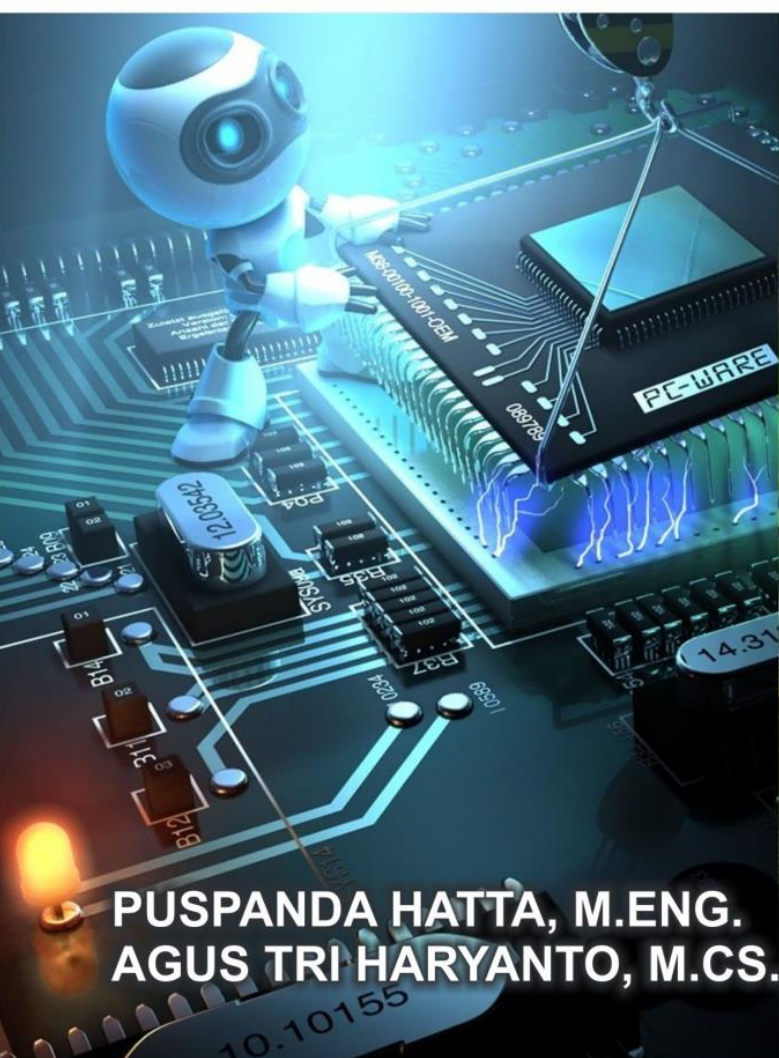




Kementerian Pendidikan Dan Kebudayaan
Republik Indonesia



SISTEM KOMPUTER



PUSPANDA HATTA, M.ENG.
AGUS TRI HARYANTO, M.CS.



SMK/MAK
KELAS
XI
SEMESTER II

Hak Cipta pada Kementerian Pendidikan dan Kebudayaan
Dilindungi Undang-Undang

Milik Negara
Tidak Diperdagangkan

Kontributor: Endra Sumartono, S.Kom., Gr.
Penyunting Materi: Dr. Agus Efendi, M.Pd
Penyunting bahasa : Badan Bahasa
Penyelia penerbitan : Politeknik Negeri Media Kreatif
Jakarta

Kotak katalog dalam terbitan (KDT)

Cetakan Ke-1, 2013
Disusun dengan huruf Palatino 14 pt.

KATA PENGANTAR

Dengan mengucapkan puji syukur Alhamdulillah kepada Tuhan YME, karena berkat limpahan rahmat dan kasih-Nya, akhirnya penulis dapat menyelesaikan penulisan modul untuk mata pelajaran SISTEM KOMPUTER yang merupakan bagian dari paket keahlian REKAYASA PERANGKAT LUNAK, TEKNIK KOMPUTER JARINGAN dan MULTIMEDIA untuk kelas X. Dalam penyusunan buku ini tidak terlepas dari kendala dan hambatan, namun berkat bimbingan dan motivasi dari semua pihak yang telah membantu maka kami dapat menyelesaikannya

Perkembangan ilmu komputer yang sangat cepat dewasa ini perlu diimbangi dengan pengetahuan dasar teorinya. Buku ini membahas tentang sistem bilangan, operasi aritmatika, gerbang logika, arsitektur komputer, media penyimpanan dan memori, serta memori semikonduktor.

Penulisan modul mata pelajaran SISTEM KOMPUTER ini bertujuan untuk membantu siswa kelas X RPL TKJ maupun MULTIMEDIA dalam memahami dan mempelajari konsep dasar teknologi sistem komputer dari sisi perangkat keras serta komponen-komponen pendukungnya. Buku ini dirancang untuk Kurikulum 2013 SMK untuk memperkuat kompetensi peserta didik dari sisi pengetahuan, ketrampilan, dan sikap secara utuh.

Dalam kesempatan ini penulis mengucapkan dan menyampaikan terima kasih yang sebesar-besarnya kepada semua pihak yang tidak dapat disebutkan satu persatu. Semoga amal baiknya mendapatkan balasan yang sesuai dari Allah SWT.

Penulis menyadari bahwa dalam menyusun modul ini masih jauh dari sempurna. Hal itu tidak lepas dari keterbatasan kemampuan kami. Oleh karena itu saran dan kritik yang membangun sangatlah penulis harapkan demi kebaikan dan kesempurnaan dalam penulisan modul selanjutnya. Akhir kata, semoga buku ini dapat bermanfaat dan dapat memenuhi harapan sebagaimana mestinya,

Surakarta, Nopember 2014

Penulis

DAFTAR ISI

KATA PENGANTAR	iii
DAFTAR ISI	iv
DAFTAR TABEL	x
DAFTAR GAMBAR	xi
PENDAHULUAN.....	xii
1. Deskripsi.....	xii
2. Prasyarat	xii
3. Petunjuk Penggunaan	xii
4. Tujuan Akhir	xiii
5. Kompetensi Inti dan Kompetensi Dasar.....	xiii
6. Peta konsep.....	xiv
PEMBELAJARAN	1
Bab 1 – Memahami Struktur dan fungsi CPU	1
1.1 Kegiatan belajar 1 – Struktur dan fungsi CPU	1
1.1.1 Tujuan Pembelajaran	1
1.1.2 Aktivitas belajar siswa.....	1
1.1.2.1. Mengamati/ observasi	1
1.1.2.2. Menanya	1
1.1.2.3. Mencoba/ Mengumpulkan informasi	1
1.1.2.4. Mengasosiasi/ menalar	2
1.1.2.5. Mengkomunikasikan.....	3
1.1.3. Rangkuman	3
1.1.4. Tugas	3
1.1.5. Penilaian diri	4
1.1.6. Uji Kompetensi/Ulangan	5
1.2 Kegiatan belajar 2 – Control Unit.....	7
1.2.1. Tujuan Pembelajaran	7
1.2.2. Aktivitas belajar siswa.....	7
1.2.2.1. Mengamati/ observasi	7
1.2.2.2. Menanya	8
1.2.2.3. Mencoba/ Mengumpulkan informasi	8
1.2.2.4. Mengasosiasi/ menalar	10
1.2.2.5. Mengkomunikasikan.....	10
1.2.3. Rangkuman	10
1.2.4. Tugas	11

1.2.5. Penilaian diri	11
1.2.6. Uji Kompetensi/Ulangan	12
1.3 Kegiatan belajar 3 – Register	14
1.3.1. Tujuan Pembelajaran	14
1.3.2. Aktivitas belajar siswa	14
1.3.2.1. Mengamati/ observasi	14
1.3.2.2. Menanya	14
1.3.2.3. Mencoba/ Mengumpulkan informasi	14
1.3.2.4. Mengasosiasi/ menalar	15
1.3.2.5. Mengkomunikasikan	16
1.3.1. Rangkuman	16
1.3.4. Tugas	16
1.3.5. Penilaian diri	17
1.3.6. Uji Kompetensi/Ulangan	18
Bab 2 – Memahami Karakteristik set Instruksi	19
2.1 Kegiatan belajar 1 – Elemen elemen instruksi	19
2.1.1. Tujuan Pembelajaran	19
2.1.2. Aktivitas belajar siswa	19
2.1.2.1. Mengamati/ observasi	19
2.1.2.2. Menanya	19
2.1.2.3. Mencoba/ Mengumpulkan informasi	20
2.1.2.4. Mengasosiasi/ menalar	21
2.1.2.5. Mengkomunikasikan	21
2.1.3. Rangkuman	21
2.1.4. Tugas	21
2.1.5. Penilaian diri	22
2.1.6. Uji Kompetensi/Ulangan	23
2.2 Kegiatan belajar 2 – Tipe – tipe Instruksi	25
2.2.1. Tujuan Pembelajaran	25
2.2.2. Aktivitas belajar siswa	25
2.2.2.1. Mengamati/ observasi	25
2.2.2.2. Menanya	25
2.2.2.3 Mencoba/ Mengumpulkan informasi	25
2.2.2.4. Mengasosiasi/ menalar	27
2.2.2.5. Mengkomunikasikan	28
2.2.3. Rangkuman	28

2.2.4. Tugas	29
2.2.5. Penilaian diri	29
2.2.6. Uji Kompetensi/Ulangan	30
2.3 Kegiatan belajar 3 – Tipe – tipe operand	32
2.3.1. Tujuan Pembelajaran	32
2.3.2. Aktivitas belajar siswa	32
2.3.2.1. Mengamati/ observasi	32
2.3.2.2. Menanya	32
2.3.2.3. Mencoba/ Mengumpulkan informasi	32
2.3.2.4. Mengasosiasi/ menalar	34
2.3.2.5. Mengkomunikasikan	34
2.3.3. Rangkuman	34
2.3.4. Tugas	35
2.3.5. Penilaian diri	36
2.3.6. Uji Kompetensi/Ulangan	36
2.4 Kegiatan belajar 4 – Tipe – tipe operasi data	39
2.4.1. Tujuan Pembelajaran	39
2.4.2. Aktivitas belajar siswa	39
2.4.2.1. Mengamati/ observasi	39
2.4.2.2. Menanya	39
2.4.2.3. Mencoba/ Mengumpulkan informasi	39
2.4.2.4. Mengasosiasi/ menalar	41
2.4.2.5. Mengkomunikasikan	42
2.4.3. Rangkuman	42
2.4.4. Tugas	42
2.4.5. Penilaian diri	43
2.4.6. Uji Kompetensi/Ulangan	44
2.5 Kegiatan belajar 5 – Instruksi percabangan	46
2.5.1. Tujuan Pembelajaran	46
2.5.2. Aktivitas belajar siswa	46
2.5.2.1. Mengamati/ observasi	46
2.5.2.2. Menanya	47
2.5.2.3. Mencoba/ Mengumpulkan informasi	47
2.5.2.4. Mengasosiasi/ menalar	50
2.5.2.5. Mengkomunikasikan	51
2.5.3. Rangkuman	51

2.5.4. Tugas	51
2.5.5. Penilaian diri	52
2.5.6. Uji Kompetensi/Ulangan	53
Bab 3 – Mode dan format pengalaman.....	54
3.1 Kegiatan belajar 1 – Inherent.....	54
3.1.1. Tujuan Pembelajaran	54
3.1.2. Aktivitas belajar siswa.....	54
3.1.2.1. Mengamati/ observasi	54
3.1.2.2. Menanya	55
3.1.2.3. Mencoba/ Mengumpulkan informasi	55
3.1.2.4. Mengasosiasi/ menalar	56
3.1.2.5. Mengkomunikasikan	56
3.1.3. Rangkuman	56
3.1.4. Tugas	56
3.1.5. Penilaian diri	57
3.1.6. Uji Kompetensi/Ulangan	58
3.2 Kegiatan belajar 2 – <i>Immediate</i>	59
3.2.1. Tujuan Pembelajaran	59
3.2.2. Aktivitas belajar siswa.....	59
3.2.2.1. Mengamati/ observasi	59
3.2.2.2. Menanya	59
3.2.2.3. Mencoba/ Mengumpulkan informasi	59
3.2.2.4. Mengasosiasi/ menalar	60
3.2.2.5. Mengkomunikasikan	60
3.2.3. Rangkuman	60
3.2.4. Tugas	61
3.2.5. Penilaian diri	61
3.2.6. Uji Kompetensi/Ulangan	62
3.3 Kegiatan belajar 3 – <i>Direction</i>	64
3.3.1. Tujuan Pembelajaran	64
3.3.2. Aktivitas belajar siswa.....	64
3.3.2.1. Mengamati/ observasi	64
3.3.2.2. Menanya	65
3.3.2.3. Mencoba/ Mengumpulkan informasi	65
3.3.2.4. Mengasosiasi/ menalar	66
3.3.2.5. Mengkomunikasikan	66

