalam menentukan bagaimana berkas dapat diakses pada media penyimpanan sekunder, seperti disk atau flash drive.
 - e. Membuat backup berkas - Sistem operasi juga dapat melakukan backup pada berkas ke media penyimpanan yang permanen (non-volatile), seperti CD, DVD, atau media penyimpanan online. Hal ini penting untuk melindungi berkas dari kehilangan atau kerusakan.
4. Manajemen Input/Output

Driver merupakan program atau modul yang memungkinkan komunikasi antara sistem operasi dan perangkat keras seperti keyboard, mouse, printer, disk drive, dan perangkat I/O lainnya. Device manager adalah komponen sistem operasi yang memungkinkan pengguna untuk melihat dan mengelola perangkat keras yang terhubung ke komputer. Dalam device manager, pengguna dapat melihat informasi tentang perangkat keras yang terinstal, termasuk driver yang terkait dengan perangkat tersebut, dan melakukan berbagai tindakan seperti menghapus, memperbarui, atau menonaktifkan perangkat.

Berikut adalah penjelasan singkat mengenai komponen Sistem Operasi untuk sistem Masukan/Keluaran:

- a. **Penyangga:** digunakan untuk menampung data sementara dari atau ke perangkat Masukan/Keluaran sehingga dapat dioptimalkan proses transfer datanya. Penyangga ini berfungsi untuk mengurangi waktu penundaan dalam operasi Masukan/Keluaran dengan menampung data sementara sebelum atau setelah diproses oleh perangkat Masukan/Keluaran.
- b. **Spooling:** kependekan dari Simultaneous Peripheral Operations On-Line, yang merupakan sebuah teknik untuk menunda pengolahan data Masukan/Keluaran pada perangkat keras dengan menyimpannya dalam antrian. Spooling berfungsi untuk mengoptimalkan penggunaan perangkat Masukan/Keluaran, karena antrian akan mengatur aliran data sehingga perangkat Masukan/Keluaran dapat dipakai secara lebih efisien.
- c. **Driver:** adalah program yang menyediakan antarmuka antara perangkat keras Masukan/Keluaran dengan sistem operasi. Driver ini berfungsi untuk mengaktifkan fungsi dan kemampuan perangkat keras sehingga dapat diakses dan digunakan oleh sistem operasi dan program aplikasi.

1.2 Peranan dan Kategori Sistem Operasi

Sistem operasi bertindak sebagai pengatur dan manajer dari sumber daya yang tersedia pada sebuah komputer, seperti CPU time, memory space, file storage, I/O device, dan lain-lain. Sebagai pengatur, sistem operasi menentukan alokasi sumber daya bagi program-program dan pengguna agar berjalan dengan efisien dan adil, serta mencegah penggunaan sumber daya yang tidak tepat. Selain itu, sistem operasi juga bertindak sebagai pengontrol terhadap perangkat I/O dan program-program pengguna untuk

mencegah terjadinya error dan masalah dalam penggunaan komputer (Romindo, et al., 2023).

Sistem Operasi dapat dikategorikan:

1. *Single User – Single Tasking* (SU-ST)
Satu komputer hanya bisa digunakan oleh satu user dan hanya bisa menjalankan satu program di satu waktu. contoh: DOS (Disk Operating System).
2. *Multi User – Single Tasking* (MU-ST)
Satu komputer dapat digunakan oleh banyak user, namun tiap user hanya bisa menjalankan 1 program (aplikasi) di satu waktu. contoh: Novell Netware (3.x, 4.x)
3. *Single User – Multi Tasking* (SU-MT)
Satu komputer dipakai oleh satu user dan dapat menjalankan banyak program disatu waktu. (Tampilan Desktop GUI). contoh: Windows, MacOS, Linux, Symbian, dan lain-lain.
4. *Multi User – Multi Tasking* (MU-MT)
Satu komputer dipakai bersamaan oleh banyak user yang dapat menjalankan banyak program di satu waktu. contoh: Unix, Linux, FreeBSD, SunSolaris.

1.3 Jenis-Jenis Sistem Operasi Komputer

Seperti yang kita ketahui bahwa Sistem operasi komputer adalah perangkat lunak yang mengontrol dan mengkoordinasikan semua sumber daya perangkat keras pada komputer dan menyediakan lingkungan bagi aplikasi dan pengguna untuk berinteraksi dengan perangkat tersebut (Stalling, 2012).

1. *DOS (Disk Operating System)*
DOS, atau Disk Operating System, adalah sistem operasi yang dikembangkan untuk digunakan pada media penyimpan disk seperti disket atau harddisk. DOS termasuk dalam golongan sistem operasi, yang berfungsi sebagai pengatur dan pengelola komputer, serta memberikan antarmuka bagi pengguna untuk

berinteraksi dengan perangkat keras dan perangkat lunak komputer.

Sistem operasi PC DOS, atau Personal Computer Disk Operating System, khususnya ditujukan untuk digunakan pada mikrokomputer IBM, atau IBM PC. Awalnya, pada tahun 1980, tim Pettersen membuat sebuah sistem operasi bernama QDOS (*Quick and Dirty Operating System*) untuk Seattle Computer Product, yang membutuhkan S-100 system dengan menggunakan Intel 8086. Kemudian, pada akhir tahun 1980, QDOS dikembangkan lebih lanjut dan diberi nama 86-DOS, yang khusus dikembangkan untuk digunakan pada arsitektur 8086.

DOS memiliki keunggulan dalam penggunaan sumber daya yang ringan dan cepat, serta mudah dipelajari dan digunakan oleh pengguna awam. Namun, sistem operasi ini memiliki kelemahan dalam keterbatasan fungsionalitas, khususnya dalam hal dukungan grafis dan kemampuan jaringan. Seiring dengan perkembangan teknologi, DOS pun kini telah digantikan oleh sistem operasi yang lebih canggih dan modern.

Beberapa produk DOS sampai saat ini adalah :

- a. DOS 1.0 adalah versi DOS paling awal yang muncul bersamaan dengan kemunculan IBM PC. Versi ini sangat sederhana dan terbatas dalam kemampuannya sehingga tidak dapat lagi mendukung perkembangan software yang ada.
- b. DOS 2.0 dikeluarkan pada bulan Maret 1983 bersamaan dengan munculnya IBM PC/XT. Meskipun versi ini memiliki beberapa kesalahan program, tetapi kemudian diperbaiki dengan versi 2.10.
- c. DOS 3.0 adalah PC-DOS 3.0 dan MSDOS 3.05 yang dirilis untuk mendukung komputer baru saat itu yang disebut *Advanced Technology* (AT). Versi ini mendukung penggunaan disk drive dengan kapasitas 1,2 megabyte dan penggunaan RAMDisk (VDISK) dengan menggunakan

extended memory yang tidak dapat langsung digunakan oleh DOS versi sebelumnya.

- d. DOS 3.1 diluncurkan pada bulan Maret 1985 dan memiliki dukungan terhadap sistem jaringan komputer.
- e. DOS 3.2 adalah PC DOS 3.2 yang dirilis bersamaan dengan peluncuran komputer laptop. Versi ini mampu memanfaatkan disk drive dengan ukuran 3,5 inci dan kapasitas 720 kilobyte yang biasanya digunakan pada komputer laptop.
- f. DOS 3.3 diluncurkan oleh IBM pada tahun 1987. Kemampuan yang ditambahkan pada DOS 3.3 adalah kemampuan untuk menggunakan harddisk hingga 32 megabyte per logical drive.
- g. DOS 4.0 memiliki tiga versi, yaitu PC-DOS 4.0, PC-DOS 4.01, dan MS-DOS 4.01. PC DOS 4.0 merupakan versi pertama yang dirilis pada bulan Juli 1988. Versi ini memiliki banyak kesalahan sehingga kemudian dirilis PC-DOS 4.01. Versi DOS 4.0 memiliki fasilitas Shell dari DOS untuk mengontrol kerja komputer dengan system menu.
- h. DOS 5.0 diluncurkan pada tahun 1991 dengan fasilitas baru seperti MIRROR, UNDELETTE, dan UNFORMAT.
- i. DOS 6.0 diluncurkan pada bulan Maret 1993. Versi ini ditambahkan beberapa utilitas baru dan perintah-perintah konfigurasi sistem dan batch file.

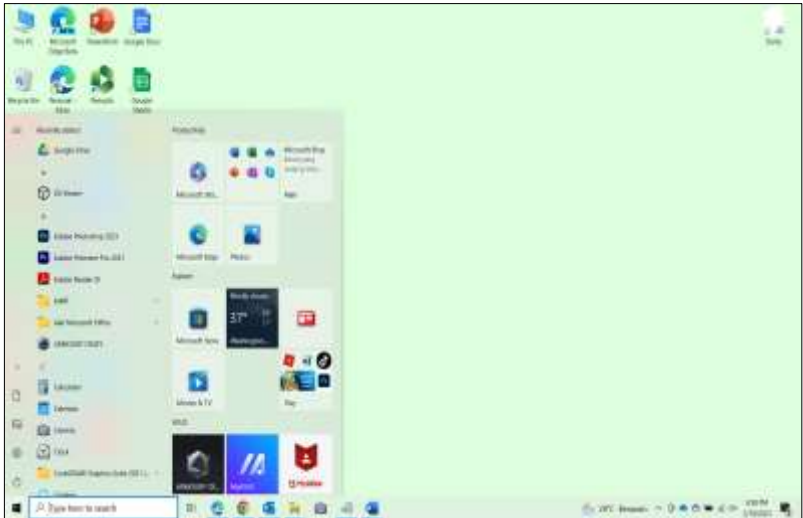
MS-DOS memiliki kelebihan dan kekurangan. Kelebihan MS-DOS adalah ukurannya yang kecil dan ringan sehingga dapat dijalankan di berbagai perangkat dengan spesifikasi yang rendah. Selain itu, karena MS-DOS berbasis teks, pengguna dapat memiliki kontrol penuh terhadap sistem dan memperoleh fleksibilitas yang lebih besar dalam mengatur sistem operasi. Kelebihan lainnya termasuk kemampuan untuk memboot komputer dengan cepat dan mengakses sistem operasi dengan

lebih cepat dibandingkan dengan sistem operasi lainnya pada saat itu.

Namun, MS-DOS juga memiliki beberapa kekurangan. Salah satunya adalah kurangnya antarmuka grafis (GUI) yang user-friendly, sehingga pengguna harus mengingat perintah-perintah teks dan formatnya untuk dapat mengoperasikan sistem operasi dengan efektif. Selain itu, pengguna juga membutuhkan pengetahuan teknis yang cukup untuk dapat memperbaiki masalah sistem operasi yang muncul. Selain itu, MS-DOS juga memiliki keterbatasan dalam mendukung aplikasi dan software yang mengutamakan grafis, sehingga kurang cocok untuk penggunaan sehari-hari yang membutuhkan interaksi antarmuka grafis yang lebih mudah dipahami.

2. Sistem Operasi Windows

Windows merupakan salah satu sistem operasi yang paling banyak digunakan di seluruh dunia. Windows dikembangkan oleh *Microsoft Corporation* dan pertama kali dirilis pada tahun 1985 sebagai pengganti dari MS-DOS yang berbasis teks. Windows menggunakan antarmuka berbasis GUI (*Graphical User Interface*) yang memungkinkan pengguna untuk berinteraksi dengan sistem operasi menggunakan mouse dan ikon grafis. Sejak awal pengembangannya, Windows telah mengalami banyak perubahan dan saat ini tersedia dalam berbagai versi seperti Windows 10, Windows 8, Windows 7, dan sebagainya. Windows juga mendukung berbagai macam aplikasi dan perangkat keras, sehingga banyak digunakan baik untuk keperluan pribadi maupun bisnis.



Gambar 1.3 : Sistem Operasi Windows
(Romindo, et al., 2023)

Selain harganya yang mahal, Windows juga memiliki beberapa kekurangan lainnya, di antaranya:

- a. Rentan terhadap serangan virus dan malware: Karena popularitasnya yang tinggi, Windows menjadi target empuk bagi para penjahat siber yang mencoba mencuri data atau merusak sistem pengguna. Selain itu, beberapa fitur keamanan bawaan Windows juga dianggap kurang memadai, sehingga pengguna harus memasang software antivirus tambahan.
- b. Sering mengalami kesalahan dan crash: Windows kadang-kadang mengalami kesalahan atau crash yang dapat merusak data atau membuat pengguna kehilangan pekerjaannya yang belum disimpan. Hal ini bisa terjadi karena berbagai faktor, termasuk program yang buruk atau tidak kompatibel, driver yang rusak, atau masalah dengan hardware.

- c. Terlalu banyak bloatware: Windows sering kali dianggap memiliki terlalu banyak program bawaan atau bloatware yang tidak terlalu berguna bagi pengguna. Ini tidak hanya memakan ruang penyimpanan yang berharga, tetapi juga dapat membuat sistem lebih lambat dan rentan terhadap masalah.
 - d. Masalah privasi: Beberapa orang khawatir tentang privasi mereka ketika menggunakan Windows, karena sistem operasi ini sering kali mengumpulkan data pengguna dan mengirimkannya ke Microsoft. Meskipun Microsoft telah memperbaiki kebijakan privasinya dalam beberapa tahun terakhir, masih ada beberapa kekhawatiran tentang penggunaan data pengguna.
3. Sistem Operasi Linux

Linux adalah sistem operasi yang bersifat open source atau sumber terbuka yang memungkinkan penggunanya untuk mengakses dan mengubah kode programnya. Linux sendiri dikembangkan oleh Linus Torvalds pada tahun 1991 dan diperkenalkan ke publik pada tahun 1994. Sistem operasi ini banyak digunakan oleh para pengembang dan administrator jaringan karena kestabilannya dan kemampuannya dalam menjalankan server.

Salah satu kelebihan utama Linux adalah kemampuan untuk diadaptasi ke berbagai macam lingkungan sistem, mulai dari perangkat keras yang kecil hingga jaringan besar. Selain itu, Linux juga tidak memerlukan spesifikasi komputer yang tinggi untuk menjalankannya, sehingga banyak dijadikan pilihan untuk penggunaan pada komputer lama atau server.

Kelebihan lainnya dari Linux adalah banyaknya aplikasi dan perangkat lunak open source yang tersedia secara gratis, serta tersedianya berbagai macam distro atau distribusi Linux yang dapat disesuaikan dengan kebutuhan pengguna. Namun, kelemahan dari Linux adalah kurangnya dukungan hardware

dari produsen terkait driver dan software khusus yang tidak tersedia di Linux. Selain itu, struktur direktori di Linux yang kompleks juga membuat pengguna awam harus lebih banyak belajar dan beradaptasi.

4. Sistem Operasi MACINTOSH

Macintosh adalah salah satu produk komputer pribadi yang dikembangkan oleh perusahaan Apple Inc. pada tahun 1984. Macintosh sering disebut juga sebagai Mac atau Macintosh computer, dan menjalankan sistem operasi macOS yang dikembangkan oleh Apple. Macintosh awalnya dikenal dengan desainnya yang unik dan berbeda dari komputer pribadi pada umumnya. Beberapa fitur yang membedakan Macintosh dari komputer pribadi lainnya adalah tampilan antarmuka pengguna grafis (GUI) yang menarik dan mudah digunakan, dukungan terhadap font dan grafik yang lebih baik, dan penggunaan mouse sebagai alat pengendali. Saat ini, Macintosh masih diproduksi dan dikenal sebagai salah satu merek komputer pribadi yang populer, khususnya di kalangan pengguna yang membutuhkan perangkat dengan performa dan desain yang lebih khusus, seperti para profesional kreatif.

5. OS/2

OS/2 merupakan sistem operasi yang dikembangkan oleh IBM dan Microsoft untuk digunakan pada komputer IBM PS/2 sebagai pengganti sistem operasi DOS. OS/2 didesain untuk memanfaatkan kemampuan penuh dari mikroprosesor Intel 80286, termasuk modus terproteksi, multitasking, dan memori virtual. OS/2 juga memiliki kemampuan untuk membaca semua cakram yang diformat menggunakan format DOS dan mendukung banyak perangkat lunak MS-DOS yang beredar saat itu.

Kelebihan dari OS/2 adalah kemampuan multitasking-nya yang sangat baik dan kompatibilitas dengan perangkat lunak DOS. Selain itu, OS/2 juga memiliki kemampuan memori virtual yang

baik, sehingga dapat mengelola penggunaan memori dengan lebih efisien. OS/2 juga memiliki sistem file yang terstruktur dengan baik, serta dukungan untuk beberapa protokol jaringan yang populer saat itu.

Namun, kelemahan OS/2 adalah kurangnya dukungan untuk aplikasi Windows. Meskipun Microsoft berpartisipasi dalam pengembangan OS/2, Microsoft akhirnya beralih ke pengembangan Windows, sehingga OS/2 kehilangan dukungan dari Microsoft. Hal ini menyebabkan pengembangan aplikasi Windows menjadi lebih populer, sementara aplikasi OS/2 menjadi semakin langka. Akhirnya, IBM menghentikan dukungan untuk OS/2 pada tahun 2006.

6. Unix

Sistem operasi Unix adalah sistem operasi yang sangat populer dalam lingkungan bisnis dan industri, terutama sebagai sistem operasi server. Unix dikembangkan oleh Bell Labs pada tahun 1960-an dan 1970-an. Unix memiliki beberapa kelebihan utama, salah satunya adalah kemampuan untuk digunakan sebagai sistem operasi multiuser yang memungkinkan beberapa pengguna untuk menggunakan sistem secara bersamaan.

Selain itu, Unix juga memiliki sistem file yang hirarkial dan mudah diakses, yang membuatnya lebih mudah untuk mengakses informasi atau data. Karena Unix digunakan secara luas dalam lingkungan server, maka Unix sangat stabil jika digunakan untuk aplikasi seperti database, file server, server internet, intranet, dan internet-client pengembangan java. Unix juga dianggap lebih baik dari Windows NT dalam hal stabilitas dan keamanan.

Namun, seperti kebanyakan sistem operasi lainnya, Unix memiliki beberapa kelemahan. Unix tidak memiliki antarmuka pengguna yang menarik dan intuitif, yang membuatnya kurang cocok untuk pengguna yang tidak berpengalaman dalam mengoperasikan sistem operasi. Selain itu, Unix juga

membutuhkan ruang penyimpanan yang lebih besar daripada sistem operasi lainnya, tidak kompatibel dengan banyak game, tidak semua hardware bisa digunakan dengan Unix, dan terkadang sulit dioperasikan oleh pengguna awam.

7. Sistem Operasi Android

Android adalah sistem operasi berbasis Linux yang dirancang untuk perangkat seluler seperti smartphone, tablet, dan smartwatch. Sistem operasi ini dikembangkan oleh Google dan berfokus pada fleksibilitas, keamanan, dan kinerja yang baik. Android menggunakan kernel Linux sebagai basisnya dan memungkinkan para pengembang untuk membuat aplikasi dengan menggunakan bahasa pemrograman Java atau Kotlin. Android juga memiliki toko aplikasi resmi yaitu Google Play Store, yang menyediakan ribuan aplikasi yang dapat diunduh dan digunakan oleh pengguna Android. Seiring waktu, Android telah mengalami banyak perubahan dan peningkatan fitur, termasuk peningkatan keamanan, peningkatan kinerja, dan integrasi dengan teknologi terbaru seperti kecerdasan buatan dan *Internet of Things* (IoT).

DAFTAR PUSTAKA

- Rizal, C. et al., 2023. *ORGANISASI KOMPUTER*. Padang: PT GLOBAL EKSEKUTIF TEKNOLOGI.
- Romindo, R. et al., 2023. *REKAYASA PERANGKAT LUNAK*. Padang: PT GLOBAL EKSEKUTIF TEKNOLOGI.
- Stalling, W., 2012. *Operating System Internals and Design Principles*. 7th ed. s.l.:Prentice Hall.

BAB 2

EVOLUSI PERKEMBANGAN KOMPUTER

Oleh Rini Nuraini

2.1 Pengertian Komputer

Kata "komputer" berasal dari kata Latin "computare" yang berarti menghitung. Sejak zaman dahulu, manusia sudah memiliki alat perhitungan. Komputer merupakan mesin yang dapat dimanfaatkan untuk memproses data berdasarkan perintah dalam program. Awalnya, istilah "komputer" dipakai untuk seseorang yang akan melakukan perhitungan, baik menggunakan peralatan maupun tidak, tetapi istilah ini selanjutnya berganti makna ke mesin tersebut. Awalnya, pemrosesan informasi hampir seluruhnya berkaitan dengan perhitungan matematika, tetapi komputer saat ini digunakan dalam berbagai pekerjaan yang bukan berkaitan dengan matematika.

Dalam arti yang lebih umum komputer dapat memiliki arti yaitu perangkat elektronika yang memiliki komponen-komponen utama kemudian bekerja bersama agar dapat dihasilkan sebuah informasi yang didasari pada program dan data. Konstruksi komputer arsitekturnya dirancang oleh von Neumann yang diajukan oleh John von Neumann pada tahun 1945.

2.2 Awal Penemuan Komputer

Kira-kira 30 tahun setelah mesin kalkulator diciptakan oleh Pascal, adalah Gottfried Wilhelm Leibniz seorang matematikawan Jerman yang memperbaikinya dengan menambahkan sistem bilangan biner yang membuat mesin lebih mudah melakukan

perhitungan dibandingkan sistem desimal. Selain itu Leibniz juga menambahkan roda gigi untuk melakukan perkalian dan pembagian. Selanjutnya seorang yang bernama George Boole memperluas sistem biner untuk matematika dan menerapkannya dalam Aljabar Boolean dan Logika Boolean. Sementara itu, Charles Babbage memainkan peran penting dalam munculnya komputer modern.

Dalam modulnya “Dunia di Ujung Jari” yang diterbitkan oleh Kemdikbud pada tahun 2020 menyatakan bahwa Charles Babbage, seorang matematikawan Inggris, dikenal sebagai penemu komputer pertama melalui karya mesin hitungnya yaitu Difference Engine No. 1. Mesin ini dirancang pada tahun 1820-an dan merupakan mesin penghitung mekanik yang dapat melakukan analisis dan diprogram oleh manusia. Karena hasil karyanya, Babbage diakui sebagai "Bapak Komputer". Seiring berjalannya waktu, perkembangan komputer mengalami perubahan dari bentuk besar menjadi lebih portabel seperti laptop dan memiliki banyak kemampuan untuk digunakan dalam berbagai hal.

2.3 Cikal Bakal Komputer Digital

Modul Tema 16 yang berjudul “Dunia di Ujung Jari” yang diterbitkan oleh Kemdikbud pada tahun 2020 menyebutkan bahwa Alan Turing, seorang matematikawan Inggris, adalah penemu komputer digital pertama pada tahun 1930. Ia membuat mesin yang mampu melakukan serangkaian perintah yang kemudian dikenal dengan nama mesin Turing, termasuk ide-ide simulasi yang dikenal sebagai Uji Turing.

Sementara itu, insinyur mesin asal Jerman, Konrad Zuse, dikenal sebagai pencipta komputer digital pertama yang bisa diprogram. Ia membuat komputer yang dinamakan Z1 sebelum pecahnya Perang Dunia II. Tahun 1936 di Berlin, Zuse membuat mesin dengan plat logam, pin dan mampu melakukan perhitungan penjumlahan dan pengurangan. Walaupun model awal

komputernya hancur ketika Perang Dunia II, namun Zuse masih diakui sebagai orang yang menciptakan komputer digital pertama.

Pada tahun 1943 seorang yang bernama John Mauchly menciptakan mesin yang dikenal sebagai ENIAC yang merupakan kepanjangan dari *Electronic Numerical Integrator and Calculator*. Perangkat tersebut dapat dimanfaatkan untuk membantu Angkatan Darat dalam memprediksi serangan selama terjadinya perang dunia ke-II. ENIAC memiliki kemampuan analisa yang sangat baik dan mampu menyelesaikan ribuan masalah dengan cepat. Meskipun demikian, ENIAC sangat besar dan berat, seberat 30 ton, dan membutuhkan ruangan yang luas, seluas 457-meter persegi, karena banyaknya komponen pendukung seperti 40-unit lemari kabinet, 6.000-unit sakelar serta tabung hampa yang berjumlah 18.000 unit.

2.4 Lahirnya Bahasa Pemrograman

Pada tahun 1954, Grace Hopper yang merupakan seorang ilmuwan komputer menciptakan bahasa pemrograman pertama, yaitu COBOL. Tujuannya adalah memudahkan pengguna komputer dalam memberikan intruksi menggunakan bahasa Inggris, yang mana sebelumnya perintah hanya dapat diberikan dengan menggunakan angka. Seiring perkembangan komputer, bahasa pemrograman pun terus berkembang. Setahun kemudian, tim pemrograman yang dipimpin oleh John Backus dari IBM menciptakan bahasa pemrograman baru, FORTRAN.

Sebagai perusahaan teknologi yang ambisius, IBM bertujuan untuk menjadi pemimpin dunia dalam hal perkembangan komputer. Untuk mewujudkan hal ini, mereka memproduksi sebuah mesin bernama IBM 650 yang akan dijual secara luas.

Pada tahun 1965, komputer mulai diakui sebagai alat yang penting bagi para ahli matematika, insinyur, dan juga masyarakat umum. Contohnya adalah Programma 101, yang lebih kecil dan

ringan daripada ENIAC, ukurannya seukuran mesin tik dengan bobot 29 kg dan telah didukung dengan internal printer.

2.5 Personal Computer (PC)

Personal Computer atau Komputer pribadi lahir pada 1970-an, ini ditandai dengan hadirnya Xerox Alto, yang merupakan jenis komputer yang mampu melakukan tugas-tugas seperti mengirim email dan mencetak dokumen dengan desain seperti komputer modern. Komputer ini sudah dilengkapi dengan keyboard, mouse dan juga layar. Pada tahun yang sama, sejumlah temuan penting juga terjadi seperti pembuatan disket, ethernet, dan chip DRAM (*Dynamic Access Memory*).

Pada tahun berikutnya, tepatnya ditahun 1976, Steve Jobs bersama dengan Steve Wozniak memulai pendirian perusahaan Apple dan menghasilkan sebuah komputer circuit tunggal pertama yang dinamai sebagai Apple I. Selanjutnya IBM meningkatkan desain komputer pribadi dengan memperkenalkan Acorn. Apple I sudah didukung dengan chip Intel, 2 slot disk, keyboard dan juga layar monitor berwarna.

Pada tahun 1983, sejumlah inovasi lain terjadi, termasuk munculnya CD-ROM sebagai media penyimpanan data yang mampu menyimpan hingga 550 MB dan menjadi standar umum untuk komputer.

Selanjutnya masih ditahun 1983, Microsoft meluncurkan program Word dan Apple meluncurkan komputer Macintosh. Macintosh terkenal sebagai komputer pertama yang bisa dikontrol menggunakan mouse dan memiliki antarmuka pengguna grafis. Microsoft juga meluncurkan sistem operasi Windows dengan fitur multi-tasking dan antarmuka grafis. Kemudian, Apple memperkenalkan Powerbook, yang merupakan seri dari sebuah laptop portabel. Yang merupakan model laptop yang dapat dibawa ke mana saja.

2.6 Komputer era 2000-an

Di awal milenium baru, kemajuan teknologi terjadi bersamaan dengan perkembangan komputer yang berkecepatan tinggi. Floppy disk dan CD-ROM telah digantikan oleh perangkat penyimpanan portabel terbaru seperti USB drive. Sementara Microsoft meluncurkan Windows XP, Apple merilis Mac OS X, sistem operasi terbarunya. Dengan menciptakan perangkat lunak pemutar musik terkenal iTunes, Apple mendominasi pasar. Program lain juga muncul, termasuk MySpace, Firefox, dan YouTube. Karena dirilisnya MacBook Pro dan MacBook Air pada tahun 2007, penggunaan laptop meningkat drastis pada tahun 2006. Steve Jobs meluncurkan iPhone dan iPad pada tahun 2010.

2.7 *Internet of Things (IoT)*

Tahun 2011 dikenang sebagai tahun dimana munculnya berbagai inovasi dalam teknologi IoT. Pertama kali, muncul perangkat IoT bernama Nest Learning Thermostat. Kemudian, banyak lagi produk IoT lain yang muncul dan populer di pasaran, seperti Apple Watch yang hadir pada tahun 2015. Pada tahun yang sama, Apple juga mengumumkan sistem operasi khusus untuk perangkat iPad, yaitu iPadOS.

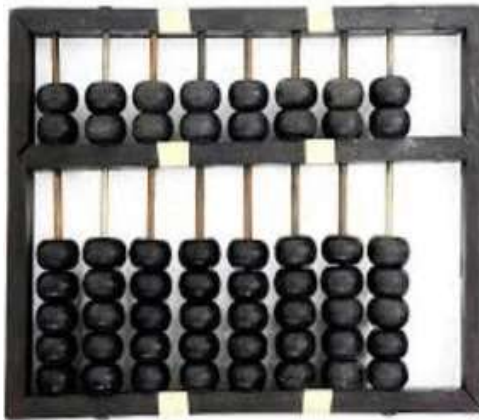
2.8 Sejarah dan Evolusi Perangkat Komputasi

Pada zaman dahulu, orang menggunakan tongkat, batu, dan tulang sebagai peralatan untuk menghitung. Namun, seiring dengan meningkatnya nalar, pikiran, dan teknologi manusia, semakin banyak perangkat komputasi yang diciptakan. Sejarah dan perkembangan perangkat komputasi populer, mulai dari yang pertama sampai yang terbaru, dapat dilihat sebagai berikut:

1. Sempoa (Abacus)

Sejarah komputer dimulai dengan munculnya sempoa atau abacus, yang dipercayai sebagai komputer pertama. Banyak catatan dalam literatur menyebutkan bahwa sempoa

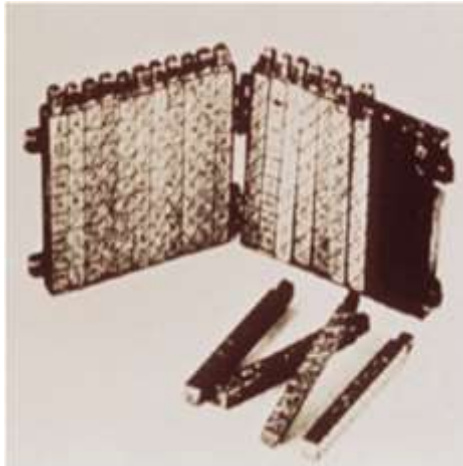
ditemukan oleh orang Cina sekitar 4.000 tahun yang lalu. Sempoa adalah sebuah rak kayu dengan batang logam bermanik-manik yang dipindahkan oleh operator untuk melakukan perhitungan aritmatika. Meski sudah ada perkembangan teknologi komputer yang lebih maju, sempoa masih digunakan hingga saat ini di beberapa negara seperti China, Rusia, dan Jepang. Contoh sempoa dapat ditemukan pada Gambar 2.1.



Gambar 2.1 : Merupakan Bentuk Dari Sempoa

2. Napier's Bone

Napier's bone merupakan alat hitung manual yang ditemukan oleh John Napier dari Merchiston pada tahun 1550-1617. Alat ini memakai sembilan strip gading atau tulang yang diberi tanda angka untuk melakukan perkalian dan pembagian. Sehingga alat ini sering disebut sebagai "Tulang Napier". Alat ini juga merupakan mesin pertama yang memanfaatkan sistem desimal. Gambar 2.2 merupakan tampilan dari napier's bone.



Gambar 2.2 : Merupakan Bentuk Dari Napier's Bone

3. Pascaline

Pascaline juga disebut Mesin Aritmatika atau Mesin Penjumlah. Alat ini ditemukan oleh matematikawan-filsuf Prancis, Blaise Pascal, antara tahun 1642 dan 1644. Pascaline merupakan jenis kalkulator mekanis dan otomatis pertama yang ada.



Gambar 2.3 : Merupakan Bentuk Dari Pascaline

Gambar 2.3 adalah bentuk dari mesin hitung yang diciptakan oleh Pascal. Ia menciptakan alat ini untuk membantu ayahnya, yang merupakan seorang akuntan pajak. Mesin hanya dapat melakukan operasi penambahan dan pengurangan. Terdiri dari kotak kayu dengan rangkaian roda gigi dan roda, saat salah satu roda berputar satu putaran, roda tetangganya juga akan berputar. Ada jendela pada bagian atas roda untuk membaca hasil akhir.

4. Stepped Reckoner atau Leibnitz wheel

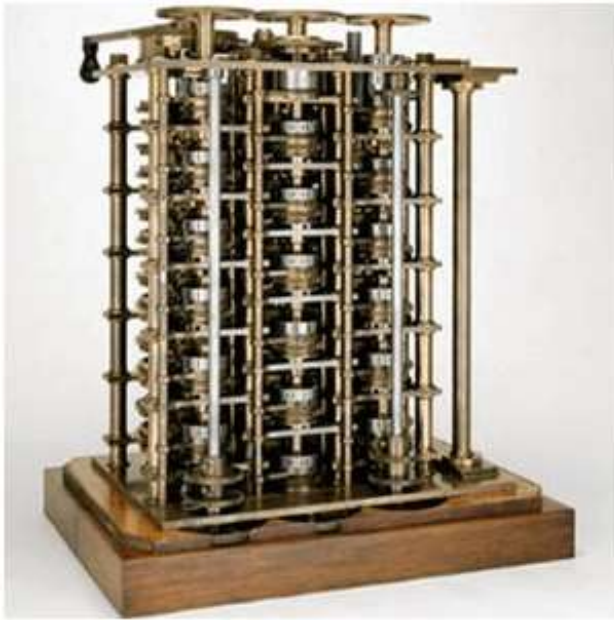
Perangkat komputasi ini ditemukan oleh matematikawan-filsuf Jerman, Gottfried Wilhelm Leibnitz, pada tahun 1673. Ia memperbaiki penemuan Pascal dan menciptakan mesin yang lebih baik. Alat yang dibuat oleh Leibnitz adalah sebuah kalkulator digital mekanik yang disebut penghitung langkah (stepped reckoner) karena tidak menggunakan roda gigi, melainkan menggunakan drum bergalur. Gambar 4 adalah bentuk dari stepped reckoner atau leibnitz wheel.



Gambar 2.4 : Tampilan Dari Stepped Reckoner Atau Leibnitz Wheel

5. Difference Engine

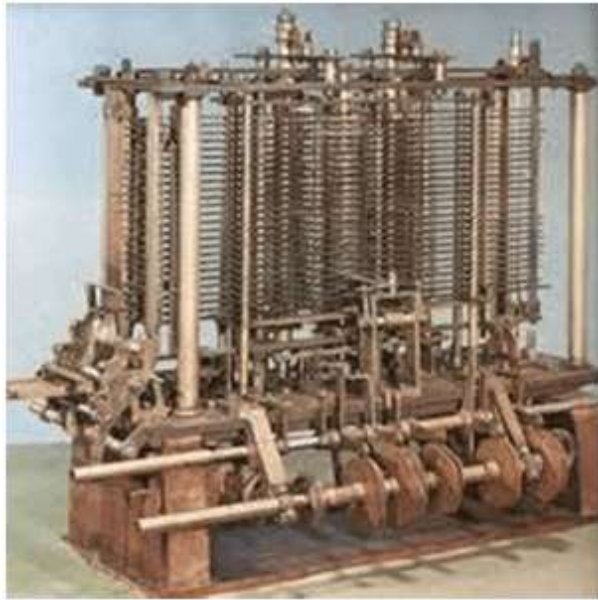
Pada awal tahun 1820-an, sebuah alat diciptakan oleh Charles Babbage, yang dikenal sebagai "Bapak Komputer Modern". Alat tersebut merupakan mesin hitung mekanik yang dinamai dengan sebutan difference engine dan dapat melakukan perhitungan sederhana. Ia diter powered oleh uap dan dirancang untuk menghitung tabel angka seperti tabel logaritma.



Gambar 2.5 : Bentuk Dari Difference Engine

6. Analytical Engine

Charles Babbage juga menciptakan mesin hitung pada tahun 1830. Ini adalah komputer mekanik yang menggunakan kartu punch sebagai masukan dan mampu memecahkan masalah matematika serta menyimpan informasi sebagai memori tetap. Gambar 6 merupakan bentuk dari analytical engine.



Gambar 2.6 : Tampilan Dari Analytical Engine

7. Mesin Tabulasi

Pada tahun 1890 ditemukan Mesin Tabulasi oleh Herman Hollerith, beliau merupakan seorang ahli statistik yang berasal dari Amerika. Mesin ini merupakan tabulator mekanik yang berbasis kartu punch. Mesin komputer ini mampu menghitung statistik dan mencatat atau mengurutkan data atau informasi.

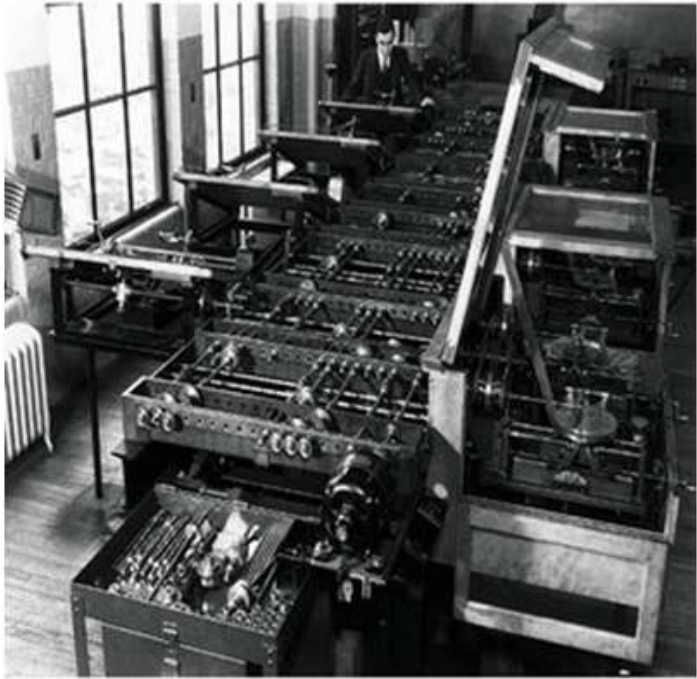
Pada tahun 1890 Mesin Tabulasi ini digunakan dalam sensus di Amerika. Herman Hollerith, ahli statistik Amerika, memulai perusahaan Hollerith's Tabulating Machine yang kemudian berkembang menjadi IBM pada tahun 1924. Gambar. 7 merupakan tampilan dari mesin tabulasi.