3.3.3. Rangkuman	66
3.3.4 Tugas	67
3.3.5. Penilaian diri	68
3.3.6. Uji Kompetensi/Ulangan	68
3.4 Kegiatan belajar 4 – Extended	70
3.4.1. Tujuan Pembelajaran	70
3.4.2. Aktivitas belajar siswa	70
3.4.2.1. Mengamati/ observasi	70
3.4.2.2. Menanya	70
3.4.2.3. Mencoba/ Mengumpulkan informasi	70
3.4.2.4. Mengasosiasi/ menalar	71
3.4.2.5. Mengkomunikasikan	72
3.4.3. Rangkuman	72
3.4.4. Tugas	72
3.4.5. Penilaian diri	73
3.4.6. Uji Kompetensi/Ulangan	73
3.5 Kegiatan belajar 5 – Indexed	75
3.5.1. Tujuan Pembelajaran	75
3.5.2. Aktivitas belajar siswa	75
3.5.2.1. Mengamati/ observasi	75
3.5.2.2. Menanya	75
3.5.2.3. Mencoba/ Mengumpulkan informasi	75
3.5.2.4. Mengasosiasi/ menalar	77
3.5.2.5. Mengkomunikasikan	78
3.5.3. Rangkuman	78
3.5.4. Tugas	79
3.5.5. Penilaian diri	80
3.5.6. Uji Kompetensi/Ulangan	80
3.6 Kegiatan belajar 6 – Relative	82
3.6.1. Tujuan Pembelajaran	82
3.6.2. Aktivitas belajar siswa	82
3.6.2.1. Mengamati/ observasi	82
3.6.2.2. Menanya	82
3.6.2.3. Mencoba/ Mengumpulkan informasi	82
3.6.2.4. Mengasosiasi/ menalar	84
3.6.2.5. Mengkomunikasikan	84

3.6.3. Rangkuman	84
3.6.4. Tugas	84
3.6.5. Penilaian diri	85
3.6.6. Uji Kompetensi/Ulangan	86
PENUTUP	88
RANGKUMAN.....	88
DAFTAR PUSTAKA.....	96

DAFTAR TABEL

Tabel 3.1.Instruksi-instruksi Mode Pengalamatan INHERENT	55
Tabel 3.3.Instruksi-instruksi Mode Pengalamatan DIRECT	65
Tabel 3.4.Instruksi-instruksi Mode Pengalamatan EXTENDED	71
Tabel 3.5.INDEXED-NO OFFSET dan INDEXED-8 BIT OFFSET	76
Tabel 3.6.Instruksi-instruksi Mode Pengalamatan INDEXED-16 BIT OFFSET	77
Instruksi-instruksi Mode Pengalamatan RELATIVE	83

DAFTAR GAMBAR

Gambar 1.1 Komponen CPU dan Struktur Internal CPU (Sumber : Stalling. 2013)	2
Gambar 1.2.CU 1	9
Gambar 1.3.CU 2	9
Gambar 1.4.CU 3	10
Gambar 2.1. Alur logika Instruksi IF	48
Gambar 2.2. Logika instruksi if else	49
Gambar 2.3. Konfigurasi nested if	50

PENDAHULUAN

1. Deskripsi

Sistem komputer merupakan suatu jaringan elektronik yang terdiri dari perangkat lunak (*software*) dan perangkat keras (*hardware*) yang melakukan tugas tertentu (menerima input, memproses input, menyimpan perintah-perintah, dan menyediakan output dalam bentuk informasi). Selain itu sistem komputer dapat pula diartikan sebagai elemen-elemen yang terkait untuk menjalankan suatu aktivitas dengan menggunakan komputer.

Komputer dapat membantu manusia dalam pekerjaan sehari-harinya, pekerjaan itu seperti: pengolahan kata, pengolahan angka, dan pengolahan gambar.

Elemen dari sistem komputer terdiri dari manusianya (*brainware*), perangkat lunak (*software*), set instruksi (*instruction set*), dan perangkat keras (*hardware*). Dengan demikian komponen tersebut merupakan elemen yang terlibat dalam suatu sistem komputer. Tentu saja *hardware* tidak berarti apa-apa jika tidak ada salah satu dari dua lainnya (*software* dan *brainware*).

2. Prasyarat

Pada mata pelajaran sistem komputer kelas XI ini mempunyai keterkaitan dengan mata pelajaran sistem komputer kelas X dan sistem komputer kelas XI semester satu. Sistem komputer pada kelas XI semester dua ini merupakan lanjutan dari sistem komputer kelas X semester satu.

Untuk dapat mengoperasikan, menggunakan komputer, mengetahui lebih lanjut tentang perangkat lunak (*software*), set instruksi (*instruction set*), dan perangkat keras (*hardware*), maka diperlukan satu set sistem komputer yang berfungsi dengan baik dan dapat memahami sistem komputer dasar. Tahapan untuk menyiapkan bagaimana seperangkat sistem komputer dapat berjalan dengan baik, dan sistem komputer materi dasar telah diuraikan dalam mata pelajaran perakitan komputer dan sistem komputer yang telah dipelajari pada kelas X semester I dan sistem komputer kelas XI semester satu.

3. Petunjuk Penggunaan

Buku siswa ini disusun berdasarkan kurikulum 2013, dimana kurikulum 2013 mempunyai ciri khusus yaitu penggunaan metode ilmiah. Buku siswa ini terdiri dari dua bab yaitu bab 1 pendahuluan dan bab 2 materi pembelajaran.

Dalam bab pendahuluan ada beberapa hal yang harus dipelajari peserta didik antara lain adalah deskripsi mata pelajaran yang berisi informasi umum, prasyarat sebelum mempelajari sistem komputer dan tujuan umum.

Selanjutnya pada Bab 2 menuntun peserta didik untuk memahami deskripsi umum tentang topik yang akan dipelajari dan rincian kegiatan belajar sesuai dengan kompetensi dan tujuan yang akan dicapai. Setiap kegiatan belajar terdiri dari tujuan dan uraian materi topik pembelajaran, tugas serta test formatif. Uraian pembelajaran berisi tentang deskripsi pemahaman topik materi untuk memenuhi kompetensi pengetahuan. Uraian pembelajaran juga menjelaskan deskripsi unjuk kerja atau langkah-langkah logis untuk memenuhi kompetensi skill.

Kemudian tugas dikerjakan oleh peserta didik dapat berupa tugas pengamatan, atau praktek. Setiap tugas yang dilakukan melalui beberapa tahapan ilmiah yaitu : 1) melakukan pengamatan setiap tahapan unjuk kerja 2) melakukan praktek sesuai dengan unjuk kerja 3) mengumpulkan data yang dihasilkan setiap tahapan 4) menganalisa hasil data menggunakan analisa deskriptif 5) mengasosiasikan beberapa pengetahuan dalam uraian materi pembelajaran untuk membentuk suatu kesimpulan 5) mengkomunikasikan hasil dengan membuat laporan portofolio. Laporan tersebut merupakan tagihan yang akan dijadikan sebagai salah satu referensi penilaian.

4. Tujuan Akhir

Setelah mempelajari materi dalam bab pembelajaran dan kegiatan belajar diharapkan peserta didik dapat memiliki kompetensi sikap, pengetahuan dan ketrampilan yang berkaitan dengan materi:

1. Memahami sistem input proses-output
2. Memahami Perangkat Eksternal/Peripheral
3. Memahami struktur dan Interkoneksi Bus
4. Memahami dan menganalisis kaidah dan simbol-simbol yang digunakan pada flowchart atau struktogram
5. Memahami organisasi prosesor, register, dan siklus instruksi (fetching, decoding, executing)
6. Memahami Struktur CPU dan fungsi CPU
7. Memahami karakteristik Set Instruksi (operand dan operasi)
8. Memahami mode dan format pengalamatan

5. Kompetensi Inti dan Kompetensi Dasar

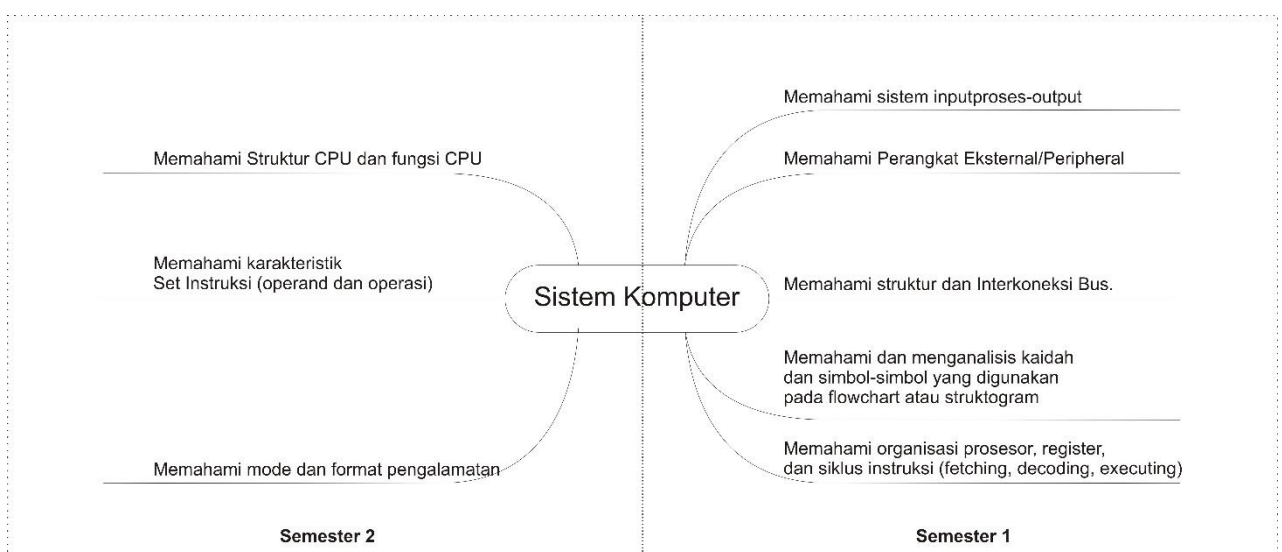
1. KI 1 : Menghayati dan mengamalkan ajaran agama yang dianutnya
 - 1.1. Memahami nilai-nilai keimanan dengan menyadari hubungan keteraturan dan kompleksitas alam dan jagad raya terhadap kebesaran Tuhan yang menciptakannya
 - 1.2. Mendeskripsikan kebesaran Tuhan yang menciptakan berbagai sumber energi di alam
 - 1.3. Mengamalkan nilai-nilai keimanan sesuai dengan ajaran agama dalam kehidupan sehari-hari
2. KI 2 : Menghayati dan mengamalkan perilaku jujur, disiplin, tanggung jawab, peduli (gotong royong, kerjasama, toleran, damai), santun, responsif dan proaktif dan menunjukkan sikap sebagai bagian dari solusi atas berbagai permasalahan dalam berinteraksi secara efektif dengan lingkungan sosial dan alam serta dalam menempatkan diri sebagai cerminan bangsa dalam pergaulan dunia.
 - 2.1. Menunjukkan perilaku ilmiah (memiliki rasa ingin tahu; objektif; jujur; teliti; cermat; tekun; hati-hati; bertanggung jawab; terbuka; kritis; kreatif; inovatif dan peduli lingkungan) dalam aktivitas sehari-hari sebagai wujud implementasi sikap dalam melakukan percobaan dan berdiskusi.
 - 2.2. Menghargai kerja individu dan kelompok dalam aktivitas sehari-hari sebagai wujud implementasi melaksanakan percobaan dan melaporkan hasil percobaan
3. KI 3 : Memahami, menerapkan dan menganalisis pengetahuan faktual, konseptual, prosedural, dan metakognitif berdasarkan rasa ingin tahunya tentang ilmu pengetahuan, teknologi, seni, budaya, dan humaniora dalam wawasan

kemanusiaan, kebangsaan, kenegaraan, dan peradaban,terkait penyebab fenomena dan kejadian dalam bidang kerja yang spesifik untuk memecahkan masalah.

- 3.1. Memahami sistem inputproses-output
- 3.2. Memahami Perangkat Eksternal/Peripheral
- 3.3. Memahami struktur dan Interkoneksi Bus.
- 3.4. Memahami dan menganalisis kaidah dan simbol-simbol yang digunakan pada flowchart atau struktogram
- 3.5. Memahami organisasi prosesor, register, dan siklus instruksi (fetching, decoding, executing)
- 3.6. Memahami Struktur CPU dan fungsi CPU
- 3.7. Memahami karakteristik Set Instruksi (operand dan operasi)
- 3.8. Memahami mode dan format pengalamatan
4. KI 4 : Mengolah, menalar dan menyaji dalam ranah konkret dan ranah abstrak terkait dengan pengembangan dari yang dipelajarinya di sekolah secara mandiri, bertindak secara efektif dan kreatif, serta mampu melaksanakan tugas spesifik di bawah pengawasan langsung.
 - 4.1. Merencanakan dan membuat perangkat input – output dengan menggunakan modul I/O terprogram
 - 4.2. Merangkai perangkat eksternal dengan console unit
 - 4.3. Menggunakan struktur dan interkoneksi bus untuk membuat jaringan
 - 4.4. Menggunakan kaidah dan simbol-simbol yang digunakan pada flowchart atau struktogram untuk memecahkan masalah
 - 4.5. Menggunakan organisasi prosesor, register dan siklus instruksi (fetching, decoding, executing) untuk memecahkan masalah
 - 4.6. Membuat rangkaian internal CPU
 - 4.7. Menggunakan set instruksi (operand dan operasi) untuk memecahkan masalah
 - 4.8. Menggunakan mode dan format pengalamatan untuk memecahkan masalah

6. Peta konsep

Peta konsep pada mata pelajaran C2 sistem komputer dapat digambarkan sebagai berikut :



Gambar Peta Konsep Sistem Komputer

PEMBELAJARAN

Bab 1 – Memahami Struktur dan fungsi CPU

1.1 Kegiatan belajar 1 – Struktur dan fungsi CPU

1.1.1 Tujuan Pembelajaran

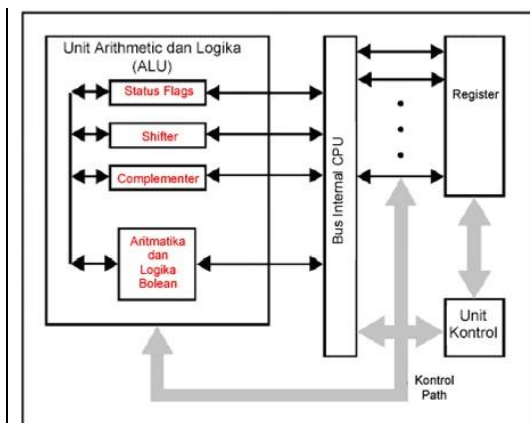
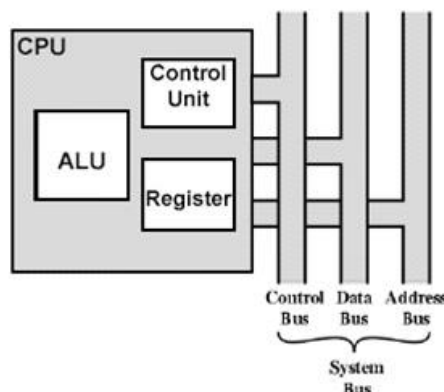
Setelah mengikuti pembelajaran, siswa mampu:

- Memahami struktur CPU dan
- Memahami Fungsi CPU (ALU)

1.1.2 Aktivitas belajar siswa

1.1.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah gambar berikut ini:



1.1.2.2. Menanya

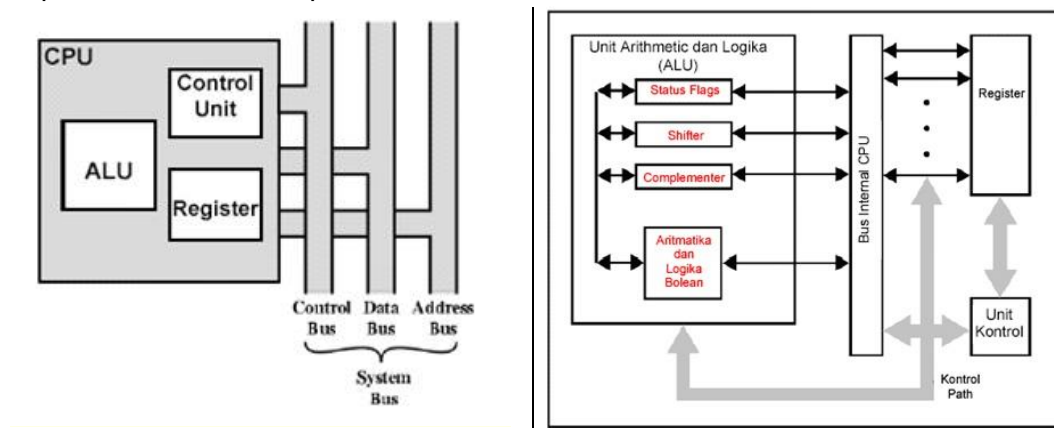
Buatlah pertanyaan kepada Gurumu tentang

1. Struktur CPU dan
2. Fungsi CPU.

1.1.2.3. Mencoba/ Mengumpulkan informasi

CPU (prosesor) merupakan komponen terpenting dari sistem komputer. CPU adalah komponen pengolah data berdasarkan instruksi – instruksi yang diberikan kepadanya. Dalam mewujudkan fungsi dan tugasnya, CPU tersusun atas beberapa komponen sebagai bagian dari struktur CPU, seperti terlihat pada gambar komponen CPU dan struktur internal CPU. CPU tersusun atas beberapa komponen, yaitu :

1. Arithmetic and Logic Unit (ALU), bertugas membentuk fungsi – fungsi pengolahan data komputer. ALU sering disebut mesin bahasa (machine language) karena bagian ini mengerjakan instruksi – instruksi bahasa mesin yang diberikan padanya. Seperti istilahnya, ALU terdiri dari dua bagian, yaitu unit aritmetika dan unit logika boolean, yang masing – masing memiliki spesifikasi tugas tersendiri.
2. Control Unit, bertugas mengontrol operasi CPU dan secara keseluruhan mengontrol computer sehingga terjadi sinkronisasi kerja antar komponen dalam menjalankan fungsi – fungsi operasinya. Termasuk dalam tanggung jawab unit kontrol adalah mengambil instruksi – instruksi dari memori utama dan menentukan jenis instruksi tersebut.
3. Registers, adalah media penyimpanan internal CPU yang digunakan saat proses pengolahan data. Memori ini bersifat sementara, biasanya digunakan untuk menyimpan data saat diolah ataupun data untuk pengolahan selanjutnya.
4. CPU Interconnections, adalah sistem koneksi dan bus yang menghubungkan komponen internal CPU, yaitu ALU, unit kontrol dan register – register dan juga dengan bus – bus eksternal CPU yang menghubungkan dengan sistem lainnya, seperti memori utama, piranti masukan/keluaran.



Gambar 1.1 Komponen CPU dan Struktur Internal CPU (Sumber : Stalling. 2013)

Hal-hal yang dilakukan CPU adalah sebagai berikut :

1. Fetch instruction (mengambil instruksi) yaitu CPU membaca instruksi dari memori
2. Interpret instruction (menterjemahkan instruksi) yaitu CPU menerjemahkan instruksi untuk menentukan langkah yang diperlukan.
3. Fetch data (mengambil data) yaitu eksekusi instruksi yang mungkin memerlukan pembacaan data dari memori data atau dari modul I/O
4. Process data (mengolah data) yaitu eksekusi instruksi yang memerlukan operasi aritmatik
5. Write data (menulis data) yaitu hasil eksekusi yang mungkin memerlukan penulisan data ke memori data atau ke modul I/O.

1.1.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang struktur dan fungsi CPU pada tabel berikut ini

No	Komponen CPU	Fungsi

Buatlah kesimpulan tentang :

1. Struktur CPU dan
2. Fungsi CPU.

1.1.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri mengenai :

1. Struktur CPU dan
2. Fungsi CPU.

1.1.3. Rangkuman

CPU adalah komponen pengolah data berdasarkan instruksi – instruksi yang diberikan kepadanya, yang tersusun atas beberapa komponen, yaitu:

1. Arithmetic and Logic Unit (ALU) atau language machine, bertugas membentuk fungsi – fungsi pengolahan data computer, yang terdiri dari dua bagian, yaitu unit aritmetika dan unit logika Boolean.
2. Control Unit, bertugas mengontrol operasi CPU dan secara keseluruhan mengontrol computer, termasuk mengambil instruksi – instruksi dari memori utama dan menentukan jenis instruksi tersebut.
3. Registers, adalah media penyimpanan internal CPU yang digunakan saat proses pengolahan data, yang bersifat sementara.
4. CPU Interconnections, adalah sistem koneksi dan bus yang menghubungkan komponen internal CPU, yaitu ALU, unit kontrol dan register – register dan juga dengan bus – bus eksternal CPU.

Hal-hal yang dilakukan CPU adalah sebagai berikut :

1. Fetch instruction (mengambil instruksi).
2. Interpret instruction (menterjemahkan instruksi).
3. Fetch data (mengambil data).
4. Process data (mengolah data).
5. Write data (menulis data).

1.1.4. Tugas

1. Apa yang kamu ketahui CPU Interconnections?

.....

-
.....
.....
2. CPU tersusun atas beberapa komponen apa saja? Jelaskan!

-
.....
.....
.....
.....
3. Tugas dari Control Unit adalah?

-
.....
.....
.....
.....
4. Tugas dari Arithmetic and Logic Unit (ALU) adalah?

-
.....
.....
.....
.....
5. Apa saja yang dilakukan CPU?

1.1.5. Penilaian diri

Nama :
Nama-nama anggota kelompok :
Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.

- 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....
- Mendengarkan orang lain
 - Mengajukan pertanyaan
 - Mengorganisasi ide-ide saya
 - Mengorganisasi kelompok
 - Mengacaukan kegiatan
 - Melamun
6. Apa yang kamu lakukan selama kegiatan?
-
-
-
-
-
-
-

1.1.6. Uji Kompetensi/Ulangan

1. Control Unit bertugas untuk...
 - a. Mengontrol computer sehingga terjadi sinkronisasi kerja antar komponen dalam menjalankan fungsi – fungsi operasinya
 - b. Membentuk fungsi – fungsi pengolahan data computer
 - c. Menyimpan data saat diolah ataupun data untuk pengolahan selanjutnya
 - d. Mengontrol computer sehingga terjadi sinkronisasi kerja antar komponen dalam menyimpan data
 - e. Menghubungkan komponen internal CPU
2. ALU terdiri dari dua bagian, yaitu...
 - a. Unit aritmatika dan unit logaritma Boolean
 - b. Unit logaritma dan unit logika Boolean
 - c. Unit fisika dan unit aritmatika
 - d. Unit jalur dan unit data
 - e. Unit arithmetika dan unit logika Boolean
3. Membentuk fungsi – fungsi pengolahan data komputer merupakan tugas dari...
 - a. Microsoft
 - b. ALU
 - c. CU
 - d. Bus
 - e. Peripheral
4. Yang dimaksud dengan Fetch data (mengambil data) adalah...
 - a. CPU membaca instruksi dari memori
 - b. Eksekusi instruksi yang mungkin memerlukan pembacaan data dari memori datau dari modul I/O
 - c. Hasil eksekusi yang mungkin memerlukan penulisan data ke memori datau ke modul I/O
 - d. CPU menerjemahkan instruksi untuk menentukan langkah yang diperlukan

- e. Eksekusi instruksi yang memerlukan operasi aritmatik
- 5. Di bawah ini merupakan beberapa komponen di dalam CPU, kecuali...
 - a. Central Unit
 - b. ALU
 - c. CU
 - d. Register
 - e. CPU Interconnections

1.2 Kegiatan belajar 2 – Control Unit

1.2.1. Tujuan Pembelajaran

Setelah mengikuti pembelajaran, siswa mampu :

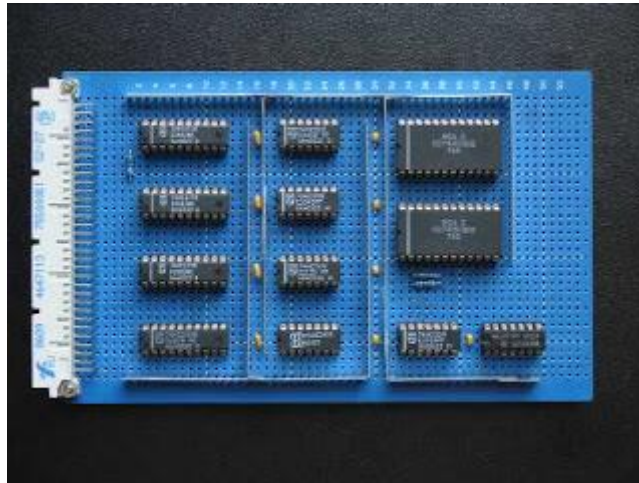
- Menjelaskan Control Unit
- Menjelaskan Fungsinya

1.2.2. Aktivitas belajar siswa

1.2.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah gambar dibawah ini :





Sumber : <http://kj-jogja.blogspot.com/2013/09/apa-itu-cu-control-unit.html>

1.2.2.2. Menanya

Bertanyalah pada gurumu apabila ada materi yang belum kamu pahami tentang Control Unit.