Gambar 2.7 : Tampilan Dari Mesin Tabulasi

8. Differential Analyzer

Komputer elektronik pertama yang diperkenalkan di AS pada tahun 1930 adalah Differential Analyzer. Ini merupakan mesin analog yang ditemukan oleh seorang yang bernama Vannevar Bush. Mesin ini mempunyai tabung vakum untuk mengubah sinyal listrik menjadi perhitungan. Komputer ini mampu menyelesaikan 25 perhitungan dalam beberapa menit. Gambar. 8 merupakan tampilan dari differential analyzer.



Gambar 2.8 : Tampilan Dari Differential Analyzer

9. Mark I

Pada tahun 1937 merupakan tahap perkembangan selanjutnya dalam sejarah komputer yang dimulai ketika Howard Aiken merencanakan untuk mengembangkan mesin yang dapat melaksanakan perhitungan yang besar. Pada tahun 1944, Mark I yang merupakan hasil kerjasama antara IBM dan Harvard telah berhasil dibangun. Komputer digital pertama yang bisa diprogram adalah Mark I. Gambar. 9 merupakan tampilan dari Mark I.



Gambar 2.9 : Tampilan Dari Mark I

2.9 Sejarah Perkembangan Komputer Secara Umum

Sejarah pengembangan komputer terbagi menjadi lima generasi, meliputi diantaranya adalah:

1. Generasi Pertama (1940-1956) - Generasi ini dicirikan oleh komputer elektronik vacuum tube. Komponen ini memberikan kemampuan yang sangat besar pada saat itu dan mengubah cara kerja komputer selamanya. Beberapa contoh komputer pada generasi ini adalah ENIAC dan UNIVAC.
2. Generasi Kedua (1956-1963) - Generasi ini dicirikan oleh pemakaian transistor untuk menggantikan vacuum tube. Ini membuat komputer lebih kecil, efisien, dan hemat energi. Contohnya adalah IBM 1400 series dan IBM 7000 series.

3. Generasi Ketiga (1964-1971) - Generasi ini dicirikan oleh pemakaian Integrated Circuit (IC) yang menggabungkan banyak transistor dalam satu chip. Ini membuat komputer menjadi lebih kecil, lebih cepat, dan lebih hemat energi. Contohnya adalah DEC PDP-8 dan IBM System/360.
4. Generasi Keempat (1971-2010) - Generasi ini dicirikan oleh pemakaian teknologi mikroprosesor, yang memungkinkan seluruh komputer dalam satu chip. Ini membuat komputer lebih mudah dibuat, lebih murah, dan lebih aksesibel bagi masyarakat. Contohnya adalah IBM PC, Apple Macintosh, dan komputer pribadi (PC).
5. Generasi Kelima (2010-sekarang) - Generasi ini dicirikan oleh pemakaian teknologi *cloud computing*, *Internet of Things* (IoT), dan komputasi *mobile*. Ini memungkinkan komputer untuk terhubung dan bekerja secara real-time melalui jaringan internet. Beberapa contoh komputer pada generasi ini adalah *smartphone*, tablet, dan *wearable device*.

2.10 Ciri-Ciri Komputer Generasi Awal Hingga Kelima

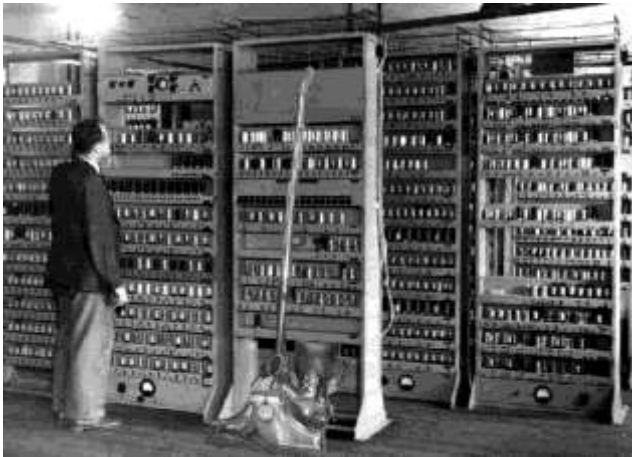
Dalam perkembangannya, terdapat pembagian sejarah perkembangan komputer dimana terdapat lima generasi didalamnya. Setiap generasi memiliki karakteristik yang unik dan berbeda.

1. Komputer Generasi Pertama (1940-1959)

Pada masa PD II, berbagai negara yang terlibat memiliki tujuan untuk menggunakan potensi strategis dari komputer dengan mengembangkan teknologinya. Hal ini memicu peningkatan dana untuk mengembangkan riset terkait komputer dan mempercepat perkembangan teknik. Seorang insinyur bernama Conrad Zuse yang bersal dari Jerman yang telah membangun komputer yang dinamai

dengan Z3 yang difungsikan untuk mendesain pesawat dan peluru kendali pada tahun 1941.

Sekutu juga membuat kemajuan pada pengembangan komputer, dengan membuat mesin pemecah kode rahasia Colossus pada tahun 1943 oleh Inggris. Efek dari pembuatan Colossus tidak terlalu memengaruhi perkembangan industri komputer karena Colossus tidak merupakan komputer umum yang bisa digunakan untuk hal lain, hanya didesain untuk memecahkan kode rahasia, dan juga kerahasiaannya tetap terjaga hingga beberapa tahun setelah perang berakhir. Gambar. 2.10 merupakan tampilan dari komputer Colossus.



Gambar 2.10 : Tampilan Dari Colossus

Pada masa itu, AS melakukan usaha yang menghasilkan kemajuan dalam bidang teknologi komputer. Howard Aiken, yang merupakan insinyur dari Harvard melakukan kerjasama dengan IBM, Aiken telah berhasil membuat sebuah kalkulator elektronik yang selanjutnya digunakan oleh Angkatan Laut AS. Kalkulator itu memiliki ukuran $\frac{1}{2}$ lapangan sepak bola dan memiliki panjang kabel sejauh 500 mil. Mark I atau The

Harvard-IBM *Automatic Sequence Controlled Calculator* adalah komputer relai elektronik yang mengendalikan komponen mekanik dengan menggunakan sinyal elektromagnetik. Perangkat ini bekerja lambat dan tidak fleksibel (setiap perhitungan memakan waktu 3-5 detik), urutan perhitungan tidak dapat diubah. Kalkulator ini dapat memecahkan persamaan yang lebih sulit serta masalah aritmatika sederhana.

Selanjutnya adalah *Electronic Numerical Integrator and Computer* (ENIAC), komputer pertama yang dibuat dalam kolaborasi antara pemerintah AS dan University of Pennsylvania, adalah contoh lebih lanjut dari sejarah komputer. ENIAC sangat besar dan membutuhkan daya hingga 160 kW. Itu memiliki 18.000 tabung vakum, 70.000 resistor, dan 5 juta sambungan solder. ENIAC adalah komputer serba guna yang dibuat oleh John Presper Eckert dan John Mauchly yang seribu kali lebih cepat dibandingkan dengan Mark I. Ciri-ciri Komputer Generasi Pertama:

- a) Program dirancang menggunakan bahasa mesin.
- b) Menerapkan konsep penyimpanan program.
- c) Memanfaatkan tabung hampa udara sebagai komponen utamanya.
- d) Memiliki ukuran fisik yang besar dan membutuhkan banyak daya listrik.
- e) Dukungan terhadap *magnetic tape* dan *magnetic disk* sebagai media penyimpanannya.
- f) Seperti komputer IBM 701 yang diproduksi pada tahun 1953 yang merupakan komputer komersial berukuran besar dan IBM 705 pada tahun 1959 untuk industri. Berikut ini contoh dari komputer generasi pertama yang terkenal adalah: *Electronic Discrete Variable Automatic Computer* (EDVAC), IBM-701, *Electronic Numerical*

Integrator and Computer (ENIAC), *Universal Automatic Computer* (UNIVACI) dan IBM-650.

2. Komputer Generasi Kedua (1959-1965)

Transistor telah ditemukan pada tahun 1948. Kondisi ini mempengaruhi perkembangan komputer secara signifikan. Keberadaan tabung vakum telah digantikan dengan transistor yang digunakan pada berbagai peralatan elektronik seperti radio, televisi, dan komputer, sehingga menyebabkan volume dari mesin-mesin elektronik menjadi lebih kecil. Tahun 1956 adalah awal digunakannya transistor dalam komputer.



Gambar 2.11 : Tampilan Dari Komputer Generasi Kedua

Penemuan teknologi memori inti-magnetik membantu pembuatan komputer generasi kedua yang lebih kecil, cepat, handal, dan hemat energi daripada generasi sebelumnya. IBM membuat superkomputer pertama yang menggunakan teknologi ini yang bernama *Stretch*, sementara Sprery-Rand membuat LARC.

Komputer ini dikembangkan untuk keperluan penelitian atom dan mampu menangani data yang sangat banyak, tetapi biayanya sangat tinggi dan terlalu kompleks untuk digunakan dalam bisnis sehingga membatasi popularitasnya.

Komputer *Long Range Aid to Navigation* (LARC) dibuat dua perangkat saja. Perangkat pertama berada di Lawrence Radiation Labs di Livermore, California. Sedangkan untuk perangkat yang kedua terletak di US Navy Research and Development Center di Washington D.C. Pada perkembangan berikutnya, yaitu generasi kedua dari komputer, keberadaan bahasa mesin digantikan oleh bahasa Assembly yang merupakan bahasa yang menerapkan penggunaan singkatan untuk difungsikan dalam mengkonversi *code* biner. Fitur Komputer Generasi Ke-dua:

- a) Memori utama memiliki kapasitas yang cukup besar.
- b) Perangkat transistor yang diterapkan berukuran lebih kecil perbandingannya dengan tabung vakum.
- c) Menerapkan tape magnetik serta disk magnetik yang dapat dilepas.
- d) Mampu memproses data secara *real-time* dan berbagi waktu.
- e) Memiliki kecepatan yang baik dalam prosesnya.
- f) Perangkat difokuskan pada implementasi dunia bisnis serta perteknikan.
- g) Komputer mini komersial pertama adalah PDP-5 dan PDP-8 yang dibuat pada tahun 1963. Ada juga komputer IBM 7070, IBM 1400, NCR 300, dan lainnya. Komputer generasi kedua yang terkenal adalah: IBM 7094, CDC 3600, UNIVAC 1108, IBM 1620 dan CDC 1604

3. Komputer Generasi Ketiga (1965-1970)

Meskipun transistor memiliki beberapa keunggulan dibandingkan tabung vakum, transistor juga menghasilkan banyak panas yang dapat merusak komponen internal komputer. Masalah ini bisa diatasi dengan batu kuarsa. Pada tahun 1958, insinyur Texas Instruments Jack Kilby menciptakan sirkuit terintegrasi. IC adalah chip silikon kecil dari pasir kuarsa yang menggabungkan tiga komponen elektronik.



Gambar 2.12 : Komputer Generasi Ketiga

Belakangan, para peneliti memasukkan lebih banyak komponen ke dalam satu chip, sebuah semikonduktor. Akibatnya, komputer dapat memuat lebih banyak komponen pada area permukaan yang lebih kecil. Menggunakan sistem operasi yang memungkinkan mesin menjalankan banyak aplikasi secara bersamaan melalui program utama yang memantau dan mengoordinasikan memori komputer adalah kemajuan lain dalam komputer generasi ketiga. Ciri-ciri Komputer generai ke-tiga:

- a) Menggunakan Integrated Circuits (IC) yang terdiri atas ratusan hingga ribuan transistor, baik hybrid maupun monolithic IC.
- b) Proses operasi lebih cepat dan tepat, dengan kapasitas memori yang jauh lebih besar.
- c) Penggunaan listrik lebih hemat karena ukuran fisik lebih kecil.
- d) Menggunakan magnetic disk dengan akses acak.
- e) Mampu melakukan multiprocessing dan multiprogramming.
- f) Alat masukan/keluaran mengalami peningkatan dengan penggunaan terminal tampilan visual.
- g) Mampu melakukan komunikasi data antar komputer.
- h) Contohnya adalah IBM S/30, NOVA, CDC 3000, PDP-11, dan lainnya. Beberapa contoh dari komputer generasi ketiga yang populer adalah: Honeywell-6000 series, IBM-360 series, TDC-316, IBM-370/168 dan PDP (Personal Data Processor).

4. Komputer Generasi Keempat (dimulai dari tahun 1970)

Setelah pengembangan Integrated Circuits, tujuannya adalah untuk memperkecil ukuran dari sirkuit dan berbagai komponen elektronik. LSI (Large Scale Integration) mampu menampung ratusan komponen dalam sebuah chip. Selanjutnya di era 1980-an, VLSI atau Very Large-Scale Integration telah berhasil memasukkan ribuan komponen dalam satu chip tunggal.

Setelah adanya LSI dan VLSI, teknologi pengembangan berlanjut ke ULSI atau *Ultra-Large-Scale Integration* yang memungkinkan banyaknya komponen dalam sebuah chip mencapai jutaan. Keberhasilan memasang jumlah besar komponen dalam ukuran chip yang semakin kecil membuat harga dan ukuran komputer semakin terjangkau dan mengecil.

Chip Intel 4004 membawa revolusi dalam pengembangan Integrated Circuit (IC) dengan memasukkan semua komponen sebuah komputer (CPU, memori, dan kontrol I/O) kedalam sebuah chip yang sangat kecil. Sebelumnya, Integrated Circuits hanya diperuntukkan untuk tugas tertentu (spesifik). Namun, saat ini mikroprosesor dapat diproduksi dan diprogram sesuai dengan kebutuhan. Bahkan, banyak perangkat rumah tangga seperti televisi, microwave, oven dan mobil dengan sistem injeksi bahan bakar elektronik (EFI) sudah dilengkapi dengan mikroprosesor.

Setelah perkembangan IC, komputer dapat digunakan oleh masyarakat umum. Sebelumnya, hanya perusahaan besar dan lembaga pemerintah yang menggunakan komputer. Pada tahun 1970-an, produsen komputer memperkenalkan produk minikomputer mereka ke pasaran umum dengan menyediakan *software* yang mudah digunakan bagi pengguna awam.



Gambar 2.13 : Komputer Generasi Keempat

Pada tahun 1980-an, perangkat lunak yang memiliki pengguna paling banyak adalah program pengolahan kata dan spreadsheet. Sejak saat itu, video game seperti Atari 2600 juga memperkenalkan komputer rumahan yang lebih canggih dan dapat diprogram kepada konsumen. Ciri-ciri Komputer generasi ke-empat:

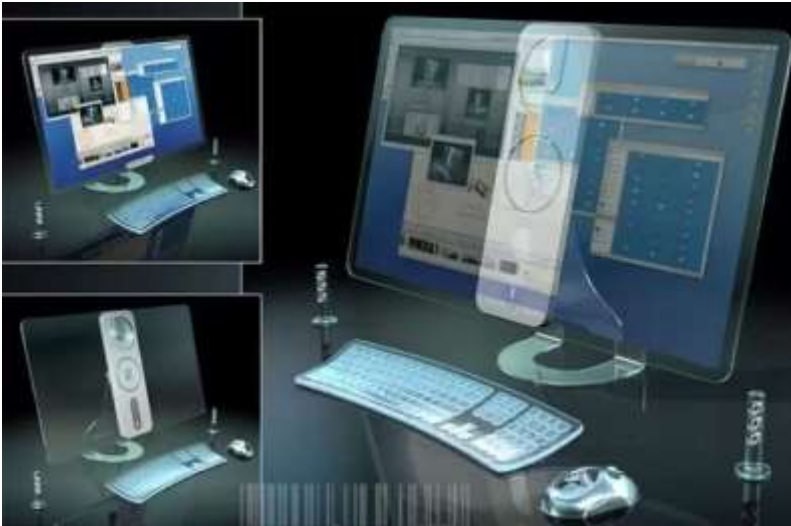
- a) Menggunakan LSI juga dikenal sebagai *Bipolar Large-Scale Integration*.
- b) Komputer memanfaatkan mikroprosesor dan semikonduktor yang berbentuk chip sebagai memori.
- c) Misalnya, komputer IBM 370 menggunakan mikroprosesor Intel 4004 yang pertama kali dikembangkan pada tahun 1971 oleh Intel Corporation.
- d) Komputer pribadi (PC) mulai populer sejak tahun 1977, contohnya Apple II dan komputer desktop oleh Xerox Corporation.
- e) Pada tahun 1981, komputer mulai menggunakan sistem Windows dan mouse secara luas. Beberapa komputer generasi keempat yang terkenal adalah: STAR 1000, PDP 11, DEC 10, RAY-X-MP (*Supercomputer*), CRAY-1(*Supercomputer*) dan STAR 1000)

5. Komputer Generasi Kelima

Komputer generasi kelima yang diterapkan dalam dunia fiksi adalah HAL9000 yang terdapat dalam novel 2001: Space Odyssey karya Arthur C. Clarke. HAL memiliki karakteristik yang dicari dari sebuah komputer generasi kelima, termasuk kecerdasan buatan yang memungkinkannya untuk berkomunikasi dengan manusia, memproses informasi visual, dan belajar dari pengalaman.

Walaupun komputer seperti HAL9000 masih jauh dari kenyataan saat ini, beberapa fitur yang dimilikinya sudah tercapai, seperti komputer yang dapat menerima perintah

suara dan menirukan nalar manusia. Kemampuan untuk menerjemahkan bahasa juga sudah ada. Namun, meskipun terlihat mudah, memahami konteks dan makna dalam bahasa manusia ternyata lebih sulit dibandingkan hanya menerjemahkan kata-kata.



Gambar 2.14 : Tampilan Dari Komputer Generasi Kelima

Beberapa perkembangan penting dalam bidang teknologi dan desain komputer membuat tampaknya komputer generasi kelima semakin dekat. Terdapat dua perkembangan utama dalam rekayasa, yaitu mampu melakukan proses secara paralel yang dapat menggantikan pemodelan non-Neumann dan teknologi superconductor yang memungkinkan aliran listrik yang tidak menggunakan hambatan, sehingga informasi yang dihasilkan akan lebih cepat.

Jepang dikenal sebagai negara yang sangat mempromosikan komputer generasi ke-5. Institut ICOT

(*Institute for New Computer Technology*) dibentuk untuk mewujudkan hal tersebut. Ada informasi yang menyatakan bahwa proyek ini telah gagal, namun ada juga yang menyatakan bahwa jika berhasil, proyek ini akan mengubah paradigma komputerisasi baru di seluruh dunia. Ciri-ciri Komputer generasi ke-lima:

- a) Penggunaan perangkat VLSI (*Very Large-Scale Integration*).
- b) Kemampuan komputasi ditingkatkan dengan bantuan AI (*Artificial Intelligence*) untuk memecahkan masalah secara mandiri, bahkan dapat digunakan untuk pengendalian robot.
- c) Mulai mengembangkan komputer yang dapat menggantikan chip.
- d) Jepang merupakan negara yang memimpin dalam pengembangan komputer generasi kelima. Berikut ini contoh dari komputer generasi kelima yang populer adalah: Laptop, UltraBook, Desktop, NoteBook dan ChromeBook

DAFTAR PUSTAKA

- Dsit. 2021. *Sejarah Komputer dan Perkembangannya dari Masa ke Masa*. <https://unida.ac.id/teknologi/artikel/sejarah-komputer-dan-perkembangannya-dari-masa-ke-masa-.html>
- Fandy. 2021. *Sejarah Perkembangan Komputer dari Generasi Pertama Hingga Kelima*. <https://www.gramedia.com/literasi/sejarah-perkembangan-komputer/#:~:text=Secara%20garis%20besar%2C%20sejarah%20perkembangan%20komputer%20terbagi%20dalam,transistor%20meningkat%20menjadi%20sirkuit%20terpadu.%20...%20More%20items>
- Trivusi. 2022. *Sejarah dan Evolusi Komputer dari Masa ke Masa*. <https://www.trivusi.web.id/2022/08/sejarah-dan-evolusi-komputer.html?m=1>

BAB 3

METODE DAN TEKNOLOGI

TRANSFER DATA ANTAR KOMPONEN

DI DALAM KOMPUTER

Oleh Rita Komalasari

3.1 Struktur Interkoneksi

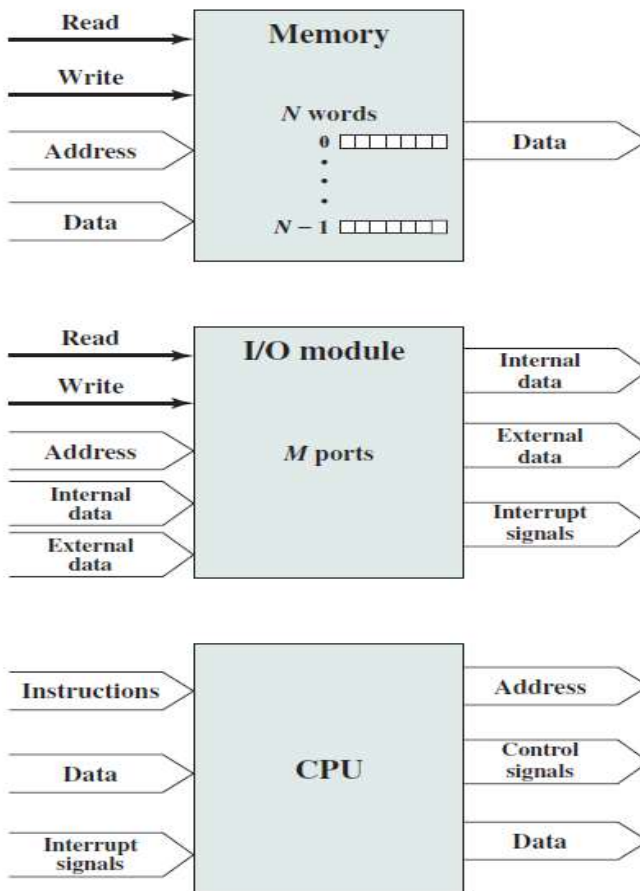
Komputer terdiri dari sekumpulan bagian ataupun modul dari 3 tipe dasar (prosesor, memori, I atau O) yang berinteraksi satu serupa lain. Pada dasarnya, pc merupakan jaringan materi bawah. Dengan begitu, wajib terdapat rute buat mengaitkan modul- modul itu. Berkas rute yang mengaitkan bermacam modul disebut bentuk interkoneksi. rancangan susunan ini hendak terkait pada pergantian yang wajib dicoba dampingi modul. Gambar 3.1 memperlihatkan tipe- tipe pergantian yang dibutuhkan dengan membuktikan struktur penting input serta output buat tiap tipe modul:

Memori: Umumnya, modul memori terdiri dari N kata dengan berjarak yang serupa. Tiap kata diberi tujuan numerik yang khas(0, 1, c, N- 1). Suatu kata informasi bisa dibaca dari ataupun ditulis ke dalam memori. Sifat operasi ditunjukkan oleh tanda pengawasan baca- tulis. Posisi buat operasi didetetapkan oleh tujuan.

Materi I atau O: Dari sudut penglihatan dalam (ke sistem komputer), I atau O dengan cara fungsional serupa dengan memori. Terdapat 2 operasi; baca serta catat. Lebih jauh lagi, materi I atau O bisa mengendalikan lebih dari satu fitur eksternal. Kita bisa merujuk ke tiap-tiap antarmuka ke fitur eksternal selaku port serta membagikan tiap- tiap alamat khas(misalnya, 0, 1, c, M- 1). Tidak

hanya itu, terdapat rute informasi eksternal buat input serta output informasi dengan fitur eksternal. Terakhir, materi I atau O bisa jadi bisa mengirim tanda interupsi ke prosesor.

Prosesor: Prosesor membaca instruksi serta informasi, menulis informasi setelahnya pemrosesan, serta memakai tanda pengawasan guna mengendalikan operasi totalitas sistem. Prosesor pula menyambut tanda interupsi.



Gambar 3.1 : Modul Komputer
 Sumber : (Stallings, 2016)

Catatan lebih dahulu mendeskripsikan informasi yang hendak dipertukarkan. Bentuk interkoneksi wajib mensupport tipe transfer selanjutnya ini:

1. Memori ke prosesor: Prosesor membaca instruksi ataupun bagian informasi dari memori.
2. Prosesor ke memori: Prosesor menulis bagian informasi ke memori.
3. I atau O ke prosesor: Prosesor membaca informasi dari fitur I atau O lewat modul I atau O.
4. Prosesor ke I atau O: Prosesor mengirim informasi ke fitur I atau O.
5. I atau O ke ataupun dari memori: Guna kedua permasalahan ini, materi I atau O diizinkan buat beralih informasi dengan cara langsung dengan memori, tanpa lewat prosesor, memakai akses memori langsung.

Sepanjang bertahun-tahun, beberapa bentuk interkoneksi pernah dicoba. Sepanjang ini yang sangat lazim yaitu(1) bus serta bermacam bentuk sekian banyak bus, serta(2) bentuk interkoneksi titik- ke- titik dengan memindahkan informasi yang dikemas.

3.2 Interkoneksi Bus

Dalam sistem komputer, berbagai subsistem harus memiliki antarmuka. Sebagai contoh, memori dan prosesor harus berkomunikasi, begitu pula prosesor dan perangkat I/O. Selama bertahun-tahun, hal ini telah dilakukan dengan bus. Bus adalah tautan komunikasi bersama, yang menggunakan satu set kabel untuk menghubungkan beberapa subsistem. Dua keuntungan signifikan dari organisasi bus adalah keserbagunaan dan biaya rendah. Dengan mendefinisikan skema koneksi tunggal, perangkat baru dapat dengan mudah ditambahkan, dan periferal bahkan dapat dipindahkan di antara sistem komputer yang menggunakan

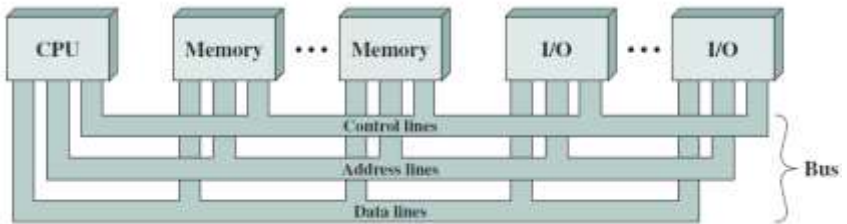
jenis bus yang sama. Selain itu, bus hemat biaya karena satu set kabel digunakan bersama dalam berbagai cara.

Kerugian utama dari sebuah bus adalah bahwa hal itu menciptakan hambatan komunikasi, yang mungkin membatasi throughput I/O maksimum. Ketika I/O harus melewati satu bus, bandwidth bus tersebut membatasi keluaran I/O maksimum.

Merancang sistem bus yang mampu memenuhi permintaan prosesor serta menghubungkan sejumlah besar perangkat I/O ke mesin menghadirkan tantangan besar. Salah satu alasan mengapa desain bus sangat sulit adalah karena kecepatan bus maksimum sebagian besar dibatasi oleh faktor fisik: panjang bus dan jumlah perangkat.

Batasan fisik ini mencegah kita menjalankan bus dengan kecepatan tinggi. Selain itu, kebutuhan untuk mendukung berbagai perangkat dengan latensi dan kecepatan transfer data yang sangat bervariasi juga membuat desain bus menjadi menantang. Karena menjadi sulit untuk menjalankan banyak kabel paralel dengan kecepatan tinggi karena kemiringan jam dan refleksi, industri sedang dalam transisi dari bus bersama paralel ke interkoneksi titik-ke-titik serial berkecepatan tinggi dengan sakelar. Dengan demikian, jaringan semacam itu secara bertahap menggantikan bus dalam sistem kami. Sebagai hasil dari transisi ini, bagian ini telah direvisi dalam edisi ini untuk menekankan masalah umum dalam menghubungkan perangkat I/O, prosesor, dan memori daripada berfokus secara eksklusif pada bus.

Suatu bus sistem umumnya terdiri dari kurang lebih 5 puluh sampai ratusan jalur terpisah. Tiap jalur diberi maksud ataupun guna khusus. Walaupun terdapat banyak konsep bus yang berlainan, pada bus apa pun, jalur bisa diklasifikasikan jadi 3 fungsional golongan (Gambar 3.2): informasi, tujuan, serta rute pengawasan. Tidak hanya itu, bisa jadi terdapat terdapat jalur penyaluran daya yang menyediakan daya ke modul yang terpasang.



Gambar 3.2 : Skema Interkoneksi Bus

Sumber : (Stallings 2016)

Jalur data sediakan jalur guna memindahkan data di antara modul sistem. Ini disebut bus data. Bus data bisa terdiri dari 32, 64, 128, ataupun terlebih lagi lebih banyak garis terpisah, jumlah garis disebut selaku luas bus data. Sebab tiap baris cuma bisa membawa satu bit pada satu durasi, jumlah memastikan berapa banyak bit yang bisa ditransfer pada satu durasi. Lebar data data merupakan aspek kunci dalam memastikan kemampuan sistem dengan cara totalitas. Misalnya, bila bus data mempunyai lebar 32 bit serta tiap instruksi ada berjarak 64 bit, sehingga prosesor wajib mengakses modul memori 2 kali sepanjang tiap daur instruksi. Baris alamat dipakai buat menunjuk pangkal ataupun tujuan data pada bus data. Misalnya, bila prosesor mau membaca suatu kata(8, 16, ataupun 32 bit) data dari memori, prosesor hendak menempatkan alamat kata yang diinginkan pada baris alamat. Jelas, luas bus alamat memastikan kapasitas memori tertinggi yang bisa jadi tertinggi dari sistem. Tidak hanya itu, jalur alamat biasanya pula dipakai buat menepatkan I atau O port. Umumnya, bit sistem besar dipakai buat memilah modul khusus pada bus, serta bit sistem kecil memilah posisi memori ataupun port I atau O dalam modul. Misalnya, pada bus alamat 8- bit, alamat 01111111 serta di bawahnya bisa jadi rujukan posisi dalam modul memori(modul 0) dengan 128 kata memori, serta alamat 10000000 serta di atasnya merujuk pada fitur yang terpasang pada modul I atau O(modul 1).

Jalur kontrol dikenakan guna mengendalikan akses serta pemakaian jalur data serta alamat. Sebab jalur data serta jalur alamat dikenakan bersama oleh seluruh komponen, hingga wajib terdapat metode guna mengendalikan penggunaannya. Sinyal kontrol mengirimkan data perintah serta pengaturan durasi di antara modul sistem. Tanda pengaturan durasi membuktikan kesahan data serta data alamat. Tanda perintah memastikan operasi yang hendak dicoba. Jalur kontrol yang biasa mencakup:

1. *Memory write*: mengakibatkan data pada bus ditulis ke posisi yang dituju.
2. *Memory read*: mengakibatkan data dari posisi yang dituju ditempatkan pada bus.
3. I atau O write: mengakibatkan data pada bus sebagai output ke port I atau O yang dituju.
4. I atau O read: mengakibatkan data dari port I atau O yang dituju ditempatkan pada bus.
5. Transfer ACK: memperlihatkan jika data sudah diperoleh dari ataupun ditempatkan pada bus.
6. Bus request: membuktikan kalau modul perlu memperoleh pengawasan bus.
7. Bus grant: membuktikan kalau modul yang meminta sudah diserahkan kontrol atas bus.
8. Interrupt request: membuktikan kalau suatu interupsi tengah menunggu.
9. Interrupt ACK: melaporkan kalau interupsi yang tertunda sudah dikenali.
10. Clock: digunakan untuk menyinkronkan operasi.
10. Reset: menginisialisasi semua modul.

Pengoperasian bus adalah sebagai berikut. Jika satu modul ingin mengirim data ke lain, ia harus melakukan dua hal: (1) mendapatkan penggunaan bus, dan (2) mentransfer data melalui bus. Jika satu modul ingin meminta data dari modul lain, modul

tersebut harus (1) mendapatkan penggunaan bus, dan (2) mentransfer permintaan ke modul lain melalui jalur kontrol dan alamat yang sesuai. Kemudian harus menunggu modul kedua itu mengirim data.

Sebuah bus dapat berupa titik-ke-titik, yang menghubungkan dua komponen tertentu. Atau bisa juga berupa jalur umum yang menghubungkan sejumlah perangkat, yang mengharuskan perangkat-perangkat ini untuk berbagi bus (disebut sebagai bus multipoint) (Null, 2003)

Jalur kontrol digunakan untuk mengontrol akses dan penggunaan data dan jalur alamat. Karena jalur data dan alamat digunakan bersama oleh semua komponen,

Secara klasik, sebuah bus umumnya berisi satu set jalur kontrol dan satu set jalur data. Jalur kontrol digunakan untuk memberi sinyal permintaan dan pengakuan, dan untuk menunjukkan jenis informasi apa yang ada di jalur data. Jalur data bus membawa informasi antara sumber dan tujuan. Informasi ini dapat terdiri dari data, perintah yang rumit, atau alamat. Sebagai contoh, jika disk ingin menulis beberapa data ke dalam memori dari sektor disk, jalur data akan digunakan untuk menunjukkan alamat dalam memori untuk menempatkan data serta membawa data aktual dari disk. Garis kontrol akan digunakan untuk menunjukkan jenis informasi yang terkandung pada jalur data bus pada setiap titik dalam transfer. Beberapa bus memiliki dua set jalur sinyal untuk mengkomunikasikan data dan alamat dalam transmisi bus tunggal. Dalam kedua kasus tersebut, jalur kontrol digunakan untuk menunjukkan apa yang terkandung dalam bus dan untuk mengimplementasikan protokol bus.

Transaksi bus mencakup dua bagian: mengirim alamat dan menerima atau mengirim data. Transaksi bus biasanya ditentukan oleh apa yang dilakukannya pada memori. Transaksi baca mentransfer data dari memori (ke prosesor atau perangkat I/O), dan transaksi tulis menulis data ke memori. Jelas, terminologi ini

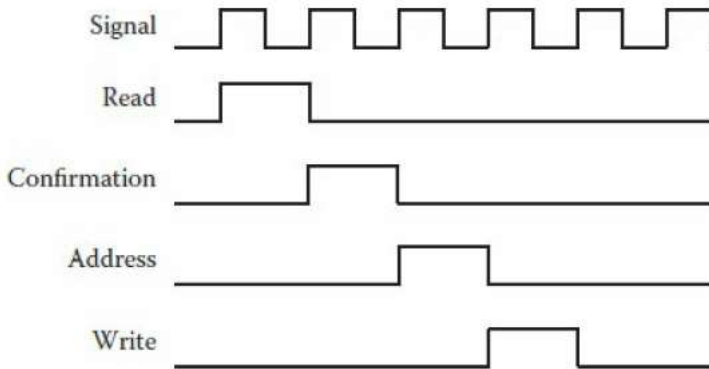
membingungkan. Untuk menghindari hal ini, kita akan mencoba menggunakan istilah input dan output, yang selalu didefinisikan dari sudut pandang prosesor: operasi input adalah memasukkan data dari perangkat ke memori, di mana prosesor dapat membacanya, dan operasi output adalah mengeluarkan data ke perangkat dari memori di mana prosesor menuliskannya.

Bus secara tradisional diklasifikasikan sebagai bus memori prosesor. Bus prosesor-memori pendek, umumnya berkecepatan tinggi, dan disesuaikan dengan sistem memori untuk memaksimalkan bandwidth prosesor-memori. Sebaliknya, bus I/O dapat berukuran panjang, dapat memiliki banyak jenis perangkat yang tersambung, dan sering kali memiliki rentang lebar dalam bandwidth data perangkat yang tersambung. Bus I/O biasanya tidak berinteraksi langsung dengan memori, tetapi menggunakan memori prosesor atau bus bidang belakang untuk terhubung ke memori. Bus lain dengan karakteristik yang berbeda telah muncul untuk fungsi khusus, seperti bus grafis.

Bus I/O berfungsi sebagai cara untuk memperluas mesin dan menghubungkan periferal baru. Untuk mempermudah hal ini, industri komputer telah mengembangkan beberapa standar. Standar tersebut berfungsi sebagai spesifikasi untuk produsen komputer dan produsen periferal. Standar memastikan perancang komputer bahwa periferal akan tersedia untuk mesin baru, dan memastikan pembuat periferal bahwa pengguna akan dapat menghubungkan peralatan baru mereka.

Dua skema dasar untuk komunikasi pada bus adalah sinkron dan asinkron. Jika bus sinkron, bus ini menyertakan jam di jalur kontrol dan protokol tetap untuk berkomunikasi yang relatif terhadap jam. Misalnya, untuk bus memori prosesor yang melakukan pembacaan dari memori, kita mungkin memiliki protokol yang mentransmisikan alamat dan perintah baca pada siklus clock pertama, menggunakan jalur kontrol untuk menunjukkan jenis permintaan. Memori kemudian mungkin

diminta untuk merespons dengan kata data pada clock kelima. Jenis protokol ini dapat diimplementasikan dengan mudah dalam mesin keadaan terbatas yang kecil. Karena protokol telah ditentukan sebelumnya dan melibatkan sedikit logika, bus dapat berjalan sangat cepat, dan logika antarmuka akan menjadi kecil.

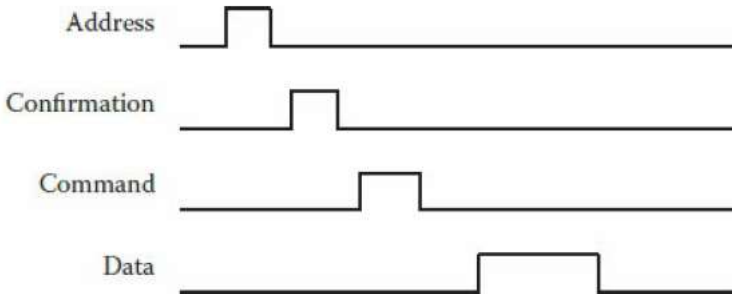


Gambar 3.3 : Synchronous bus
Sumber : (Yadin, 2016)

Namun, bus sinkron memiliki dua kelemahan utama. Pertama, setiap perangkat pada bus harus berjalan pada kecepatan clock yang sama. Kedua, karena masalah kemiringan clock, bus sinkron tidak dapat bertahan lama jika mereka cepat. Bus memori prosesor sering kali sinkron karena perangkat yang berkomunikasi berdekatan, jumlahnya sedikit, dan disiapkan untuk beroperasi pada kecepatan clock yang tinggi.

Bus asinkron tidak memiliki clock. Karena tidak memiliki clock, bus asinkron dapat mengakomodasi berbagai macam perangkat, dan bus dapat diperpanjang tanpa mengkhawatirkan kemiringan clock atau masalah sinkronisasi. Untuk mengoordinasikan transmisi data antara pengirim dan penerima, bus asinkron menggunakan protokol *handshaking*. Protokol *handshaking* terdiri dari serangkaian langkah di mana pengirim

dan penerima melanjutkan ke langkah berikutnya hanya jika kedua belah pihak setuju. Protokol ini diimplementasikan dengan serangkaian jalur kontrol tambahan.

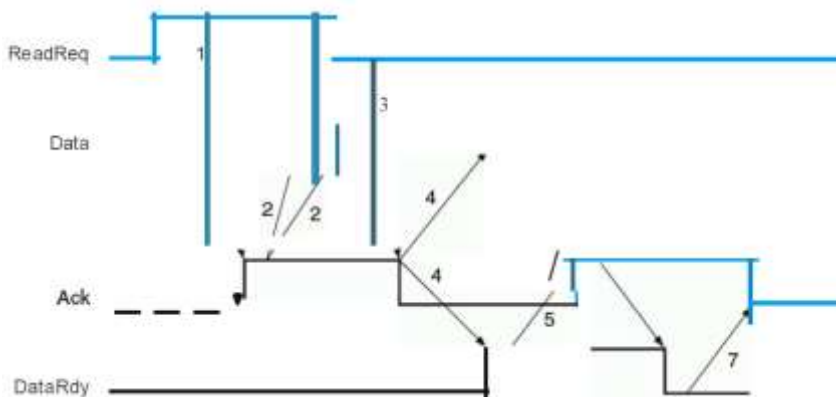


Gambar 3.4 : Asynchronous bus
Sumber : (Yadin, 2016)

Sebuah contoh sederhana akan mengilustrasikan cara kerja bus asinkron. Mari kita pertimbangkan sebuah perangkat yang meminta sebuah kata data dari sistem memori. Asumsikan bahwa ada tiga jalur kontrol:

1. ReadReq: Digunakan untuk mengindikasikan permintaan baca memori. Alamatnya diletakkan pada baris data pada saat yang bersamaan.
2. DataRdy: Digunakan untuk mengindikasikan bahwa kata data sekarang siap pada jalur data. Dalam transaksi output, memori akan menegaskan sinyal ini karena menyediakan data. Dalam transaksi input, perangkat I/O akan menegaskan sinyal ini, karena ia akan menyediakan data. Dalam kedua kasus tersebut, data ditempatkan pada jalur data pada saat yang bersamaan.
3. Ack: Digunakan untuk mengakui sinyal ReadReq atau DataRdy dari pihak lain.

Dalam protokol asinkron, sinyal kontrol ReadReq dan DataRdy ditegaskan hingga pihak lain (memori atau perangkat) mengindikasikan bahwa jalur Kontrol telah dilihat dan jalur data telah dibaca; indikasi ini dibuat dengan menegaskan jalur Ack. Proses lengkap ini disebut *handshaking*. Gambar 3.5 menunjukkan bagaimana protokol tersebut beroperasi dengan menggambarkan langkah-langkah dalam komunikasi. Meskipun sebagian besar bandwidth bus ditentukan oleh pilihan protokol sinkron atau asinkron dan karakteristik waktu bus, beberapa faktor lain mempengaruhi bandwidth yang dapat dicapai oleh satu transfer. Yang paling penting adalah lebar bus data, dan apakah mendukung transfer blok atau mentransfer satu kata pada satu waktu.



Gambar 3.5 : Protokol *handshaking* asinkron terdiri dari tujuh langkah untuk membaca kata dari memori dan menerimanya dalam perangkat I/O.

Sumber : (Hennessy, 2005)

Sinyal dalam warna adalah sinyal yang ditegaskan oleh perangkat I/O, sedangkan memori menegaskan sinyal yang ditunjukkan dalam warna hitam. Tanda panah melabeli tujuh langkah dan peristiwa yang memicu setiap langkah. Simbol yang

menunjukkan dua garis (tinggi dan rendah) pada saat yang sama pada jalur data menunjukkan bahwa jalur data memiliki data yang valid pada saat ini. (Simbol ini menunjukkan bahwa data valid, tetapi nilainya tidak diketahui).

Langkah-langkah dalam protokol dimulai segera setelah perangkat memberi sinyal permintaan dengan menaikkan ReadReq dan meletakkan alamat pada baris Data:

1. Ketika memori melihat baris ReadReq, memori akan membaca alamat dari bus data dan menaikkan Ack ke mengindikasikan bahwa alamat tersebut telah dilihat.
2. Perangkat I/O melihat jalur Ack tinggi dan melepaskan jalur ReadReq dan data.
3. Memori melihat bahwa ReadReq rendah dan melepaskan jalur Ack untuk mengakui sinyal ReadReq.
4. Langkah ini dimulai ketika memori telah menyiapkan data. Ini menempatkan data dari permintaan baca pada jalur data dan membangkitkan DataRdy.
5. Perangkat I/O melihat DataRdy, membaca data dari bus, dan memberi sinyal bahwa ia memiliki data dengan menaikkan Ack.
6. Memori melihat sinyal Ack, menurunkan DataRdy, dan melepaskan jalur data.
7. Akhirnya, perangkat I / O, melihat DataRdy menjadi rendah, menjatuhkan garis Ack, yang menunjukkan bahwa transmisi selesai.
8. Transaksi bus baru sekarang dapat dimulai.

DAFTAR PUSTAKA

- Hennessy, D. A. P. J. L. 2005. *Computer Organization and Design, Third Edition_ The Hardware Software Interface*. San Farnscisco: Morgan Kaufmann.
- Null, L. 2003. *The Essentials of Computer Organization and Architecture*. London: Jones and Bartlett.
- Stallings, W. 2016. *Computer Organization and Architecture Designing for Performance*. 10th edn. New Jersey: Pearson Education,.
- Yadin, A. 2016. *Computer systems architecture*. Boca Raton: CRC Press,.

BAB 4

MEMORI INTERNAL

Oleh Fajar Sari Kurniawan

4.1 Pendahuluan

Sistem Memori (Memori) adalah tempat penyimpanan data atau komponen komponen elektronik yang menyimpan perintah-perintah yang menunggu untuk di eksekusi oleh prosesor, data yang diperlukan oleh insruksi (perintah) tersebut dan hasil-hasil dari data yang diproses (informasi) (Patterson & Hennessy, 2018). Memori dapat dianggap sebagai komponen penting karena berperan dalam menentukan efisiensi dan efektivitas perangkat komputasi. Semakin tinggi kapasitas dan kecepatan, maka semakin baik kinerja perangkat.

Memori komputer adalah media penyimpanan berupa perangkat keras komputer (*hardware*) untuk menyimpan data yang bersifat sementara atau permanen. Menurut sistem komputer memori diklasifikasikan untuk tujuan dan kegunaan yang berbeda. Memori dikelompokkan menjadi dua jenis yaitu memori internal dan memori eksternal. Memori internal atau biasa disebut dengan istilah memori utama yaitu memori yang diakses langsung oleh prosesor untuk merespon perintah dari pengguna (*user*). Memori ini berisi semua instruksi dari berbagai sumber untuk membuat program bekerja sebagaimana mestinya.

Dalam satuan bahasan ini akan dipelajari mengenai memori internal yang meliputi, karakteristik sistem memori, keandalan memori, kapasitas memori, dan modul memori.

4.2 Pengertian Memori Internal

Mengapa pengaturan (*setting*) pada perangkat elektronik yang memiliki program otomatis, tidak berubah saat dinyalakan berulang kali? Atau pernahkah anda memperhatikan setiap kali menyalakan komputer bahwa pengaturan pada komputer tidak berubah? Dimana pengaturan itu disimpan? Yap, disitulah sebuah memori digunakan.

Memori internal adalah jenis memori yang dipasang dalam satu rangkaian dengan prosesor. Lokasinya menyatu di dalam perangkat papan utama (*motherboard*), sehingga tidak bisa dicabut. Adapun memori internal yang dapat dicabut (diganti), namun secara teknis memori internal harus terpasang saat komputer digunakan. Fungsi utamanya adalah sebagai tempat penyimpanan data atau program yang pernah digunakan komputer. Maka dari itulah, pengaturan di komputer tetap tersimpan meski komputer dalam keadaan mati sekalipun. Setelah komputer dinyalakan, memori internal memanggil semua data atau program yang tersimpan, menjalankan pengaturan awal antarmuka pengguna, kemudian menjalankan sistem operasi dasar (OS), termasuk semua utilitas perangkat lunak (software) yang ter-*install*.

Memori internal juga digunakan untuk menyimpan instruksi atau data yang sedang diproses. Data yang diolah komputer tersimpan didalam memori internal, untuk kemudian dapat digunakan lagi pada proses berikutnya. Data yang tersimpan tersebut bersifat sementara, dan akan tergantikan oleh data baru jika sudah tidak digunakan. Hal tersebut berjalan terus menerus selama komputer digunakan, dan akan berhenti jika komputer telah dimatikan.

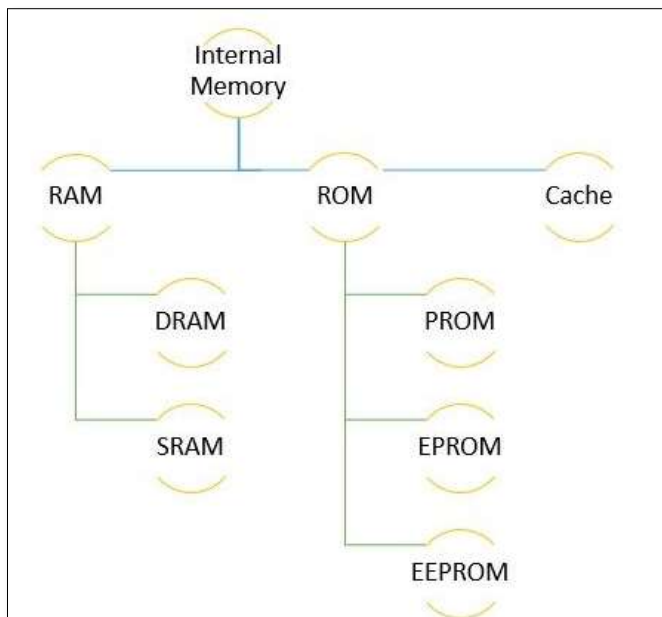
Memori internal juga ada yang bertindak sebagai penyimpan data (*storage*) yang bersifat tetap atau tidak dapat diubah. Sekali data tersimpan maka data tersebut akan tersimpan

selamanya, atau dapat diperbarui dengan menghapus data yang lama.

4.3 Jenis-jenis Memori Internal

Memori internal komputer terbuat dari bahan semikonduktor (silikon). Memori ini lebih mahal dan biasanya berukuran kecil dibandingkan dengan memori eksternal. Disebut memori internal karena memori ini diakses oleh prosesor melalui bus sistem secara langsung tanpa melalui *input-output* komputer.

Memori internal dibedakan menjadi dua macam: *Read Only Memory* (ROM) dan *Random Access Memory* (RAM). Selain itu, terdapat juga memori yang disebut *Cache Memory*.



Gambar 4.1 : Jenis Memori Internal

4.3.1 Read Only Memory (ROM)

ROM adalah sebuah memori semikonduktor berupa perangkat keras chip sebagai media penyimpanan yang hanya dapat dibaca. Memori ini biasanya digunakan untuk menyimpan program utama dari sistem. Memori ini bersifat *Non-Volatile* atau tidak tergantung oleh catu daya. Data akan tetap tersimpan meski komputer mati atau ketika daya terputus secara tidak terduga, sehingga lebih aman karena anda tidak perlu khawatir kehilangan data di tengah jalan.

ROM digunakan untuk menyimpan instruksi atau informasi secara permanen oleh pembuat (*vendor*). Berisi data atau program permanen yang sering digunakan untuk mengontrol fungsi dari utilitas perangkat keras yang dimaksudkan, dan tidak memerlukan modifikasi atau personalisasi.



Gambar 4.2 : ROM BIOS
(Sumber : wikipedia.org)

Salah satu jenis data yang biasa disimpan dalam chip ROM adalah *Basic Input-Output System* (BIOS). Secara umum, tugas BIOS disebut sebagai proses *booting*. *Bootting* adalah aktivitas perangkat

lunak komputer yang mencakup proses identifikasi, konfigurasi, pengujian, dan menghubungkan perangkat keras ke sistem operasi setelah komputer dinyalakan.

ROM memiliki sifat permanen yang tidak bisa diubah, dan isian datanya telah ditentukan oleh pembuat (*vendor*). Diperlukan pemrograman ulang jika ada kesalahan atau perubahan data pada ROM. Kelemahan ini dapat diantisipasi dengan jenis ROM yang dapat diprogram. Berikut beberapa jenis ROM pada komputer yaitu :

1. *Programmable ROM (PROM)*

PROM pada dasarnya adalah versi kosong dari ROM, yang kemudian dapat diisi dengan program oleh pemakai. Penulisan program dapat dilakukan secara elektrik dengan menggunakan perangkat tambahan. Sama halnya dengan ROM, PROM tidak bisa dihapus dan hanya bisa ditulis satu kali. Sehingga jika terjadi kesalahan pada saat pemrograman, maka PROM tidak bisa digunakan lagi dan harus ganti dengan yang baru.

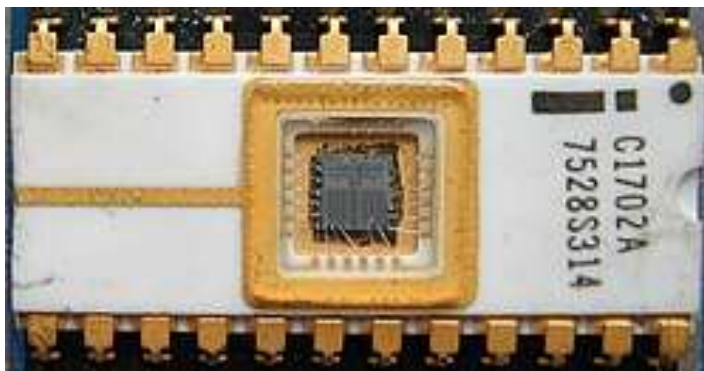


Gambar 4.3 : Chip PROM Texas Instrumen TBP18SA030N
Sumber : wikipedia.org

2. *Erasable Programmable ROM (EPROM)*

EPROM adalah ROM yang dapat dihapus dan diperbarui berulang kali. Memiliki jendela kuarsa untuk menghapus isi

data atau program dan mengembalikan ke keadaan awalnya (*reset*) dengan cara memaparkan memori ke sinar ultraviolet. Sinar ini akan membakar semua muatan kecil di EPROM untuk membuka kembali sirkuitnya dan membuat memori kembali kosong. Setelah itu memori dapat diprogram ulang dengan cara seperti pemrograman pada PROM. Penggunaan (hapus-isi) berulang-ulang dapat menyebabkan keausan pada sirkuit memori, namun EPROM memiliki masa pakai lebih dari 1000 kali penghapusan.



Gambar 4.4 : Chip EPROM Intel C1702A
Sumber : wikipedia.org

3. *Electrically Erasable Programmable* ROM (EEPROM)

Berbeda dengan EPROM dimana seluruh data terhapus saat terkena sinar ultraviolet. EEPROM merupakan memori yang dapat dihapus secara selektif dan ditulisi kapan saja tanpa menghapus isi sebelumnya. Sehingga tidak perlu menghapus dan menulis ulang semuanya untuk membuat satu suntingan. Penulisan data atau program pada memori dapat dilakukan tanpa peralatan tambahan dan juga dapat dihapus tanpa melepas chip memori dari perangkat kerasnya.



Gambar 4.5 : Chip EEPROM STMicro M24C02
Sumber : wikipedia.org

4.3.2 *Random Access Memory (RAM)*

RAM adalah memori semikonduktor berupa perangkat keras chip sebagai media penyimpanan yang diakses secara acak. Karakteristik penyimpanannya dapat dibaca dan ditulis. Digunakan untuk menyimpan data di lokasi yang dapat diakses dengan cepat, memungkinkan program berjalan dengan cepat, dan penghitungan data yang besar dapat dilakukan secara efisien

Berfungsi untuk menerima instruksi atau program, mengatur data yang sedang diproses, dan menyimpan data yang sedang menunggu untuk dikirim ke media keluaran (*output*) penyimpanan sekunder, atau perangkat komunikasi. Hal tersebut dapat dilihat dari struktur RAM sebagai berikut :

- *Input storage*, digunakan untuk menampung masukan dari perangkat *input*.
- *Program Storage*, bagian dari RAM yang berfungsi sebagai tempat penyimpanan instruksi program yang akan digunakan.

- *Working Storage*, bagian dari memori yang bertugas menyimpan data yang akan diolah dan hasil pengolahannya.
- *Output Storage*, berfungsi untuk menyimpan hasil akhir pengolahan data yang akan di tampilkan ke perangkat *output*.

RAM dapat dianggap sebagai wilayah kerja komputer, dan juga salah satu faktor yang mempengaruhi kecepatan sebuah komputer. RAM memiliki kecepatan tertentu yang menentukan seberapa cepat informasi dapat dibaca dan ditulis ke dalamnya, sehingga pemrosesan data dapat dilakukan secara cepat dan efisien. Semakin besar kapasitas memori maka kinerja komputer akan semakin cepat.



Gambar 4.6 : Jenis RAM
(Sumber : selasar.com)

Data yang tersimpan dapat hilang selama proses berlangsung karena memori ini bersifat volatile, artinya penyimpanan data tergantung pada catu daya, dimana data yang tersimpan akan hilang ketika aliran listrik hilang. RAM terkadang juga disebut sebagai *main memory*, *primary storage*, atau *internal storage*.

Berdasarkan dari cara kerjanya, RAM dibagi menjadi dua jenis yaitu *Static* RAM (SRAM) dan *Dynamic* RAM (DRAM).

1. *Dynamic* RAM (DRAM)

RAM dinamik menyimpan data berbentuk muatan listrik pada kapasitor yang tersusun sebagai sel-sel elektrik. Nilai kapasitor ditafsirkan sebagai bilangan biner, bernilai 1 jika bertegangan tinggi dan bernilai 0 jika bertegangan rendah. Kapasitor memiliki sifat alamiah melepaskan muatannya, sehingga data yang disimpan akan hilang. Oleh karena itu, kapasitor harus disegarkan (*refresh*) secara teratur untuk menjaga penyimpanan data, hal ini yang membuatnya sangat dinamis.

Dapat dikategorikan dalam beberapa tipe, diantaranya adalah

a. *First Page Mode* DRAM (FPM DRAM)

Merupakan bentuk awal dari RAM yang populer saat ini, mempunyai kecepatan transfer data lambat yaitu sekitar 176 MB per detik. Rentang frekuensi kinerja FPM DRAM di antara 16 MHz sampai 66 MHz.

b. *Extended Data Output* DRAM (EDO DRAM)

Merupakan jenis chip yang mampu mengoptimalkan waktu membaca memori pada prosesor komputer. Dikembangkan tahun 1995, EDO RAM memiliki 72 pin dan kecepatan transfer data mencapai 66 Mhz. Menggantikan primary memori terdahulu yakni FPM RAM. Memori jenis ini dapat melakukan proses simpan dan ambil data dalam waktu bersamaan, sehingga kecepatan baca tulis pada memori ini bisa lebih cepat.

c. *Synchronous* DRAM (SDRAM)

Digunakan antara tahun 1996 hingga 2003, merupakan jenis RAM yang disinkronisasikan oleh *clock system* pada bus CPU. Kecepatannya antara 100 MHz sampai 133 MHz dan dapat menampung memori hingga 1GB.

Memiliki 168 Pin saluran transfer data dan terdapat dua celah (*notch*) dibagian kaki atau konektornya.

d. Rambus DRAM (RDRAM)

Merupakan teknologi memori dinamis buatan Rambus yang diperkenalkan pada tahun 1995. RAM ini bekerja pada frekuensi sekitar 133 MHz dengan kecepatan akses data sampai 1.6 GB per detik. RDRAM biasanya digunakan pada kartu grafis akselerator (video dan permainan), untuk memori *cache*, atau proses penyimpanan memori pada sistem yang mempunyai kinerja tinggi seperti server dan workstation. Slot memori untuk RDRAM adalah 184 pin.

e. *Single Data Rate* SDRAM (SDR SDRAM)

Merupakan perkembangan dari SDRAM konvensional dengan kecepatan data yang lebih tinggi. Proses instruksi memori ini yaitu satu kali baca dan satu kali tulis per siklus waktu. Biasa digunakan pada konsol permainan video. Memiliki slot memori 184 pin dan dua celah (*notch*) pada bagian konektornya.

f. *Double Data Rate* SDRAM (DDR SDRAM)

DDR (Double Data Rate) SDRAM merupakan teknologi lanjutan dari SDR SDRAM, yang dapat bekerja dua kali lebih cepat. RAM ini mampu memproses dua instruksi sekaligus (dua kali baca dan dua kali tulis) dalam satuan siklus waktu. Konsumsi daya listrik yang lebih rendah sehingga lebih efisien, dan kapasitas memori yang cukup besar hingga 4 GB per chipnya. Memiliki 184 pin dan satu celah pada konektornya.

DDR SDRAM mengalami peningkatan pada generasi penerusnya, yaitu DDR2 dan DDR3 dan DDR4. Generasi terbaru dari DDR SDRAM ini dapat bekerja semakin cepat pada kecepatan *clock* yang lebih tinggi (hingga

800MHz), dan lebih efisien karena penggunaan daya yang semakin kecil (1,5 V).

2. *Static* RAM (SRAM).

Sesuai namanya SRAM adalah jenis RAM dengan metode statis untuk penyimpanan datanya, dan data tetap terjaga (tidak berubah) selama terdapat daya listrik. Tidak memerlukan kapasitor yang membutuhkan penyegaran secara berulang-ulang, sehingga kinerja bisa lebih cepat dan daya yang dibutuhkan lebih kecil dibandingkan DRAM. Penyimpanan data RAM statik dengan cara menyimpan nilai biner menggunakan konfigurasi gerbang logika flip-flop. Harga yang mahal dan ketersediaanya hanya dalam kapasitas kecil maka penggunaan memori ini pada bagian yang terpenting saja, seperti memori *cache* CPU dan *buffer Hard Drive*.

Berdasarkan penjelasan diatas, maka dapat dilihat perbedaan fitur tiap tipe memori sebagai berikut:

Tabel 4.1 : Perbedaan Tipe Memori

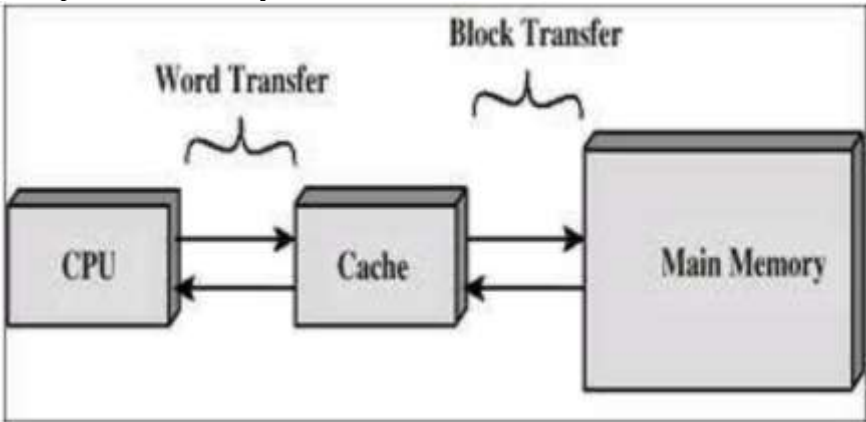
Type	Volatility	Writeable	Max Erase Cycles	Speed	Category	Erasure
ROM	NV	N	NA	Fst	Read only	Not Posible
PROM	NV	Once, with device	NA	Fast	Read only	Not Posile
EPROM	NV	Yes, with device	Limited	Fast	Read-mostly	UV light, chip-level
EEPROM	NV	Y	Limited	Fast to read, slow to write	Read mostly	Electricall y, byte-level
SRAM	V	Y	Unlimited	Fast	Readwrite	Electricall y, byte-level
DRAM	V	Y	Unlimited	Moderate	Readwrite	Electricall y, byte-

Type	Volatility	Writeable	Max Erase Cycles	Speed	Category	Erasure level

Sumber:barrgroup.com

4.3.3 Cache Memory

Memori *cache* dapat didefinisikan sebagai memori yang lebih cepat, lebih kecil, namun lebih mahal di bandingkan memori utama maupun memori sekunder komputer. Berfungsi menyimpan informasi dari program, aplikasi, dan data yang sering digunakan. Memori *cache* berada antara memori utama dan prosesor. Hal ini berfungsi agar memori utama tidak langsung diakses oleh CPU, tetapi melalui *cache* memori dahulu yang mempunyai kecepatan akses lebih tinggi. Setiap prosessor mengakses data pada komputer pertama kali, komputer membuat salinan data pada *cache*, kemudian ketika hendak mengakses kembali data tersebut, prosessor akan mengecek salinan data tersebut dan mengaksesnya duluan tanpa harus mengakses memori utama, sehingga lebih cepat dan efisien. Metode ini akan meningkatkan kinerja sistem dan membuat tampilan memori utama kelihatan lebih cepat dari kecepatan sebenarnya.



Gambar 4.7 : Letak Memori Cache

Cache memory digunakan untuk sinkronisasi data dengan CPU dan memastikan bahwa data segera tersedia saat CPU membutuhkan data. Berikut jenis-jenis memori *cache*:

1. *L1 Cache*

Cache Level 1 adalah jenis memori *cache* level pertama yang bekerja dengan kecepatan yang sama dengan prosesor. *Cache* ini terintegrasi ke dalam prosesor, misalnya sebuah CPU memiliki empat inti prosesor (*quad-core*) maka *cache* level 1 nya juga berjumlah empat yang tertanam di setiap inti prosesor-nya. Ukuran memori berkisar 2-64 KB. Memiliki dua jenis *cache*, yaitu *cache* instruksi yang menyimpan instruksi yang dibutuhkan oleh CPU dan *cache* data yang menyimpan data yang dibutuhkan oleh CPU.

2. *L2 Cache*

Disebut *cache* level 2 atau *cache* L2. Terletak pada internal CPU ataupun eksternal CPU. Semua inti CPU dapat memiliki *cache* L2 terpisah atau dapat saling berbagi. Jika berada di luar (eksternal), maka L2 *cache* dihubungkan dengan CPU menggunakan sistem bus berkecepatan tinggi. L2 *cache* memiliki kecepatan yang sedikit lebih lambat dari *cache* L1, dan mempunyai ukuran bervariasi antara 256-512 KB.

3. *L3 Cache*

Dikenal sebagai *cache* level 3 atau *cache* L3. Tidak semua prosesor terdapat *cache* L3, hanya beberapa prosesor kelas atas yang dapat memiliki jenis *cache* ini. Digunakan untuk menambah kinerja *cache* L1 dan L2. Biasa terletak di luar CPU (eksternal) dan digunakan bersama-sama oleh semua inti CPU. Ukuran memori bervariasi dari 1-8 MB. *Memory cache* ini lebih cepat dari RAM (*Random Access Memory*), namun lambat dari *cache* L1 dan L2.

Saat CPU membutuhkan data, tahap awal CPU akan mengakses *cache* L1. Apabila di L1 tidak ditemukan datanya, maka pemeriksaan data di lanjutkan ke *cache* L2. Jika masih tidak ditemukan juga maka pencarian beralih ke *cache* L3. Apabila data yang dicari dapat ditemukan di dalam memori *cache*, maka disebut sebagai *cache hit*. Sebaliknya, jika data yang dibutuhkan tidak ditemukan di dalam memori *cache* disebut *cache miss*.

Tahap berikutnya jika terjadi *cache miss* (data tidak ada dalam salah satu *cache memory*), maka penelusuran data dilanjutkan ke *Random Access Memory* (RAM). Jika tidak ada juga data tersebut dalam RAM, maka data tersebut diambil dari *Hard Disk Drive*.

4.4 Modul Memori Internal

Modul memori adalah papan sirkuit tempat memori terintegrasi atau dipasang. Modul memori memungkinkan pemasangan dan penggantian yang mudah dalam sistem elektronik, terutama komputer seperti komputer pribadi, workstation, dan server. Modul memori pertama dibuat untuk model komputer dari produsen tertentu, sehingga tidak semua sistem bisa menggunakannya. *Joint Electron Device Engineering Council* (JEDEC) yaitu badan internasional yang menentukan standar IC (*Integrated Circuit*) kemudian membuat standarisasi pada modul memori sehingga dapat digunakan dalam sistem apa pun yang menggunakannya. Karakteristik yang membedakan setiap modul memori antara lain tegangan, kapasitas, kecepatan (bit rate), dan faktor bentuk. Jenis - jenis modul memori dapat dibagi sebagai berikut:

1. *Dual Inline Package* (DIP)

DIP adalah dua baris paralel pin penghubung. Memori dapat dipasang melalui lubang ke *Printed Circuit Board* (PCB) atau dimasukkan ke dalam soket. Pin DIP sangat halus dan sulit untuk memasangnya, karena harus ditusukkan ke dalam

soket dan pin mudah tidak sejajar dan cenderung bengkok. Jadi, ketika pin ini dikeluarkan dari soket, pin harus diluruskan kembali yang dapat menyebabkan kerusakan pada memori dan membuatnya tidak dapat digunakan kembali.

2. *Single In-Line Memory Modules (SIMM)*

SIMM adalah papan sirkuit kecil yang memiliki konektor tepi tempat chip RAM ditempatkan. Slot tersedia pada motherboard untuk memasukkan SIMM. Konektor SIMM dan slot yang terletak di motherboard harus terbuat dari logam yang sama, yaitu emas atau timah. Jika konektor SIMM dari emas maka konektor slot juga harus dari emas, bukan dari logam lain. Konektor logam yang ada di setiap sisi tepi bawah bekerja secara efektif melalui papan kartu, dan hanya satu set konektor yang berfungsi pada satu waktu. Terdapat dua varian SIMM, yaitu 30 pin dan 72 pin.

- SIMM 30 pin memiliki lebar alamat 8 bit dan RAM 1MB atau 4 MB. Namun modul ini juga mempunyai bit paritas untuk deteksi kesalahan yang membuat lebar alamat menjadi 9 bit. Pemasangan SIMM yang benar dapat memperhatikan lekukan di kiri bawah.
- SIMM 72 pin berisi lebar alamat 32 bit atau 36 bit termasuk bit paritas. Setiap byte diberikan bit paritas (32 bit data, 4 bit paritas). Besaran memori RAM yang dimilikinya bisa 4, 8, 16, 32, atau 64 MB. Pemasangan dengan memperhatikan lekuk di samping dan tengah modul.

3. *Dual In-Line Memory Module (DIMM)*

DIMM berisi serangkaian sirkuit terpadu DRAM. Modul terpasang ke papan sirkuit tercetak (PCB), dengan beberapa chip RAM pada papan sirkuit tunggal, yang terhubung ke motherboard. Memiliki konektor logam yang mirip dengan SIMM, tetapi salah satu sisi konektor tidak bergantung pada

yang lain. Selain itu modul ini menggunakan jalur data 64-bit, karena prosesor yang digunakan pada komputer saat ini memiliki lebar data 64-bit.

DIMM juga memiliki papan sirkuit yang jauh lebih kecil dan penyisipan yang lebih mudah dibandingkan dengan SIMM. Motherboard tingkat lanjut menggunakan DIMM 168, 184, 240 pin. Ini mengkonsumsi daya 3,3 volt dan dapat menyimpan dari 32 MB hingga 1GB memori.

4. *Small outline Dual Inline Memory Module (SODIMM)*

Pada dasarnya SO-DIMM adalah versi kecil dari DIMM, diperuntukkan untuk perangkat yang memiliki ukuran kecil seperti model laptop/notebook, high-end printer dan perangkat keras jaringan seperti router. SO-DIMM mempunyai jumlah pin yang bervariasi yaitu 72, 100, 144, 200 dan pin 204, dimana setiap variasi jumlah pin memiliki performa yang berbeda. SO-DIMM mendukung transfer data 32-bit data pada modul yang memiliki pin berjumlah 72 dan 100, sedangkan transfer data 64-bit data pada modul yang memiliki pin 144, 200 dan 204.

Speksifikasi SO-DIMM sama seperti DIMM pada umumnya, hanya saja terdapat versi *low voltage* seperti DDR3L, DDR4L dan sejenisnya. Dari segi fisik ukurannya lebih kecil dari memori PC biasa dikarenakan penyesuaian pada perangkat yang akan dipasang (laptop), adapun ukurannya setengah ukuran dari DIMM biasa.

Tabel 4.2 : Perbandingan SIMM dan DIMM

Perbandingan	SIMM	DIMM
Slot konektor	Pin dikedua sisi terhubung.	Pin bersifat mandiri
Transfer data	32bit	64bit
Daya	5 Volt	3.3 Volt
Kapasitas	4MB sampai 64MB	32MB sampai 1GB
Pengaplikasian	CPU 486 dan komputer	PC Pentium

	pentium generasi awal	modern
--	-----------------------	--------

Sumber : nooblasto.com

DAFTAR PUSTAKA

- Ardyanto. 2023, 25 Januari. SIMM dan DIMM: Pengertian, Perbedaan dan Jenis-Jenisnya. Nooblasto.com
- Barr, Michael. 2023, 26 Januari. Types of Memory in Embedded Systems. Barrgroup.com
- Kusumah, Geolana Wijaya. 2023, 25 Januari. Jenis RAM. Selasar.com
- Patterson, D. A., & Hennessy, J. L. 2018. Computer Organization and Design: The Hardware Software Interface - RISC-V Edition. In Elsevier.
- Setiawan, Eko Budi. 2010. Computer Organization. Bandung : Indonesian Computer University
- Seliwati. 2022. Pengenalan Teknologi Komputer. Indie Press.
- Stallings, William. 2016. Computer Organization and Architecture Designing for Performance, 10th Edition. New Jersey: Prentice-Hall

BAB 5

EKSTERNAL MEMORY

Oleh Zul Rachmat

5.1 Pendahuluan

Komputer merupakan kumpulan perangkat elektronik yang saling berhubungan antara satu komponen dengan komponen lainnya yang dapat digunakan untuk memasukkan data dan memprosesnya menjadi informasi. Komputer terdiri dari beberapa komponen penunjang utama, diantaranya yaitu perangkat *input*, *Central Processing Unit*, *Storage Device* atau media penyimpanan, dan perangkat *output*.

Memori merupakan istilah umum yang merujuk pada media penyimpanan data sementara pada komputer. Setiap data yang diproses oleh processor akan disimpan di dalam memori yang bersifat sementara, karena data yang disimpan di dalamnya tersimpan selama komputer masih dialiri daya, ketika komputer dimatikan maka data yang disimpan di dalam memori fisik akan hilang. Oleh karena itu, sebelum mematikan komputer, seluruh data yang belum disimpan agar disimpan ke dalam media penyimpanan permanen yang umumnya berbasis disk, sehingga data tersebut dapat dibuka kembali.

Memori biasa disebut juga dengan istilah *storage* yang digunakan sebagai media penyimpanan data dan informasi pada saat komputer digunakan. Memori termasuk bagian yang sangat penting di dalam komputer yang berada di dalam CPU (*Central Processing Unit*). Media penyimpanan tambahan disebut dengan penyimpanan eksternal yang dapat menyimpan data walaupun aliran listrik dimatikan. Setiap penyimpanan eksternal memiliki

kelebihan dan kekurangan, serta bersifat relatif sesuai dengan kebutuhan pemakai.

Dari sekian banyaknya komponen yang ada pada komputer, memori menjadi salah satu perangkat yang penting untuk menyimpan dan menganalisis data, jadi dapat dikatakan bahwa memori merupakan komponen vital karena berperan penting untuk menentukan kinerja dan performa perangkat komputer. Semakin tinggi kapasitas dan kecepatan memori, maka semakin baik pula kinerja sebuah perangkat pada komputer.

5.2 Pengertian Memori Eksternal

Memori menjadi salah satu komponen yang sangat penting di dalam *Central Processor Unit* (CPU) untuk mengoperasikan perangkat komputer. Setiap data dan program yang akan diproses oleh *processor* semuanya akan disimpan ke dalam memori. Data yang sifatnya sementara akan hilang secara otomatis jika perangkat dimatikan, sedangkan data yang bersifat permanen akan tetap ada walaupun aliran listrik dimatikan.

Memori komputer secara umum dapat diartikan sebagai perangkat keras atau *hardware* yang digunakan untuk menyimpan data baik berupa informasi atau instruksi yang bersifat sementara atau permanen. Memori merupakan kumpulan dari beberapa *storage units* yang menyimpan data binary dalam bentuk bit. Jika prosesor tidak memerlukan item tertentu dalam jangka waktu yang lebih lama, program atau data maka akan secara otomatis tersimpan ke dalam memori permanen atau memori sekunder, kemudian data akan dipanggil dari memori sekunder ke memori utama untuk dilakukan eksekusi.

Pada dasarnya memori secara fisik yang digunakan pada komputer atau laptop dibagi menjadi 2 (dua) bagian yaitu memori internal yang disimpan secara langsung oleh bagian prosesor dan memori eksternal sebagai perangkat tambahan atau pendukung dari komputer. Memori eksternal merupakan memori tambahan

yang berfungsi sebagai penyimpanan data atau program dalam media fisik yang berbentuk kaset atau *disk* agar tetap mengalir transistor sehingga tetap dapat menyimpan data. Memori eksternal pada umumnya melakukan operasi penulisan, pembacaan, dan penyimpanan data di luar dari memori utama komputer dengan tujuan untuk membantu fungsi RAM. (Jarti and Hutabri, 2022)

Memori eksternal adalah perangkat penyimpanan data yang terpisah dari memori internal pada perangkat elektronik seperti smartphone, komputer, tablet, dan lainnya. Fungsinya adalah sebagai penambahan memori untuk menyimpan data, seperti file foto, video, musik, dokumen, dan lainnya.

Beberapa kegunaan memori eksternal antara lain :

- a. Kapasitas penyimpanan yang besar
Memori eksternal memiliki kapasitas penyimpanan yang lebih besar dibandingkan memori internal, sehingga dapat menyimpan lebih banyak data.
- b. Mudah dibawa
Memori eksternal dapat dengan mudah dibawa-bawa ke mana saja, sehingga dapat membantu anda menyimpan data pribadi dan membawa data penting kemana saja.
- c. Fleksibilitas
Memori eksternal dapat digunakan pada berbagai perangkat elektronik seperti smartphone, komputer, tablet, dan lainnya.
- d. Keamanan Data
Memori eksternal memiliki fitur proteksi data seperti password, sehingga data yang disimpan aman dari pihak yang tidak berhak.