1. Control Unit
2. Fungsi Control Unit

1.2.2.3. Mencoba/ Mengumpulkan informasi

Tugas dari CU adalah sebagai berikut:

1. Mengatur dan mengendalikan alat-alat input dan output.
2. Mengambil instruksi-instruksi dari memori utama.
3. Mengambil data dari memori utama kalau diperlukan oleh proses.
4. Mengirim instruksi ke ALU bila ada perhitungan aritmatika atau perbandingan logika serta mengawasi kerja.
5. Menyimpan hasil proses ke memori utama.

Proses tiga langkah karakteristik unit control:

1. Menentukan elemen dasar prosesor
2. Menjelaskan operasi mikro yang akan dilakukan prosesor
3. Menentukan fungsi-fungsi yang harus dilakukan unit control agar menyebabkan pembentukan operasi mikro

Masukan-masukan unit control:

1. Clock / pewaktu
Pewaktu adalah cara unit control dalam menjaga waktunya. Unit control menyebabkan sebuah operasi mikro (atau sejumlah operasi mikro yang bersamaan) dibentuk bagi setiap pulsa waktu. Pulsa ini dikenal sebagai waktu siklus prosesor.
2. Register instruksi
Opcode instruksi saat itu digunakan untuk menentukan operasi mikro mana yang akan dilakukan selama siklus eksekusi.
3. Flag

Flag ini diperlukan oleh unit control untuk menentukan status prosesor dan hasil operasi ALU sebelumnya.

4. Sinyal control untuk mengontrol bus

Bagian bus control bus system memberikan sinyal-sinyal ke unit control, seperti sinyal-sinyal interupsi dan acknowledgement.

Keluaran-keluaran unit control:

Sinyal control didalam prosesor terdiri dari dua macam:

1. sinyal-sinyal yang menyebabkan data dipindahkan dari register yang satu ke register yang lainnya,
2. sinyal-sinyal yang dapat mengaktifasi fungsi-fungsi ALU tertentu.



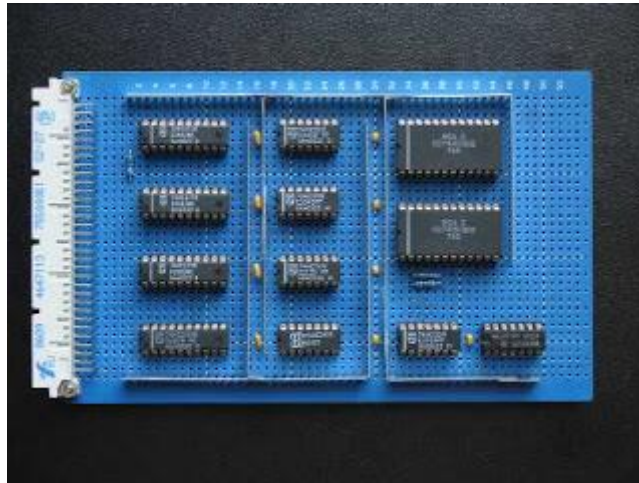
Sumber : <http://kj-jogja.blogspot.com/2013/09/apa-itu-cu-control-unit.html>

Gambar 1.2.CU 1



Sumber : <http://kj-jogja.blogspot.com/2013/09/apa-itu-cu-control-unit.html>

Gambar 1.3.CU 2



Sumber : <http://kj-jogja.blogspot.com/2013/09/apa-itu-cu-control-unit.html>

Gambar 1.4.CU 3

1.2.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang control unit pada tabel berikut ini:

Tugas CU	
Fungsi CU	

Buatlah kesimpulan tentang

1. Control Unit
2. Fungsi Control Unit

1.2.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri mengenai

1. Control Unit
2. Menjelaskan Control Unit

1.2.3. Rangkuman

Tugas dari CU : Mengatur dan mengendalikan alat-alat input dan output, Mengambil instruksi-instruksi dari memori utama, Mengambil data dari memori utama kalau diperlukan oleh proses, Mengirim instruksi ke ALU bila ada perhitungan aritmatika atau perbandinganlogika serta mengawasi kerja serta Menyimpan hasil proses ke memori utama.

Masukan-masukan unit control: Clock / pewaktu, Register instruksi, Flag, Sinyal control untuk mengontrol bus

1.2.4. Tugas

1. Tugas dari CU adalah!

.....

.....

.....

.....

.....

2. Karakteristik unit control memiliki langkah apa saja? Sebutkan!

.....

.....

.....

.....

.....

3. Pewaktu dalam masukan unit control adalah?

.....

.....

.....

.....

.....

4. Sinyal control didalam prosesor terdiri dari dua macam, sebutkan!

.....

.....

.....

.....

.....

5. Sebutkan macam macam Keluaran-keluaran unit control!

.....

.....

.....

.....

.....

1.2.5. Penilaian diri

Nama :

Nama-nama anggota kelompok :

Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.

- 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
 3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
 4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
 5. Selama kerja kelompok, saya....
 - Mendengarkan orang lain
 - Mengajukan pertanyaan
 - Mengorganisasi ide-ide saya
 - Mengorganisasi kelompok
 - Mengacaukan kegiatan
 - Melamun
 6. Apa yang kamu lakukan selama kegiatan?

.....

.....

.....

.....

.....

.....

1.2.6. Uji Kompetensi/Ulangan

2. Yang bukan merupakan tugas unit control adalah...
 - a. Mengatur dan mengendalikan alat-alat input dan output
 - b. Mengambil instruksi-instruksi dari ALU
 - c. Mengambil data dari memori utama kalau diperlukan oleh proses
 - d. Mengirim instruksi ke ALU bila ada perhitungan aritmatika atau perbandingan logika serta mengawasi kerja
 - e. Mengambil instruksi-instruksi dari memori utama
3. Di bawah ini merupakan masukan-masukan unit control, kecuali...
 - a. Sinyal control untuk mengontrol bus
 - b. Sinyal control untuk mengontrol ALU
 - c. Flag
 - d. Clock
 - e. Register instruksi
4. Menentukan operasi mikro mana yang akan dilakukan selama siklus eksekusi, merupakan fungsi dari...
 - a. Sinyal control untuk mengontrol bus
 - b. Sinyal control untuk mengontrol ALU
 - c. Flag
 - d. Clock
 - e. Register instruksi
5. Pulsa pada clock dikenal sebagai...
 - a. Daya siklus prosesor
 - b. Alir siklus prosesor

- c. Waktu siklus prosesor
 - d. Sinyal siklus prosesor
 - e. Masukan siklus prosesor
6. Langkah terakhir karakteristik unit control, adalah...
- a. Menjelaskan operasi mikro yang akan dilakukan prosesor
 - b. Menentukan elemen dasar prosesor
 - c. Menentukan fungsi-fungsi yang harus dilakukan unit control agar menyebabkan pembentukan operasi makro
 - d. Menentukan fungsi-fungsi yang harus dilakukan unit control agar menyebabkan pembentukan operasi mikro
 - e. Menentukan elemen khusus prosesor

1.3 Kegiatan belajar 3 – Register

1.3.1. Tujuan Pembelajaran

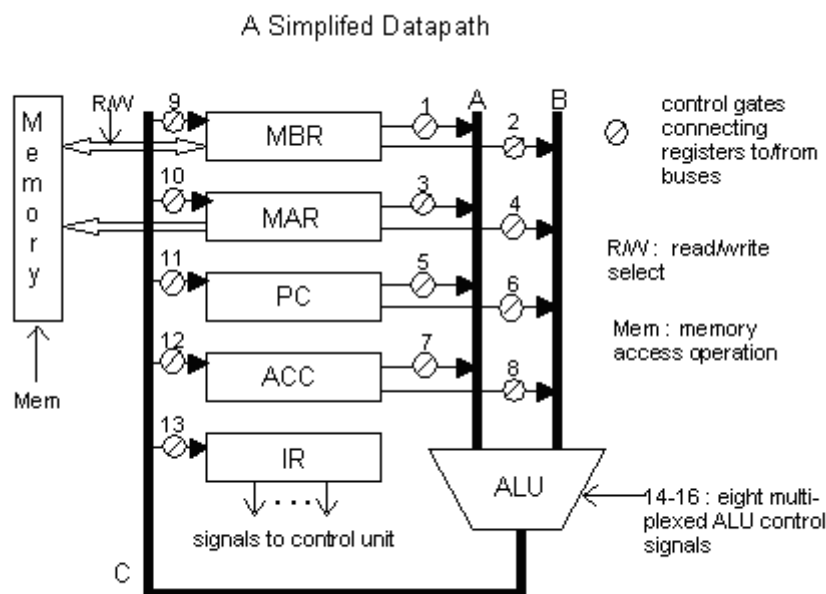
Setelah mengikuti pembelajaran, siswa mampu :

- Memahami register
- Memahami fungsi Register

1.3.2. Aktivitas belajar siswa

1.3.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah gambar berikut ini



Sumber : <http://www.c-jump.com/>

1.3.2.2. Menanya

Buatlah pertanyaan kepada gurumu tentang :

1. Register
2. Fungsi Register

1.3.2.3. Mencoba/ Mengumpulkan informasi

Sistem Komputer menggunakan hirarki memori pada tingkatan yang atas, memori lebih cepat, lebih kecil, lebih mahal. Di dalam CPU, terdapat sekumpulan register yang tingkatan memorinya berada di atas hirarki memori utama dan cache. Register dalam CPU memiliki dua fungsi:

1. User-visible Register

User-visible Register adalah register yang dapat direferensikan dengan menggunakan bahasa mesin yang dieksekusi CPU, User-visible terdiri dari :

- a. General Purpose Register
 - Digunakan untuk mode pengalamatan dan data.
 - Akumulator (aritmatika, *Shift*, Rotate)
 - *Base Register* (*Rotate*, *Shift*, aritmatika)
 - *Counter Register* (Looping)
 - *Data Register* (menyimpan alamat I/O device).
 - b. Register Alamat
 - Digunakan untuk mode pengalamatan
 - Segment Register (Code Segment, Data Segment, Stack Segment, Extra Segment)
 - Register Index / Stack Index, Data Index (untuk menyimpan alamat-alamat yang terindeks)
 - Stack Pointer (register yang dedicated menunjuk kebagian teratas stack)
 - c. Register Data
 - Digunakan untuk menampung data
 - d. Register Kode Status Kondisi (Flag)
 - Kode yang menggambarkan hasil operasi sebelumnya
2. Control and Register
- Control and register adalah register-register yang digunakan oleh unit kontrol untuk mengontrol operasi CPU dan oleh program sistem operasi untuk mengontrol eksekusi program. Terdapat empat register yang penting adalah :
- a. *Program Counter* (PC)
 - Berisi alamat instruksi yang akan diambil
 - b. *Instruction Register* (IR)
 - Berisi alamat instruksi terakhir
 - c. *Memory Address Register* (MAR)
 - Berisi alamat penyimpanan dalam memori.
 - d. *Memori Buffer Register* (MBR)
 - Berisi data yang dibaca dari memori atau yang diyliskan ke memori
 - e. *Memori Data Register* (MDR)
 - Merupakan register yang digunakan untuk menampung data atau instruksi hasil pengiriman dari memori utama ke CPU atau menampung data yang akan direkam ke momori utama dari hasil pengolahan CPU.

Fungsi CPU

1. Menjalankan program – program yang disimpan dalam memori utama dengan cara mengambil instruksi – instruksi, menguji instruksi tersebut dan mengeksekusinya satu persatu sesuai alur perintah.
2. Sehingga langkah-langkah pada operasi CPU, yaitu operasi :
 - Pembacaan instruksi (fetch) dan
 - Pelaksanaan instruksi (execute)

1.3.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang register pada tabel berikut ini

No	Macam Register	Penjelasan

Buatlah kesimpulan tentang

1. Register
2. Fungsi Register

1.3.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri tentang :

1. Register
2. Fungsi Register

1.3.1. Rangkuman

Sistem Komputer menggunakan hirarki memori pada tingkatan yang atas, memori lebih cepat, lebih kecil, lebih mahal. Di dalam CPU, terdapat sekumpulan register yang tingkatan memorinya berada di atas hirarki memori utama dan cache.

Register dalam CPU memiliki dua fungsi:

1. User-visible Register, yaitu register yang dapat direferensikan dengan menggunakan bahasa mesin yang dieksekusi CPU, User-visible terdiri dari: General Purpose Register; Register Alamat; Register Data; dan Register Kode Status Kondisi (Flag)
2. Control and Register, yaitu register-register yang digunakan oleh unit kontrol untuk mengontrol operasi CPU dan oleh program sistem operasi untuk mengontrol eksekusi program. Terdapat empat register yang penting, yaitu: Program Counter (PC); Instruction Register (IR); Memory Address Register (MAR); Memori Buffer Register (MBR); Memori Data Register (MDR)

Fungsi CPU, yaitu Menjalankan program – program yang disimpan dalam memori utama dengan cara mengambil instruksi – instruksi, menguji instruksi tersebut dan mengeksekusinya satu persatu sesuai alur perintah.

1.3.4. Tugas

1. Register dalam CPU memiliki dua fungsi, sebutkan!

.....

.....

.....

.....

.....

.....

2. User-visible Register adalah?

.....

.....

.....

.....

.....

.....

3. Control and register adalah?

.....

.....

.....

.....

.....

4. Terdapat empat register yang penting, sebutkan!

.....

.....

.....

.....

.....

5. Sebutkan Fungsi dari CPU?

.....

.....

.....

.....

.....

1.3.5. Penilaian diri

Nama :

Nama-nama anggota kelompok :

Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....
 - Mendengarkan orang lain
 - Mengajukan pertanyaan
 - Mengorganisasi ide-ide saya
 - Mengorganisasi kelompok
 - Mengacaukan kegiatan
 - Melamun
6. Apa yang kamu lakukan selama kegiatan?

.....
.....
.....
.....
.....
.....

1.3.6. Uji Kompetensi/Ulangan

1. Menerjemahkan instruksi ke dalam perintah computer merupakan siklus...
 - a. Fetch
 - b. Stretch
 - c. Execute
 - d. Decode
 - e. Store
2. Langkah-langkah yang dilakukan oleh prosesor komputer untuk setiap instruksi bahasa mesin yang diterima, disebut...
 - a. Siklus data
 - b. Siklus processor
 - c. Siklus bahasa
 - d. Siklus instruksi
 - e. Siklus mesin
3. Store merupakan proses...
 - a. Mendapatkan instruksi dari Main Memory
 - b. Menerjemahkannya ke dalam perintah computer
 - c. Proses perintah
 - d. Menulis hasilnya ke Memori Utama
 - e. Menampilkan hasil pada layar monitor
4. Yang bukan merupakan proses siklus mesin, yaitu...
 - a. Stretch
 - b. Fetch
 - c. Decode
 - d. Execute
 - e. Store
5. Yang dilakukan CPU sebelum menjemput instruksi dari memori utama melalui bus data ke memori, adalah...
 - a. Menafsirkan dan mengimplementasikan instruksi
 - b. Membentuk instruksi ini diterjemahkan oleh unit control
 - c. CPU menyajikan nilai dari program counter (PC) di bus alamat
 - d. Menulis kembali ke memori
 - e. Melewati informasi dekode sebagai urutan sinyal kontrol ke unit fungsi yang relevan dari CPU

Bab 2 – Memahami Karakteristik set Instruksi

2.1 Kegiatan belajar 1 – Elemen elemen instruksi

2.1.1. Tujuan Pembelajaran

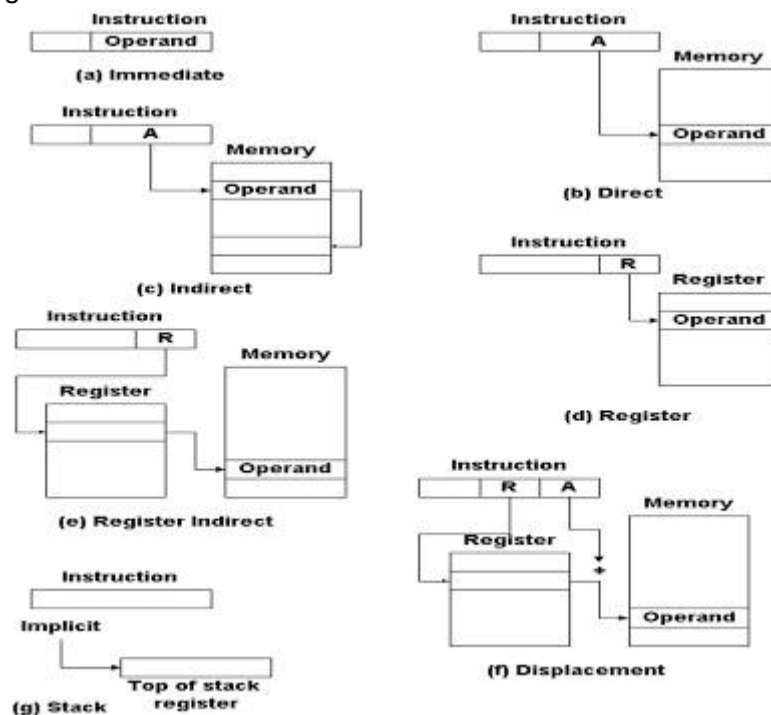
Setelah mengikuti pembelajaran, siswa mampu:

- Memahami elemen set intruksi
- menjelaskan karakteristik set instruksi

2.1.2. Aktivitas belajar siswa

2.1.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah gambar berikut ini :



Sumber : <http://syaifuldeathcore.blogspot.com/>

2.1.2.2. Menanya

Buatlah pertanyaan kepada gurumu tentang :

1. Elemen set intruksi
2. Karakteristik set instruksi

2.1.2.3. Mencoba/ Mengumpulkan informasi

Elemen Instruksi

Agar dapat dieksekusi, setiap instruksi harus berisi informasi yang diperlukan oleh CPU. Informasi itu dituangkan dalam elemen-elemen instruksi:

- Operation Code/Kode Operasi: menspesifikasikan operasi yang akan dilakukan (misalnya, ADD).
- Operasi dispesifikasikan oleh kode biner, yang dikenal sebagai kode operasi, atau opcode.
- Source Operand Reference/Referensi Operand Sumber: operasi dapat mencakup satu atau lebih sumber, operand merupakan input bagi operasi.
- Result Operand Reference/Referensi Operand Hasil: operasi dapat membuat hasil operasi.
- Next Instruction Reference/Referensi Operand Selanjutnya: elemen ini memberitahu CPU posisi instruksi berikutnya yang harus diambil setelah menyelesaikan eksekusi suatu instruksi.

Sumber dan hasil operand dapat berada di salah satu dari ketiga daerah di bawah ini:

- Memori utama atau memori virtual: dengan referensi alamat berikutnya, maka alamat memori utama atau virtual harus diketahui.
- Register CPU: instruksi harus diberi nomor register yang dimaksud.
- Perangkat I/O: instruksi harus menspesifikasikan modul I/O yang diperlukan oleh operasi.

Representasi Instruksi

Dalam komputer, instruksi direpresentasikan oleh sekumpulan bit. Dalam penulisan format instruksi (instruction format) biasanya dibagi dalam beberapa kolom berkaitan dengan elemen-elemen yang akan mengisi instruksi.

Apa Itu Mnemonic? Adalah singkatan-singkatan yang mengindikasikan suatu operasi yang merupakan representasi dari opcode.

Contoh:

Mnemonic

ADD Add (penambahan)

SUB Subtract (pengurangan)

LOAD Muatkan data dari memori

INC Increment (penambahan dengan satu)

Mnemonic Operand

ADD A

Artinya menambah secara langsung 8 bit data ke dalam isi akumulator dan menyimpan hasil di akumulator.

Operand juga dapat direpresentasikan secara simbolik.

Misalnya:

Mnemonic Operand

ADD R,Y

Artinya tambahkan nilai lokasi Y ke isi register R. Program dapat mendahului dengan definisi-definisi: X = 513, Y = 514, dst. Program akan mengkonversikan opcode dan referensi operand menjadi bentuk biner, akhirnya akan membentuk instruksi mesin biner.

2.1.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang set instruksi dan karakteristiknya pada tabel berikut ini :

No	Elemen Instruksi	Fungsi

Buatlah kesimpulan tentang

1. Elemen set intruksi
2. Karakteristik set instruksi

2.1.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri tentang :

1. Elemen set intruksi
2. Karakteristik set instruksi

2.1.3. Rangkuman

Elemen-elemen instruksi yang diperlukan oleh CPU, antara lain; Operation Code/Kode Operasi; Operasi dispesifikasikan oleh kode biner; Source Operand Reference/Referensi Operand Sumber; Result Operand Reference/Reference Operand Hasil; Next Instruction Reference/Reference Operand Selanjutnya.

Sumber dan hasil operand dapat berada di salah satu dari tiga daerah, yaitu: Memori utama atau memori virtual; Register CPU; Perangkat I/O.

Dalam komputer, instruksi direpresentasikan oleh sekumpulan bit.Mnemonic adalah singkatan-singkatan yang mengindikasikan suatu operasi yang merupakan representasi dari opcode. Contoh: ADD; SUB; LOAD; INC.

2.1.4. Tugas

1. Sebutkan Informasi yang dapat dituangkan dalam elemen-elemen instruksi?

.....

.....

.....

.....

.....

.....

2. Sumber dan hasil operand dapat berada pada area apasaja?

.....
.....
.....
.....
.....
.....
.....
3. Representasi Instruksi adalah?

.....
.....
.....
.....
.....
.....
4. Apa Itu Mnemonic?

.....
.....
.....
.....
.....
.....
5. Singkatan-singkatan yang mengindikasikan suatu operasi yang merupakan representasi dari opcode disebut?

2.1.5. Penilaian diri

Nama :

Nama-nama anggota kelompok :

Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....

- Mendengarkan orang lain
 - Mengajukan pertanyaan
 - Mengorganisasi ide-ide saya
 - Mengorganisasi kelompok
 - Mengacaukan kegiatan
 - Melamun
6. Apa yang kamu lakukan selama kegiatan?

.....

.....

.....

.....

.....

.....