Memori eksternal pada dasarnya bertujuan untuk menyimpan data yang bersifat tetap (*non volatile*) baik pada saat komputer aktif ataupun tidak. Selain itu, memori eksternal dapat membantu dalam mendapatkan memori dengan harga yang terjangkau atau cukup murah namun memiliki kapasitas yang

tinggi untuk penggunaan dalam jangka waktu lama. Contohnya seperti *harddisk*, *flashdisk*, atau *floppy disk*.

Secara umum, memori eksternal memiliki fungsi sebagai penambahan memori untuk menyimpan data dan mempermudah mobilitas data.

5.3 Jenis - Jenis Memori Eksternal

Jenis-jenis penyimpanan eksternal tidak selalu sama dan selalu mengalami perkembangan di setiap zamannya. Terdapat beberapa memori eksternal yang dulu sangat populer seperti disket, namun kini mulai dilupakan. Seiring perkembangan teknologi, bentuk dan ukuran memori saat ini berbeda dengan sebelumnya dan pada umumnya semakin kecil namun memiliki kapasitas memori yang besar.

Berikut jenis-jenis memori eksternal, diantaranya :

A. *Floppy Disk*

Floppy disk yang biasa disebut juga dengan disket memiliki bentuk persegi dan memiliki cakram tipis magnetis di dalamnya yang bertujuan untuk menyimpan data serta mudah untuk dibawa kemana-mana. *Floppy disk* sangat populer di masa dahulu namun seiring perkembangan teknologi kini sudah mulai ditinggalkan, salah satu alasannya yakni ukuran kapasitas memori dari *floppy disk* yang sangat kecil, biasanya hanya beberapa *megabyte* saja. Selain memiliki akses data yang lambat, komputer zaman sekarang juga sudah tidak menyediakan lagi pembaca disket pada CPU.



Gambar 5.1 : Floppy Disk
Sumber : Wikipedia

B. *Flashdisk*

Flashdisk merupakan media penyimpanan data atau memori eksternal yang dapat dihubungkan pada *port USB* dan mampu menyimpan berbagai format data seperti Doc, Mp3, JPG, dan lain-lain. merupakan hasil perkembangan dari pembaharuan Floppy disk yang memiliki fungsi sama yaitu sebagai alat penyimpanan data yang dapat dibawa kemana-mana dengan mudah karena bentuknya yang kecil. Dari segi ukuran, flashdisk lebih kecil dibandingkan dengan *floppy disk*, namun memiliki kapasitas penyimpanan yang lebih besar. *Flashdisk* menjadi media penyimpanan paling populer pada tahun 2010-an, namun flashdisk memiliki kekurangan yakni cukup rentan terkena oleh virus sehingga data yang ada di dalamnya tidak dapat terbaca bahkan hilang secara permanen.



Gambar 5.2 : Flashdisk
Sumber : Wikipedia

C. Hardisk

Banyak yang mengira jika *Hardisk* merupakan memori internal karena berada di dalam prosesor, namun saat ini juga dapat digunakan sebagai memori eksternal layaknya *flashdisk* yang dapat dibawa kemana-mana tetapi dengan kapasitas memori yang jauh lebih besar. *Hardisk* dapat digunakan sebagai media penyimpanan berupa aplikasi, *software*, dan juga file-file lainnya. Dengan kapasitas yang besar, hardisk sering digunakan juga untuk mem-*backup* data yang ada pada komputer atau laptop sehingga kinerja komputer dapat menjadi ringan dan cepat.



Gambar 5.3 : Harddisk
Sumber : Wikipedia

D. Solid-State Drive

Solid-State Drive (SSD) adalah jenis memori eksternal yang menggunakan teknologi *solid-state drive* untuk menyimpan data. SSD adalah alternatif untuk hard disk tradisional dan menawarkan beberapa keuntungan, seperti:

- a. Kecepatan : SSD memiliki kecepatan akses data yang lebih cepat dibandingkan dengan *hard disk* tradisional. Ini membuat SSD lebih baik untuk aplikasi yang membutuhkan akses data cepat, seperti memulai sistem operasi dan membuka aplikasi.

- b. Responsifitas : SSD memiliki waktu respon yang lebih cepat dibandingkan dengan *hard disk*, membuat perangkat yang menggunakan SSD lebih responsif dan memuaskan digunakan.
- c. Tahan terhadap kerusakan : SSD tidak memiliki bagian bergerak, sehingga lebih tahan terhadap kerusakan fisik dan vibrasi dibandingkan dengan hard disk tradisional.
- d. Konsumsi daya : SSD membutuhkan lebih sedikit daya dibandingkan dengan *hard disk* tradisional, membuat perangkat yang menggunakan SSD lebih hemat daya.
- e. Bentuk : SSD memiliki bentuk yang lebih *compact* dan ringan dibandingkan dengan *hard disk* tradisional, membuatnya lebih mudah untuk dibawa kemana saja.

Namun, SSD Eksternal juga memiliki beberapa kekurangan, seperti harga yang lebih mahal dibandingkan dengan *hard disk* tradisional dan kapasitas yang lebih terbatas. Oleh karena itu, pilihan tergantung pada kebutuhan dan budget setiap pengguna.



Gambar 5.4 : Solid-State Drive

Sumber : Wikipedia

E. *Compact Disk*

Compact disk (CD) merupakan salah satu memori eksternal yang digunakan pada komputer yang memiliki fungsi yang

sama dengan flashdisk yaitu untuk menyimpan dan memindahkan file, serta mudah dibawa kemana saja. CD umumnya berbentuk seperti cakram optik yang terdiri dari beberapa jenis seperti CD-R, CD-RW, dan CD-Worm. CD saat ini juga sudah mulai ditinggalkan sebab proses penggunaannya yang cukup rumit, selain itu tidak semua komputer atau laptop yang masih memiliki fitur CD Drive.



Gambar 5.5 : Compact Disk
Sumber : Wikipedia

F. Digital Versatile Disk

Digital Video Disk (DVD) adalah generasi lanjutan dari teknologi penyimpanan dengan menggunakan media optical disc. DVD memiliki kapasitas yang jauh lebih besar dibandingkan dengan CD, yakni dapat mencapai 9 Gigabyte. Penggunaan DVD banyak diminati dan dimanfaatkan secara luas oleh perusahaan musik dan film sejak diperkenalkan pertama kali. DVD memiliki keunggulan yaitu kemampuan untuk menghasilkan kualitas video dan audio yang lebih baik, namun hanya dapat dibuka dengan menggunakan DVD drive.



Gambar 5.6 : Digital Versatile Disk
Sumber : Wikipedia

G. Memory Card

Memory card merupakan media penyimpanan eksternal yang banyak digunakan pada smartphone dan kamera. Memori memiliki ukuran yang kecil, tetapi kapasitasnya sama dengan flash disk.



Gambar 5.7 : Memory Card
Sumber : Wikipedia

H. Cloud Storage

Cloud Storage atau penyimpanan awan merupakan salah satu media penyimpanan digital yang menggunakan koneksi jaringan untuk mengakses data yang tersimpan. *Cloud storage* adalah solusi terbaru dalam dunia penyimpanan data yang menawarkan banyak keunggulan dibandingkan media penyimpanan lainnya, diantaranya memastikan data yang tersimpan aman. Data yang tersimpan pada cloud storage

secara otomatis tersimpan pada sejumlah server yang dikelola oleh penyedia hosting. (Suparyanto dan Rosad, 2020)



Gambar 5.8 : Cloud Storage
Sumber : Wikipedia

5.4 Karakteristik Memori Eksternal

Memori eksternal merupakan media penyimpanan data yang terpisah dari memori internal pada perangkat seperti komputer, laptop, smartphone, tablet, kamera digital, dan perangkat lainnya. Memori eksternal umumnya memiliki kapasitas penyimpanan yang lebih besar dibandingkan memori internal, serta dapat dengan mudah dipindahkan dari satu perangkat ke perangkat lainnya.

Berikut adalah beberapa karakteristik dari memori eksternal:

a. Kapasitas Memori

Memori eksternal memiliki kapasitas yang lebih besar dibandingkan dengan memori internal pada perangkat

elektronik, seperti komputer atau smartphone sehingga dapat menampung lebih banyak data.

b. Portabilitas

Memori eksternal dapat mudah dipindahkan dari satu perangkat ke perangkat lain, sehingga dapat memudahkan pengguna dalam mengakses data dimana saja dan kapan saja jika diperlukan.

c. Kecepatan

Kecepatan akses data pada memori eksternal bisa lebih cepat dibandingkan dengan memori internal. Kecepatan transfer data dapat bervariasi tergantung pada jenis memori dan jenis koneksi yang digunakan, misalnya USB, microSD, atau SSD.

d. Fleksibilitas

Memori eksternal dapat menyimpan berbagai jenis file, seperti dokumen, gambar, video, dan musik.

e. Kompatibilitas

Memori eksternal dapat digunakan pada berbagai perangkat, seperti komputer, laptop, tablet, dan smartphone, selama perangkat tersebut memiliki port USB.

f. Harga

Harga memori eksternal sangat bervariasi, tergantung pada kapasitas, merk, dan spesifikasi produk.

g. Keamanan

Data pada memori eksternal bisa terlindungi dengan menggunakan *password* atau enkripsi. Namun, memori eksternal masih memiliki risiko kehilangan atau kerusakan fisik. Beberapa jenis memori eksternal dapat tahan terhadap guncangan dan getaran, sehingga data yang tersimpan di dalamnya dapat tetap aman dan tidak rusak.

DAFTAR PUSTAKA

- Jarti, N. and Hutabri, E. 2022. *Arsitektur Dan Organisasi Komputer*, CV. Eureka Media Aksara. Purbalingga: CV. Eureka Media Aksara.
- Suparyanto dan Rosad. 2020. *Manajemen Sistem Informasi*, Suparyanto dan Rosad. Edited by A. Yanto. Padang: PT. Global Eksekutif Teknologi.

BAB 6

CARA KERJA PERALATAN INPUT DAN PERALATAN OUTPUT

Oleh Wahyuddin S

6.1 Pendahuluan

Peralatan *input* dan *output* (I/O) pada komputer adalah komponen yang memungkinkan komputer untuk berkomunikasi dengan dunia luar. Peralatan *input* membantu komputer menerima data dan perintah, sementara peralatan *output* membantu menampilkan hasil dari proses komputer.

Peralatan *input* seperti *keyboard*, *mouse*, *scanner*, *microphone*, dll, mengirimkan data ke komputer melalui *port input*, seperti USB, PS/2, atau *port* lainnya. Data ini kemudian diterjemahkan oleh sistem operasi komputer menjadi sinyal digital yang dapat dikenali oleh komputer.

Peralatan *output* seperti *monitor*, *printer*, *speaker*, dll, menerima data digital dari komputer melalui *port output*, seperti VGA, HDMI, atau port lainnya, dan mengubahnya menjadi bentuk yang dapat diterima oleh manusia.

Dalam operasi normal, sistem operasi komputer mengendalikan peralatan I/O, memastikan bahwa data yang tepat diteruskan ke peralatan *input* atau *output* yang sesuai. Ini memastikan bahwa komputer dapat berinteraksi dengan dunia luar dengan cara yang efisien dan efektif (Mayefis et al. 2023).

6.2 Komponen Utama Sistem Komputer

Secara umum, pengertian komputer adalah suatu perangkat elektronik yang dapat digunakan untuk mengelola data sesuai dengan prosedur yang telah dirumuskan sebelumnya sehingga dapat menghasilkan informasi yang bermanfaat oleh pengguna. Komputer merupakan alat elektronik yang terdiri dari rangkaian berbagai komponen yang saling terhubung sehingga membentuk suatu sistem kerja. Sistem pada komputer dapat melakukan pekerjaan secara otomatis berdasarkan program yang diperintahkan kepadanya sehingga komputer mampu menghasilkan informasi berdasarkan instruksi dan program yang ada.

Pada sistem komputer terdapat tiga elemen utama yang memiliki peranan penting dalam sebuah proses pengolahan data pada komputer agar dapat menghasilkan sebuah informasi yang baik dan valid sesuai dengan kebutuhan pengguna (Wahyuddin 2023). Berikut tiga elemen pada komputer yaitu:

6.2.1 *Hardware* (Perangkat Keras)

Hardware (perangkat keras) pada komputer adalah segala hal berupa fisik yang membentuk komputer seperti CPU, *motherboard*, memori, *hardisk*, *monitor*, *keyboard*, *mouse* dan lainnya. Perangkat keras menentukan spesifikasi teknik komputer seperti kecepatan, kapasitas penyimpanan dan kapasitas grafis. Perangkat keras juga berperan dalam memastikan bahwa komputer dapat melakukan tugas yang diberikan dan berinteraksi dengan dunia luar melalui peralatan *input* dan *output*. Perangkat keras merupakan bagian terpenting dari sistem komputer dan harus bekerja dengan baik dan sesuai spesifikasi agar komputer berfungsi dengan baik.

6.2.2 Software (Perangkat Lunak)

Software (perangkat lunak) pada komputer merupakan segala hal yang tidak bisa dilihat atau diraba secara fisik seperti sistem operasi, aplikasi program dan data. Perangkat lunak memastikan bahwa komputer dapat melakukan tugas yang diberikan dan membuat interaksi dengan komputer lebih mudah bagi pengguna.

Sistem operasi seperti Windows, MacOS atau linux merupakan contoh dari perangkat lunak yang memastikan bahwa aplikasi dapat berjalan dengan lancar. Aplikasi seperti *Microsoft Office*, *Adobe Creative Suite* atau *Game* merupakan contoh dari perangkat lunak yang memungkinkan komputer dapat melakukan tugas spesifik seperti mengetik, mengedit gambar atau bermain *game*.

Perangkat lunak dapat diinstall atau dihapus dari sistem komputer sesuai dengan kebutuhan dari pengguna. Perangkat lunak sangat penting karena dapat memastikan bahwa komputer dapat melakukan tugas yang diberikan dan membuat interaksi dengan komputer lebih mudah bagi pengguna (Iskandar *et al.* 2022).

6.2.3 Brainware (Pengguna)

Brainware (pengguna) pada komputer adalah sebutan untuk manusia yang bekerja sebagai operator atau pengguna komputer. *Brainware* memainkan peran penting dalam menentukan bagaimana komputer digunakan dan bagaimana hasil yang diinginkan dicapai. *Brainware* memiliki kemampuan untuk memahami, menggunakan dan memecahkan masalah dengan menggunakan komputer.

Brainware juga bertanggung jawab untuk memastikan bahwa komputer bekerja dengan baik dan sesuai dengan yang diinginkan. Hal ini dapat dilakukan dengan memahami bagaimana perangkat keras dan perangkat lunak bekerja Bersama-sama dan

memastikan bahwa merak berfungsi dengan baik. *Brainware* juga bertanggung jawab untuk melakukan *backup data*, mempengaruhi perangkat lunak dan melakukan pemeliharaan lainnya untuk memastikan bahwa komputer berfungsi dengan baik.

Secara umum, *brainware* memainkan peran penting dalam memastikan bahwa komputer bekerja dengan baik dan dapat digunakan untuk mencapai tujuan yang diinginkan. *Brainware* memiliki kemampuan untuk memahami dan menggunakan teknologi dan bertanggung jawab untuk memastikan bahwa komputer bekerja dengan baik.

6.3 Peralatan *Input* Komputer

Peralatan *input* pada komputer adalah perangkat yang digunakan untuk memasukkan data dan informasi ke dalam komputer (Wikipedia n.d.-a). Beberapa contoh peralatan *input* komputer yaitu:

1. *Keyboard*

Keyboard adalah alat *input* yang digunakan untuk memasukkan teks dan perintah pada komputer.



Gambar 6.1 : *Keyboard* (Papan Ketik)

Sumber:

https://en.wikipedia.org/wiki/Computer_keyboard

2. *Mouse*

Mouse merupakan alat *input* yang digunakan untuk mengontrol *pointer* dan memilih item pada layar.



Gambar 6.2 : Keyboard (Papan Ketik)

Sumber: https://en.wikipedia.org/wiki/Computer_mouse

3. *Scanner*

Scanner merupakan alat *input* yang digunakan untuk memindai dokumen atau gambar dan memasukkannya ke dalam komputer.



Gambar 6.3 : Scanner (Pemindai)

Sumber: https://en.wikipedia.org/wiki/Image_scanner

4. *Webcam*

Webcam merupakan alat *input* yang digunakan untuk merekam video dan memasukkannya ke dalam komputer.



Gambar 6.4 : *Webcam* (Kamera Web)

Sumber: <https://en.wikipedia.org/wiki/Webcam>

5. *Joystick*

Joystick merupakan alat *input* yang digunakan untuk bermain game atau mengontrol objek pada layar.



Gambar 6.5 : *Webcam* (Tuas Kendali)

Sumber:

https://en.wikipedia.org/wiki/PlayStation_Analog_Joystick

6. *Touchpad*

Touchpad merupakan alat *input* yang mirip dengan *mouse*, tetapi memiliki permukaan sentuh sebagai alat kontrol *pointer*.



Gambar 6.6 : *Touchpad* (Bantalan Sentuh)

Sumber:

https://en.wikipedia.org/wiki/PlayStation_Analog_Joystick

7. *Flashdisk*

Flashdisk merupakan alat penyimpanan data memori kilat tipe NAND yang memiliki alat penghubung USB terpadu.



Gambar 6.7 : *Flashdisk* (Diska Lepas USB)

Sumber: https://en.wikipedia.org/wiki/USB_flash_drive

8. CD-ROM

CD-ROM merupakan alat yang digunakan untuk menyimpan data.



Gambar 6.8 : *CD-ROM* (Cakram Optik Digital)

Sumber: <https://en.wikipedia.org/wiki/CD-ROM>

Ini hanya beberapa contoh peralatan input komputer. Ada banyak peralatan input lain yang digunakan tergantung pada kebutuhan dan tujuan dari pengguna.

6.3 Peralatan *Output* Komputer

Peralatan *output* pada komputer adalah perangkat yang digunakan untuk menampilkan informasi dan data yang diproses oleh komputer (Detik.com n.d.-b). Beberapa contoh peralatan *output* komputer meliputi:

1. *Monitor*

Monitor merupakan alat *output* yang digunakan untuk menampilkan informasi visual pada layar.



Gambar 6.9 : Monitor

Sumber: https://id.wikipedia.org/wiki/Monitor_komputer

2. *Printer*

Printer merupakan alat *output* yang digunakan untuk mencetak dokumen atau gambar pada kertas.



Gambar 6.10 : Printer

Sumber:

https://simple.wikipedia.org/wiki/Computer_printer

3. *Plotter*

Plotter merupakan alat *output* yang digunakan untuk mencetak gambar teknis dan desain pada kertas atau film.



Gambar 6.11 : Plotter

Sumber: <https://en.wikipedia.org/wiki/Plotter>

4. *Speaker*

Speaker merupakan alat *output* yang digunakan untuk mengeluarkan suara dari komputer.



Gambar 6.12 : Speaker

Sumber:

https://id.wikipedia.org/wiki/Pengeras_suara_komputer

5. *Projector*

Projector merupakan alat *output* yang digunakan untuk memproyeksikan informasi visual pada layar atau dinding.



Gambar 6.13 : Projector

Sumber:

https://en.wikipedia.org/wiki/Digital_Light_Processing

6. *Headphones*

Headphones merupakan alat *output* yang digunakan untuk mendengarkan suara dari komputer dengan privasi.



Gambar 6.14 : Headphones

Sumber: <https://en.wikipedia.org/wiki/Headphones>

Ini hanya beberapa contoh peralatan *output* komputer. Ada banyak peralatan *output* lain yang digunakan tergantung pada kebutuhan dan tujuan dari pengguna.

Kesimpulan cara kerja peralatan *input* dan *output* pada komputer adalah sebagai berikut:

1. Peralatan *input* berfungsi untuk memasukkan data dan informasi ke dalam komputer. Contoh peralatan *input* adalah *keyboard*, *mouse*, *scanner*, *mikrofon*, dan lain-lain.
2. Peralatan *output* berfungsi untuk menampilkan informasi yang diproses oleh komputer. Contoh peralatan *output* adalah *monitor*, *printer*, *speaker*, dan lain-lain.
3. Data dan informasi yang dimasukkan melalui peralatan *input* diteruskan ke CPU (*Central Processing Unit*) yang merupakan inti dari sistem komputer. CPU memproses data dan informasi tersebut sesuai dengan instruksi yang diterima.
4. Hasil dari proses tersebut kemudian diteruskan ke peralatan *output* untuk ditampilkan pada layar atau dicetak pada kertas.
5. Peralatan *input* dan *output* bekerja secara berkesinambungan dan saling berkaitan untuk memastikan sistem komputer dapat bekerja dengan baik dan memenuhi kebutuhan pengguna.

DAFTAR PUSTAKA

- Wikipedia. n.d.-a. "Masukan/Keluaran - Wikipedia Bahasa Indonesia, Ensiklopedia Bebas." 2023. Retrieved February 5, 2023 (<https://id.wikipedia.org/wiki/Masukan/keluaran>).
- Detik.com. n.d.-b. "Mengenal Perangkat Input Dan Output Dalam Komponen Komputer." 2023. Retrieved February 5, 2023 (<https://www.detik.com/edu/detikpedia/d-5807479/mengenal-perangkat-input-dan-output-dalam-komponen-komputer>).
- Iskandar, Akbar, Bias Yulisa Geni, Citra Nurina Prabiantissa, Didi Kurnaedi, S. Wahyuddin, Khairunnisa Samosir, Tutuk Indriyani, Denny Alfian, S. Nurmuslimah, and Agus Supriyadi. 2022. *Pengantar Jaringan Komputer*. Get Press.
- Mayefis, Reska, Tri Yusranto, Nono Heryana, Allans Prima Aulia, Angga Aditya Permana, Sitti Aisa, Johni S. Pasaribu, S. Wahyuddin, and Fredy A. H. Sihombing. 2023. *Rekayasa Perangkat Lunak*. Global Eksekutif Teknologi.
- Wahyuddin, S. 2023. "1.5 Tujuan Sistem Operasi." *Sistem Operasi Komputer* 10.

BAB 7

DUKUNGAN SISTEM OPERASI DALAM KINERJA KOMPUTER

Oleh Zen Munawar

7.1 Pendahuluan

Sebagian besar tujuan utama dari Sistem Operasi (SO) adalah penggunaan sumber daya perangkat lunak dan perangkat keras sistem komputer secara efisien. Bab ini membahas tentang kinerja sistem komputer. Dimulai dengan pertimbangan tentang perlunya pemanfaatan sumber daya komputer yang seimbang, yang memberikan perspektif yang berguna. Selanjutnya melihat desain organisasi komputer kontemporer yang dimaksudkan untuk memberikan kinerja guna memenuhi permintaan saat ini dan yang akan datang. Alat dan model yang telah dikembangkan untuk menyediakan sarana penilaian kinerja sistem komputer komparatif.

Tahun demi tahun, biaya sistem komputer terus menurun drastis, sementara kinerja dan kapasitas sistem tersebut terus meningkat secara dramatis. Laptop saat ini memiliki kekuatan komputasi mainframe IBM dari beberapa belas tahun lalu. Dengan demikian, memiliki daya komputer yang “bebas” secara virtual. Prosesor sangat murah sehingga sekarang bisa memiliki mikroprosesor. Revolusi teknologi yang berkelanjutan ini telah memungkinkan pengembangan aplikasi dengan kompleksitas dan kekuatan yang mencengangkan. Misalnya, aplikasi desktop yang membutuhkan daya besar dari sistem berbasis mikroprosesor saat ini seperti pemrosesan gambar, perenderan tiga dimensi,

pengenalan ucapan, konferensi video, penulisan multimedia, anotasi file suara dan video, dan pemodelan simulasi.

Sistem workstation sekarang mendukung aplikasi teknik dan ilmiah yang sangat canggih dan memiliki kapasitas untuk mendukung aplikasi gambar dan video. Selain itu, bisnis mengandalkan server yang semakin kuat untuk menangani transaksi dan pemrosesan basis data dan untuk mendukung jaringan klien/server besar yang telah menggantikan pusat komputer mainframe besar di masa lalu. Selain itu, penyedia layanan cloud menggunakan bank server berkinerja tinggi yang sangat besar untuk memenuhi aplikasi bervolume tinggi dan bertransaksi tinggi untuk spektrum klien yang luas.

Apa yang menarik tentang semua ini dari perspektif organisasi dan arsitektur komputer adalah bahwa, di satu sisi, blok bangunan dasar untuk keajaiban komputer saat ini hampir sama dengan komputer dari setengah abad yang lalu, sementara di sisi lain tangan, teknik untuk kinerja maksimum dari bahan yang ada menjadi semakin canggih. Saat maju melalui berbagai elemen dan komponen komputer, dua tujuan dikejar. Bab ini menjelaskan fungsionalitas mendasar di setiap area yang sedang dibahas, dan kedua mengeksplorasi teknik-teknik yang diperlukan untuk mencapai kinerja maksimal.

7.2 Sistem Operasi MS Windows

Karena Sistem Operasi (SO) Microsoft Windows adalah SO yang paling banyak digunakan untuk komputer pribadi, mereka harus memenuhi kebutuhan semua jenis pengguna sistem komputer. Dibandingkan dengan Windows XP, versi SO Windows; yaitu Windows Vista, Windows 7, Windows 8 dan Windows 10, memperkenalkan sejumlah fitur dan penyempurnaan baru. Selain itu, peningkatan kinerja diberlakukan sebagai salah satu tujuan desain utama untuk Windows Vista dan Windows 7. Dalam bab ini menyajikan evaluasi kinerja dari tiga versi SO Microsoft untuk

komputer pribadi; yaitu Windows XP, Windows Vista dan Windows 7.

7.2.1 Kinerja Sistem Operasi

Pengukuran kinerja SO dilakukan melalui seperangkat aplikasi benchmark di lingkungan yang terkendali. Untuk memastikan hasil pengukuran kinerja yang akurat, andal, dan berulang, kami telah membuat proses pengukuran kinerja dan model evaluasi kinerja. Penekanan khusus diberikan pada area evaluasi dengan dampak terbesar pada performa: penjadwalan CPU, manajemen memori, manajemen subsistem grafis, manajemen hard disk drive, dan performa jaringan. Untuk mengetahui kinerja OS Windows di lingkungan yang berbeda, pengukuran kinerja dilakukan dalam tiga percobaan. Hasil percobaan menunjukkan bahwa Windows Vista dan Windows 7 memiliki beberapa peningkatan kinerja pada sistem komputer high-end yang berdiri sendiri, tetapi Windows XP mengungguli Windows Vista dan Windows 7 pada sistem komputer low-end yang berdiri sendiri. Selanjutnya, pada sistem komputer jaringan Windows Vista dan Windows 7 menunjukkan peningkatan kinerja jaringan terutama untuk lalu lintas dengan paket berukuran sedang. Sistem operasi perlu melakukan adaptasi untuk mengakomodasi tuntutan arsitektur modem berperforma tinggi (Chen *et al.*, 1995).

Sistem Operasi (SO) memperluas mesin dan memberi pemrogram cara yang lebih sederhana untuk bekerja dengan perangkat keras. Karena mengelola sumber daya dalam ruang dan waktu, pemrogram dan pengguna tidak perlu mengalokasikan sumber daya sistem ke aplikasi. SO yang menyediakan penggunaan sumber daya perangkat lunak dan perangkat keras yang lebih efisien dari sistem komputer mencapai kinerja yang lebih baik. Untuk setiap versi baru dari sebuah OS diharapkan memiliki banyak fitur baru dan performa yang lebih baik dari versi sebelumnya. Ketika Microsoft® merilis dua versi terakhir

Windows® OS untuk komputer pribadi, yaitu Windows Vista™ pada tahun 2007 dan setelahnya Windows 7™ pada tahun 2009, sejumlah besar perhatian diberikan pada fitur baru mereka, seperti yang dijelaskan dalam (Hassell and Campbell, 2007; Bott, Siechert and Stinson, 2012). Namun, evaluasi kinerja dan perbandingan dengan pendahulunya tidak dilakukan dalam literatur. Meskipun salah satu tujuan desain terpenting untuk Windows Vista, selain fitur dan keamanan baru, adalah peningkatan kinerja, setelah dirilis sebagian besar keluhan terkait dengan kinerjanya. Salah satu alasan utamanya adalah rata-rata sistem komputer tidak cukup kuat untuk memanfaatkan semua fitur Windows Vista.

Berbeda dengan yang disebutkan di atas, setelah dirilis, peningkatan kinerja adalah karakteristik Windows 7 yang paling sering disebutkan dan dipuji. Sementara itu, pertanyaan abadi muncul di antara pengguna: "SO Windows mana untuk komputer pribadi yang memiliki kinerja terbaik?" Selain itu, berbagai perbandingan berbagai versi SO Windows untuk komputer pribadi telah muncul di Internet (Schmid, 2007; M, 2009; Schmerer, 2009). Namun dari pengukuran dan perbandingan kinerja tersebut tidak dapat ditarik kesimpulan yang representatif karena tidak memiliki proses pengukuran kinerja dan metodologi evaluasi yang jelas dan terpadu. Selain itu, dalam literatur hanya ada beberapa studi ilmiah tentang kinerja OS Windows, evaluasinya dan perbandingannya dengan berbagai versi OS Windows untuk komputer pribadi. Akibatnya, tidak ada metode, proses, atau pendekatan terpadu atau standar untuk pengukuran dan evaluasi kinerja OS Windows. Oleh karena itu, motivasi utama dalam makalah ini adalah untuk mengembangkan model pengukuran, evaluasi dan perbandingan kinerja OS Windows yang akan disatukan dan dapat digunakan kembali. Selain itu, tujuannya adalah menggunakan model ini untuk evaluasi kinerja tiga versi terakhir OS Windows untuk komputer pribadi.

7.2.2 Evaluasi Kinerja

Evaluasi kinerja dari tiga versi 32-bit terbaru dari SO Windows; yaitu Windows XP® Professional SP3 (Service Pack 3), Windows Vista Business SP2 dan Windows 7 Professional. Tujuan utamanya adalah untuk menentukan mana dari tiga versi terakhir OS Windows ini yang menangani sumber daya sistem lebih efisien, dan akibatnya memiliki kinerja yang lebih baik. Masalah utama pengukuran dan evaluasi kinerja SO Windows adalah bahwa tidak ada aplikasi benchmark yang bertujuan untuk mengukur kinerja SO Windows secara langsung. Ada banyak aplikasi benchmark yang mengukur kinerja sistem komputer, tetapi sebagian besar digunakan untuk mengukur kinerja perangkat keras sistem komputer. Namun, dalam lingkungan yang terkontrol, aplikasi tolok ukur ini dapat digunakan untuk mengukur kinerja SO Windows yang berbeda.

Tujuh aplikasi tolok ukur berbeda yang tersedia di (Niemelä, 2005; Renqvist and Kallinen, 2006; Botta, Dainotti and A, 2007; SPEC, 2009). Enam aplikasi tolok ukur pertama digunakan untuk mengukur kinerja OS Windows dalam bekerja dengan komponen perangkat keras yang berdampak paling besar pada kinerja sistem komputer yang berdiri sendiri: memori, CPU (Central Processing Unit), subsistem grafis, dan HDD (Hard Disk Drive). Aplikasi benchmark ketujuh digunakan untuk mengukur kinerja jaringan OS Windows. Untuk memastikan hasil pengukuran kinerja yang andal dan konsisten, kami telah menciptakan proses pengukuran kinerja. Evaluasi kinerja OS dapat dilakukan dengan teknik yang meliputi perancangan percobaan, pemodelan analitis atau simulasi (Fortier and Michel, 2003). Model evaluasi kinerja yang memiliki sejumlah besar indikator kinerja yang dapat dibandingkan antara SO Windows yang berbeda. Untuk mengetahui kinerja SO Windows di lingkungan yang berbeda, pengukuran kinerja dilakukan dalam tiga percobaan. Windows yang lebih baru membawa peningkatan kinerja di bidang jaringan.

7.2.3 Faktor Kinerja dan Atribut Sistem

Lima faktor kinerja dalam persamaan sebelumnya (I_c , p , m , k , t) dipengaruhi oleh empat atribut sistem: desain set instruksi (dikenal sebagai arsitektur set instruksi); teknologi kompiler (seberapa efektif kompiler dalam menghasilkan program bahasa mesin yang efisien dari program bahasa tingkat tinggi); implementasi prosesor; dan hierarki cache dan memori.

Tabel 7.1 : Faktor Kinerja dan Atribut Sistem

	<i>I_c</i>	<i>p</i>	<i>m</i>	<i>k</i>	<i>t</i>
Instruction set architecture	X	X			
Compiler technology	X	X	X		
Processor implementation		X			X
Cache and memory hierarchy				X	X

Sumber: (Stallings, 2016)

Tabel 7.1 merupakan matriks dimana satu dimensi menunjukkan lima faktor kinerja dan dimensi lainnya menunjukkan empat atribut sistem. X dalam sel menunjukkan atribut sistem yang memengaruhi faktor kinerja.

7.3 Dukungan Kinerja Pada SO Windows

SO memiliki dua peran utama (Tanenbaum, 2008). Pertama, menyediakan abstraksi perangkat keras mentah untuk program aplikasi. Kedua, mengelola sumber daya sistem seperti ruang memori, waktu CPU, sumber daya grafis, ruang penyimpanan file, dan perangkat I/O (Input/Output). Tujuan utama sebagian besar OS adalah eksekusi aplikasi pengguna yang efisien. OS yang

mengelola sumber daya sistem komputer dengan lebih baik akan memiliki kinerja yang lebih baik. Struktur OS dan kode sumber berdampak besar pada kinerja, tetapi perilaku OS bergantung pada beban kerja aktual dan persyaratan eksternal (Fortier and Michel, 2003). Tujuan umum SO, seperti Windows XP, Windows Vista dan Windows 7, bekerja dengan aplikasi bisnis dan desktop dan dirancang untuk berbagai pengguna. Selain komputer pribadi, mereka menemukan tempatnya di sistem komputer tertanam .

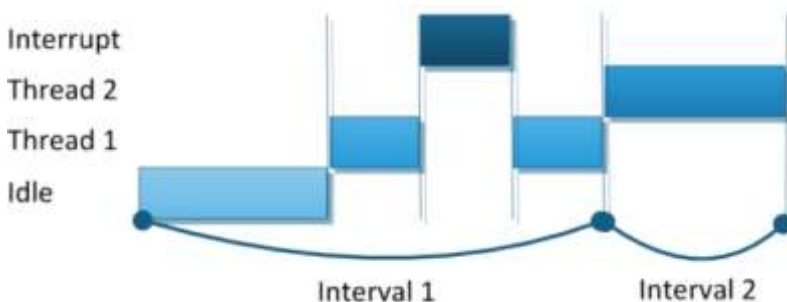
7.3.1 Teknik Kinerja Windows XP

Dibandingkan dengan versi sistem operasi Windows sebelumnya, tujuan desain Windows XP mencakup peningkatan keamanan dan keandalan, kemampuan aplikasi Windows dan POSIX (Sistem Operasi Portabel untuk UNIX), kinerja tinggi, ekstensibilitas, portabilitas, dan dukungan internasional (Tanenbaum, 2008). Salah satu tujuan desain terpenting Windows XP adalah kinerja tinggi. Windows XP meningkatkan kinerja dengan mengurangi panjang jalur kode dalam fungsi kritis, menggunakan algoritme yang lebih baik dan struktur data per prosesor, menggunakan pewarnaan memori untuk mesin NUMA (Non-Uniform Memory Access), menerapkan protokol penguncian yang lebih terukur, mengoptimalkan prioritas utas, dan memungkinkan multiprosesing simetris.

7.3.2 Peningkatan Kinerja Pada Windows Vista

Tujuan desain Windows Vista mencakup keamanan tingkat lanjut, kinerja dan keandalan yang ditingkatkan, antarmuka pengguna grafis yang ditingkatkan, transisi status daya yang ditingkatkan, dan pengenalan fitur baru seperti menyesuaikan perilaku pengguna dan menyelesaikan masalah secara otomatis. Salah satu tujuan desain terpenting adalah peningkatan kinerja tanpa meningkatkan persyaratan perangkat keras. Idennya adalah untuk meningkatkan daya tanggap SO dan mengoptimalkan manajemen memori. Windows Vista menambahkan banyak

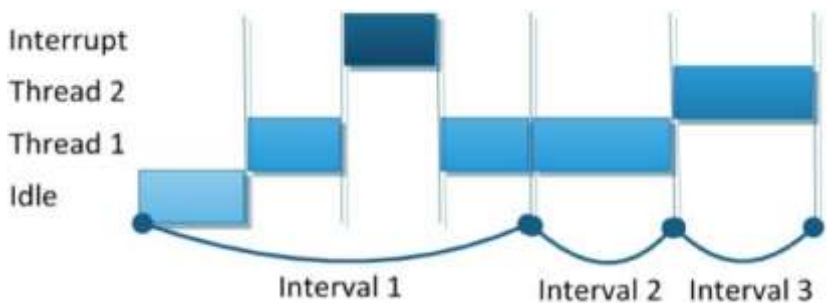
peningkatan pada proses, utas, dan bagian. Setiap proses memiliki satu atau lebih utas, yang menjadikan unit dasar yang dapat dieksekusi dan dikirim oleh kernel. Baik Windows XP maupun Windows Vista menggunakan algoritma prioritas dengan 32 tingkat prioritas di mana thread prioritas tertinggi selalu berjalan (Tanenbaum, 2008). Pada gambar. 7.1 menunjukkan penghitungan waktu CPU di Windows XP, yang tidak menyediakan penjadwalan thread yang adil (Rusinovich and Solomon, 2004).



Gambar 7.1 : Penjadwalan Thread Pada Windows XP

Sumber : (Rusinovich and Solomon, 2004)

Di Windows Vista, penjadwal menggunakan penghitung waktu siklus untuk penjadwalan CPU yang adil, seperti yang ditunjukkan pada gambar 7.2. Utas mendapat setidaknya satu putaran tanpa menghitung waktu interupsi (Rusinovich and Solomon, 2004). CPU adalah salah satu sumber daya terpenting dari sistem komputer dan kebijakan penjadwalan CPU OS memiliki pengaruh besar pada kinerja OS secara keseluruhan. Oleh karena itu, dalam pengukuran yang membutuhkan waktu CPU yang signifikan, Windows Vista akan menunjukkan hasil yang lebih baik.



Gambar 7.2 : Penjadwalan Thread Pada Windows Vista
 Sumber : (Rusinovich and Solomon, 2004)

Manajemen memori memengaruhi kinerja OS secara lebih signifikan (dalam hal throughput, waktu respons, beban kerja, dll.) daripada manajemen CPU (Peng, Li and Mili, 2007). Manajemen memori di Windows Vista benar-benar berubah dibandingkan dengan Windows XP, dan memiliki banyak peningkatan, seperti ruang alamat kernel dinamis, prioritas memori, peningkatan NUMA dan dukungan halaman besar, memori video paging, peningkatan bandwidth I/O, akses bagian, kekokohan dan diagnosa diri. Peningkatan ini akan menghasilkan operasi baca/tulis/salin memori yang lebih cepat dan latensi memori yang lebih singkat.

7.3.3 Peningkatan Kinerja Pada Windows 7

Windows 7 dibangun di atas basis kode Windows Vista dan oleh karena itu, menjaga kompatibilitas aplikasi. Selanjutnya, semua fitur dari Windows Vista dipertahankan di Windows 7 dan sebagian besar ditingkatkan untuk mendapatkan kinerja sistem yang lebih baik, efisiensi daya, skalabilitas, dan daya tanggap. Seperti dijelaskan dalam (Microsoft, 2009), tujuan desain yang mendasari Windows 7 adalah peningkatan kinerja dalam skenario pengguna utama dengan fokus pada respons pengguna. Peningkatan paling penting yang memengaruhi berbagai aspek kinerja dijelaskan di (Network, 2022).

Perubahan ReadyBoost termasuk dukungan untuk halaman yang didukung file halaman caching, penggunaan beberapa perangkat flash secara bersamaan dan dukungan untuk cache 32 GB. ReadyBoot ditingkatkan dengan menggunakan kompresi dan mengurangi jejak memori. Paralelisme yang meningkat dari inisialisasi driver dan logika dan mekanisme prefetching yang ditingkatkan akan menghasilkan proses booting yang lebih cepat. Penyempurnaan pengelola memori mencakup kebijakan penggantian halaman yang ditingkatkan, yang menggunakan nilai 3-bit (di Windows Vista ini adalah nilai 2-bit) untuk memutuskan halaman mana yang akan ditinggalkan dalam kumpulan proses kerja. Cache sistem, kumpulan paged, dan kode sistem yang dapat di-page sekarang masing-masing memiliki perangkat kerja mereka sendiri (di Windows Vista mereka berada di perangkat kerja yang sama). Juga, operasi registri ditingkatkan dengan menghapus pemetaan memori. Peningkatan DWM (Desktop Window Manager) memiliki konsumsi memori hingga 50% lebih kecil per jendela. SMT (*Simultaneous MultiThreading*) ditingkatkan dengan menggunakan SMT Parking sebagai panduan lebih lanjut untuk menghindari penggunaan pasangan logis. Karena memungkinkan kinerja yang lebih baik pada prosesor Intel multi-core hyper-threaded, ini akan meningkatkan kinerja Windows 7. Kunci operator kernel dihapus untuk beberapa operasi dan untuk yang lain digantikan oleh serangkaian teknik sinkronisasi dan status utas "pra-tunggu" yang baru. Manfaat utamanya adalah Windows 7 dapat menskalakan hingga 256 prosesor. UMS (Penjadwalan Mode Pengguna) meningkatkan kinerja dengan memisahkan utas mode pengguna dan utas mode kernel. Utas pengguna dapat dijadwalkan dalam mode pengguna tanpa transisi kernel.

Skalabilitas untuk aplikasi yang mengelola memori dalam jumlah besar ditingkatkan dengan menghapus kunci global PFN (Nomor Bingkai Fisik) pengelola memori. UBPM (*Unified Background Process Manager*) adalah komponen berbasis pemicu

yang memulai atau menghentikan layanan pada peristiwa tertentu. Ini meningkatkan kinerja dengan meminimalkan jumlah layanan yang berjalan di latar belakang.

DirectX 11 meningkatkan skalabilitas dan kinerja dengan memperkenalkan fitur-fitur baru seperti tessellation, multithreading, linkage shader dinamis, peningkatan kompresi tekstur dan shader komputer. Core Parking meningkatkan efisiensi daya dengan secara dinamis memilih satu set prosesor (socket) yang harus tetap diam, berdasarkan penggunaan terakhirnya dan dengan mempertahankan beban pada prosesor sesedikit mungkin. API manajemen pengatur waktu (*Application Programming Interface*) memiliki dua peningkatan yang mengurangi konsumsi daya. Yang pertama adalah ITTD (*Intelligent Timer Tick Distribution*), yang tidak mengizinkan interupsi timer pada prosesor yang sedang tidak digunakan, dan yang kedua adalah Penggabungan Timer, yang menggabungkan pemberitahuan timer aplikasi yang berbeda menjadi satu. MinWin adalah bagian bawah independen Windows 7 yang terdiri dari kernel, HAL (*Hardware Abstraction Layer*), TCP/IP (*Transmission Control Protocol/Internet Protocol*), sistem file, driver, dan layanan sistem inti. Manfaat utamanya adalah dapat dibangun, di-boot, dan diuji secara terpisah dari sistem lainnya.

DAFTAR PUSTAKA

- Bott, E., Siechert, C. and Stinson, C. 2012. *Windows 7 Insight Out Deluxe Edition*. Deluxe. Edited by C. Philips. Washington: Microsoft Press. Available at: <https://ptgmedia.pearsoncmg.com/images/9780735656925/samplepages/9780735656925.pdf>.
- Botta, A., Dainotti, A. and A, P. 2007. 'Multi-protocol and multi-platform traffic generation and measurement', in *Conference on Computer Communications*. Anchorage: IEEE Computer Society.
- Chen, J. B. *et al.* 1995. 'The Measured Performance of Personal Computer Operating Systems', *SIGOPS Operating Systems Review*, 29(5), pp. 299–313.
- Fortier, P. J. and Michel, H. E. 2003. *Computer Systems Performance Evaluation and Prediction*. First. Digital Press. doi: 10.1016/B978-1-55558-260-9.X5000-0.
- Hassell, J. and Campbell, T. 2007. *Windows Vista: Beyond the Manual*. New York: APress Media. LLC.
- M, W. 2009. *Windows compared: Windows 7 vs Vista vs XP, PC Plus*. Available at: <https://www.techradar.com/news/software/operating-systems/windows-compared-windows-7-vs-vista-vs-xp-615167> (Accessed: 1 November 2022).
- Microsoft. 2009. *Performance Testing Guide for Windows*, Microsoft Corporation. Available at: www.microsoft.com/whdc/system/sysperf/Win7perf.msp x.
- Network, M. D. 2022. *Microsoft Developer: Any platform. Your languages. Your tools.*, Microsoft Developer. Available at: <https://developer.microsoft.com/en-us/> (Accessed: 1 November 2022).

- Niemelä, S. 2005. 'PCMark ® 05 PC Performance Analysis', *Corporation, Futuremark*. First. Future Mark. Available at: <https://s3.amazonaws.com/download-aws.futuremark.com/pcmark05-whitepaper.pdf>.
- Peng, Y., Li, F. and Mili, A. 2007. 'Modeling the evolution of operating systems: An empirical study', *Journal of Systems and Software*, 80(1), pp. 1–15. doi: <https://doi.org/10.1016/j.jss.2006.03.049>.
- Renqvist, N. and Kallinen, M. 2006. *3DMark06 Whitepaper v1.0.2*. Finland: Futuremark Corporation. Available at: https://s3.amazonaws.com/download-aws.futuremark.com/3DMark06_Whitepaper.pdf.
- Russinovich, M. E. and Solomon, D. A. 2004. *Microsoft Windows Internals : Microsoft Windows Server 2003, Windows XP, and Windows 2000*. 4th edn. Washinton DC: Microsoft Press.
- Schmerer, K. 2009. *Benchmarks: Windows 7 RTM versus Vista, XP, ZdNet Inovation*. Available at: <https://www.zdnet.com/article/benchmarks-windows-7-rtm-versus-vista-xp/>.
- Schmid, P. 2007. *Windows XP vs. Vista: The Benchmark Rundown, Tom's Hardware*. Available at: <https://www.tomshardware.com/reviews/xp-vs-vista,1531.html> (Accessed: 5 November 2022).
- SPEC. 2009. *Standard Performance Evaluation Corporation, Spec*. Available at: <https://www.spec.org/gwpg/gpc.static/vp10results.html> (Accessed: 1 November 2022).
- Stallings, W. 2016. *Computer Organization and Architecture: Designing For Performance*. Tenth. Edited by T. Johnson. New Jersey: Pearson Education, Inc.
- Tanenbaum, A. S. 2008. *Modern Operating Systems*. 3rd edn. New York: Pearson Education, Inc.

BAB 8

DASAR ARITMATIKA KOMPUTER

Oleh Johni S Pasaribu

8.1 Pendahuluan

Struktur CPU adalah rangkaian logika sinkron yang memproses data yang dapat dibagi menjadi dua blok: *Control* dan *Datapath*. *Datapath* dan *Control* adalah kedua komponen yang bersama-sama secara kolektif dikenal sebagai prosesor (CPU) (Khan, 2018). Lima komponen klasik komputer adalah input, output, memori, *datapath* dan *control*, dengan dua komponen terakhir digabungkan dan disebut prosesor (CPU). Prosesor mendapatkan instruksi dan data dari memori. Input menulis data ke memori, dan output membaca data dari memori. *Control* mengirimkan sinyal yang menentukan operasi *datapath*, memori, input, dan output. Gambar 8.1 berikut menunjukkan organisasi standar sebuah komputer.



Gambar 8.1 : Organisasi komputer menunjukkan lima komponen klasik: input, output, memori, datapath dan control.
(Sumber: (Patterson, David A. and Hennessy, 2014))

Datapath adalah kumpulan unit fungsional seperti aritmatika logika unit (ALU), *register*, *shifter*, *counter*, *bus* dan *hardware part* yang melakukan operasi pemrosesan data. Bersamaan dengan *control unit* membentuk CPU. Setiap langkah (*fetch*, *decode*, *execute*, *save*) membutuhkan jalur komunikasi (*transfer data*) antara memori, register dan ALU. *Control* bertugas mengontrol perintah *datapath* oleh sinyal kontrol tentang kapan dan bagaimana aliran serta beroperasinya data. Sinyal kontrol dihasilkan oleh *Control Unit*.

Dua hal utama dalam aritmatika komputer adalah pengkodean maupun konversi sebuah angka baik dalam format biner, oktal, heksadesimal dan desimal dan operasi-operasi aritmatika dasar (tambah, kurang, perkalian dan pembagian). Kedua hal utama ini berlaku untuk aritmetika integer (bilangan bulat) dan floating-point (bilangan pecahan) (Stallings, 2010).

Radix (atau Basis): Gagasan di balik sistem penomoran adalah bahwa nilai numerik direpresentasikan melalui peningkatan pangkat radix (atau basis).

System	Radix	Allowable Digits
Decimal	10	0, 1, 2, 3, 4, 5, 6, 7, 8, 9
Binary	2	0, 1
Octal	8	0, 1, 2, 3, 4, 5, 6, 7

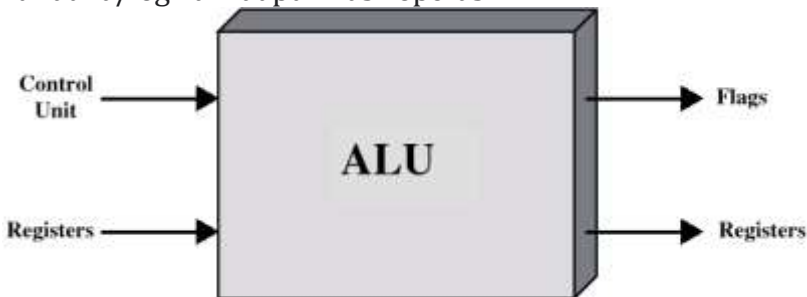
Powers of 2
$2^{-2} = \frac{1}{4} = 0.25$
$2^{-1} = \frac{1}{2} = 0.5$
$2^0 = 1$
$2^1 = 2$
$2^2 = 4$
$2^3 = 8$
$2^4 = 16$
$2^5 = 32$
$2^6 = 64$
$2^7 = 128$
$2^8 = 256$
$2^9 = 512$
$2^{10} = 1,024$
$2^{15} = 32,768$
$2^{16} = 65,536$

Decimal	4-Bit Binary	Hexadecimal
0	0000	0
1	0001	1
2	0010	2
3	0011	3
4	0100	4
5	0101	5
6	0110	6
7	0111	7
8	1000	8
9	1001	9
10	1010	A
11	1011	B
12	1100	C
13	1101	D
14	1110	E
15	1111	F

Hexadecimal	16	0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F
-------------	----	--

8.2 ALU (*Arithmetic Logic Unit*)

Salah satu bagian dalam dari mikroprosesor yang berfungsi untuk melakukan operasi aritmatika (tambah, kurang, perkalian dan pembagian) dan operasi logika (AND, OR, NOT, NAND, NOR, XOR, XNOR) pada data adalah ALU (*Arithmetic Logic Unit*). Seperti pada Gambar 8.2 berikut menunjukkan *Control Unit* memindahkan data antara register, ALU, dan memori. Input ke ALU adalah data yang akan diolah yang disebut operand dan *opcode* (kode operasi) menunjukkan operasi yang akan dilakukan; output ALU adalah hasil dari operasi yang dilakukan dimana kemudian hasilnya kembali disimpan dalam register. Register adalah lokasi penyimpanan sementara baik itu berupa operand untuk operasi aritmatika/logika maupun hasil operasi.

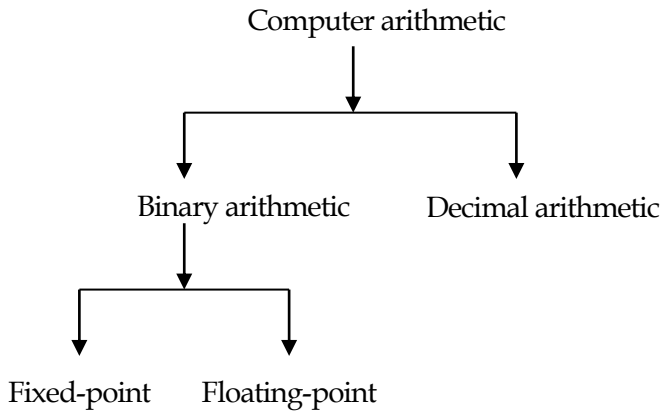


Gambar 8.2 : Input dan output dari ALU. (Sumber: (Stallings, 2010))

8.3 Representasi *Integer*

Karena rangkaian-rangkaian internal komputer merupakan dunia biner, maka komputer digital pasti melakukannya dengan aritmetika biner. Walaupun *user* menggunakan sistem bilangan desimal maka dapat dipastikan sistem desimal ini diubah ke dalam bilangan biner oleh komputer sebelum melakukan operasi-operasi aritmetika. Hasil akhir diubah kembali ke dalam desimal untuk diberikan kepada *user*. Seperti pada Gambar 8.3 menunjukkan

aritmetika biner dan desimal dimana pembahasan fokus pada aritmatika biner.



Gambar 8.3 Jenis-jenis aritmetika komputer

Sebuah bilangan bulat (*integer*) bisa bertanda (*signed integer*) ataupun tidak bertanda positif maupun negatif (*unsigned integer*).

8.3.1 Bilangan bulat Biner tak bertanda (*Unsigned Integer*)

Byte tidak bertanda menggunakan 8 bit seluruhnya dimana daerah nilainya antara 0 (0000 0000) hingga 255 (1111 1111), bukannya dari –128 hingga 127. Sedang untuk *word* (16 bit) tidak bertanda daerah nilainya antara 0 hingga 65,535.

Berikut ini beberapa istilah yang biasa digunakan dalam bilangan biner :

words	: istilah yang mewakili sebuah bilangan biner.
double word	: 32 bit bilangan biner
word	: 16 bit bilangan biner
byte	: 8 bit bilangan biner
nibble	: 4 bit bilangan biner

- bit : Tiap digit dalam bilangan biner
- MSB : Most Significant Bit (bit paling kiri)
- LSB : Least Significant Bit (bit paling kanan)

Contoh pada bilangan 16 bit (*word*):

1	0	0	1	0	1	1	1	1	1	1	1	0	0	0	1
MSB												LSB			
Nibble				Nibble				Nibble				Nibble			
Byte								Byte							
Word															

Proses mengubah atau konversi dari suatu basis bilangan ke basis bilangan lainnya:

Konversi dari bilangan biner (basis 2), oktal (basis 8) atau heksadesimal (basis 16) menjadi bilangan berbasis 10 (desimal) yaitu mengalikan setiap digit bilangan tersebut dengan basis bilangannya yang dipangkatkan 0, 1, 2 dst dimulai dari basis paling kanan. Konversi dari bilangan desimal menjadi biner, oktal atau heksadesimal yaitu membagi bilangan desimal itu dibagi dengan basis bilangan lainnya, misalnya basis n, yaitu membagi dengan n secara berulang sampai hasilnya sama dengan nol dan sisa hasil baginya disimpan. Sisa bagi ini kemudian diurutkan dari yang paling akhir hingga yang paling awal dan inilah yang merupakan hasil konversi bilangan tersebut. Konversi bilangan basis 8 (oktal) ke bilangan basis 2 (biner) yaitu setiap digit oktal akan langsung dikonversi ke biner lalu hasilnya digabungkan. Sebaliknya, konversi bilangan biner ke oktal yaitu dengan bagi setiap 3 digit biner menjadi sebuah angka oktal dimulai dari paling kanan. Konversi bilangan heksadesimal ke desimal yaitu dengan mengalikan digit bilangan heksa dengan pangkat bilangan 16 dari kanan ke kiri mulai dengan pangkat 0, 1, 2, dst. Konversi bilangan heksadesimal ke biner yaitu setiap digit heksadesimal langsung

dikonversi ke biner lalu hasilnya digabungkan. Konversi bilangan biner ke heksadesimal yaitu lakukan setiap 4 digit menjadi sebuah angka heksa dimulai dari paling kanan.

Contoh 8.1:

$$243.51_{10} = 2 * 10^2 + 4 * 10^1 + 3 * 10^0 + 5 * 10^{-1} + 1 * 10^{-2}$$

$$10110_2 = 1 * 2^4 + 0 * 2^3 + 1 * 2^2 + 1 * 2^1 + 0 * 2^0 = 22_{10}$$

$$6AF.3C = 6 * 16^2 + 10 * 16^1 + 15 * 16^0 + 3 * 16^{-1} + 12 * 16^{-2}$$

$$147_{10} = \mathbf{10010011_2}$$

2 147	1	↑
2 73	1	
2 36	0	
2 18	0	
2 9	1	
2 4	0	
2 2	0	
2 1	1	
0		

8.3.2 Bilangan bulat Biner bertanda (Signed Integer)

Semua representasi yaitu magnituda tanda, komplement satu, dan komplement dua menggunakan angka 1 pada posisi bit orde tinggi atau (*MSB: Most Significant Bit*) sebagai bilangan negatif sebaliknya jika angka 0 maka bilangan positif.

Contoh 8.2:

$$10001010 \text{ (bilangan negatif)} = -10$$

$$00001010 \text{ (bilangan positif)} = 10$$

Contoh 8.3 untuk m = 3:

Biner	Nilai	Biner	Nilai
000	0		
001	+1	101	-1
010	+2	110	-2
011	+3	111	-3

Terdapat masalah yaitu ada dua representasi dari 0:

$$00000000 = + 0_{10}$$

$$10000000 = - 0_{10}$$

Untuk menguji apakah suatu angka adalah 0 atau tidak, CPU perlu melihat apakah itu 00000000 atau 10000000. Bilangan $+ 0$ dan $- 0$ adalah sama bagi manusia sementara bagi komputer adalah beda, karena komputer membandingkan 2 buah bilangan tersebut secara bit per bit. Oleh karena itu, memiliki dua representasi 0 tidak nyaman.

Beberapa metoda yang digunakan untuk mengubah bilangan biner positif menjadi negatif:

1. Representasi magnitudo (nilai) tanda
2. Representasi komplemen satu
3. Representasi komplemen dua

8.3.2.1 Representasi Magnitudo (Nilai) Tanda

Angka magnitudo (nilai) tanda memiliki tanda sebagai bit paling kiri (juga disebut sebagai bit orde tinggi atau bit paling signifikan) sedangkan bit yang tersisa mewakili besaran (atau nilai absolut) dari nilai numerik:

- Jika bit tanda 0, bilangan positif;
- Jika bit tanda 1, bilangan negatif.

Representasi paling sederhana yaitu representasi nilai tanda dimana negasi suatu bilangan integer hanya dengan membalikkan bit tanda. Contoh 8.4:

$$\begin{array}{ll} +18 & = 00010010 \\ -18 & = 10010010 \quad (\text{sign magnitude}) \end{array}$$

N bit dapat mewakili $-(2^{n-1} - 1)$ hingga $2^{n-1} - 1$. Namun representasi magnituda tanda mempunyai persoalan yaitu:

- Operasi penjumlahan dan pengurangan membutuhkan:
- Mempertimbangkan tanda dan besaran setiap angka.

Contoh 8.5:

Penjumlahan: $01001111_2 + 00100011_2$ dengan aritmatika magnituda tanda.

$$01001111_2 (79_{10}) + 00100011_2 (35_{10}) = 01110010_2 (114_{10})$$

Tidak ada **overflow** dalam contoh ini:

$$\begin{array}{r} 0001111 \leq \text{carries} \\ 01001111 \quad (79) \\ + 00100011 \quad (35) \\ \hline 01110010 \quad (114) \end{array}$$

Contoh 8.6:

Penjumlahan: 01001111_2 to 01100011_2 dengan aritmatika magnituda tanda.

Diperoleh hasil yang salah:

$$01001111_2 (79_{10}) + 01100011_2 (99_{10}) = 0110010_2 (50_{10})$$

Terjadi kondisi **overflow** dan carry yang **dibuang**, sehingga menghasilkan jumlah yang **salah**.

$$\begin{array}{r} 1001111 \leq \text{carries} \\ 01001111 \quad (79) \\ + 01100011 \quad (99) \\ \hline 00110010 \quad (50) \end{array}$$

Pengurangan: Kurangi $01001111_2 (79_{10})$ dari $01100011_2 (99_{10})$ dengan aritmatika magnituda tanda.

Diperoleh hasil yang salah:

$$01100011_2 (99_{10}) + 11001111_2 (-79_{10}) = 0110010_2 (50_{10}) \text{ seharusnya } 00010100_2 (20_{10})$$

$$\begin{array}{r} 10001111 \leq \text{carries} \\ 01100011 \quad (99) \\ + 11001111 \quad (-79) \\ \hline 00110010 \quad (50) \end{array}$$

Contoh 8.7: Ada dua bilangan 0:

$$\begin{aligned} +0_{10} &= 00000000 \\ -0_{10} &= 10000000 \quad (\text{sign magnitude}) \end{aligned}$$

Karena kelemahan-kelemahan tadi, representasi magnitudo (nilai) tanda jarang digunakan di ALU.

8.3.2.2 Representasi Komplemen Satu (*One's complement*)

Dalam representasi yang berbeda, komplemen satu (*one's complement*) dari suatu bilangan diperoleh dengan membalikkan (menginvert) seluruh nilai bitnya ($1 \rightarrow 0$ dan $0 \rightarrow 1$). Bilangan positif direpresentasikan persis seperti dalam bentuk magnitudo tanda. Bilangan negatif direpresentasikan dalam bentuk komplemen 1. MSB akan menunjukkan tanda nomor tersebut: $0 \rightarrow$ positif, $1 \rightarrow$ negatif. Pada representasi ini, *carry* **tidak** dibuang tetapi ditambahkan.

Contoh 8.8: Nyatakan 23_{10} dan -9_{10} dalam bentuk komplemen satu biner 8-bit.

$$\begin{aligned} 23_{10} &= + (00010111_2) = 00010111_2 \\ -9_{10} &= - (00001001_2) = 11110110_2 \end{aligned}$$

Kerugian utama dari komplemen satu adalah kita masih memiliki dua representasi untuk nol: 00000000_2 dan 11111111_2 . Muncul kembali ambiguitas $+0$ dan -0

Contoh 8.9 untuk $m = 5$:

$$\begin{array}{rcl}
 \text{(a)} & \begin{array}{r} 00111 \quad (+7) \\ - \quad 00011 \quad (+3) \end{array} & \text{komplemen} \rightarrow + \begin{array}{r} 00111 \quad (+7) \\ 11100 \quad (-3) \\ \hline 1 \quad 00011 \\ + \quad \quad \quad 1 \\ \hline 00100 \quad (+4) \end{array} \\
 \text{(b)} & \begin{array}{r} 10101 \quad (-10) \\ + \quad 11100 \quad (-3) \end{array} & \begin{array}{l} \text{tetap} \rightarrow \\ \text{tetap} \rightarrow \end{array} + \begin{array}{r} 10101 \quad (-10) \\ 11100 \quad (-3) \\ \hline 1 \quad 10001 \\ + \quad \quad \quad 1 \\ \hline 10010 \quad (-13) \end{array} \\
 \text{(c)} & \begin{array}{r} 00111 \quad (+7) \\ - \quad 01011 \quad (+11) \end{array} & \text{komplemen} \rightarrow + \begin{array}{r} 00111 \quad (+7) \\ 10100 \quad (-11) \\ \hline 11011 \end{array} \\
 \text{(d)} & \begin{array}{r} 10000 \quad (-15) \\ + \quad 10000 \quad (-15) \end{array} & \begin{array}{l} \text{tetap} \rightarrow \\ \text{tetap} \rightarrow \end{array} + \begin{array}{r} 10000 \quad (-15) \\ 10000 \quad (-15) \\ \hline 1 \quad 00000 \\ + \quad \quad \quad 1 \\ \hline 00001 \quad (+1) \rightarrow ??? \end{array}
 \end{array}$$

Akibat dari kelemahan-kelemahan tersebut, komplemen 1 jarang digunakan di ALU.

8.3.2.3 Representasi Komplemen 2 (*Two's complement, Radix Complement*)

Pada representasi komplemen 2, bilangan positif direpresentasikan persis seperti dalam bentuk magnitudo tanda. Bilangan negatif direpresentasikan dalam bentuk komplemen 2. Bagaimana cara menghitung komplemen 2 dari sebuah angka? Lengkapi setiap bit angka ($1 \rightarrow 0$ dan $0 \rightarrow 1$), lalu tambahkan satu ke angka yang dihasilkan. MSB akan menunjukkan tanda angka tersebut: $0 \rightarrow$ positif, $1 \rightarrow$ negatif.

Contoh 8.10: Ekspresikan 23_{10} , -23_{10} , dan -9_{10} dalam bentuk komplement dua biner 8-bit.

$$23_{10} = + (00010111_2) = 00010111_2$$

$$-23_{10} = - (00010111_2) = 11101000_2 + 1 = 11101001_2$$

$$-9_{10} = - (00001001_2) = 11110110_2 + 1 = 11110111_2$$

Contoh 8.11: Tambahkan 9_{10} dengan -23_{10} dalam bentuk aritmatika komplement dua biner 8-bit:

$$00001001_2 (9_{10}) + 11101001_2 (-23_{10}) = 11110010_2 (-14_{10})$$

00 001001	<= Carries
00001001 ₂	(9)
<u>+ 11101001₂</u>	<u>+ (-23)</u>
11110010 ₂	(-14)

Contoh 8.12: Tambahkan 23_{10} dengan -9_{10} dalam bentuk aritmatika komplement dua biner 8-bit:

$$00010111_2 (23_{10}) + 11110111_2 (-9_{10}) = 00001110_2 (14_{10})$$

11 110111	<= Carries
00010111 ₂	(23)
<u>+ 11110111₂</u>	<u>+ (- 9)</u>
00001110 ₂	(14)

Pada penjumlahan dengan metoda two's complement, nilai carry tidak perlu lagi ditambahkan sehingga proses lebih singkat. Berikut perbandingan hasil antara bilangan biner tidak bertanda, komplement dua dan komplement satu:

$$\begin{aligned}
 1101_2 &= 13_{10} && \text{(a 4-bit unsigned number)} \\
 0\ 1101 &= +13_{10} && \text{(a positive number in 5-bit two's complement)} \\
 1\ 0010 &= -13_{10} && \text{(a negative number in 5-bit ones' complement)} \\
 1\ 0011 &= -13_{10} && \text{(a negative number in 5-bit two's complement)} \\
 \\
 0100_2 &= 4_{10} && \text{(a 4-bit unsigned number)} \\
 0\ 0100 &= +4_{10} && \text{(a positive number in 5-bit two's complement)} \\
 1\ 1011 &= -4_{10} && \text{(a negative number in 5-bit ones' complement)} \\
 1\ 1100 &= -4_{10} && \text{(a negative number in 5-bit two's complement)}
 \end{aligned}$$

8.3.3 Penjumlahan dan Pengurangan

Aritmatika Penjumlahan:

+	0	1
0	0	1
1	1	0*

* : melibatkan *carry* untuk kolom berikutnya

Contoh 8.13 untuk aritmatika m= 5 bit:

$$\begin{array}{lcl}
 \text{(a)} & \begin{array}{r} 00101 \ (+5) \\ + \quad 00110 \ (+6) \\ \hline 01011 \ (+11) \end{array} & \text{(b)} \quad \begin{array}{r} 00111 \ (+7) \\ + \quad 11110 \ (-2) \\ \hline 1\ 00101 \ (+5) \end{array} \\
 & & \text{carry register}
 \end{array}$$

$$\begin{array}{lcl}
 \text{(c)} & \begin{array}{r} 11011 \ (-5) \\ + \quad 11100 \ (-4) \\ \hline 1\ 10111 \ (-9) \end{array} & \text{Ke carry register}
 \end{array}$$

Pada bagian d berikut, komplemen 2 dari 00000 tetap hasilnya 00000.

$$\begin{array}{lcl}
 \text{(d)} & \begin{array}{r} 00000 \rightarrow \\ + \quad 11111 \\ \hline 1\ 00000 \end{array} & \rightarrow \text{tidak ada ambiguitas } +0 \text{ dan } -0 \\
 \text{(e)} & \begin{array}{r} 00101 \ (+5) \\ + \quad 01110 \ (+14) \\ \hline 10011 \ (-13) \end{array} & \rightarrow \text{Overflow}
 \end{array}$$

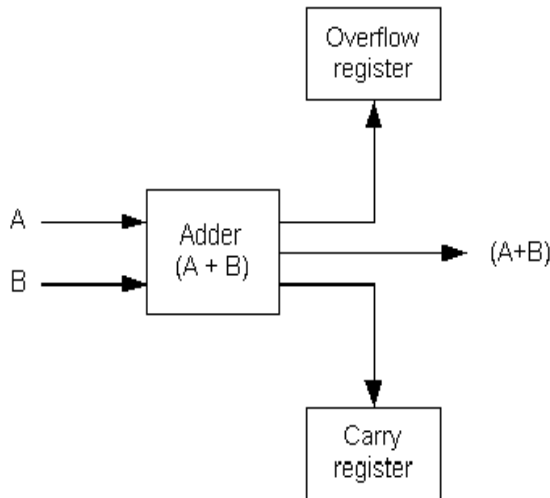
Skema yang paling umum untuk mengimplementasikan bagian *integer* dari ALU adalah representasi komplement dua.

Penjumlahan dilakukan seolah-olah kedua angka tersebut seperti bilangan bulat tak bertanda (*unsigned integers*) seperti diilustrasikan pada Gambar 8.4. Empat contoh pertama mengilustrasikan operasi yang berhasil dimana tidak terjadi *overflow*. Perhatikan bahwa bisa terdapat *carry* di luar word (ditunjukkan dengan bayangan), dimana *carry* ini diabaikan. Pada penjumlahan tertentu, hasilnya mungkin lebih besar dari ukuran word yang digunakan, ini disebut *overflow*. Bila dua bilangan ditambahkan, dan keduanya positif atau keduanya negatif, *overflow* akan terjadi bila hasilnya memiliki tanda yang berlawanan.

$1001 = -7$ $+0101 = 5$ $1110 = -2$ (a) $(-7) + (+5)$	$1100 = -4$ $+0100 = 4$ $10000 = 0$ (b) $(-4) + (+4)$
$0011 = 3$ $+0100 = 4$ $0111 = 7$ (c) $(+3) + (+4)$	$1100 = -4$ $+1111 = -1$ $11011 = -5$ (d) $(-4) + (-1)$
$0101 = 5$ $+0100 = 4$ $1001 = \text{Overflow}$ (e) $(+5) + (+4)$	$1001 = -7$ $+1010 = -6$ $10011 = \text{Overflow}$ (f) $(-7) + (-6)$

Gambar 8.4 : Penjumlahan dalam Komplemen Dua (Sumber: (Stallings, 2010)).

Organisasi fungsional untuk Penjumlahan:



Gambar 8.5 : Diagram Blok Hardware untuk Penjumlahan
(Sumber: (Stallings, 2010)).

Bagaimana dengan Pengurangan? Bagaimana cara menghitung $A - B$? Hitung komplemen 2 dari B (katakanlah, B_1). Kemudian hitung $R = A + B_1$. Jadi langkah pengurangan adalah melakukan negasi (bentuk komplemen-2) operand kedua (B) kemudian menjumlahkan kedua bilangan tersebut. Jadi, pengurangan diperoleh dengan menggunakan penambahan. Contoh 8.14 untuk $m = 4$:

(a) $(+3) - (+2)$

$$\begin{array}{r} 0011 \quad (+3) \\ - 0010 \quad (+2) \\ \hline \end{array} \Rightarrow \begin{array}{r} 0011 \quad (+3) \\ + 1110 \quad (-2) \\ \hline 1\ 0001 \quad (+1) \end{array}$$

ke carry register

(b) $(+3) - (+5)$

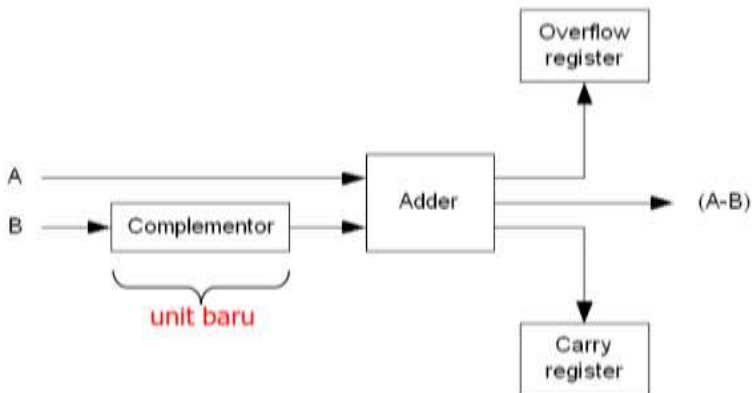
$$\begin{array}{r} 0011 \quad (+3) \\ - 0101 \quad (+5) \\ \hline \end{array} \Rightarrow \begin{array}{r} 0011 \quad (+3) \\ + 1011 \quad (-5) \\ \hline 1110 \quad (-2) \end{array}$$

(c) $(-2) - (-5)$

$$\begin{array}{r} 1110 \quad (-2) \\ - 1011 \quad (-5) \\ \hline \end{array} \Rightarrow \begin{array}{r} 1110 \quad (-2) \\ + 0101 \quad (+5) \\ \hline 1\ 0011 \quad (+3) \end{array}$$

ke carry register

Organisasi fungsional untuk Pengurangan:



Gambar 8.6 : Diagram Blok Hardware untuk Pengurangan
(Sumber: (Stallings, 2010))

Berikut diberikan hasil pengurangan seperti Gambar 8.7.

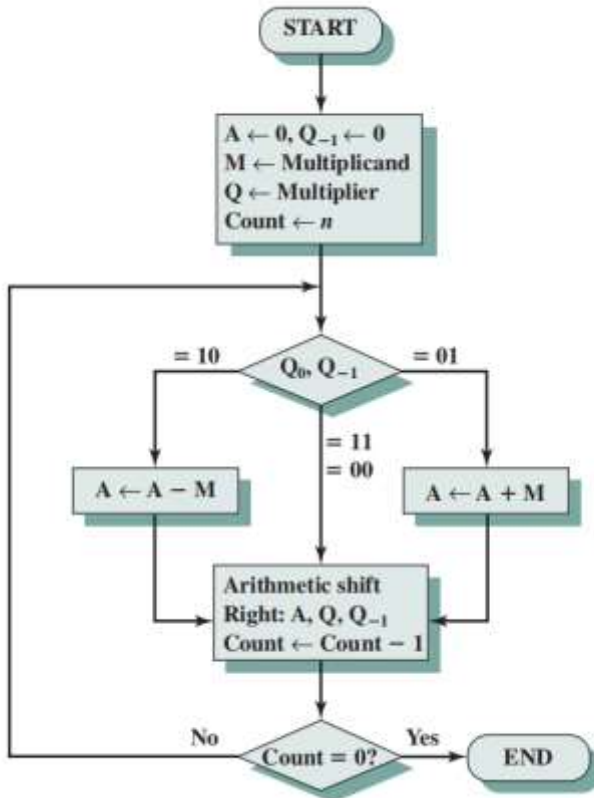
$0010 = 2$ $+1001 = -7$ $1011 = -5$ (a) $M = 2 = 0010$ $S = 7 = 0111$ $-S = 1001$	$0101 = 5$ $+1110 = -2$ $10011 = 3$ (b) $M = 5 = 0101$ $S = 2 = 0010$ $-S = 1110$
$1011 = -5$ $+1110 = -2$ $11001 = -7$ (c) $M = -5 = 1011$ $S = 2 = 0010$ $-S = 1110$	$0101 = 5$ $+0010 = 2$ $0111 = 7$ (d) $M = 5 = 0101$ $S = -2 = 1110$ $-S = 0010$
$0111 = 7$ $+0111 = 7$ $1110 = \text{Overflow}$ (e) $M = 7 = 0111$ $S = -7 = 1001$ $-S = 0111$	$1010 = -6$ $+1100 = -4$ $10110 = \text{Overflow}$ (f) $M = -6 = 1010$ $S = 4 = 0100$ $-S = 1100$

Gambar 8.7 : Pengurangan dalam Komplement Dua (Sumber: (Stallings, 2010))

8.3.4 Perkalian

Secara teoritis, perkalian dapat dilakukan dengan menjumlah secara berulang. Algoritma Booth (***Booth's algorithm***) adalah suatu metoda perkalian komplement-2 yang populer. Seperti *flowchart* implementasi algoritma Booth pada Gambar 8.8, misalkan perkalian $A \times B$ di mana A adalah ***multiplier*** (pengali)

ditempatkan di register Q dan B adalah ***multiplicand*** (terkali) ditempatkan di register M. A dan Q_{-1} (sebuah flip-flop register 1 bit untuk menyimpan bit LSB ***multiplier***) diinisialisasi ke 0. Logika kontrol akan memeriksa bit-bit pengali satu per satu. Jika bit sekarang dan bit yang sebelumnya sama (00 atau 11), tidak dilakukan penjumlahan. Isi akumulator (hasil-kali parsial) dan ***multiplier*** digeser ke kanan satu bit. Bit LSB akumulator masuk ke posisi MSB ***multiplier***. Jika kedua bit berbeda, maka ***multiplicand*** ditambahkan ke register A atau dikurangi dari register A, tergantung pada apakah kedua bit tersebut adalah 0-1 atau 1-0. Setelah penambahan atau pengurangan, terjadi pergeseran ke kanan. Pergeseran ke kanan adalah dimana bit A yang paling kiri, tidak hanya digeser satu bit ke kanan, akan tetapi juga tetap ada disitu. Hal ini diperlukan untuk menyediakan tempat bagi tanda bilangan pada A dan Q. Ketika semua bit ***multiplier*** telah mendapat giliran, maka hasil kali berada pada akumulator dan register ***multiplier***.