2.1.6. Uji Kompetensi/Ulangan

1. Berikut adalah elcmen-elemen instruksi, kecuali...
 - a. Operation Code/Kode Operasi
 - b. Source Operand Reference/ Referensi Operand Sumber
 - c. Operation word/ kata operasi
 - d. Result Operand Reference/ Reference Operand Hasil
 - e. Next Instruction Reference/ Reference Operand Selanjutnya
2. Maksud dari elemen Source Operand Reference adalah...
 - a. Operasi dapat membuat hasil operasi
 - b. Operasi dapat mencakup satu atau lebih sumber
 - c. Memberitahu CPU posisi instruksi berikutnya yang harus diambil
 - d. Menyelesaikan eksekusi suatu instruksi
 - e. Operasi dapat mencakup suatu referensi
3. Sumber dan hasil operand dapat berada di salah satu dari daerah di bawah ini, kecuali...
 - a. Memori utama
 - b. Register I/O
 - c. Memori virtual
 - d. Register CPU
 - e. Perangkat I/O
4. Mnemonic adalah...
 - a. Singkatan-singkatan yang mengindikasikan suatu operasi yang merupakan representasi dari opcode
 - b. Singkatan-singkatan yang mengindikasikan suatu operasi yang merupakan representasi dari Source Operand Reference
 - c. Angka-angka yang mengindikasikan suatu operasi yang merupakan representasi dari opcode
 - d. Symbol-simbol yang mengindikasikan suatu operasi yang merupakan representasi dari opcode
 - e. Symbol-simbol yang mengindikasikan suatu operasi yang merupakan representasi dari opcode
5. Arti dari mnemonic INC (Increment), yaitu...
 - a. Muatkan data dari memori
 - b. Penambahan
 - c. Pengurangan

- d. Penambahan dengan satu
- e. Pengurangan dengan satu

2.2 Kegiatan belajar 2 – Tipe – tipe Instruksi

2.2.1. Tujuan Pembelajaran

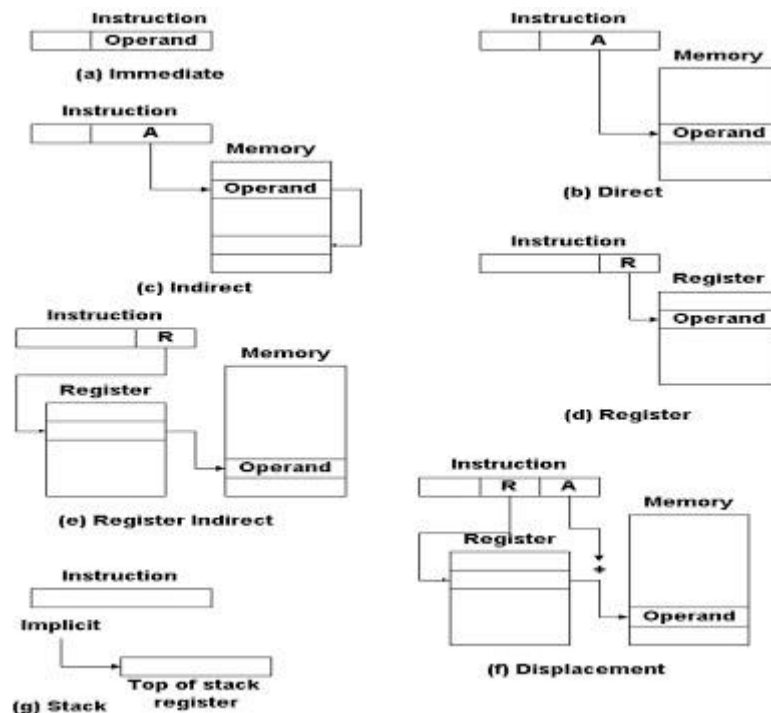
Setelah mengikuti pembelajaran, siswa mampu :

- Menyebutkan dan memahami tipe – tipe instruksi.

2.2.2. Aktivitas belajar siswa

2.2.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah gambar berikut ini :



Sumber : <http://syaifuldeathcore.blogspot.com/>

2.2.2.2. Menanya

Buatlah pertanyaan kepada gurumu tentang tipe tipe instruksi.

2.2.2.3 Mencoba/ Mengumpulkan informasi

Tipe atau jenis-jenis instruksi

1. Data processing: Arithmetic dan Logic Instructions

Data processing adalah jenis pemrosesan yang dapat mengubah data menjadi informasi atau pengetahuan. Pemrosesan data ini sering menggunakan

komputer sehingga bisa berjalan secara otomatis. Setelah diolah, data ini biasanya mempunyai nilai yang informatif jika dinyatakan dan dikemas secara terorganisir dan rapi, maka istilah pemrosesan data sering dikatakan sebagai sistem informasi.

Kedua istilah ini mempunyai arti yang hampir sama, pemrosesan data mengolah dan memanipulasi data mentah menjadi informasi (hasil pengolahan), sedangkan sistem informasi memakai data sebagai bahan masukan dan menghasilkan informasi sebagai produk keluaran.

2. Data storage: Memory instructions

Sering disebut sebagai memori komputer, merujuk kepada komponen komputer, perangkat komputer, dan media perekaman yang mempertahankan data digital yang digunakan untuk beberapa interval waktu. Penyimpanan data komputer menyediakan salah satu tiga fungsi inti dari komputer modern, yakni mempertahankan informasi. Ia merupakan salah satu komponen fundamental yang terdapat di dalam semua komputer modern, dan memiliki keterkaitan dengan mikroprosesor, dan menjadi model komputer yang digunakan semenjak 1940-an.

Dalam penggunaan kontemporer, memori komputer merujuk kepada bentuk media penyimpanan berbahan semikonduktor, yang dikenal dengan sebutan Random Access Memory (RAM), dan kadang-kadang dalam bentuk lainnya yang lebih cepat tapi hanya dapat menyimpan data secara sementara. Akan tetapi, istilah "computer storage" sekarang secara umum merujuk kepada media penyimpanan massal, yang bisa berupa cakram optis, beberapa bentuk media penyimpanan magnetis (seperti halnya hard disk) dan tipe-tipe media penyimpanan lainnya yang lebih lambat ketimbang RAM, tapi memiliki sifat lebih permanen, seperti flash memory.

3. Data Movement: I/O instructions

Proses data movement ini adalah memindahkan (dapat dikatakan membackup juga) data – data dari database yang berupa data, indeks, grand, schema, dan lain – lain ketempat baru. Tempat baru ini bisa ke dalam database baru atau memang untuk dibackup saja.

Data movement terdiri dari 2 bagian besar yaitu :

- Load & Upload [difokuskan untuk memindahkan data yang berupa indeks atau data itu sendiri alias isi dari database tersebut]
- Export & Import [memindahkan data secara lengkap, mulai dari grand, schema, dan seluruhnya]

Jika dilihat, load tersebut berhubungan dengan import dan upload berhubungan dengan export

Load berfungsi untuk memasukan data / transaksi ke sebuah table. Dapat dikatakan juga insert, replace, atau update. Sedangkan upload berfungsi untuk membuat dari data table ke fisik / file. Kelemahan load adalah dalam prosesnya bisa saja terjadi data yang tidak berpindah secara sempurna.

Upload Parameter

- Limit [membatasi beberapa record]
- Sample [mencari sample yang telah ditentukan]
- When [berdasarkan kondisi]

Dan pada upload, hanya satu parameter saja yang dapat berjalan alias tak bisa berjalan bersamaan apabila parameternya lebih dari 1.

Bulk Data Movement (Software Pendukung)

- ETL [Extrat Transform Load], software yang focus terhadap data warehouse
- Replication and Propagation, software yang memonitoring source database dan target, dan yang dihasilkan oleh software ini adalah pencatatatn log.

Perlu diperhatikan juga hak akses dalam load & unload, import & export minimal adalah akses select.

Distribution Database

Dalam distribution database terdapat 3 istilah yaitu :

- Autonomi [idependent], untuk tabel umum akses yang diberikan berbeda dari setiap user.
- Isolation [stand alone], untuk tabel khusus (privacy) itu terpisah dari user.
- Transparancy [all user], akses tabel terpisah dari user tetapi user masih dapat mengaksesnya.

Lawan dari database terdistribusi adalah database terpusat. Server yang terpusat memang diuntungkan dalam sisi maintenance sedangkan server terdistribusi lebih rumit dalam proses integrasinya.

Jika database terdistribusi paling tidak membutuhkan Sumber Daya Manusia [SDM] yang baik, network yang lebih baik karena permasalahan network itu sangat fatal dan biasanya permasalahannya tidak jauh – jauh dari permasalahan traffic network. Dan yang tidak boleh dilupakan adalah request dan respon.

4. Control: Test and branch instructions

Unit kendali (bahasa Inggris: Control Unit - CU) adalah salah satu bagian dari CPU yang bertugas untuk memberikan arahan/kendali/ kontrol terhadap operasi yang dilakukan di bagian ALU (Arithmetic Logical Unit) di dalam CPU tersebut. Output dari CU ini akan mengatur aktivitas dari bagian lainnya dari perangkat CPU tersebut.

Pada awal-awal desain komputer, CU diimplementasikan sebagai ad-hoc logic yang susah untuk didesain. Sekarang, CU diimplementasikan sebagai sebuah microprogram yang disimpan di dalam tempat penyimpanan kontrol (control store). Beberapa word dari microprogram dipilih oleh microsequencer dan bit yang datang dari word-word tersebut akan secara langsung mengontrol bagian-bagian berbeda dari perangkat tersebut, termasuk di antaranya adalah register, ALU, register instruksi, bus dan peralatan input/output di luar chip. Pada komputer modern, setiap subsistem ini telah memiliki kontrolernya masing-masing, dengan CU sebagai pemantaunya (supervisor).

2.2.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang tipe tipe instruksi pada tabel berikut ini :

Tipe Instruksi	Penjelasan

Buatlah kesimpulan tentang tipe tipe instruksi

2.2.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri mengenai tipe tipe instruksi.

2.2.3. Rangkuman

Tipe atau jenis-jenis instruksi, antara lain:

1. Data processing: Arithmetic dan Logic Instructions

Data processing adalah jenis pemrosesan yang dapat mengubah data menjadi informasi atau pengetahuan. Setelah diolah, data ini biasanya mempunyai nilai yang informative, maka istilah pemrosesan data sering dikatakan sebagai sistem informasi.

2. Data storage: Memory instructions

Sering disebut sebagai memori komputer, merujuk kepada komponen komputer, perangkat komputer, dan media perekaman yang mempertahankan data digital yang digunakan untuk beberapa interval waktu.

Dalam penggunaan kontemporer, memori komputer merujuk kepada bentuk media penyimpanan berbahan semikonduktor, yang dikenal dengan sebutan Random Access Memory (RAM). Akan tetapi, istilah "computer storage" sekarang secara umum merujuk kepada media penyimpanan massal seperti halnya hard disk.

3. Data Movement: I/O instructions

Proses data movement ini adalah memindahkan (dapat dikatakan membackup juga) data – data dari database yang berupa data, indeks, grand, schema, dan lain – lain ketempat baru.

Data movement terdiri dari 2 bagian besar yaitu: Load & Upload dan Export & Import. Load berfungsi untuk memasukan data / transaksi ke sebuah table. Sedangkan upload berfungsi untuk membuat dari data table ke fisik / file. Upload Parameter, antara lain: limit, sample, dan when.

Bulk Data Movement (Software Pendukung), antara lain: ETL [Extract Transform Load] dan Replication and Propagation. Hak akses dalam load & unload, import & export minimal adalah akses select.

Dalam distribution database terdapat 3 istilah yaitu: Autonomi [idependent]; Isolation [stand alone]; dan Transparancy [all user]. Lawan dari database terdistribusi adalah database terpusat.

4. Control: Test and branch instructions

CU diimplementasikan sebagai sebuah microprogram yang disimpan di dalam tempat penyimpanan kontrol (control store). Control Unit - CU) adalah

salah satu bagian dari CPU yang bertugas untuk memberikan arahan/kendali/kontrol terhadap operasi yang dilakukan di bagian ALU (Arithmetic Logical Unit) di dalam CPU tersebut.

2.2.4. Tugas

1. Sebutkan Tipe atau jenis-jenis instruksi!

.....

.....

.....

.....

.....

2. Data movement terdiri dari 2 bagian besar yaitu?

.....

.....

.....

.....

.....

3. Dalam distribution database terdapat 3 istilah yaitu?

.....

.....

.....

.....

.....

4. Unit kendali (bahasa Inggris: Control Unit - CU) adalah?

.....

.....

.....

.....

.....

5. CU diimplementasikan sebagai?

.....

.....

.....

.....

.....

2.2.5. Penilaian diri

Nama :

Nama-nama anggota kelompok :

Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....