Gambar 8.8 : Diagram Alir Algoritma Booth untuk Perkalian dalam Komplemen Dua (Sumber: (Stallings, 2010))

Contoh 8.15 Algoritma Booth ($7 * 3$):

A	Q	Q ₋₁	M	
0000	0011	0	0111	Initial Values
1001	0011	0	0111	$A \leftarrow A - M$
1100	1001	1	0111	Arithmetic Shift Right
1110	0100	1	0111	Arithmetic Shift Right
0101	0100	1	0111	$A \leftarrow A + M$
0010	1010	0	0111	Arithmetic Shift Right
0001	0101	0	0111	Arithmetic Shift Right

Hasil akhir terdapat dalam register A dan Q:

- $A = 0001$
- $Q = 0101$
- $AQ = 00010101_2 = (16 + 4 + 1)_{10} = 21_{10}$

Contoh 8.16 Algoritma Booth ($7 * -3$):

A	Q	Q ₋₁	M	
0000	1101	0	0111	Initial Values
1001	1101	0	0111	$A \leftarrow A - M$
1100	1110	1	0111	Arithmetic Shift Right
0011	1110	1	0111	$A \leftarrow A + M$
0001	1111	0	0111	Arithmetic Shift Right
1010	1111	0	0111	$A \leftarrow A - M$
1101	0111	1	0111	Arithmetic Shift Right
1110	1011	1	0111	Arithmetic Shift Right

Hasil akhir terdapat dalam register A dan Q:

- $A = 1110$
- $Q = 1011$
- $AQ = 11101011_2 = (-128 + 64 + 32 + 8 + 2 + 1)_{10} = -21_{10}$

Contoh 8.17 Algoritma Booth ($-7 * 3$):

A	Q	Q ₋₁	M	
0000	0011	0	1001	Initial Values
0111	0011	0	1001	$A \leftarrow A - M$
0011	1001	1	1001	Arithmetic Shift Right
0001	1100	1	1001	Arithmetic Shift Right
1010	1100	1	1001	$A \leftarrow A + M$
1101	0110	0	1001	Arithmetic Shift Right
1110	1011	0	1001	Arithmetic Shift Right

Hasil akhir terdapat dalam register A dan Q:

- $A = 1110$
- $Q = 1011$
- $AQ = 11101011_2 = (-128 + 64 + 32 + 8 + 2 + 1)_{10} = -21_{10}$

Contoh 8.18 Algoritma Booth ($-7 * -3$):

A	Q	Q ₋₁	M	
0000	1101	0	1001	Initial Values
0111	1101	0	1001	$A \leftarrow A - M$
0011	1110	1	1001	Arithmetic Shift Right
1100	1110	1	1001	$A \leftarrow A + M$
1110	0111	0	1001	Arithmetic Shift Right
0101	0111	0	1001	$A \leftarrow A - M$
0010	1011	1	1001	Arithmetic Shift Right
0001	0101	1	1001	Arithmetic Shift Right

Hasil akhir terdapat dalam register A dan Q:

- $A = 0001$
- $Q = 0101$
- $AQ = 00010101_2 = (16 + 4 + 1)_{10} = 21_{10}$

8.3.5 Pembagian

Pembagian dilakukan dengan melakukan pergeseran dan pengurangan. Gambar 8.9 menunjukkan algoritma Booth untuk pembagian biner tidak bertanda (+/+, -/-). Pembagi (penyebut) ditempatkan di register M, yang dibagi (pemecahan, pembilang) di register Q. Pemecahan (yang dibagi) harus diekspresikan sebagai bilangan komplemen dua $2n$ -bit. Jadi, sebagai contoh, $(+7) = 0111$ 4-bit menjadi 00000111 , dan $(-7) = 1001$ 4-bit menjadi 11111001 .

Pada awalnya, pembagi n bit digeser ke kiri, sedangkan n bit 0 ditambahkan. Dengan cara ini panjang pembagi dan pembagiannya adalah $2n$ bit. Algoritma dan implementasi algoritma Booth hanya berurusan dengan angka positif. Jika angka negatif terlibat, tanda hasil bagi diset MINUS jika pembagi dan pembagiannya berbeda tanda. Namun, tanda sisa lebih sulit diatur. Sisanya akan berada di A. Jika tanda pembagi dan pemecahan sama, maka hasil bagi akan berada di dalam Q; sedangkan bila tidak sama, hasil bagi yang benar adalah komplemen dua dari Q.

Ikuti aturan: $\text{Dividen} = \text{Hasil Bagi} \times \text{Pembagi} + \text{Sisa}$

Contoh: $+7/+2 = +3(+1)$ karena $+7 = +2*(+3)+1$

$+7/-2 = -3(+1)$ karena $+7 = -2*(-3)+1$

$-7/+2 = -3(-1)$ karena $-7 = +2*(-3)-1$

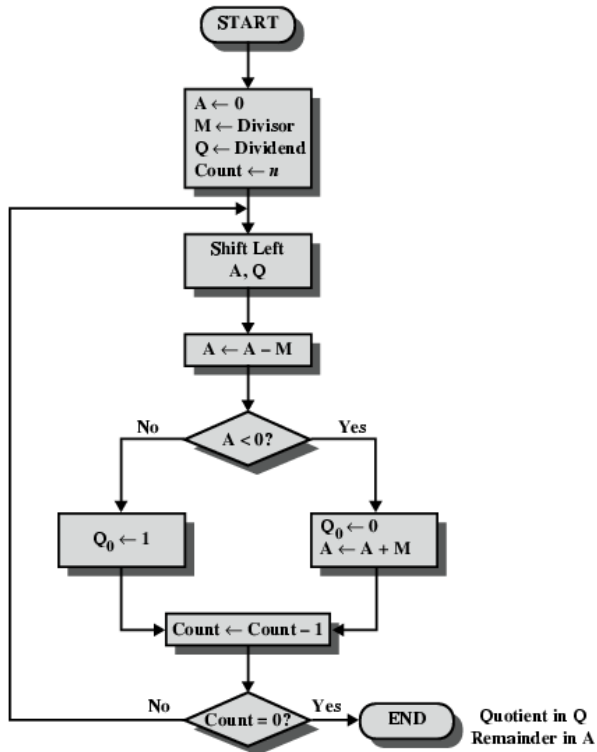
$-7/-2 = +3(-1)$ karena $-7 = -2*(+3)-1$

Geser A, Q ke kiri 1 bit. Jika M dan A mempunyai tanda yang sama, lakukan $A \rightarrow A-M$; bila tandanya M dan A berbeda, lakukan $A \rightarrow A+M$. Operasi diatas akan berhasil jika tanda A sesudah dan sebelum operasi tetap sama.

a) Jika operasi berhasil atau $A=0$, maka set $Q_0 \leftarrow 1$.

b) Jika operasi tidak berhasil dan $A \neq 0$, maka set $Q_0 \leftarrow 0$ dan simpan nilai A sebelumnya.

Ulangi langkah mulai dari Geser A, Q ke kiri 1 bit sampai operasi berhasil atau tidak beberapa kali sampai terdapat posisi bit di Q.



Gambar 8.9 : Algoritma Booth untuk Pembagian Tidak Bertanda (+/+, -/-) (Sumber: (Stallings, 2010))

Contoh 8.19 Algoritma Booth untuk $(+7) / (+3)$, $(+7) / (-3)$, $(-7) / (+3)$ dan $(-7) / (-3)$:

a. $(+7)/(+3)$:			b. $(+7)/(-3)$:		
A	Q	M=0011	A	Q	M=1101
0000	0111	Nilai awal	0000	0111	Nilai awal
0000	1110	geser	0000	1110	geser
1101		kurangkan	1101		tambahkan
0000	1110	simpan kembali	0000	1110	simpan kembali
①			①		
0001	1100	geser	0001	1100	geser
1110		kurangkan	1110		tambahkan
0001	1100	simpan kembali	0001	1100	simpan kembali
②			②		
0011	1000	geser	0011	1000	geser
0000		kurangkan	0000		tambahkan
0000 (-	1001	set $Q_0 \leftarrow 1$	0000	1001	set $Q_0 \leftarrow 1$
③			③		
0001	0010	geser	0001	0010	geser
1110		kurangkan	1110		tambahkan
0001	0010	simpan kembali	0001	0010	simpan kembali
④			④		
Hasil bagi = 0010 ($=Q$) Sisa bagi = 0001 ($=A$) $7/3 = 2$ sisa 1			Hasil bagi = 1110 ($=Q$) Sisa bagi = 0001 ($=A$) $7/-3 = -2$ sisa 1		
c. $(-7)/(+3)$:			d. $(-7)/(-3)$:		
A	Q	M=0011	A	Q	M=1101
1111	1001	Nilai awal	1111	1001	Nilai awal
1111	0010	geser	1111	0010	geser
0010		tambahkan	0010		kurangkan
1111	0010	simpan kembali	1111	0010	simpan kembali
①			①		
1110	0100	geser	1110	0100	geser
0001		tambahkan	0001		kurangkan
1110	0100	simpan kembali	1110	0100	simpan kembali
②			②		
1100	1000	geser	1100	1000	geser
1111		tambahkan	1111		kurangkan
1111	1001	set $Q_0 \leftarrow 1$	1111	1001	set $Q_0 \leftarrow 1$
③			③		
1111	0010	geser	1111	0010	geser
0010		tambahkan	0010		kurangkan
1111	0010	simpan kembali	1111	0010	simpan kembali
④			④		
Hasil bagi = 1110 ($=Q$) Sisa bagi = 1111 ($=A$) $-7/3 = -2$ sisa -1			Hasil bagi = 0010 ($=Q$) Sisa bagi = 1111 ($=A$) $-7/-3 = 2$ sisa -1		

8.4 Representasi Floating Point

Dalam dunia ilmiah, bilangan pecahan (floating point) diekspresikan dalam dua bagian: bagian pecahan disebut mantissa, dan bagian eksponensial yang menunjukkan pangkat sepuluh yang harus dinaikkan mantissa tersebut untuk mendapatkan nilai yang kita butuhkan. Dalam komputer digital, angka floating-point terdiri dari tiga bagian: bit tanda, bagian eksponen (mewakili eksponen dengan pangkat 2), dan bagian pecahan yang disebut significand (yang merupakan untuk mantissa). Bilangan floating-point (F) dinyatakan sebagai bilangan (significand, S atau mantissa, M) dikalikan dengan konstanta (basis, B) yang dipangkatkan dengan bilangan bulat (eksponen, E). Notasi floating point F:

$$F = \pm M \times B^{\pm E}$$

M = Mantissa

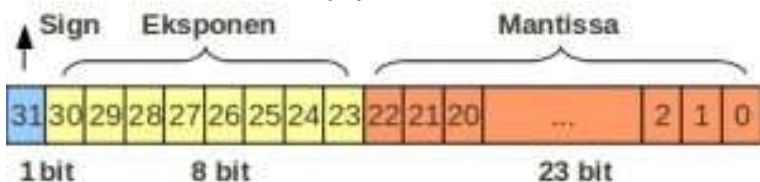
E = Eksponen

B = Basis dari eksponen

B $\neq$ basis bilangan

Bilangan *floating-point* 32-bit (presisi tunggal, single precision) tersusun atas:

- 1 bit tanda (S),
- 8 bit eksponen (E), dan
- 23 bit untuk mantisa (M)



Gambar 8.10 : Format floating point presisi tunggal 32 bit
(Sumber: (Stallings, 2010))

Basis bilangan (Mantissa) selalu bernilai 2 (biner), sementara basis eksponen B tidak harus 2. Standarisasi bilangan

floating point yaitu melakukan dalam bentuk ternormalisasi dengan cara *Most Significant Digit* dari mantissa bukan nol (non-zero) dengan bentuknya adalah: $\pm 1, bbb \dots b \times 2^{\pm E}$. Bit pertama significand selalu 1 sehingga tidak perlu disimpan (implisit) dalam field significand. Dan b adalah bilangan biner (1 atau 0). Jika semua bilangan floating point direpresentasikan dalam komputer dengan bentuk ternormalisasi, maka posisi bit satu dapat disimpan dengan mengabaikan MSB (selalu 1). Ini disebut *hidden 1 principle*. Biased eksponen (E') = $E + 127$. Representasi nilai = $\pm 1.M \times 2^{E'-127}$.

Contoh 8.20: Tuliskan bilangan pecahan desimal $B = 35.625_{10}$ dalam format floating-point 32-bit dalam format presisi tunggal.

$$\begin{aligned}
 B &= (35.625)_{10} \\
 &= (35)_{10} + (0.625)_{10} \\
 &= (100011)_2 + (0.1001)_2 \\
 &= (100011.1001)_2 \\
 &= (1.000111001)_2 \times 2^5
 \end{aligned}$$

Dari perhitungan di atas, nilai eksponen $E = 5 + 127 = 132 = 10000100$ dan mantissa $M = 000111001$



Contoh 8.21: Tuliskan bilangan 0.0 , 1.0_{10} , 0.5_{10} , 0.75_{10} , 3.0_{10} dan -0.375_{10} dalam format floating-point 32-bit dalam format presisi tunggal (S, E, M).

0.0 : 0 00000000 000000000000000000000000

$1.0 (1.0 \times 2^0)$: 0 01111111 000000000000000000000000

0.5	(0.1 biner = 1.0 x 2 ⁻¹):	0	01111110
000000000000000000000000			
0.75	(0.11 biner = 1.1 x 2 ⁻¹):	0	01111110
100000000000000000000000			
3.0	(11 biner = 1.1*2 ⁻¹):	0	10000000
100000000000000000000000			
- 0.375	(- 0.011 biner = -1.1*2 ⁻²):	1	01111101
100000000000000000000000			

Contoh 8.22: Tuliskan 1.6328125×2^{20} dalam format presisi tunggal.

$1.6328125 = 1 + (2^{-1} + 2^{-3} + 2^{-7})$ dinormalisasi: 1.1010001

S = 0 (positif), M = 101000100000000000000000

Biased exponent:

$$\begin{array}{rcl}
 20_{(10)} & := & 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \\
 + 127_{(10)} & := & 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \\
 \hline
 & & 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1
 \end{array}$$

$1.1010001 \times 2^{10100} = 0 \ 10010011 \ 101000100000000000000000$

Contoh 8.23: Tuliskan -1.6328125×2^{20} dalam format presisi tunggal.

$-1.6328125 = -1 + (2^{-1} + 2^{-3} + 2^{-7})$, dinormalisasi: -1.1010001

S = 1 (negatif), M = 101000100000000000000000

Biased exponent:

$$\begin{array}{rcl}
 20_{(10)} & := & 0 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 \quad 0 \quad 0 \\
 + 127_{(10)} & := & 0 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \quad 1 \\
 \hline
 & & 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 0 \quad 1 \quad 1
 \end{array}$$

$-1.1010001 \times 2^{10100} = 1 \ 10010011 \ 101000100000000000000000$

Contoh 8.24: Tuliskan 1.6328125×2^{-20} dalam format presisi tunggal:

$1.6328125 = 1 + (2^{-1} + 2^{-3} + 2^{-7})$ dinormalisasi: 1.1010001

S = 0 (positif), M = 101000100000000000000000

Biased exponent:

$$\begin{array}{rcccccccc} -20_{(10)} := & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ + 127_{(10)} := & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \end{array}$$

$$1.1010001 \times 2^{-10100} = 0 \quad 01101011$$

101000100000000000000000

Contoh 8.25: Tuliskan $-1.6328125 \times 2^{-20}$ dalam format presisi tunggal.

$-1.6328125 = -1 + (2^{-1} + 2^{-3} + 2^{-7})$, dinormalisasi: -1.1010001

S = 1 (negatif), M = 101000100000000000000000

Biased exponent:

$$\begin{array}{rcccccccc} -20_{(10)} := & 1 & 1 & 1 & 0 & 1 & 1 & 0 & 0 \\ + 127_{(10)} := & 0 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ \hline & 0 & 1 & 1 & 0 & 1 & 0 & 1 & 1 \end{array}$$

$$-1.1010001 \times 2^{-10100} = 1 \quad 01101011$$

101000100000000000000000

DAFTAR PUSTAKA

- Khan, S. R. 2018. *A Brief Analysis of Processor Application: Build Data Path, Control and Instructions*. Available at: https://www.academia.edu/37048334/A_Brief_Analysis_of_Processor_Application_Build_Data_Path_Control_and_Instructions.
- Patterson, David A. and Hennessy, J. L. 2014. *Computer Organization and Design: The Hardware/Software Interface*. Fifth Edit. Elsevier, Inc.
- Stallings, W. 2010. *Computer Organization and Architecture: Designing for Performance*. 8th Editio. New Jersey: Prentice Hall. Available at: <https://ocw.upj.ac.id/files/Textbook-TIF203-Ebook-William-Stallings-Designing-for-Performance-8th-Edition-Prentice-Hall-2011.pdf>.

BAB 9

INSTRUCTION ADDRESSING DAN FORMAT

Oleh Novianti Indah Putri

9.1 Pendahuluan

Dalam bab ini, membahas prinsip-prinsip dasar yang terlibat dalam arsitektur dan desain set instruksi. Dimulai dengan pertimbangan lokasi memori dan alamat. Model abstrak dari memori utama yang dianggap sebagai urutan sel yang masing-masing mampu menyimpan n bit. Cara mengatasi masalah menyimpan dan mengambil informasi ke dalam dan dari memori. Informasi yang disimpan dan/atau diambil dari memori perlu ditangani. Pembahasan tentang sejumlah cara berbeda untuk mengalamatkan lokasi memori (mode pengalamatan) adalah topik berikutnya yang akan dibahas dalam bab ini. Sebuah program terdiri dari sejumlah instruksi yang harus diakses dalam urutan tertentu. Itu memotivasi untuk menjelaskan masalah eksekusi instruksi dan pengurutan secara rinci. Kemudian menunjukkan penerapan mode pengalamatan dan karakteristik instruksi yang disajikan dalam menulis kode segmen sampel untuk melakukan sejumlah tugas pemrograman sederhana. Fokus di bab ini adalah pada memori (utama) dari sudut pandang pemrogram. Secara khusus, kami fokus pada cara informasi disimpan dan diambil dari memori.

9.2 Addressing Modes

Addressing Mode (Mode Pengalamatan) – Istilah mode pengalamatan mengacu pada cara di mana operan instruksi ditentukan. Mode pengalamatan menentukan aturan untuk menafsirkan atau memodifikasi bidang alamat instruksi sebelum operan benar-benar dieksekusi. Mode pengalamatan adalah metode untuk menentukan operan dari suatu instruksi. Tugas mikroprosesor adalah menjalankan serangkaian instruksi yang disimpan dalam memori untuk melakukan tugas tertentu. Operasi memerlukan hal-hal berikut: operator atau opcode yang menentukan apa yang akan dilakukan, operan yang menentukan data yang akan digunakan dalam operasi. Misalnya, jika kita ingin menjumlahkan angka 1 dan 2 dan mendapatkan hasil, secara matematis kita akan menuliskannya sebagai $1 + 2$. Dalam hal ini, operator kita adalah (+), atau penjumlahan, dan operan kita adalah angka 1 dan 2. Dalam mikroprosesor, mesin perlu diberi tahu cara mendapatkan operan untuk melakukan operasi. Alamat efektif adalah istilah yang menjelaskan alamat operan yang disimpan dalam memori. Ada beberapa metode untuk menentukan alamat efektif dari operan tersebut atau mendapatkannya langsung dari register. Metode ini dikenal sebagai mode pengalamatan.

Dalam bab ini, kita akan mendefinisikan beberapa mode pengalamatan yang dapat digunakan untuk menunjukkan cara mengambil operan. Juga akan mengilustrasikan contoh bagaimana setiap mode pengalamatan digunakan dan menunjukkan beberapa karakteristik dari mode pengalamatan ini. Berbagai cara untuk menentukan lokasi operan dalam instruksi disebut sebagai mode pengalamatan: mode pengalamatan tersirat / implisit, mode pengalamatan tumpukan, mode pengalamatan langsung, mode pengalamatan langsung, mode pengalamatan tidak langsung, mode pengalamatan daftar langsung, daftar mode pengalamatan tidak langsung, mode pengalamatan relatif, mode pengalamatan terindeks, mode pengalamatan daftar basis, mode pengalamatan

peningkatan otomatis, mode pengalamatan pengurangan otomatis (Hayes, 1988).

9.2.1 Klasifikasi Instruksi

Instruksi menentukan fungsi apa yang dapat dilakukan mikroprosesor. Instruksi ini dapat diklasifikasikan ke dalam lima kategori fungsional berikut: operasi transfer data (penyalinan), operasi aritmatika, operasi logis, operasi percabangan, dan operasi kontrol mesin (Kart, 2022).

Dapat dijelaskan bahwa kelas-kelas ini bersama dengan contoh sederhana di paragraf berikut. Perlu dicatat bahwa dalam menyajikan contoh-contoh ini, kami akan menggunakan operasi konvensi, sumber, tujuan untuk menyatakan instruksi apa pun. Ini juga menunjukkan bahwa nilai yang akan ditambahkan adalah yang disimpan di register R1 dan R2 yang hasilnya harus disimpan di register R3.

Contoh instruksi tiga alamat yang mengacu pada lokasi memori dapat berupa ADD A,B,C. Instruksi menambahkan isi lokasi memori A ke isi lokasi memori B dan menyimpan hasilnya di lokasi memori C. Instruksi dua alamat berbentuk operasi add-1, add-2. Dalam bentuk ini, masing-masing add-1 dan add-2 mengacu pada register atau lokasi memori. Pertimbangkan, misalnya, instruksi ADD R1,R2. Instruksi ini menambahkan isi register R1 ke isi register R2 dan menyimpan hasilnya di register R2. Isi asli register R2 hilang karena operasi ini sedangkan isi asli register R1 tetap utuh. Instruksi ini setara dengan instruksi tiga alamat dalam bentuk ADD R1,R2,R2. Instruksi serupa yang menggunakan lokasi memori dan bukan register dapat berbentuk ADD A,B. Dalam hal ini, isi dari lokasi memori A ditambahkan ke isi dari lokasi memori B dan hasilnya digunakan untuk menimpa isi asli dari lokasi memori B.

9.2.2 Operasi Instruksi Tiga Alamat

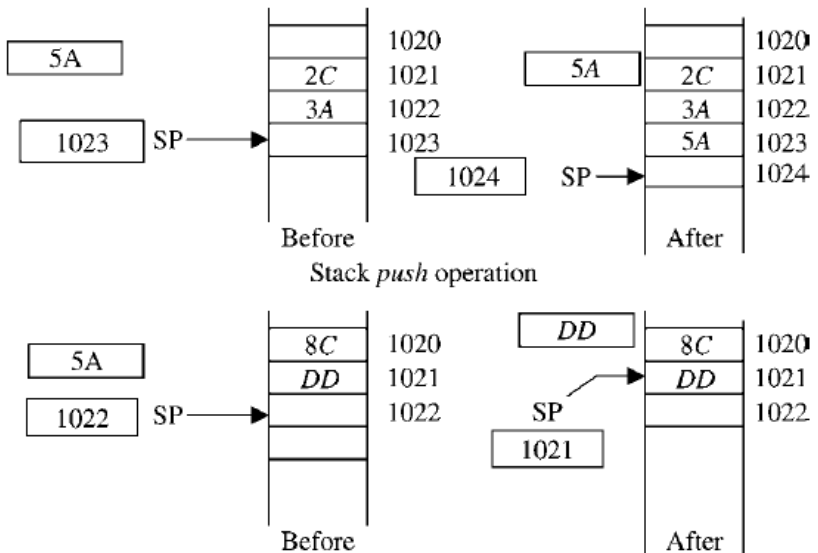
Operasi yang dilakukan oleh instruksi tiga alamat ADD A,B,C dapat dilakukan oleh dua instruksi dua alamat MOVE B,C dan ADD A,C. Ini karena instruksi pertama memindahkan isi lokasi B ke lokasi C dan instruksi kedua menambahkan isi lokasi A ke isi lokasi C (isi lokasi B) dan menyimpan hasilnya di lokasi C.

Instruksi satu alamat berbentuk ADD R1. Dalam hal ini instruksi secara implisit mengacu pada register, yang disebut Racc Akumulator, sehingga isi akumulator ditambahkan ke isi register R1 dan hasilnya disimpan kembali ke Racc akumulator. Jika lokasi memori digunakan sebagai pengganti register, maka instruksi dalam bentuk ADD B digunakan. Dalam hal ini, instruksi menambahkan konten Racc akumulator ke konten lokasi memori B dan menyimpan hasilnya kembali ke Racc akumulator. Instruksi ADD R1 setara dengan instruksi tiga alamat ADD R1,Racc,Racc atau dengan instruksi dua alamat ADD R1,Racc.

Di antara instruksi dua alamat dan satu alamat, bisa terdapat instruksi satu setengah alamat. Pertimbangkan, misalnya, instruksi ADD B,R1. Dalam hal ini, instruksi menambahkan isi register R1 ke isi lokasi memori B dan menyimpan hasilnya dalam register R1. Karena fakta bahwa instruksi menggunakan dua jenis pengalamatan, yaitu register dan lokasi memori, ini disebut instruksi satu setengah alamat. Ini karena pengalamatan register membutuhkan jumlah bit yang lebih kecil daripada yang dibutuhkan oleh pengalamatan memori.

Sangat menarik untuk menunjukkan bahwa ada instruksi alamat nol. Ini adalah instruksi yang menggunakan operasi tumpukan. Tumpukan adalah mekanisme organisasi data di mana item data terakhir yang disimpan adalah item data pertama yang diambil. Dua operasi spesifik dapat dilakukan pada tumpukan. Ini adalah operasi push dan pop. Gambar 9.1 mengilustrasikan dua operasi ini.

Seperti dapat dilihat, register khusus, yang disebut stack pointer (SP), digunakan untuk menunjukkan lokasi stack yang dapat dialamatkan. Dalam operasi push tumpukan, nilai SP digunakan untuk menunjukkan lokasi (disebut bagian atas tumpukan) di mana nilai (5A) akan disimpan (dalam hal ini adalah lokasi 1023). Setelah menyimpan (mendorong) nilai ini, SP dinaikkan untuk menunjukkan ke lokasi 1024. Dalam operasi stack pop, SP pertama-tama dikurangi menjadi 1021. Nilai yang disimpan di lokasi ini (DD dalam kasus ini) diambil (dimunculkan) dan disimpan dalam register yang ditunjukkan.



Gambar 9.1 : Operasi stack push dan pop
Sumber : (AfterAcademy, 2022)

Operasi yang berbeda dapat dilakukan dengan menggunakan struktur stack. Pertimbangkan, misalnya, instruksi seperti `ADD (SP), (SP)`. Instruksi menambahkan isi dari lokasi stack yang ditunjuk oleh SP ke yang ditunjuk oleh `SP + 1` dan

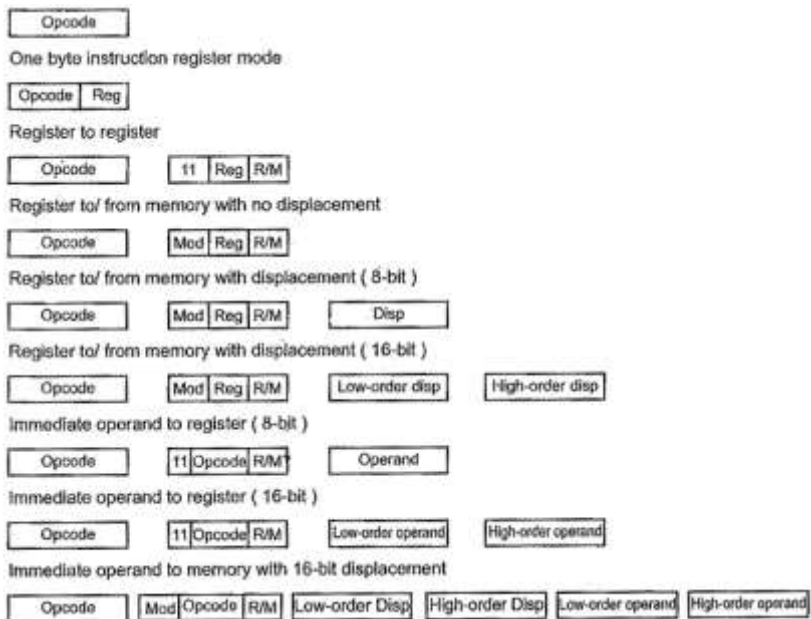
menyimpan hasilnya pada stack di lokasi yang ditunjuk oleh nilai SP saat ini.

9.3 Intruction Format

Instruksi mencakup sekumpulan kode operasi dan operan yang dikelola dengan kode operasi. Format instruksi mendukung desain bit dalam instruksi. Ini berisi bidang termasuk opcode, operan, dan mode pengalamatan. Panjang instruksi umumnya dipertahankan dalam kelipatan panjang karakter, yaitu 8 bit. Ketika panjang instruksi adalah permanen, beberapa bit ditugaskan ke opcode, operan, dan mode pengalamatan. Fungsi mengalokasikan bit dalam instruksi dapat diinterpretasikan dengan mempertimbangkan elemen-elemen berikut - Jumlah mode pengalamatan, Jumlah operan, Jumlah register CPU, Jumlah set register, Jumlah baris alamat. Menampilkan format instruksi umum berupa format instruksi yang dapat digunakan oleh mikroprosesor Intel yang paling menonjol. Format instruksi ini mencakup empat bidang, seperti bidang opcode, bidang mode pengalamatan, bidang perpindahan, dan bidang langsung.

9.3.1 Format Instruksi 8086

Format Instruksi 8086 Instruksi 8086 bervariasi dari 1 sampai 6 byte panjangnya. Gambar 9.2 menunjukkan format instruksi untuk instruksi 1 sampai 6 byte. Seperti ditunjukkan pada Gambar 9.2, perpindahan dan operan dapat sepanjang 8-bit atau 16-bit tergantung pada instruksinya. Opcode dan mode pengalamatan ditentukan menggunakan dua byte pertama dari sebuah instruksi.



Gambar 9.2 : Contoh Format Instruksi 8086

Sumber : (Guide, 2014)

Opcode/byte mode pengalamatan dapat diikuti oleh :

- Tidak ada byte tambahan
- Dua byte EA (Hanya untuk pengalamatan langsung).
- Perpindahan satu atau dua byte
- Satu atau dua byte operan langsung
- Perpindahan satu atau dua byte diikuti oleh operan langsung satu atau dua byte

Pemindahan dua byte dan alamat segmen dua byte (hanya untuk pengalamatan intersegmen langsung). Sebagian besar opcode di 8086 memiliki indikasi 1-bit khusus.

W-bit : Beberapa instruksi 8086 dapat beroperasi pada byte atau kata. W-bit dalam opcode instruksi tersebut menentukan

apakah instruksi adalah instruksi byte ($W = 0$) atau instruksi kata ($W = 1$).

D-bit : D-bit dalam opcode instruksi menunjukkan bahwa register yang ditentukan dalam instruksi adalah register sumber ($D = 0$) atau register tujuan ($D = 1$).

S-bit : Bilangan komplemen 8-bit 2 dapat diperluas menjadi bilangan komplemen 16-bit 2 dengan menjadikan semua bit dalam byte orde tinggi sama dengan bit paling signifikan dalam byte orde rendah. Ini dikenal sebagai perpanjangan tanda. S-bit bersama dengan W-bit menunjukkan:

Tabel 9.1 : S-Bit Bersama Dengan W-Bit

S	W	Operasi
0	0	operasi 8-bit
0	1	Operasi 16-bit dengan operan langsung 16-bit
1	0	
1	1	Operasi 16-bit dengan tanda diperpanjang operan langsung 8-bit

Sumber: (Guide, 2014)

V-bit : V-bit memutuskan jumlah shift untuk instruksi rotate dan shift. Jika $V = 0$, maka hitung = 1; jika $V = 1$, hitungannya ada di register CL. Misalnya, jika $V = 1$ dan $CL = 2$ maka instruksi bergeser atau diputar bergeser atau diputar 2-bit

Z-bit : Digunakan untuk string primitif seperti REP untuk perbandingan dengan ZF Flag.

9.3.2 Alokasi Bit

Alokasi Bit, berikut faktor-faktor yang saling terkait masuk ke dalam menentukan penggunaan pengalamatan bit: Jumlah mode pengalamatan: Kadang-kadang mode pengalamatan dapat ditunjukkan secara implisit. Misalnya, opcode tertentu mungkin selalu memerlukan pengindeksan. Dalam kasus lain, mode pengalamatan harus eksplisit dan satu atau lebih bit mode akan dibutuhkan (Technology, 2019).

Jumlah operan: Instruksi umum pada mesin saat ini menyediakan dua operan. Setiap alamat operan dalam instruksi mungkin memerlukan indikator modenya sendiri atau penggunaan indikator mode dapat dibatasi hanya pada salah satu bidang alamat.

Daftar versus memori: Hanya beberapa bit yang diperlukan untuk menentukan register. Semakin banyak register yang dapat digunakan untuk referensi operan, semakin sedikit bit yang dibutuhkan. Sejumlah penelitian menunjukkan bahwa total 8 hingga 32 register yang terlihat oleh pengguna diinginkan.

Jumlah set register: Beberapa arsitektur memiliki koleksi dua atau lebih set khusus (seperti data dan perpindahan). Perpecahan fungsional membutuhkan lebih sedikit bit untuk digunakan dalam instruksi.

Rentang alamat: Untuk alamat yang mereferensikan memori, rentang alamat yang dapat dirujuk terkait dengan jumlah bit alamat. Dengan pengalamatan perpindahan, jangkauan dibuka hingga panjang register alamat.

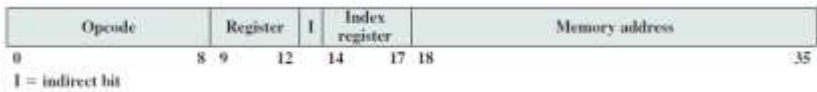
Perincian alamat: Dalam sistem dengan kata-kata 16- atau 32-bit, alamat dapat mereferensikan kata atau byte sesuai pilihan perancang. Pengalamatan byte nyaman untuk manipulasi karakter tetapi membutuhkan, untuk memori berukuran tetap, lebih banyak bit alamat.

Sebagai contoh, mengutip satu studi yang membandingkan berbagai pendekatan format instruksi, termasuk penggunaan stack,

register serba guna, akumulator, dan hanya pendekatan memori-ke-register (Cragon, 1979). Menggunakan serangkaian asumsi yang konsisten, tidak ada perbedaan signifikan dalam ruang kode atau waktu eksekusi yang diamati. Mari kita lihat secara singkat bagaimana dua desain mesin bersejarah menyeimbangkan berbagai faktor ini. PDP-8 Salah satu desain instruksi paling sederhana untuk komputer tujuan umum adalah untuk PDP-8 (Bell and Siewiorek, 1978).

9.3.3 Instruksi Panjang Variabel

Dalam arsitektur di mana tidak semua instruksi memiliki panjang yang sama, bagaimana komputer mengetahui berapa banyak yang harus dibaca untuk satu instruksi (Overflow, 2019).



Gambar 9.3 : Format Instruksi PDP-10
 Sumber : (Baker, 2019)

PDP-10 memiliki panjang kata 36-bit dan panjang instruksi 36-bit.

Besaran yang harus dibayar untuk instruksi dengan panjang variabel adalah peningkatan kompleksitas prosesor. Namun, akan melihat bahwa mesin RISC dan superskalar dapat mengeksplorasi penggunaan panjang tetap instruksi untuk memberikan kinerja yang lebih baik. Prosesor RISC menggunakan instruksi dengan panjang tetap. Instruksi dengan panjang variabel dapat menyebabkan inefisiensi implementasi dan eksekusi (Books, 2019). Format instruksi dengan panjang variabel yang ada memberikan kepadatan kode yang lebih tinggi daripada format dengan panjang tetap, tetapi tidak cocok untuk pengambilan dan dekode instruksi paralel atau pipelined (Pan and Asanovi’c, 2001).

DAFTAR PUSTAKA

- AfterAcademy, A. 2022. *Stack and its basic Operations*, *afteracademy.com/blog*. Available at: <https://afteracademy.com/blog/stack-and-its-basic-operations/>.
- Baker, H. 2019. *PDP-10 Machine Language, PDP 10*. Available at: <http://pdp10.nocrew.org/docs/instruction-set/pdp-10.html> (Accessed: 1 November 2022).
- Bell, C. G. and Siewiorek, D. P. 1978. 'Structural Levels of the PDP-8', in *Computer Engineering: A DEC View of Hardware Systems Design*. Digital Equipment Corporation, pp. 209–213. Available at: http://iiif.library.cmu.edu/file/Newell_box00047_fld04051_doc0001/Newell_box00047_fld04051_doc0001.pdf.
- Books, N. 2019. 'RISC Processors 1', in *Architecture Books*. Arch Book, pp. 1–28. Available at: http://service.scs.carleton.ca/sivarama/org_book/org_book_web/solution_manual/org_soln_sep/arch_book_solution_ch14_sep.pdf.
- Cragon, H. G. 1979. 'An Evaluation of Code Space Requirements and Performance of Various Architectures', *Computer Architecture News*, pp. 1–17. doi: 10.1145/859445.859446.
- Guide, E. 2014. *8086 Instruction Format*, *EEEGuide.com*. Available at: <https://www.eeeguide.com/8086-instruction-format/> (Accessed: 3 November 2022).
- Hayes, J. P. 1988. *Computer Architecture and Organization*. Second. New York: McGraw-Hill, Inc.
- Kart, B. 2022. *Instruction Set Classification*, *BrainKart.com*. Available at: https://www.brainkart.com/article/Instruction-Set-Classification-of-8085-Processor_7820/ (Accessed: 1 November 2022).

- Overflow, S. 2019. *With variable length instructions how does the computer know the length of the instruction being fetched?*, *stackoverflow.com*. Available at: <https://stackoverflow.com/questions/24269368/with-variable-length-instructions-how-does-the-computer-know-the-length-of-the-i> (Accessed: 1 November 2022).
- Pan, H. and Asanović, K. 2001. 'Heads and Tails : A Variable-Length Instruction Format Supporting Parallel Fetch and Decode', *International Journal Programming*, 18(2), pp. 1–8. doi: 10.1145/502217.502244.
- Technology, I. I. of 2019. *Instruction Formats: Allocation of Bits, SCLM Presentation*. Available at: <http://www.cs.iit.edu/~cs561/cs350/InstructionFormat/allocationofbits.html> (Accessed: 1 January 2023).

BAB 10

CPU

Oleh Niko Akbar

10.1 Apa Itu CPU?

1. Nama lain dari *Processor/ Central Processor/ Microprocessor*.
2. Menangani semua instruksi/perintah yang diterimanya dari perangkat keras dan perangkat lunak yang berjalan di komputer.
3. Otaknya komputer
4. Ditempatkan berdampingan dengan *Motherboard*
5. Fungsi utamanya Mengambil sebuah inputan dari periferal (Keyboard, Mouse, Printer, dll) / program, menafsirkan dan mengeluarkan hasil output ke monitor atau perangkat lain.

10.2 Kegunaan CPU pada komputer

1. Bisa melakukan *Image-Processing*
2. *3-Dimensional Rendering*
3. *Speech Recognition* (Pengenalan Suara)
4. *Video Conferencing*
5. *Multimedia authoring*
6. *Voice and video annotation of files*
7. *Simulation modeling*

10.3 Hal yang diketahui tentang Microprocessor

Untuk membuat Kecepatan sebuah komputer meningkat. Maka Desainer/Pencipta Microprocessor harus Mempunyai Persyaratan untuk Microprocessor dapat berjalan dengan lancar di dalam komputer.

1. Komputer harus bisa menerima Instruksi.
2. Mampu Menafsirkan Pesan yang dikirim (*decoding*) dari bahasa Mesin (*opcode*).
3. Dapat Mengerjakan sebuah perhitungan/kalkulasi dalam komputer.

Bila Processor dapat berkembang dengan memfokuskan kecepatan performa yang tinggi sampai batas tertentu. Ada Faktor lain yang perlu diperhatikan, yaitu Persyaratan pada komponen dalam komputer.

1. Balance/Keseimbangan : harus ada penyesuaian/ penyetelan dalam komputer yang menutupi kelebihan dari Processor dalam komputer.

Seorang Arsitek/Perancang Komputer harus mempunyai solusi pada desain pembuatan komponen dalam komputer. Termasuk diantaranya,

1. Meningkatkan Kecepatan Transfer antar bit yang ada didalam DRAM
2. Ubah antarmuka DRAM agar lebih efisien dengan menyertakan cache. atau skema buffering lainnya pada chip DRAM.
3. Kurangi frekuensi akses memori dengan menggabungkan struktur cache yang semakin kompleks dan efisien antara prosesor dan memori utama
4. Meningkatkan bandwidth interkoneksi antara prosesor dan memori dengan menggunakan bus berkecepatan lebih tinggi dan hierarki bus untuk buffer dan struktur aliran data

10.4 Hukum dalam Peningkatan Performa sebuah Sistem

Hukum disini tidak terkait tetapi keduanya memberikan wawasan tentang kinerja sistem paralel dan Sistem *Multicore*

1. Hukum Amdahl (*Ahmdahl's Law*)

Perancang sistem komputer harus mencari cara untuk meningkatkan kinerja sistem sesuai dengan kemajuan sebuah teknologi atau perubahan desain.

Hukum Amdahl pertama kali diusulkan oleh Gene Amdahl pada tahun 1967 dan berkaitan dengan adanya potensi percepatan suatu program dengan menggunakan banyak prosesor dibandingkan satu prosesor.

2. *Little's Law*

Dalam mengevaluasi perangkat keras prosesor dan persyaratan pengaturan untuk sistem baru, Maka Performa merupakan salah satu parameter utama yang perlu dipertimbangkan bersama dengan biaya, ukuran, keamanan, keandalan, dan, dalam beberapa kasus, konsumsi daya.

10.5 Cara Mengukur Performa Komputer

1. Kecepatan jam (*Clock Speed*)

Operasi yang dilakukan pada prosesor, seperti mengambil instruksi, mendekode, memberi instruksi, melakukan operasi aritmatika, dan sebagainya, diatur oleh suatu sistem pada jam. Biasanya, semua operasi dimulai dengan hitungan jam. Pada dasarnya, kecepatan prosesor ditentukan oleh frekuensi *pulse* yang dihasilkan oleh jam, diukur dalam siklus per detik, atau Hertz (Hz).

2. Tingkat Eksekusi Instruksi

Sebuah program terdiri dari sejumlah instruksi komputer yang dieksekusi untuk program sampai program selesai atau tergantung seberapa waktu yang ditentukan.

10.6 Rangkaian Instruksi

Sebuah CPU (*Control Processing Unit*) merupakan Sekumpulan Instruksi Mesin.

Karakteristik Instruksi Mesin

Sebuah operasi pada processor ditentukan oleh instruksi yang dieksekusi, disebut sebagai instruksi mesin atau instruksi komputer. Kumpulan instruksi berbeda yang dapat dieksekusi oleh prosesor disebut sebagai set instruksi prosesor.

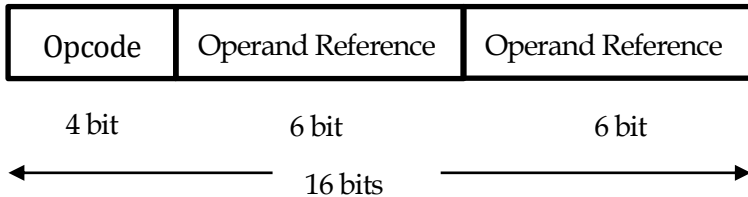
10.7 Elemen instruksi mesin

Setiap instruksi harus berisi informasi yang dibutuhkan oleh prosesor untuk eksekusi. Unsur-unsur tersebut adalah sebagai berikut,

- *Operation Code* : menentukan operasi yang akan dilakukan. Operasi ditentukan oleh kode biner, yang dikenal sebagai kode operasi(Opcode).
- *Source Operand Reference* : operasi dapat melibatkan satu atau lebih operand sumber, yang merupakan input untuk operasi.
- *Result operand reference* : operand menghasilkan suatu hasil
- *Next instruction reference* : memberi tahu prosesor tempat mengambil instruksi berikutnya setelah eksekusi instruksi ini selesai.

10.8 Perwujudan Instruksi

Setiap instruksi merupakan perwujudan dari beberapa urutan dari beberapa bit, Instruksi terbagi menjadi beberapa field, yang berhubungan dengan elemen penyusun dari beberapa isian. Contoh sederhana dari instruksi pada gambar di bawah



Selama eksekusi instruksi, instruksi dibaca ke dalam *Instruction Register* (IR) dalam prosesor. Dan kemudian mengekstrak data dari berbagai bidang instruksi untuk melakukan operasi yang diperlukan.

Sulit bagi programmer untuk berurusan dengan representasi biner dari instruksi mesin. Dengan demikian, sehingga menggunakan representasi simbol atau instruksi mesin

Opcode diwakili oleh singkatan, yang disebut mnemonik, yang menunjukkan operasi. Contoh umum

Add	Add
Sub	Substract
Mul	Multiply
Div	Divided
Load	Memuat data dari memori
Stor	Menyimpan data dari memory

Contoh Instruksi :

Add R, X berarti menambahkan nilai pada lokasi data X ke isi register R. Pada contoh ini, X mengacu pada alamat lokasi di memori, dan R mengacu pada register tertentu. Operasi dilakukan pada isi lokasi, bukan pada alamatnya

10.9 Tipe Instruksi

Instruksi tingkat tinggi. Sebagai contoh

$X = X + Y$

Pernyataan ini menginstruksikan memori untuk menambahkan nilai yang tersimpan di Y ke nilai yang tersimpan pada X dan menaruh hasilnya di X. Asumsikan bahwa variabel X dan Y berhubungan dengan lokasi pada nomor 513 dan 514. Kemudian, asumsikan satu set instruksi mesin yang sederhana, operasi ini dapat diselesaikan dengan tiga instruksi; yakni

1. Memuat register dengan isi lokasi memori 513
2. Tambahkan isi lokasi memori 514 ke register tersebut.
3. Simpan isi register di lokasi memori 513.

Bahasa tingkat tinggi mengekspresikan operasi dalam aljabar secara ringkas menggunakan variabel. Bahasa mesin mengekspresikan operasi dalam bentuk dasar yang melibatkan pergerakan data ke atau dari register.

Kumpulan instruksi mesin harus cukup untuk mengekspresikan instruksi apa pun dari bahasa tingkat tinggi. Dengan mengingat hal ini, kita dapat mengkategorikan jenis instruksi sebagai berikut:

- Data Processing : Instruksi untuk algoritma dan aritmatika.
- Data Storage : Pergerakan data ke dalam atau ke luar lokasi register dan atau memori.
- Data movement: Instruksi Input/Output (I/O)
- Control : Instruksi untuk tes dan percabangan (*test and branch*)

Arithmetics Instruksi ini menyediakan kemampuan komputasi untuk memproses data numerik. Instruksi logika (*boolean*) beroperasi pada bit-bit kata sebagai bit, bukan sebagai angka; dengan demikian, instruksi ini menyediakan kemampuan

untuk memproses jenis data lain yang mungkin ingin digunakan oleh pengguna.

Instruksi I/O diperlukan untuk mentransfer program dan data ke dalam memori dan hasil perhitungan kembali ke pengguna. Instruksi I/O diperlukan untuk mentransfer program dan data ke dalam memori dan hasil perhitungan kembali ke pengguna. Instruksi cabang (*branch*) kemudian digunakan untuk bercabang ke serangkaian instruksi yang berbeda tergantung pada keputusan yang dibuat.

Number of Addresses

Salah satu cara tradisional untuk menggambarkan arsitektur prosesor adalah dalam hal jumlah alamat yang terkandung dalam setiap konstruksi.

Ternyata, instruksi aritmatika dan logika membutuhkan paling banyak operan. Hampir semua operasi aritmatika dan logika adalah unary (satu operan sumber) atau biner (dua operan sumber). Dengan demikian, kita membutuhkan maksimal dua alamat untuk mereferensikan operan sumber. Hasil dari sebuah operasi disimpan, menunjukkan alamat ketiga, yang mendefinisikan operan tujuan. Akhirnya, setelah menyelesaikan sebuah instruksi, instruksi berikutnya diambil dan memerlukan sebuah diperlukan.

Contoh Sebuah Program mengeksekusi $Y = \frac{A-B}{C+(D \times E)}$

Instruksi		Keterangan
SUB	Y, A, B	$Y \leftarrow A - B$
MPY	T, D, E	$T \leftarrow D \times E$
ADD	T, T, C	$T \leftarrow T + C$
DIV	Y, Y, T	$Y \leftarrow Y \div T$

Tabel 10.3 : instruksi Alamat ke-tiga

Instruksi		Keterangan
MOVE	Y,A	$Y \leftarrow A$
SUB	Y,B	$Y \leftarrow Y - B$
MOVE	T,D	$T \leftarrow D$
MPY	T,E	$T \leftarrow T \times E$
ADD	T,C	$T \leftarrow T + C$
DIV	Y,T	$Y \leftarrow Y \div T$

Tabel 10.4 : Instruksi Alamat ke dua

Instruksi		Keterangan
LOAD	D	$AC \leftarrow D$
MPY	E	$AC \leftarrow AC \times E$
ADD	C	$AC \leftarrow AC + C$
STOR	Y	$Y \leftarrow AC$
LOAD	A	$AC \leftarrow A$
SUB	B	$AC \leftarrow AC - B$
DIV	Y	$AC \leftarrow AC \div Y$
STOR	Y	$Y \leftarrow AC$

Disimpulkan bahwa penggambaran yang dilakukan adalah instruksi dengan 1, 2, ataupun 3 alamat. Dalam setiap kasus pada tabel diasumsikan bahwa alamat instruksi berikutnya bersifat implisit dan bahwa satu operasi dengan dua operan sumber dan 1 hasil operan harus di perlihatkan.

10.10 Rancangan rangkaian instruksi

Desain rangkaian instruksi sangat kompleks karena mempengaruhi begitu banyak aspek sistem komputer. Sebuah rangkaian instruksi mendefinisikan banyak fungsi yang dilakukan oleh prosesor dan memiliki efek yang signifikan bila diimplementasi pada prosesor

Masalah yang paling mendasar berkaitan dengan rancangan rangkaian instruksi yang menjadi perdebatan. Isu-isu rancangan fundamental yang penting meliputi hal berikut :

- Repertoar Operasi : Berapa banyak dan operasi mana yang harus disediakan dan seberapa kompleks operasi yang seharusnya.
- Jenis Data : Berbagai jenis data menjadi dasar operasi dapat dilakukan.
- Format Instruksi : Panjang Instruksi , Jumlah alamat , dan berbagai ukuran field
- Registers : Jumlah register pada prosesor yang bisa mengangap sebuah instruksi, dan
- *Addressing* (Pengalamatan) : Mode yang digunakan untuk menentukan alamat operan.

10.11 Jenis Operan

Instruksi mesin beroperasi pada data. Kategori umum data yang paling penting adalah

- Alamat
- Bilangan/ Angka
- Karakter
- Data Logis

Dalam banyak kasus, beberapa perhitungan harus dilakukan pada referensi operan dalam sebuah instruksi. Untuk menentukan alamat memori utama atau virtual. Alamat dapat digunakan sebagai bilangan bulat tak bertanda (*Unsigned Integer*).

Jenis data umum lainnya adalah karakter pada sebuah angka.

10.12 Angka

Semua bahasa mesin menyertakan tipe data numerik. Bahkan pemrosesan data non-numerik, ada kebutuhan pada angka untuk bertindak sebagai penghitung, lebar bidang, dan sebagainya. Ada beberapa jenis data numerik yang umum di komputer.

- Bilangan Bulat Biner
- Titik ambang Biner (*Floating Point Binary*)
- Desimal

Meskipun semua operasi internal komputer bersifat biner, Pengguna sistem masih berurusan dengan angka desimal. Oleh karena itu, ada kebutuhan untuk mengkonversi dari desimal ke biner pada suatu inputan dari biner menjadi desimal sebagai sebuah output. Lebih baik menyimpan dan mengoperasikan angka dalam bentuk desimal.

10.13 Karakter

Bentuk data yang umum adalah teks atau karakter string. Meskipun data tekstual nyaman bagi manusia, namun tidak demikian halnya dengan data karakter. Dalam bentuk karakter dapat dengan mudah atau ditransmisikan oleh pemrosesan data dan sistem komunikasi. Sistem tersebut dirancang dimana karakter diwakili oleh urutan bit. Kode karakter yang paling umum digunakan di Amerika Serikat disebut sebagai American Standard Code for Information Interchange (ASCII). Setiap karakter dapat diwakili dengan pola 7-bit yang unik; dengan demikian, 128 karakter yang berbeda dapat diwakili.

10.14 Jenis – Jenis Operasi pada CPU

Banyaknya Opkode yang berbeda tersebar antar mesin. Namun demikian, Jenis operasi Umum yang sama dapat ditemukan pada semua mesin. Kategorisasi yang berguna dan umum adalah sebagai berikut:

1. Transfer Data
2. Aritmatika
3. Logika
4. Konversi
5. I/O
6. Kontrol Sistem
7. Transfer Kontrol

Dari tabel berikut [1] Daftar jenis instruksi umum dalam setiap kategori bersama dengan tindakan yang diambil oleh prosesor untuk menjalankan jenis operasi tertentu.

Jenis	Nama Operasi	Deskripsi
Data Transfer	<i>Move (Transfer)</i>	Mentransfer word atau blok dari sumber ke tujuan
	<i>Store</i>	Mentransfer word dari prosesor ke memori
	<i>Load(Fetch)</i>	Mentransfer word dari memori ke prosesor
	<i>Exchange</i>	Menukar konten sumber dan tujuan
	<i>Clear (reset)</i>	Mentransfer word 0s ke tujuan
	<i>Set</i>	Mentransfer kata 1s ke tujuan
	<i>Push</i>	Mentransfer word dari sumber ke bagian atas tumpukan(<i>stack</i>)

Jenis	Nama Operasi	Deskripsi
	<i>Pop</i>	Mentransfer word dari atas tumpukan(<i>stack</i>) ke tujuan
Arithmetic	<i>Add</i>	Menghitung jumlah dua operan
	<i>Subtract</i>	Mengurangi inputan dari dua operan
	<i>Multiply</i>	Mengkalikan inputan dari dua operan
	<i>Divide</i>	Menghitung hasil bagi dari dua operan
	<i>Absolute</i>	Mengganti operan dengan nilai absolutnya
	<i>Negate</i>	Mengubah tanda operan
	<i>Increment</i>	Tambahkan 1 ke operan
	<i>Decrement</i>	Substrak 1 dari operan
Logical	<i>AND</i>	Mengerjakan Logika AND
	<i>OR</i>	Mengerjakan Logika OR
	<i>NOT</i>	(pelengkap) Mengerjakan logika NOT
	<i>Exclusive-OR</i>	Mengerjakan XOR logis
	<i>Test</i>	Menguji kondisi yang ditentukan, atur flag berdasarkan hasil
	<i>Compare</i>	Membuat perbandingan logis atau aritmatika dua atau lebih operan, merangkai flag berdasarkan hasil
	<i>Set Control Variables</i>	Kelas instruksi untuk mengatur kontrol untuk

Jenis	Nama Operasi	Deskripsi
		tujuan proteksi, penanganan Interupsi, Kontrol
	<i>Shift</i>	Bagaian sift Kiri (Kanan) digunakan Menggeser operan, dan menetapkan di akhir
	<i>Rotate</i>	Kiri (Kanan) Menggeser operan, dengan ujung pembungkus
	<i>Jump (branch)</i>	Transfer tanpa syarat; memuat PC dengan alamat yang ditentukan
	<i>Jump Conditional</i>	Menguji kondisi yang ditentukan; memuat PC dengan alamat tertentu atau tidak melakukan apa pun, berdasarkan kondisi
	<i>Jump to subroutine</i>	Tempatkan informasi kontrol program saat ini di lokasi yang diketahui; lompat ke alamat yang ditentukan
	<i>Return</i>	Tempatkan informasi kontrol program saat ini di lokasi yang diketahui; lompat ke alamat yang ditentukan
	<i>Execute</i>	Mengambil operasi dari lokasi yang ditentukan dan menjalankannya sebagai instruksi; jangan

Jenis	Nama Operasi	Deskripsi
		memodifikasi PC
	<i>Skip</i>	Penambahan PC untuk melewati instruksi berikutnya
	<i>Skip Conditional</i>	Menguji kondisi yang ditentukan; baik untuk melewati instruksi berikutnya
	<i>Halt</i>	Menghentikan Eksekusi Program
	<i>Wait (Hold)</i>	Menghentikan Eksekusi Program, uji kondisi yang ditentukan berulang kali; lanjutkan eksekusi ketika kondisi terpenuhi
	<i>No Operation</i>	Tidak ada Operasi yang dilakukan, tetapi eksekusi program dilanjutkan
Input/Output	<i>Input (read)</i>	Mentransfer data dari port I/O atau perangkat tertentu ke tujuan (misalnya, Memori Utama atau register prosesor)
	<i>Output (write)</i>	Mentransfer data dari sumber yang ditentukan ke port I/O atau perangkat
	<i>Start I/O</i>	Transfer instruksi ke prosesor I/O untuk memulai Operasi I/O
	<i>Test I/O</i>	Informasi status transfer dari Sistem I/O ke tujuan tertentu

Jenis	Nama Operasi	Deskripsi
Conversion	<i>Translate</i>	Menerjemahkan Nilai di bagian memori berdasarkan tabel korespondensi
	<i>Convert</i>	Mengonversi isi kata dari satu bentuk ke bentuk lainnya (misalnya, desimal terkemas ke biner)

Tindakan Prosesor untuk Berbagai Jenis Operasi

Transfer Data	Mentransfer Data dari satu lokasi ke lokasi lain
	Jika memori saya terlibat : - Menentukan alamat memori - Melakukan transformasi alamat memori virtual ke aktual - Memeriksa Cache - Memulai baca/tulis memori
Aritmatika	Mungkin melibatkan transfer data, sebelum dan/atau sesudah
	Menjalankan fungsi dalam ALU
	Mengatur kode kondisi dan flag
Logis	Sama seperti aritmatika
Konversi	Mirip dengan aritmatika dan logika. Mungkin melibatkan logika khusus untuk melakukan konversi
Pengalihan Kontrol	Memperbarui penghitung program. Untuk pemanggilan/kembali subrutin, mengelola pengoperan parameter dan hubungan
I/O	Mengeluarkan perintah ke Modul I / O
	Jika I/O yang dipetakan dalam memori, tentukan alamat yang dipetakan dalam memori

10.15 Transfer data

Jenis instruksi mesin yang paling mendasar adalah instruksi transfer data.

1. Lokasi operan sumber dan tujuan harus ditentukan. Setiap lokasi dapat berupa memori, register, atau bagian atas stack.
2. Panjang data yang akan ditransfer harus ditunjukkan.
3. Seperti halnya semua instruksi dengan operan, mode pengalamatan untuk setiap operan harus ditentukan.

Dalam hal tindakan prosesor, operasi transfer data mungkin merupakan jenis yang paling sederhana. Jika sumber dan tujuan adalah register, maka prosesor hanya menyebabkan data ditransfer dari satu register ke register lainnya. Jika salah satu atau kedua operan berada di memori, maka prosesor harus melakukan beberapa atau semua tindakan berikut: Calculate the memory address, based on the address mode.

1. Jika alamat merujuk ke memori virtual, terjemahkan dari alamat memori virtual ke memori nyata.
2. Tentukan apakah item yang dituju ada dalam cache.
3. Jika tidak, keluarkan perintah ke modul memori.[2]

DAFTAR PUSTAKA

- J. P. Hayes, *Computer Architecture and Organization*, 3rd ed. United States of America: The McGraw-Hill Companies, Inc, 1998.
- W. Stallings, *COMPUTER ORGANIZATION AND ARCHITECTURE: Designing for performance*, 10th ed., vol. 148. United States of America, 2016.

BAB 11

REDUCE INSTRUCTION SET COMPUTER (RISC)

Oleh Listra Firgia

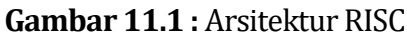
11.1 Pendahuluan

Reduce Instruction Set Computer (RISC) adalah pendekatan desain arsitektur komputer yang bertujuan untuk menyederhanakan set instruksi yang digunakan oleh unit pemrosesan sentral (CPU) sebuah komputer. Tujuan RISC adalah untuk mengurangi jumlah instruksi yang harus dieksekusi oleh CPU untuk melakukan tugas, sehingga membuat CPU lebih cepat dan efisien.

Beberapa fitur utama dari arsitektur RISC meliputi:

1. Set Instruksi Sederhana: Arsitektur RISC menggunakan set instruksi yang direduksi dan disederhanakan, sehingga mempermudah CPU untuk mengdecode dan mengeksekusi instruksi. Ini memungkinkan untuk kecepatan pemrosesan yang lebih cepat dan kinerja yang lebih baik.
2. Arsitektur berbasis register: Dalam arsitektur RISC, sebagian besar pemrosesan data dilakukan melalui register, bukan melalui memori. Ini membuat pemrosesan data lebih cepat dan efisien.
3. Arsitektur load-store: Arsitektur RISC menggunakan arsitektur load-store, yang memisahkan pemrosesan data dari penyimpanan data. Ini mempermudah CPU untuk mengakses data dan mengurangi jumlah instruksi yang diperlukan untuk melakukan tugas.

- Secara ringkas, arsitektur RISC dirancang untuk meningkatkan kinerja dan efisiensi CPU sebuah komputer. Dengan mengurangi jumlah instruksi yang diperlukan untuk melakukan tugas dan menggunakan set instruksi yang disederhanakan, arsitektur RISC membuat pemrosesan data lebih cepat dan efisien.



11.2 Evolusi RISC

180

instruksi yang tersedia sangat kompleks dan beragam. Namun, arsitektur ini membuat CPU komputer menjadi kurang efisien dan lambat, karena membutuhkan waktu yang lebih lama untuk mengdecode dan mengeksekusi instruksi.

Untuk memecahkan masalah ini, para ahli komputer memperkenalkan konsep RISC. Dalam RISC, set instruksi yang tersedia direduksi dan disederhanakan, sehingga CPU dapat lebih mudah mengdecode dan mengeksekusi instruksi. Arsitektur ini juga menggunakan pipeline processing, operasi antar register, dan load-store architecture, sehingga kinerja CPU menjadi lebih efisien dan cepat.

Pada tahun 1981, IBM mengeluarkan produk pertama yang menggunakan arsitektur RISC, yaitu IBM 801. Kemudian, beberapa perusahaan lain seperti Sun Microsystems dan Hewlett-Packard juga memperkenalkan produk yang menggunakan arsitektur RISC.

Pada tahun 1990-an, RISC menjadi arsitektur yang paling populer untuk komputer workstation dan superkomputer. Saat ini, banyak produsen smartphone dan tablet juga menggunakan arsitektur RISC untuk mempercepat kinerja CPU dan membuat sistem lebih efisien.

Evolusi RISC tidak berhenti di sana. Dalam beberapa tahun terakhir, banyak perusahaan mengembangkan arsitektur RISC yang lebih canggih, seperti RISC-V, yang memiliki fitur-fitur baru dan lebih fleksibel untuk memenuhi kebutuhan pasar yang berubah. (Li, 2020)

Evolusi RISC menunjukkan bahwa arsitektur ini sangat penting bagi kinerja dan efisiensi komputer. Dengan arsitektur RISC, para pengguna dapat menikmati kinerja yang lebih baik dan lebih efisien dalam memproses data dan menjalankan aplikasi.

11.2.1 Kelebihan dan Kekurangan RISC

Berikut adalah beberapa kelebihan dan kekurangan *Reduce Instruction Set Computer* (RISC):

Kelebihan:

1. Proses decoding dan eksekusi instruksi yang lebih cepat dan efisien.
2. Set instruksi yang lebih sederhana dan terbatas.
3. Kinerja yang lebih baik dan lebih stabil.
4. Biaya produksi yang lebih rendah.
5. Lebih mudah untuk dikembangkan dan ditingkatkan.

Kekurangan:

1. Terbatasnya fitur dan fleksibilitas.
2. Kemampuan yang lebih rendah untuk menangani tugas kompleks dan beragam.
3. Lebih sedikit instruksi yang tersedia untuk digunakan.
4. Kemampuan yang lebih rendah untuk mengatasi tugas yang tidak dapat diterjemahkan ke set instruksi sederhana.

Secara umum, RISC memiliki kinerja yang lebih baik dan lebih stabil, tetapi memiliki terbatasnya fitur dan fleksibilitas. Ini membuat RISC cocok untuk aplikasi yang membutuhkan kinerja yang cepat dan efisien, tetapi tidak cocok untuk aplikasi yang membutuhkan fitur dan fleksibilitas tinggi.

11.2.2 Konsep RISC

Konsep RISC memiliki beberapa keuntungan seperti kinerja yang lebih baik dan lebih stabil, biaya produksi yang lebih rendah, dan lebih mudah untuk dikembangkan dan ditingkatkan. (Bansal, 2021) Berikut adalah tabel yang menjelaskan beberapa poin penting mengenai *Reduce Instruction Set Computer* (RISC):

Tabel 11.1 : Konsep *Reduce Instruction Set Computer* (RISC)

No.	Konsep	Deskripsi
1.	Set Instruksi Sederhana	Dalam RISC, set instruksi yang tersedia direduksi dan disederhanakan untuk mempermudah CPU mengdecode dan mengeksekusi instruksi. Setiap instruksi hanya memiliki satu tugas utama.
2.	Pipeline Processing	RISC menggunakan pipeline processing, di mana CPU memproses instruksi dalam waktu bersamaan. Ini membuat CPU lebih efisien dan mempercepat waktu eksekusi.
3.	Register-to-register Operation	Dalam RISC, operasi dilakukan antar register dalam CPU, bukan antara register dan memori. Ini membuat operasi lebih cepat dan efisien.
4.	Load-Store Architecture	RISC menggunakan load-store architecture, di mana operasi pemindahan data dilakukan melalui register, bukan melalui alamat memori. Ini membuat operasi pemindahan data lebih cepat dan efisien.
5.	Reduced Instruction Overhead	RISC mengurangi overhead instruksi, seperti instruksi kondisional dan pengulangan, untuk membuat CPU lebih cepat dan efisien.

Sumber: Book Computer Organization and Design: The Hardware/Software Interface

Ini adalah beberapa poin penting dalam RISC yang membantu meningkatkan kinerja dan efisiensi CPU sebuah komputer. Arsitektur ini sering digunakan dalam sistem yang membutuhkan kinerja tinggi, seperti superkomputer, sistem embedded, dan sistem mobile.

11.3 RISC vs CISC

Reduce Instruction Set Computer (RISC) dan *Complex Instruction Set Computer* (CISC) adalah dua arsitektur yang berbeda dalam prosesor komputer. Berikut adalah beberapa perbedaan antara kedua arsitektur tersebut:

1. Set Instruksi: RISC memiliki set instruksi yang lebih sederhana dan terbatas, sementara CISC memiliki set instruksi yang lebih kompleks dan beragam.
2. Ukuran Instruksi: Instruksi dalam RISC lebih kecil dan sederhana, sehingga membuat proses decoding dan eksekusi instruksi menjadi lebih cepat dan efisien. Sementara instruksi dalam CISC lebih besar dan kompleks, sehingga membuat proses decoding dan eksekusi instruksi menjadi lebih lambat.
3. Kinerja: RISC memiliki kinerja yang lebih cepat dan efisien dibandingkan CISC, karena memiliki set instruksi yang lebih sederhana dan memiliki pipeline processing yang lebih baik.
4. Kustomisasi: CISC memiliki lebih banyak fitur dan fleksibilitas untuk dikustomisasi sesuai dengan kebutuhan, sementara RISC lebih terbatas.
5. Harga: RISC biasanya lebih murah dibandingkan CISC, karena memiliki set instruksi yang lebih sederhana dan memiliki kinerja yang lebih baik.

Secara umum, RISC dan CISC memiliki kelebihan dan kekurangan masing-masing. Pilihan terbaik antara kedua arsitektur ini tergantung pada kebutuhan dan tujuan pengguna. Misalnya, RISC lebih cocok untuk aplikasi yang membutuhkan kinerja yang cepat dan efisien, sedangkan CISC lebih cocok untuk aplikasi yang membutuhkan fleksibilitas dan kustomisasi tinggi.

11.3.1 Definisi CISC

Complex Instruction Set Computing (CISC) adalah sebuah arsitektur komputer yang memiliki set instruksi yang kompleks dan banyak. Dalam arsitektur ini, sebuah instruksi dapat melakukan banyak tugas sekaligus, seperti melakukan beberapa operasi aritmatika atau mengakses memori. Banyaknya instruksi yang tersedia membuat prosesor CISC memiliki fleksibilitas yang lebih besar dalam mengatasi berbagai tugas dan membuat kode program menjadi lebih ringkas dan mudah dipahami. (Blem *et al.*, 2015)

Namun, banyaknya instruksi juga mengakibatkan prosesor CISC menjadi lebih sulit untuk diterjemahkan ke dalam bahasa mesin, sehingga memperlambat proses eksekusi. Oleh karena itu, arsitektur CISC seringkali dikritik karena kurang efisien dan lebih lambat dibandingkan dengan arsitektur lain, seperti *Reduced Instruction Set Computing* (RISC). Dalam arsitektur CISC, instruksi-instruksi yang tersedia memiliki panjang yang berbeda-beda, sehingga membutuhkan pengolahan dan penerjemahan yang lebih rumit dan waktu yang lebih lama dibandingkan dengan arsitektur RISC.

Seiring berjalannya waktu, teknologi komputer berkembang dan menyebabkan arsitektur CISC mengalami beberapa perbaikan dan modifikasi. Salah satunya adalah memperbaiki kecepatan eksekusi dengan mengoptimalkan pengolahan dan penerjemahan instruksi. Beberapa produsen prosesor juga mengadopsi konsep CISC dan memodifikasinya untuk memperbaiki efisiensi dan kecepatan eksekusi. Namun, meskipun sudah mengalami perbaikan, arsitektur CISC masih kurang populer dibandingkan dengan arsitektur RISC karena masih dikenal sebagai arsitektur yang kurang efisien dan lebih lambat.

Kesimpulannya, arsitektur CISC adalah suatu arsitektur komputer yang memiliki set instruksi yang kompleks dan

banyak. Kelebihan utama arsitektur ini adalah fleksibilitas yang lebih besar dalam mengatasi berbagai tugas dan membuat kode program menjadi lebih ringkas dan mudah dipahami. Namun, arsitektur ini juga dikenal sebagai arsitektur yang kurang efisien dan lebih lambat dibandingkan dengan arsitektur RISC.

11.3.2 Evolusi CISC

Evolusi CISC (*Complex Instruction Set Computing*) melibatkan perkembangan sistem komputer yang memiliki instruksi yang lebih kompleks dan lebih banyak dibandingkan dengan sistem komputer sebelumnya. Awalnya, komputer menggunakan sistem instruksi sederhana yang memerlukan banyak waktu dan usaha untuk melakukan tugas yang sederhana. Kemudian, pengembang komputer mulai membuat sistem instruksi yang lebih kompleks dan efisien untuk mempercepat proses pemrosesan. Hal ini mengarah pada munculnya CISC, yang memiliki instruksi yang lebih banyak dan kompleks dibandingkan dengan sistem komputer sebelumnya.

Pada tahun 1970-an, banyak perusahaan komputer mulai mengembangkan mikroprosesor CISC yang memiliki banyak instruksi yang mempermudah tugas yang berulang dan mempercepat proses pemrosesan. Salah satu contohnya adalah perusahaan komputer Intel, yang memperkenalkan mikroprosesor 8080 pada tahun 1974. Mikroprosesor ini memiliki instruksi yang lebih kompleks dan mempermudah tugas yang berulang seperti perhitungan matematika dan manipulasi data.

Namun, seiring dengan berkembangnya teknologi komputer, sistem CISC mulai mengalami keterbatasan dalam hal kecepatan dan efisiensi. Oleh karena itu, munculah pendekatan RISC (*Reduced Instruction Set Computing*) yang memiliki instruksi yang lebih sederhana dan efisien dalam hal

waktu dan sumber daya. Meskipun begitu, CISC masih banyak digunakan pada beberapa perangkat komputer dan mikroprosesor tertentu hingga saat ini.

Sebagai kesimpulan, evolusi CISC melibatkan perkembangan sistem komputer yang memiliki instruksi yang lebih kompleks dan lebih banyak dibandingkan dengan sistem komputer sebelumnya. Hal ini membantu mempercepat proses pemrosesan dan mempermudah tugas yang berulang, namun juga memiliki keterbatasan dalam hal kecepatan dan efisiensi.

Beberapa contoh prosesor yang menggunakan arsitektur CISC meliputi prosesor Intel 80386 dan 80486, dan seri Motorola 68000. Prosesor ini biasa digunakan pada komputer pribadi dan workstation awal dan dikenal karena kemampuannya untuk melakukan banyak instruksi dalam satu siklus.

Salah satu fitur utama dari prosesor CISC adalah dukungan mereka untuk instruksi yang kompleks yang melakukan beberapa operasi dalam satu siklus. Misalnya, prosesor Intel 80386 mencakup instruksi yang dapat melakukan operasi seperti manipulasi string dan transfer blok, yang mengurangi jumlah instruksi yang diperlukan untuk melakukan tugas-tugas ini.

Prosesor CISC juga dikenal karena dukungan mereka untuk set instruksi berukuran berubah-ubah, yang memungkinkan penggunaan memory yang lebih fleksibel dan efisien. Ini berarti bahwa prosesor dapat mengeksekusi instruksi dengan ukuran berbeda, memungkinkan fleksibilitas yang lebih besar dalam desain dan optimasi perangkat lunak.

Namun, dengan munculnya prosesor RISC, penggunaan arsitektur CISC telah menurun, karena prosesor RISC terbukti lebih sederhana, efisien, dan lebih mudah didesain dan diproduksi. Namun, beberapa prosesor modern, seperti

arsitektur x86 yang digunakan dalam banyak komputer pribadi, masih mengandung elemen prinsip desain CISC.

DAFTAR PUSTAKA

- Bansal, M. 2021. 'Reduced Instruction Set Computer (RISC): A Survey', in *Journal of Physics: Conference Series*. IOP Publishing, p. 12040.
- Blem, E. *et al.* 2015. 'ISA wars: Understanding the relevance of ISA being RISC or CISC to performance, power, and energy on modern architectures', *ACM Transactions on Computer Systems (TOCS)*, 33(1), pp. 1–34.
- Li, J. 2020. 'Design of a four-stage pipelined reduced instruction set computing microprocessor', *Available at SSRN 3615850*.

BAB 12

PROSES PARALEL DAN SUPERSCALAR DALAM OPERASIONAL KOMPUTER

Oleh Amna

12.1 Proses Paralel

Proses Paralel atau *Parallel Processing* merupakan proses yang terurut yang digunakan ketika akan memproduksi sebuah barang atau produk jadi. Beberapa unit sub komponen akan disusun dan dibentuk untuk dapat menghasilkan suatu produk akhir. Dua proses tersebut dikerjakan secara paralel sehingga pada akhir proses kedua proses akan digabungkan untuk mendapatkan suatu penyelesaian (Asmadi & Rahmawati, 2021). Menurut (Sinduningrum, 2020) digunakan lebih dari satu CPU dalam suatu pemrosesan paralel yang berguna untuk menggerakkan suatu program secara terus menerus. Semakin banyak CPU yang digunakan dalam suatu proses paralel, maka program yang dapat berjalan akan dilakukan dengan lebih cepat.