<ul style="list-style-type: none"> ▪ Mendengarkan orang lain ▪ Mengajukan pertanyaan ▪ Mengorganisasi ide-ide saya 	<ul style="list-style-type: none"> ▪ Mengorganisasi kelompok ▪ Mengacaukan kegiatan ▪ Melamun
---	--
6. Apa yang kamu lakukan selama kegiatan?

.....

2.2.6. Uji Kompetensi/Ulangan

1. Yang bukan merupakan tipe/ jenis instruksi, yaitu...
 - a. Data processing
 - b. Data storage
 - c. Data result
 - d. Data movement
 - e. Control
2. Memori komputer merujuk kepada bentuk media penyimpanan berbahan semikonduktor, yaitu...
 - a. CPU
 - b. ROM
 - c. Hardisk
 - d. Flashdisk
 - e. RAM
3. Pada data movement, Load berfungsi untuk...
 - a. Memasukan data / transaksi ke sebuah table
 - b. Mengolah data / transaksi dalam sebuah table
 - c. Mengeksekusi data / transaksi dalam sebuah table
 - d. Menampilkan data / transaksi pada sebuah table
 - e. Membuat dari data table ke fisik / file
4. Membuat dari data table ke fisik / file, merupakan fungsi dari...

- a. Upload
 - b. Load
 - c. Browse
 - d. Create
 - e. Write
5. Yang dimaksud dengan transparency [all user] dalam distribution database, adalah...
- a. Tabel umum akses yang diberikan berbeda dari setiap user
 - b. Tabel khusus (privacy) itu terpisah dari user
 - c. Tabel terpisah dari user dan user tidak dapat mengaksesnya
 - d. Tabel terpisah dari user tetapi user masih dapat mengaksesnya
 - e. Table khusus untuk user
6. Tugas dari Control Unit adalah...
- a. Memasukan data / transaksi ke sebuah table
 - b. Memberikan arahan/kendali/ kontrol terhadap operasi yang dilakukan di bagian ALU (Arithmetic Logical Unit)
 - c. Mengubah data menjadi informasi atau pengetahuan
 - d. Mempertahankan data digital yang digunakan untuk beberapa interval waktu
 - e. Mengubah arahan/kendali/ kontrol terhadap operasi yang dilakukan di bagian ALU (Arithmetic Logical Unit)

2.3 Kegiatan belajar 3 – Tipe – tipe operand

2.3.1. Tujuan Pembelajaran

Setelah mengikuti pembelajaran, siswa mampu :

- Menyebutkan dan Memahami Tipe tipe operand.

2.3.2. Aktivitas belajar siswa

2.3.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah tabel gambar berikut ini :

Operator	Operasi	Tipe Operand	Tipe Hasil
*	Perkalian	real,real integer,integer real,integer	real integer real
DIV	Pembagian Bulat	integer,integer	integer
/	Pembagian real	real,real integer,integer real,integer	real real real
MOD	Sisa Pembagian	integer,integer	integer
+	Pertambahan	real,real integer,integer real,integer	real integer real
-	Pengurangan	real,real integer,integer real,integer	real integer real

Sumber : <http://dheardheani-2011130051-sistemoperasi.blogspot.com/>

2.3.2.2. Menanya

Buatlah pertanyaan kepada gurumu tentang tipe tipe operand.

2.3.2.3. Mencoba/ Mengumpulkan informasi

Tipe-tipe Operand

1. Pengertian Operand

Operand adalah sebuah objek yang ada pada operasi matematika yang dapat digunakan untuk melakukan operasi. Operand atau operator dalam bahasa C berbentuk simbol bukan berbentuk keyword atau kata yang biasa ada di bahasa pemrograman lain. Simbol yang digunakan bukan karakter yang ada dalam abjad tapi ada pada keyboard kita seperti = , * dan sebagainya.

2. Tipe Operand

a. Tipe Data

Tipe Data setiap data memiliki tipe data, apakah merupakan angka bulat (integer), angka biasa (real), atau berupa karakter (char), dan sebagainya.

Ada dua kategori dari tipe data yaitu tipe dasar dan tipe bentukan.

a. Tipe dasar adalah tipe data yang selalu tersedia pada setiap bahasa pemrograman, antara lain :

a. Bilangan bulat

- Integer (-32768 s/d +32768)

Bilangan atau angka yang tidak memiliki titik desimal atau pecahan seperti +10,-1024,+32767,+255. Tipe data ini dapat dituliskan sebagai integer atau int. Operasi aritmatiknya terdiri dari : tambah +, kurang -, kali *, bagi /, sisa hasil bagi %. Operasi Perbandingan terdiri dari : lebih kecil <, lebih kecil atau sama , lebih besar atau sama >=

- Long (-16 juta s/d 16 juta)
- Byte (0 s/d 255)

b. Bilangan pecahan

- Double ($3,7 \times 10^{-308}$ s/d $3,7 \times 10^{+308}$)
- Float ($3,4 \times 10^{-38}$ s/d $3,4 \times 10^{+38}$)

c. Karakter

Karakter adalah data tunggal yang mewakili semua huruf, simbol baca dan juga simbol angka yang tidak dapat dioperasikan secara sistematis. Tipe ini dapat dituliskan sebagai char.

b. Tipe bentukan adalah tipe data yang dibentuk dari kombinasi tipe dasar, antara lain :

a. Array (larik)

Selama ini kita menggunakan satu variabel untuk menyimpan 1 buah nilai dengan tipe data tertentu.

misalnya : `int a1,a2,a3,a4,a5;`

Deklarasi variabel diatas menggunakan menyimpan 5 data integer dimana masing-masing variabel diberi nama a1,a2,a3,a4,a5. Jika kita memiliki 10 data integer atau mungkin 100 data integer bahkan mungkin kita tidak ketahui atau mungkin bersifat dinamis. Saat ini lah kita menggunakan tipe bentukan Array.

Array adalah tipe data bentukan yang merupakan wadah untuk menampung beberapa nilai data yang sejenis

b. String

String adalah tipe data bentukan yang merupakan deretan karakter yang membentuk satu kata atau satu kalimat, yang biasanya dapat dua tanda kutip.

b. Variabel

Variabel adalah nama yang mewakili suatu elemen data seperti : jenis untuk jenis kelamin, t4lahir untuk tempat lahir, alamat untuk alamat dan sebagainya.

Ada aturan tertentu yang wajib diikuti dalam pemberian nama variabel, antara lain :

- Harus dimulai dengan abjad tidak boleh dengan angka atau simbol.
- Tidak boleh ada spasi diantaranya
- Jangan menggunakan simbol-simbol yang bisa membingungkan seperti titik dua, titik koma, kima, dan sebagainya.
- Sebaiknya memiliki arti yang sesuai dengan elemen data

Sebaiknya tidak terlalu panjang

Contoh variabel yang benar : Nama, Alamat, Nilai_Ujian

Contoh Variabel yang salah : 4XYZ, IP rata, Var :+xy,458;

c. Operator dan Operand

Operand adalah data, tetapan, perubah atau hasil dari suatu fungsi sedangkan Operator merupakan simbol-simbol yang memiliki fungsi untuk menghubungkan operand sehingga menjadi transformasi.

Jenis-jenis operator adalah sebagai berikut :

a. Operator Aritmetika

Operator untuk melakukan fungsi aritmetika seperti : +(penjumlahan), – (mengurangkan), * (mengalikan), / (membagi).

b. Operator relational

Operator untuk menyatakan relasi atau perbandingan antara dua operand, seperti > (lebih besar), =(lebih besar atau sama), <= (lebih kecil atau sama), == (sama), != (tidak sama).

c. Operator Logik

Operator untuk merelasikan operand secara logis seperti && (and), || (or), !(not).

2.3.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang tipe tipe operand pada tabel berikut ini :

Tipe Operand	Penjelasan

Buatlah kesimpulan tentang tipe tipe operand.

2.3.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri tentang tipe tipe operand.

2.3.3. Rangkuman

Operand adalah sebuah objek yang ada pada operasi matematika yang dapat digunakan untuk melakukan operasi. Operand atau operator dalam bahasa C berbentuk symbol yang berupa karakter yang ada pada keyboard seperti = , * dan sebagainya.

Tipe Operand, antara lain:

- Tipe Data, dapat berupa angka bulat (integer), angka biasa (real), atau berupa karakter (char), dan sebagainya. Ada dua kategori dari tipe data yaitu tipe dasar (berupa Bilangan bulat Integer, Long, Byte; Bilangan pecahan double dan float; dan karakter) dan tipe bentukan (berupa Array (larik) dan string).

- b. Variabel, yaitu nama yang mewakili suatu elemen data seperti : jenis untuk jenis kelamin, tanggal lahir untuk tempat lahir. Aturan yang wajib diikuti dalam pemberian nama variabel, antara lain: Harus dimulai dengan huruf abjad; Tidak boleh ada spasi diantaranya; tidak menggunakan simbol-simbol; Sebaiknya memiliki arti yang sesuai dengan elemen data; Sebaiknya tidak terlalu panjang
- c. Operator dan Operand. Operand adalah data, tetapan, perubah atau hasil dari suatu fungsi sedangkan Operator merupakan simbol-simbol yang memiliki fungsi untuk menghubungkan operand sehingga menjadi transformasi. Jenis-jenis operator adalah: Operator Aritmetika; Operator relational; Operator Logik.

2.3.4. Tugas

1. Operand adalah?

.....

.....

.....

.....

.....

2. Sebutkan Tipe Operand!

.....

.....

.....

.....

.....

3. Ada dua kategori dari tipe data yaitu tipe dasar dan tipe bentukan, jelaskan!

.....

.....

.....

.....

.....

4. Variabel adalah?

.....

.....

.....

.....

.....

5. Jenis-jenis operator adalah?

.....

.....

.....

.....

.....

2.3.5. Penilaian diri

Nama :
 Nama-nama anggota kelompok :
 Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....

▪ Mendengarkan orang lain	▪ Mengorganisasi kelompok
▪ Mengajukan pertanyaan	▪ Mengacaukan kegiatan
▪ Mengorganisasi ide-ide saya	▪ Melamun
6. Apa yang kamu lakukan selama kegiatan?

.....

2.3.6. Uji Kompetensi/Ulangan

1. Operand adalah...
 - a. Simbol-simbol yang memiliki fungsi untuk menghubungkan operand sehingga menjadi tranformasi
 - b. Data yang selalu tersedia pada setiap bahasa pemograman
 - c. Sebuah objek yang ada pada operasi sistem yang dapat digunakan untuk melakukan operasi
 - d. Data yang tersedia pada setiap operasi
 - e. Sebuah objek yang ada pada operasi matematika yang dapat digunakan untuk melakukan operasi
2. Data tunggal yang mewakili semua huruf, simbol baca dan juga simbol angka yang tidak dapat dioperasikan secara sistematis, disebut...
 - a. Karakter
 - b. Bilangan bulat
 - c. Bilangan pecahan

- d. Bilangan decimal
 - e. Symbol
3. Di bawah ini yang bukan merupakan bentuk dari tipe data pada tipe operand, yaitu...
 - a. Angka bulat
 - b. Angka biasa
 - c. Integer
 - d. Outreger
 - e. Karakter
 4. Tipe data yang selalu tersedia pada setiap bahasa pemrograman adalah...
 - a. Tipe bentukan
 - b. Tipe dasar
 - c. Tipe bahasa
 - d. Tipe program
 - e. Tipe file
 5. Tipe bentukan adalah...
 - a. Tipe data yang selalu tersedia pada setiap bahasa pemrograman
 - b. Tipe data yang dibentuk dari kombanisasi tipe dasar
 - c. Tipe data yang merupakan deretan karakter yang membentuk satu kata atau satu kalimat
 - d. Tipe data yang dibentuk dari kombanisasi tipe program
 - e. Tipe data yang hanya tersedia pada satu program
 6. Tipe data bentukan yang merupakan wadah untuk menampung beberapa nilai data yang sejenis disebut...
 - a. Array
 - b. String
 - c. Operator
 - d. Variable
 - e. Data value
 7. String adalah...
 - a. Tipe data bentukan yang merupakan wadah untuk menampung beberapa nilai data yang sejenis
 - b. Tipe data bentukan yang merupakan deretan karakter yang membentuk satu kata atau satu kalimat, yang biasanya dapat dua tanda kutip
 - c. Tipe data bentukan yang merupakan deretan karakter yang membentuk satu kata atau satu kalimat, yang biasanya dapat dua tanda strip
 - d. Tipe data bentukan yang merupakan wadah untuk menampung beberapa nilai data yang berbeda jenis
 - e. Tipe data yang mewakili sutau elemen data
 8. Nama yang mewakili sutau elemen data seperti jenkel untuk jenis kelamin, t4lahir untuk tempat lahir, alamat untuk alamat, disebut...
 - a. Array
 - b. String
 - c. Operator
 - d. Variable
 - e. Data value
 9. Di bawah ini merupakan aturan yang wajib diikuti dalam pemberian nama variabel, kecuali...