Sebelum adanya proses paralel, digunakan *batch processing* dalam memproses data. Proses tersebut membutuhkan waktu yang relatif lebih lama karena memakai konsep antrian pada data sehingga suatu data akan dapat diproses setelah proses data sebelumnya selesai. Namun, dengan menggunakan proses paralel komputer dapat melakukan pekerjaan besar yaitu memproses data sekaligus pada beberapa terminal yang ada di setiap host komputer. Dalam proses paralel, dibutuhkan penggabungan antara teknologi komputer dan komunikasi (Agustini, 2021).

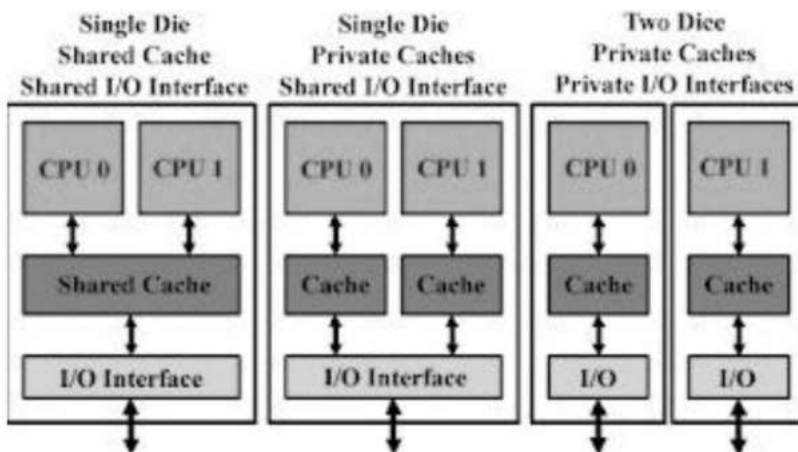
Salah satu contoh penggunaan proses paralel yaitu pada *MapReduce* yang diciptakan oleh *Google*. Pada *MapReduce*, terdapat dua proses yang dilakukan secara paralel yaitu proses *Map* dan proses *Reduce*. Kedua proses tersebut akan dilakukan secara bersamaan tanpa saling bergantung antar satu dengan yang lain. Proses *Map* akan dijalankan untuk mengumpulkan informasi yang berasal dari data yang tersebar pada komputer-komputer dimana suatu *MapReduce* dijalankan. Hasil akhir yang didapatkan oleh proses *Map* akan diberikan kepada proses *Reduce*. Setelah data selesai di proses oleh proses *Reduce*, informasi olahan yang didapatkan akan dikirimkan kepada pengguna (Wijaya, 2019).

Proses paralel dapat diimplementasikan dalam berbagai tingkat dalam sistem komputer menggunakan perangkat keras dan perangkat lunak (Gebali, 2011):

1. Tingkat data paralel. Data yang berjumlah besar dioperasikan secara simultan. Contohnya adalah penambahan bit paralel, perkalian, dan penambahan bilangan biner, prosesor vektor dan *sistolic array* karena berurusan dengan berbagai sampel data.
2. Tingkat instruksi paralel (ILP). Secara simultan dijalankan instruksi yang banyak oleh prosesor. Sebagai contoh dalam instruksi *pipeline*.
3. Tingkat *thread* paralel (TLP). Suatu *thread* adalah porsi dari program yang berbagi sumber prosesor dengan *thread* lainnya. Suatu *thread* juga disebut sebagai proses yang ringan. Dalam TLP, berbagai perangkat lunak dieksekusikan secara simultan dalam satu atau beberapa prosesor.
4. Tingkat proses paralel. Suatu proses adalah program yang menjalankan komputer. Proses memerlukan sumber sendiri, seperti ruang memori dan register. Komputasi *multitasking* dan berbagi waktu dilaksanakan secara simultan dalam satu atau beberapa mesin.

12.2 Teknik Pemrosesan Paralel

Teknik yang digunakan dalam suatu pemrosesan paralel yaitu *Multiprocessing*. *Multiprocessing* merupakan penggunaan dua atau lebih prosesor secara bersamaan untuk mengeksekusi atau menjalankan suatu tugas (Murdoko *et al*, 2019). *Multiprocessing* membantu komputer untuk memproses data dengan lebih cepat. Salah satu implementasi dari *multiprocessing* yaitu dalam proses *multiprogramming*. Pada *multiprogramming*, *multiprocessing* menjadi sumber daya utama untuk menjalankan beberapa tugas ganda atau lebih dari satu proses pada suatu program. *Multiprogramming* menggunakan beberapa CPU yang berbeda dengan fokus untuk menyelesaikan beberapa proses program dengan waktu yang lebih cepat (Mair, 2018).



Gambar 12.1 : *Multiprocessing* (Wahyuddin *et al*, 2023)

Menurut (Anggraeni & Irviani, 2017) multiprocessing mendukung dilakukannya suatu proses multitasking yaitu berjalannya beberapa program dalam waktu yang sama. Setiap satu prosesor akan mengeksekusi satu program. Penggunaan

beberapa prosesor membuat proses eksekusi dapat melakukan beberapa tugas program sekaligus secara paralel. Secara spesifik, *multiprocessing* juga dapat digunakan dalam suatu multithread. Seperti yang digunakan pada *multitasking*, *multiprocessor* dapat digunakan dalam *multithread* yaitu satu prosesor dapat mewakili atau menjalankan satu *thread* dalam *multithread*.

12.3 Kepentingan Proses Paralel

Proses paralel berguna dalam menangani kumpulan tugas program yang ingin diselesaikan. Berbeda dengan teknik *Work-pool* yaitu sekumpulan tugas yang akan diberikan kepada proses-proses *slave* dan kemudian akan berkomunikasi dengan proses *master* untuk mengakses tugas lainnya. Teknik *work-pool* diimplementasikan pada permasalahan yang dapat dibagi menjadi tugas-tugas kecil. Pada teknik *work-pool*, dibutuhkan waktu untuk menunggu tugas-tugas kecil untuk diselesaikan. Namun, hal ini berbeda dengan pemrosesan program menggunakan proses paralel (Wilkinson & Allen, 2010).

Pada proses paralel, program dieksekusi sekaligus sehingga waktu yang dibutuhkan oleh program akan menjadi lebih singkat. Waktu eksekusi yang lebih singkat juga dapat membantu penghematan daya pada prosesor dan pada bagian sirkuit. Dalam memproses data, waktu merupakan salah satu aspek yang terpenting. Mendapatkan waktu yang sesingkat mungkin untuk memproses data menjadi sesuatu yang penting dan dapat menguntungkan. Sebagai gantinya, dibutuhkan memori yang lebih besar pada proses paralel karena program diproses secara bersamaan sehingga dibutuhkan ruang penyimpanan yang lebih luas untuk data yang ada (Sabry, 2023).

12.4 Superscalar

Salah satu cara untuk meningkatkan laju dari *prosesor* adalah dengan menggabungkan sifat paralel sementara yang digunakan dalam proses *pipeline* dan data paralel dengan mengeluarkan beberapa instruksi dari sambungan instruksi secara simultan dalam setiap siklus. Instruksi dapat diberikan secara tidak berurutan dalam sambungan tetapi dapat berurutan ketika memaksimalkan penggunaan sumber perangkat keras. Hal ini disebut sebagai proses *superscalar* (Rajaraman dan Murthy, 2016). *Superscalar* merupakan suatu sebutan teknologi yang dimiliki oleh intel. *Superscalar* dapat mengeksekusi lebih dari satu instruksi program. Teknologi superscalar ini terkandung di dalam chip intel salah satunya yaitu chip RISC (*Reduced Instruction Set Computer*) yang memiliki *output* tinggi. Instruksi pada *chip* tersebut tidak begitu sulit namun *clock speed* yang dimilikinya tinggi sehingga performa memproses data yang dimiliki chip tersebut lebih cepat (Daulay, 2007).

Menurut Clements (2014), struktur *prosesor superscalar* terbagi menjadi lima elemen yaitu:

1. Instruksi *fetch*

Instruksi *fetch* berfungsi untuk mengambil instruksi dari memori (*cache* atau tujua khusus dengan mekanisme lainnya untuk memegang instruksi) yang akan dieksekusi.

2. Instruksi *decode*

Instruksi *decode* dan menamakan kembali register membuka kode instruksi dan menamakan kembali sumber yang dibutuhkan oleh instruksi tersebut. Menamakan kembali register akan mereduksi efek leher botol yang disebabkan oleh jumlah register pengguna terlihat yang terbatas dengan menggunakan register sementara, di mana nama dari register tersebut diganti pada saat *runtime* untuk mengelakkan konflik di antara instruksi dengan register yang sama.

3. Instruksi isu

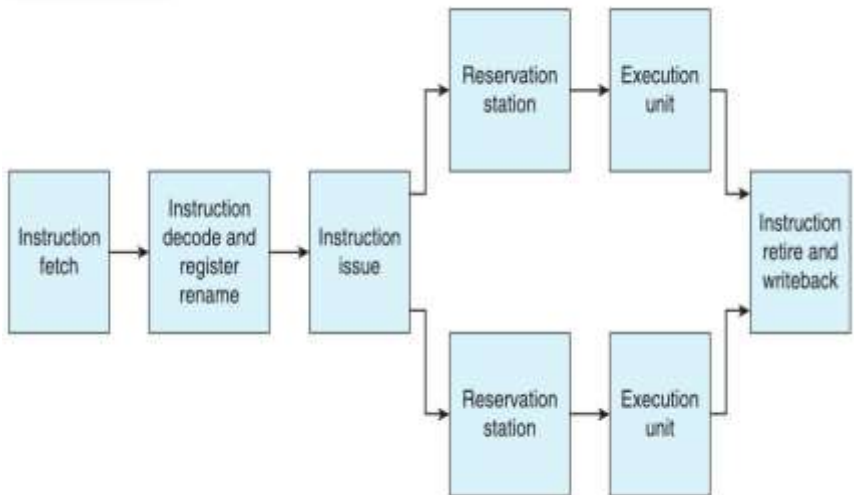
Tahap instruksi isu meneruskan instruksi kepada unit eksekusi dan memastikan bahwa jumlah maksimal dari instruksi memungkinkan untuk dijalankan secara paralel. Dua istilah isu dan *dispatch* mendeskripsikan penerusan instruksi untuk eksekusi. *Dispatch* mengindikasikan bahwa instruksi diberikan ke unit fungsional untuk eksekusi dan isu yang mengindikasikan bahwa suatu instruksi diberika kepada unit eksekusi meskipun instruksi tersebut belum tersedia untuk eksekusi langsung.

4. Stasiun reservasi

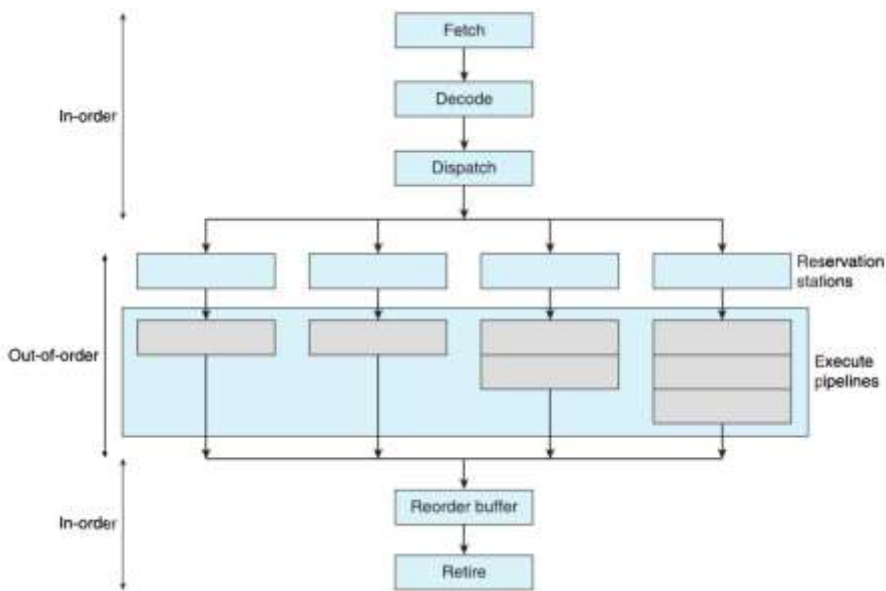
Stasiun reservasi dan unit eksekusi adalah penyangga instruksi dan ALU (atau alat fungsional lainnya) yang menginterpretasikan aliran dari *kode-op*. Stasiun reservasi mengeluarkan instruk yang berkaitan dengan unit proses saja ketika seluruh sumber yang dibutuhkan tersedia.

5. Instruksi *retire*

Tahap instruksi *retire* dan *writeback* bertanggung jawab dalam menuliskan kembali hasil dari register dan memastikan bahwa instruksi yang ada lengkap dan dalam susunan yang benar. Tahap ini juga dikomunikasikan dengan tahap resevasi ketika instruksi yang sudah selesai membebaskan sumber.



Gambar 12.2 : *Prosesor superscalar* secara umum
(Clements, 2014)



Gambar 12.3 : Aliran informasi dalam *prosesor superscalar* (Clements, 2014)

Prosesor superscalar dapat dibagi menjadi tiga jenis yaitu *Statically Scheduled superscalar processors*, *VLIQ (very long instruction Word) processors* dan *Dynamically scheduled superscalar processors*. Dua dari ketiga jenis *prosesor superscalar* tersebut membedakan instruksi per jam yang menggunakan suatu jadwal atau *out-oforder* eksekusi sesuai yang telah terjadwalkan secara dinamis. Pada *prosesor VLIWs*, fungsionalnya bersifat independen. VLIW menyusun suatu instruksi yang sangat panjang dari suatu paket berganda dari operasi yang akan dilakukan. Petunjuk yang ada di dalam paket itu program harus memiliki suatu kendala yang sama. Hal ini dapat membantu beberapa program ditempatkan dalam suatu instruksi yang sama sehingga dapat dieksekusi secara paralel (Malik *et al*, 2018).

12.5 Proses Paralel dan *Superscalar*

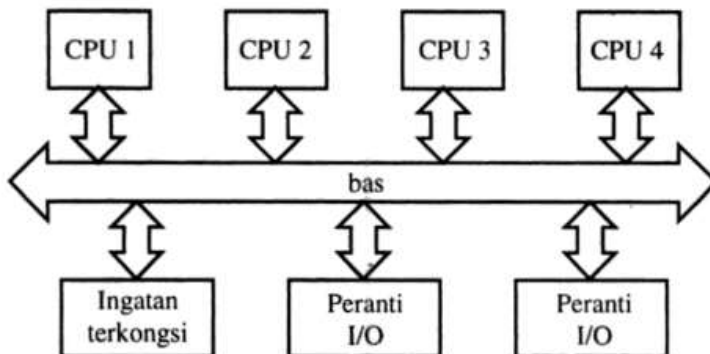
Terdapat beberapa evolusi prosesor yang dikeluarkan oleh intel. Namun secara spesifik, teknologi superscalar baru digunakan dan diterapkan pada prosesor pada tahun 1993. Prosesor pertama yang menggunakan teknologi superscalar yaitu pada prosesor Pentium. Prosesor tersebut memungkinkan instruksi pada suatu program dijalankan secara paralel. Selanjutnya, pada tahun 1995 prosesor baru yaitu Pentium Pro diluncurkan. Terdapat suatu perkembangan dari prosesor Superscalar yang sebelumnya. Pada prosesor Pentium Pro, terdapat fitur yang baru ditemukan seperti sistem untuk memprediksi cabang dari suatu program, menganalisa alur dari data program dan sistem cache pada memori yang semakin maju (Hardi, 2021).

Pada versi perkembangan selanjutnya dari prosesor Pentium ini dilanjutkan dengan Pentium II, Pentium III dan Pentium IV. Terdapat beberapa peningkatan fitur yang dimiliki setiap prosesor tersebut. Mulai dari penerapan teknologi MMX yang dapat memenuhi kebutuhan multimedia dan telah menerapkan teknologi RISC. Lalu diikuti Pentium III yang memiliki instruksi *floating point* sehingga dapat mengolah grafis 3D dan terakhir Pentium IV yang diproduksi tahun 2000 yang memiliki kecepatan proses yang telah mencapai 3,06 GHz (Fachri et al, 2020).



Gambar 12.3 : Pentium-4 (Noersasongko & Andono, 2010)

12.6 Proses Paralel dan *Superscalar* dalam Operasional Komputer



Gambar 12.4 : Operasi komputer paralel dengan 4 CPU (Zabidi, 1996)

Di dalam pemrosesan paralel, terdapat ratusan maupun ribuan proses yang berada dalam suatu sistem. Kesusahan yang dapat dimiliki oleh komputer operasional yaitu bagaimana proses yang begitu banyak dapat disambungkan dan di eksekusi secara paralel. Upaya untuk memindahkan data-data tersebut sangat penting karena setiap tugas pada sistem dilakukan secara bersamaan. Ketika menggunakan proses paralel, harus dipastikan bahwa semua pemroses sedang aktif pada saat itu. Hal ini karena aktivitas proses yang tidak beraturan dan pemrosesan yang tidak melakukan tugas apapun pada program dapat menurunkan kegunaan dari sistem (Zabidi, 1996).

12.7 Proses Paralel dan *Superscalar* dalam Peningkatan Kinerja Komputer

Dilakukan perubahan struktural pada program untuk meningkatkan kinerjanya. Dapat dilakukan optimisasi pada komputer seperti pada program prosesor tunggal maupun optimisasi paralel. Hal ini dapat dijalankan dengan sistem multiprosesor (Wilkinson & Allen, 2010). Menurut (Rock, 2006) dua chip yang dapat bekerja dua kali lebih cepat dapat mengerjakan proses yang berbeda dengan dua kali waktu lebih cepat ketika digunakan secara bersamaan.

Menurut Gebali (2011), prosesor superscalar memiliki kemampuan untuk mengeksekusi instruksi yang berjumlah besar secara paralel. Teknik kompilasi dan perangkat keras digunakan untuk memaksimalkan jumlah instruksi yang dapat digunakan secara paralel. Namun masih terdapat batasan untuk mencapai peningkatan laju seperti berikut:

1. Ketergantungan terhadap *true data (read after write (RAW))*
Baca setelah tulis atau *read after write* mengimplikasikan bahwa instruksi *I* harus membaca nilai baru dari register setelah instruksi lain *j* yang dilaksanakan sebagai operasi menulis. Diasumsikan instruksi I_0 memproduksi hasil dan

instruksi I_i menggunakan hasil tersebut. Dapat dikatakan bahwa I_1 memiliki ketergantungan terhadap *true data* terhadap I_0 dan eksekusi dari I_1 harus ditunda sehingga I_0 selesai.

2. Ketergantungan terhadap prosedur

Permasalahan terbesar terhadap instruksi komputer adalah keberadaan instruksi cabang. Penyelesaian instruksi cabang harus ditunda setelah instruksi pertama menghasilkan *output*, sehingga fase *fetch* untuk instruksi cabang diperlambat.

3. Konflik sumber

Konflik sumber terjadi ketika dua atau lebih instruksi membutuhkan sumber prosesor yang sama. Sebagai contoh prosesor yang berbagi sumber adalah memori, *cache*, *buses*, dan *file register*. Konflik sumber dapat terjadi ketika eksekusi dari instruksi yang berkompetisi tertunda. Berbeda dengan ketergantungan terhadap *true data*, konflik sumber dapat dieliminasi dengan menduplikasi sumber yang terbagi. Hal ini dapat menjadi solusi yang kurang praktis dan mahal. Misalnya, dengan mengeliminasi konflik unit *floating-point*, ia menyangkut desain dua unit *floating point* yang berkaitan dengan setiap ALU. Hal ini dapat membutuhkan nilai *silicon real estate* yang sedikit. Konflik *cache* dapat dihilangkan dengan mendesain *cache* dengan dua *port* atau menduplikasi *cache* tersebut meskipun kurang praktis.

4. Ketergantungan kepada output (*write after write* (WAW))

WAW mengimplikasikan bahwa instruksi I menulis suatu operandi setelah instruksi j tertulis pada operandi lain di register. Urutan ini penting dikarenakan register seharusnya mengandung nilai tertulis dari instruksi j setelah kedua instruksi I dan j telah selesai dan mengeksekusi. Ketergantungan terhadap *output* terjadi ketika dua instruksi, I_0 dan I_1 menyimpan hasil *output* di dalam register yang sama. Dalam kasus tersebut, kandungan register pada waktu

tertentu bergantung kepada instruksi yang selesai terakhir, I_0 atau I_1 .

5. Tidak memiliki ketergantungan (*write after read* (WAR))

WAR mengimplikasikan bahwa instruksi i menulis suatu operandi setelah instruksi j telah membacakan kandungan dari register. Tidak ketergantungan diilustrasikan dengan bantuan pecahan kode RTL umum untuk menjelaskan ketergantungan *output*.

DAFTAR PUSTAKA

- Agustini, K. 2021. *Komunikasi Data dan Jaringan Komputer serta Analoginya dalam Konsep Subak - Rajawali Pers*. PT Rajagrafindo Persada.
- Anggraeni, E. Y., dan Irviani, R. 2017. *Pengantar Sistem Informasi*. Penerbit Andi.
- Asmadi, D. dan Rahmawati, S. 2021. *Analisis dan Estimasi Biaya*. Syiah Kuala University Press.
- Clements, A. 2014. *Computer Organization and Architecture*. Cengage Learning.
- Daulay, M. S. 2007. *Mengenal Hardware-Software & Pengelolaan Instalasi Komputer*. Penerbit Andi.
- Fachri, B., Parinduri, I., Hutagalung, S. N., dan Harahap, R. R. 2020. *Arsitektur dan Organisasi Komputer*. Penerbit Yayasan Kita Menulis.
- Gebali, F. 2011. *Algorithms and Parallel Computing*. John Wiley and Sons.
- Hardi, R. 2021. *Arsitektur dan Organisasi Komputer*. Penerbit Uwais Inspirasi Indonesia.
- Mair, Z. R. 2018. *Teori dan Praktek Sistem Operasi*. Deepublish.
- Malik, R. F., Passarella, R., Exaudi, K., dan Zulfahmi, R. 2018. *Buku Ajar Mata Kuliah Organisasi dan Arsitektur Komputer*. Unsri Press.
- Murdoko, J., Pangaribuan, G., dan Waluyo, R. 2019. *Pemrograman Java Untuk Sains dan Teknik*. Penerbit Cahaya Informatika.
- Noersasongko, E., dan Andono, P. N. 2010. *Mengenal Dunia Komputer*. PT Elex Media Komputindo.
- Rajaraman, V. dan Murthy, C.S.R. 2016. *Parallel Computers: Architecture and Programming*. Edisi kedua. PHI Learning Private Limited.

- Rock, D. 2006. *Quiet Leadership: Enam Langkah Mengubah Kinerja demi Kesuksesan Perusahaan Anda*. Penerbit PT Gramedia Pustaka Utama.
- Sabry, F. 2023. *Komputasi Tujuan Umum Pada Unit Pemrosesan Grafik*. Penerbit Satu Miliar Berpengetahuan.
- Sindiningrum, E. 2020. *Teori Organisasi Arsitektur Komputer & Praktik Assembler Untuk Pemula*. Deepublish.
- Wahyuddin, S., Nurmuslimah, S., Ariantoro, T. R., Yanto, G., Yuswardi, Dewantara, R., Jamaludin, Sihombing, F. A. H., Harlina, S., Yusuf, A. M., dan Rachman, A. 2023. *Sistem Operasi Komputer*. PT Global Eksekutif Teknologi.
- Wijaya, W. M. 2019. *Teknologi Big Data: Sistem Canggih di Balik Google, Yahoo!, Facebook, IBM*. Penerbit Nilacakra.
- Wilkinson, B., dan Allen, M. 2010. *Teknik & Aplikasi Menggunakan Jaringan Workstation dan Komputer Paralel*. Penerbit Andi.
- Zabidi, M. M. A. 1996. *Asas Organisasi Sistem Komputer*. Penerbit Universiti Teknologi Malaysia.

BIODATA PENULIS



Romindo, S.Kom., M.Kom., CLMA., CBPA
Dosen Program Studi Sistem Informasi
Fakultas Ilmu Komputer

Romindo, S.Kom., M.Kom., CLMA., CBPA lahir di Bekasi, 11 November 1991. Saat ini bertempat tinggal di Medan - Sumatera Utara, Indonesia. Gelar non akademik Certified Leadership Management Associate (CLMA) dan Certified Book and Paper Authorship (CBPA) pada tahun 2020. Saat ini Penulis menjadi Dosen di Universitas Pelita Harapan. Sejak itu Penulis mendalami dan mengajar mata kuliah Web Programming terutama dalam bidang Internet of Things, serta Penulis juga aktif dalam menulis buku, berbagai artikel ilmiah dan menjadi narasumber dalam beberapa seminar dan workshop nasional.

BIODATA PENULIS



Rini Nuraini, S.T., M. Kom

Dosen Universitas Nasional Jakarta dan di tempatkan di Fakultas Teknologi Komunikasi dan Informatika pada Prodi Informatika

Rini Nuraini, S.T., M. Kom, menyelesaikan kuliah dan mendapat gelar Sarjana Teknik Komputer (S.T.) untuk S1 dan Magister Ilmu Komputer (M. Kom) untuk S2. Karir sebagai Dosen Universitas Nasional Jakarta dan di tempatkan di Fakultas Teknologi Komunikasi dan Informatika pada Prodi Informatika. Dosen Tersertifikasi pada Tahun 2011. Jabatan Fungsional Dosen Lektor. Penerima penelitian hibah dikti pada tahun 2009 dengan tema *E-Learning*. Kajian Riset: *E-Learning, Robotic, Artificial Intelligence (Deep Learning), Machine Learning* dan *Decision Support System*.

BIODATA PENULIS



Rita Komalasari, M.Kom

Saat ini penulis sudah memiliki jabatan fungsional akademik lektor, merupakan alumni dari Ilmu Komputer Universitas Padjadjaran dan Sistem Informasi STMIK LIKMI. Penulis aktif menjadi editor dan reviewer jurnal nasional terakreditasi Sinta serta aktif dalam menulis artikel di jurnal nasional maupun internasional serta menulis buku. Selain itu pula penulis aktif sebagai pengurus bidang publikasi dan HAKI pada organisasi Asosiasi Pendidikan Tinggi Informatika dan Komputer (APTIKOM) Propinsi Jawa Barat, sebagai Bendahara Umum pada organisasi Perkumpulan Auditor Teknologi Informasi dan Komunikasi Indonesia (PASTIKINDO). Penulis memiliki kompetensi Information Technology Expert Auditor, ICT Project Manager dan System Analyst.

BIODATA PENULIS



Fajar Sari Kurniawan, S.T., M.Kom.

Dosen Program Studi Transportasi Laut
Jurusan Bisnis Maritim Politeknik Maritim Negeri Indonesia

Lahir di Semarang tanggal 4 Desember 1983. Menyelesaikan pendidikan S1 pada Jurusan Teknik Elektro dan melanjutkan S2 Magister Sistem Informasi di Universitas Diponegoro. Penulis adalah dosen PNS Politeknik Maritim Negeri Indonesia pada Program Studi Transportasi Laut Jurusan Bisnis Maritim. Mengampu mata kuliah Aplikasi Komputer dan Teknologi Informasi. Pernah mengajar di kampus Universitas Ivet dan Akademi Teknik Perkapalan pada mata kuliah Teknologi Informasi, Gambar Teknik, dan Sistem Telekomunikasi dan Navigasi. Penulis juga pernah bekerja pada bidang telecommunication engineering di perusahaan internasional ZTE Company.

BIODATA PENULIS



Zul Rachmat, S.Kom., M.M.

Ketua Sekolah Tinggi Manajemen Informatika
dan Komputer (STMIK) Amika Soppeng Program Studi Manajemen
Informatika

Penulis lahir di Ujung Pandang tanggal 22 Maret 1990. Penulis adalah dosen pada Program Studi Manajemen Informatika di STMIK Amika Soppeng dan saat ini diamanahkan sebagai Ketua di perguruan tinggi tersebut. Menyelesaikan pendidikan S1 pada program studi sistem informasi di Universitas Dipa Makassar pada tahun 2012 dan melanjutkan pendidikan S2 pada Jurusan Manajemen di Universitas Muslim Indonesia pada tahun 2015. Memiliki ketertarikan pada dunia Pendidikan khususnya di berbagai penelitian ilmiah dan tulisan yang dimuat pada berbagai jurnal ilmiah. Telah menjadi narasumber di berbagai kegiatan seminar dan pelatihan khususnya yang berkaitan dengan manajemen informatika. Motto hidup yakni sebaik-baiknya manusia ialah yang bermanfaat bagi sekelilingnya.

BIODATA PENULIS



Wahyuddin S, S.Kom., M.Kom
Dosen STMIK Amika Soppeng

Wahyuddin S. was born at Malaka-Bone-Sulawesi Selatan in 1992. In 2011 he attended STMIK Dipanegara Makassar and was completed in 2015. He was completed after attending 7 semesters and active on an XPcom (Extreme Programmer Computer) campus organization. He was also active as a lecturer assistant for three semesters and taught several courses on programming. He continued his Master of Information systems at UNIKOM Bandung in 2016 and was completed in April 2019. He worked as a lecturer at a campus (STMIK AMIKA Soppeng) 2019 to present and also a Freelance Web Programmer. Has competence in the field of software engineer, application developer, multimedia, web developer, network security, and data analyst.

BIODATA PENULIS



Zen Munawar.

Dosen Program Studi Manajemen Informatika
Politeknik LP3I Bandung

Penulis memiliki jabatan akademik lektor, dan alumni dari TI UNPAS, Pasca Sarjana AdPen UNINUS, Sistem Informasi dari STMIK IM dan Pasca Sarjana Sistem Informasi STMIK LIKMI Bandung serta mengikuti Program Doktorat Teknologi Informasi dan Komunikasi di UTeM Melaka Malaysia. Penulis aktif menjadi editor board dan reviewer jurnal nasional terakreditasi Sinta serta aktif dalam menulis artikel di jurnal nasional maupun internasional serta menulis buku. Selain itu pula penulis aktif sebagai pengurus bidang kerjasama industri pada organisasi APTIKOM Prov. Jabar, sebagai sekretaris DPW Jawa Barat pada Ikatan Ahli Informatika Indonesia (IAII) sebagai Dewan Pengawas pada organisasi Perkumpulan Auditor Teknologi Informasi dan Komunikasi Indonesia (PASTIKINDO). Penulis memiliki kompetensi Information Technology Expert Auditor, ICT Project Manager, System Analyst dan Assesor Of Competency dari BNSP.

BIODATA PENULIS



Ir. Johni S. Pasaribu, MT.

Dosen Program Studi Sistem Informasi
Fakultas IT Politeknik Piksi Ganesha

Penulis lahir di Medan tanggal 1 Juni 1966 dari pasangan HA Pasaribu (alm) dan Taruli Tambunan (alm). Penulis adalah anak keempat dari enam bersaudara. Pendidikan yang ditempuh yaitu SD Methodist Medan lulus tahun 1979, SMPN 88 Jakarta lulus tahun 1982 dan SMAN 3 Jakarta lulus tahun 1985. Kemudian melanjutkan studi di ITB jurusan Teknik Fisika spesialisasi Instrumentasi dan Kontrol. Penulis melakukan penelitian untuk tugas akhir di PT IPTN dengan judul Perancangan Sistem Kendali Fly By Wire Gerak Longitudinal Pada Engineering Flight Simulator Pesawat N250-100 dan menyelesaikan gelar sarjana teknik Insinyur (Ir) tahun 1992.

Kemudian penulis bekerja di PT. IPTN (sekarang PT Dirgantara Indonesia) di Departemen Iron Bird kemudian berpindah di Departemen Engineering Flight Simulator (EFS). Penulis terlibat dalam Operation and Maintenance untuk EFS sehingga simulator N250-100 ini dapat digunakan saat dilakukan pengujian-pengujian pesawat melalui simulator sebelum pengujian langsung pesawat di udara. Tahun 1998 penulis melanjutkan studi

S2 di ITB pada jurusan Teknik Informatika untuk spesialisasi Software Engineering, dimana ilmu yang diperoleh dapat diaplikasikan dalam pekerjaan di lapangan. Tahun 2001 penulis menyelesaikan pendidikan S2 dengan gelar MT mengambil penelitian yang berjudul Pengembangan Perangkat Lunak Simulasi Sistem Perasa Kendali Elevator Pesawat N250-100 dimana penelitian ini juga dilakukan di Engineering Flight Simulator (EFS) N250-100. Tahun 2004 penulis mulai aktif mengajar di Perguruan Tinggi di Bandung diantaranya pernah mengajar di Unikom, ITHB, STT Telkom (kini TelU). Kini penulis penuh mengajar dan menjadi dosen tetap di Politeknik Piksi Ganesha. Bidang keahlian mengajar dan penelitian penulis adalah Artificial Intelligence (AI), Decision Support System (DSS), Software Engineering dan Software Testing. Penulis juga aktif melakukan bimbingan mahasiswa S1 dan publikasi di jurnal-jurnal nasional maupun internasional.

BIODATA PENULIS



Novianti Indah Putri

Dosen Program Studi Sistem Informasi
Universitas Kebangsaan Republik Indonesia

Penulis merupakan dosen tetap di program studi Sistem Informasi program sarjana Fakultas Ilmu Komputer dan Sistem Informasi (FIKSI), Universitas Kebangsaan Republik Indonesia (UKRI) di Bandung. Penulis merupakan alumni dari Teknik Informatika Universitas Islam Negeri Sunan Gunung Djati (UIN SGD) Bandung dengan predikat Cumlaude, Pasca Sarjana Teknik Informatika Universitas Langlangbuana (UNLA) Bandung. Penulis aktif dalam menulis artikel di jurnal nasional terakreditasi SINTA. Penulis sebagai editor di Penerbit Kaizen Multimedia Publishing. Penulis juga aktif dalam melaksanakan penelitian dan publikasi ilmiah yang ditunjukkan dengan nilai saat ini pada H-indeks Google Scholar 12. Selain itu pula penulis sebagai anggota pada organisasi Asosiasi Pendidikan Tinggi Informatika dan Komputer (APTIKOM) Provinsi. Jawa Barat.

BIODATA PENULIS



Niko Akbar, M.Kom

Dosen Tetap Ilmu Komputer Universitas Dinamika Bangsa Jambi

Penulis lahir di jambi pada tanggal 24 Desember 1994. Saat ini merupakan dosen tetap Ilmu Komputer Program Studi Teknik Informatika , Telah menyelesaikan Pendidikan S1 pada tahun 2017 dan telah menempuh Pendidikan S2 di Sekolah Tinggi Ilmu Komputer (Sekarang Universitas Dinamika Bangsa).

Penulis juga telah menulis beberapa Book Chapter sebagai author, salah satunya adalah *Pigeon Hole Principle*(di buku Logika dan Struktur Diskrit). Selain itu, penulis juga aktif di berbagai penelitian dan Pengabdian yang penulis jalankan.

BIODATA PENULIS



Listra Firgia, S.Kom., M.Kom.

Dosen Program Studi Teknologi Informasi
Institut Shanti Bhuana

Penulis lahir di Pontianak tanggal 13 Mei 1990. Penulis adalah dosen tetap pada Program Studi Teknologi Informasi dan menjadi Kepala Pusat Riset dan Pengabdian Masyarakat Institut Shanti Bhuana. Menyelesaikan pendidikan S1 pada Jurusan Sistem Informasi di STMIK Insan Pembangunan yang sekarang menjadi Universitas Insan Pembangunan Indonesia dan melanjutkan S2 pada Jurusan Ilmu Komputer di Universitas Budi Luhur. Penulis aktif dalam membuat artikel ilmiah dan buku.

BIODATA PENULIS



Amna

Dosen prodi teknik informatika di STTP Paya kumbuh

Memulai pendidikan dasar dan menengah di kota kelahirannya Takengon Aceh Tengah dan kemudian melanjutkan pendidikan tinggi di bidang matematika di USK Banda Aceh pada tahun 1988 dan memperoleh gelar sarjana pada 1992. Tahun 1993 melanjutkan program Pra S2 Matematika di ITB Bandung. Program S2 pula di tempuh pada Mathematics Department Fakulti Sains matematik dan komputer Universiti Kebangsaan Malaysia (UKM). Akhir tahun 1994 menukar jurusan ke Sains Komputer seiring dengan terbentuknya Fakulti Teknologi dan Sains Maklumat UKM. Selama melanjutkan studi S2 sangat aktif sebagai Tutor sambilan untuk mata kuliah matematik, statistik dan ilmu komputer, juga sebagai Asisten Research untuk bidang ilmu komputer dan rekayasa perangkat lunak. Setelah menamatkan S2 tahun 1997 dunia Dosen pun digeluti dengan memulai sebagai Dosen program degree sains sistem di kolej Politek MARA Bangi untuk franchise program UKM-KYPM. Kemudian sebagai dosen D3 sains komputer prancis dengan UiTM Shah Alam, UPM Serdang juga program degree software engineering dengan MMU Cyberjaya di KYPM Kuala Lumpur. Tahun 2001 kembali ke Kampus UKM

Bangi sebagai dosen sains sistem. Tahun 2003 bergabung dengan Kolej SAL Kuala Lumpur sebagai dosen program prancise degree in software engineering dan degree in networking dengan Edith Cowan University Western Australia. Tahun 2004 bergabung dengan Universiti Tenaga Nasional (UNITEN) sebagai asisten research bidang parallel processing dan dosen matematik di college Of engineering sambil melanjutkan studi S3. Tahun 2008 bergabung dengan College of IT UNITEN sebagai dosen system and network departement. Tahun 2010 hingga sekarang bergabung dengan Universitas Gajah Putih sebagai dosen teknik informatika. Banyak jurnal dan proseding baik internasional dan nasional yang telah di hasilkan dalam jangka waktu tersebut. Juga mengisi berbagai macam konferensi, simposium, dan webinar. Sejak 2017 hingga sekarang sebagai dosen tamu prodi teknik informatika di STTP Paya kumbuh. Empat book chapter yang telah di hasilkan: *The art of digital marketing* strategi pemasaran generasi milanial, *Fintech innovation essence, position & strategy*, Sistem operasi dan Tahapan rekayasa perangkat lunak. Hal terbaru dari riset pada 27 Desember 2021 mendapat tempat harapan 1 lomba Inovasi LISIK 2021menerusi judul riset Kamus Bahasa Gayo Berbasis android. Lahir dari ayah Abdurrahman AS dan ibu Fatimah dan menikah dengan Dr. Anna Permatasari Kamarudin, S.Tp, MBA sejak tahun 2000 dan dikaruniakan dengan 3 putri diberi nama Adiba Imani Salsabila, Najla Raihana Kamila dan Khayla Rahadatul Fakhira. Email Penulis: amnaa98@hotmail.com