- a. Harus dimulai dengan abjad tidak boleh dengan angka atau simbol
 - b. Tidak boleh ada spasi diantaranya
 - c. Menggunakan simbol-simbol seperti titik dua, titik koma, koma
 - d. Tidak menggunakan simbol-simbol yang bisa membingungkan seperti titik dua, titik koma, koma
 - e. Sebaiknya memiliki arti yang sesuai dengan elemen data
10. Jenis-jenis operator antara lain operator aritmatika, relational, dan...
- a. Dasar
 - b. Bentukkan
 - c. Logic
 - d. Real
 - e. Fiction

2.4 Kegiatan belajar 4 – Tipe – tipe operasi data

2.4.1. Tujuan Pembelajaran

Setelah mengikuti pembelajaran, siswa mampu :

- Menyebutkan dan Memahami Tipe – tipe operasi data

2.4.2. Aktivitas belajar siswa

2.4.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah tabel gambar berikut ini :

Operator	Operasi	Tipe Operand	Tipe Hasil
*	Perkalian	real,real integer,integer real,integer	real integer real
DIV	Pembagian Bulat	integer,integer	integer
/	Pembagian real	real,real integer,integer real,integer	real real real
MOD	Sisa Pembagian	integer,integer	integer
+	Pertambahan	real,real integer,integer real,integer	real integer real
-	Pengurangan	real,real integer,integer real,integer	real integer real

Sumber : <http://dheardheani-2011130051-sistemoperasi.blogspot.com/>

2.4.2.2. Menanya

Buatlah pertanyaan tentang tipe tipe operasi.

2.4.2.3. Mencoba/ Mengumpulkan informasi

TRANSFER DATA

- Menetapkan lokasi operand sumber dan operand tujuan.
- Lokasi-lokasi tersebut dapat berupa memori, register atau bagian paling atas daripada stack.
- Menetapkan panjang data yang dipindahkan.
- Menetapkan mode pengalamatan.

Tindakan CPU untuk melakukan transfer data adalah :

- Memindahkan data dari satu lokasi ke lokasi lain.
- Apabila memori dilibatkan :

- Menetapkan alamat memori.
- Menjalankan transformasi alamat memori virtual ke alamat memori aktual.
- Mengawali pembacaan / penulisan memori

Operasi set instruksi untuk transfer data :

- MOVE : memindahkan word atau blok dari sumber ke tujuan
- STORE : memindahkan word dari prosesor ke memori.
- LOAD : memindahkan word dari memori ke prosesor.
- EXCHANGE : menukar isi sumber ke tujuan.
- CLEAR / RESET : memindahkan word 0 ke tujuan.
- SET : memindahkan word 1 ke tujuan.
- PUSH : memindahkan word dari sumber ke bagian paling atas stack.
- POP : memindahkan word dari bagian paling atas sumber

ARITHMETIC

Tindakan CPU untuk melakukan operasi arithmetic :

- Transfer data sebelum atau sesudah.
- Melakukan fungsi dalam ALU.
- Menset kode-kode kondisi dan flag.

Operasi set instruksi untuk arithmetic :

1. ADD : penjumlahan
2. SUBTRACT : pengurangan
3. MULTIPLY : perkalian
4. DIVIDE : pembagian
5. ABSOLUTE
6. NEGATIVE
7. DECREMENT
8. INCREMENT

Nomor 5 sampai 8 merupakan instruksi operand tunggal.

LOGICAL

Tindakan CPU sama dengan arithmetic.

Operasi set instruksi untuk operasi logical :

- AND, OR, NOT, EXOR
- COMPARE : melakukan perbandingan logika.
- TEST : menguji kondisi tertentu.
- SHIFT : operand menggeser ke kiri atau kanan menyebabkan konstanta pada ujung bit.
- ROTATE : operand menggeser ke kiri atau ke kanan dengan ujung yang terjalin.

CONVERSI

Tindakan CPU sama dengan arithmetic dan logical.

Instruksi yang mengubah format instruksi yang beroperasi terhadap format data.

Misalnya pengubahan bilangan desimal menjadi bilangan biner.

Operasi set instruksi untuk konversi :

- TRANSLATE : menterjemahkan nilai-nilai dalam suatu bagian memori berdasarkan tabel korespondensi.
- CONVERT : mengkonversi isi suatu word dari suatu bentuk ke bentuk lainnya.

INPUT / OUTPUT

Tindakan CPU untuk melakukan INPUT /OUTPUT :

- Apabila memory mapped I/O maka menentukan alamat memory mapped.

- Mengawali perintah ke modul I/O

Operasi set instruksi Input / Output :

- INPUT : memindahkan data dari perangkat I/O tertentu ke tujuan
- OUTPUT : memindahkan data dari sumber tertentu ke perangkat I/O
- START I/O : memindahkan instruksi ke prosesor I/O untuk mengawali operasi I/O
- TEST I/O : memindahkan informasi dari sistem I/O ke tujuan

TRANSFER CONTROL

Tindakan CPU untuk transfer control :

- Mengupdate program counter untuk subrutin , call / return.

Operasi set instruksi untuk transfer control :

- JUMP (cabang) : pemindahan tidak bersyarat dan memuat PC dengan alamat tertentu.
- JUMP BERSYARAT : menguji persyaratan tertentu dan memuat PC dengan alamat tertentu atau tidak melakukan apa tergantung dari persyaratan.
- JUMP SUBRUTIN : melompat ke alamat tertentu.
- RETURN : mengganti isi PC dan register lainnya yang berasal dari lokasi tertentu.
- EXECUTE : mengambil operand dari lokasi tertentu dan mengeksekusi sebagai instruksi
- SKIP : menambah PC sehingga melompati instruksi berikutnya.
- SKIP BERSYARAT : melompat atau tidak melakukan apa-apa berdasarkan pada persyaratan
- HALT : menghentikan eksekusi program.
- WAIT (HOLD) : melanjutkan eksekusi pada saat persyaratan dipenuhi.
- NO OPERATION : tidak ada operasi yang dilakukan.

CONTROL SYSTEM

Hanya dapat dieksekusi ketika prosesor berada dalam keadaan khusus tertentu atau sedang mengeksekusi suatu program yang berada dalam area khusus, biasanya digunakan dalam sistem operasi.

Contoh : membaca atau mengubah register kontrol.

2.4.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang tipe tipe operasi pada tabel berikut ini :

Tipe Operasi	Penjelasan

Buatlah kesimpulan tentang tipe tipe operasi.

2.4.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri tentang tipe tipe operasi.

2.4.3. Rangkuman

Transfer data, meliputi kegiatan: Menetapkan lokasi operand sumber dan operand tujuan; Lokasi-lokasi tersebut dapat berupa memori, register atau bagian paling atas daripada stack; Menetapkan panjang data yang dipindahkan; Menetapkan mode pengalamatan. Tindakan CPU untuk melakukan transfer data adalah: Memindahkan data dari satu lokasi ke lokasi lain; Menetapkan alamat memori, Menjalankan transformasi alamat memori virtual ke alamat memori actual, Mengawali pembacaan / penulisan memori (Apabila memori dilibatkan). Operasi set instruksi untuk transfer data, antara lain: MOVE, STORE, LOAD, EXCHANGE, CLEAR / RESET, SET, PUSH, POP.

ARITHMETIC. Tindakan CPU untuk melakukan operasi arithmetic, antara lain: Transfer data sebelum atau sesudah; Melakukan fungsi dalam ALU; dan Menset kode-kode kondisi dan flag. Operasi set instruksi untuk arithmetic, yaitu: ADD, SUBTRACT, DIVIDE, ABSOLUTE, NEGATIVE, DECREMENT, INCREMENT.

LOGICAL. Tindakan CPU untuk melakukan operasi logical, sama dengan arithmetic. Operasi set instruksi untuk operasi logical, yaitu: AND, OR, NOT, EXOR; COMPARE; TEST; SHIFT; ROTATE.

CONVERSI. Tindakan CPU untuk melakukan operasi Conversi, sama dengan arithmetic dan logical. Operasi set instruksi untuk conversi, yaitu: TRANSLATE dan CONVERT.

I/O. Tindakan CPU untuk melakukan INPUT /OUTPUT, yaitu: Apabila memory mapped I/O maka menentukan alamat memory mapped , dan Mengawali perintah ke modul I/O. Operasi set instruksi Input / Ouput, antara lain: INPUT, OUTPUT, START I/O, TEST I/O.

TRANSFER CONTROL. Tindakan CPU untuk transfer control, yaitu Mengupdate program counter untuk subrutin , call / return. Operasi set instruksi untuk transfer control, meliputi: JUMP (cabang); JUMP BERSYARAT; JUMP SUBRUTIN; RETURN; EXECUTE; SKIP; SKIP BERSYARAT; HALT; WAIT (HOLD); NO OPERATION.

CONTROL SYSTEM. Hanya dapat dieksekusi ketika prosesor berada dalam keadaan khusus tertentu atau sedang mengeksekusi suatu program yang berada dalam area khusus, biasanya digunakan dalam sistem operasi. Contoh : membaca atau mengubah register kontrol.

2.4.4. Tugas

1. Tindakan CPU untuk melakukan transfer data adalah?

.....

.....

.....

-

 2. Sebutkan dan Jelaskan Operasi set instruksi untuk transfer data!

-

 3. Sebutkan Operasi set instruksi untuk conversi!

-

 4. Sebutkan Operasi set instruksi untuk transfer control?

-

 5. Sebutkan Operasi set instruksi Input / Ouput?

2.4.5. Penilaian diri

Nama :
 Nama-nama anggota kelompok :
 Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.

- 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....
- Mendengarkan orang lain
 - Mengajukan pertanyaan
 - Mengorganisasi ide-ide saya
 - Mengorganisasi kelompok
 - Mengacaukan kegiatan
 - Melamun
6. Apa yang kamu lakukan selama kegiatan?
-
-
-
-
-
-

2.4.6. Uji Kompetensi/Ulangan

1. Transfer data meliputi kegiatan di bawah ini, kecuali...
 - a. Menetapkan lokasi operand sumber dan operand tujuan
 - b. Lokasi-lokasi tersebut dapat berupa memori, register atau bagian paling atas daripada stack
 - c. Menetapkan panjang data yang dipindahkan
 - d. Menetapkan mode pengalamatan
 - e. Menetapkan kualitas data yang dipindahkan
2. Di bawah ini merupakan tindakan CPU untuk melakukan transfer data apabila memori dilibatkan, kecuali...
 - a. Mengawali penulisan memori
 - b. Menetapkan alamat memori
 - c. Menjalankan transformasi alamat memori aktual ke alamat memori virtual
 - d. Menjalankan transformasi alamat memori virtual ke alamat memori aktual
 - e. Mengawali pembacaan memori
3. Memindahkan word dari bagian paling atas sumber disebut instruksi...
 - a. PUSH
 - b. SET
 - c. STORE
 - d. EXCHANGE
 - e. POP
4. Di bawah ini merupakan operasi set instruksi untuk arithmetic, kecuali...
 - a. SUBTRACT
 - b. NEGATIVE
 - c. POSITIVE
 - d. DECREMENT
 - e. ABSOLUTE
5. Yang bukan merupakan operasi set instruksi untuk operasi logical, yaitu...
 - a. AND
 - b. COMPARE
 - c. ROTATE
 - d. SHIFT
 - e. IF

6. Fungsi dari operasi set instruksi TRANSLATE, yaitu...
 - a. Menterjemahkan nilai-nilai dalam suatu bagian memori berdasarkan tabel translation
 - b. Menterjemahkan nilai-nilai dalam suatu bagian memori berdasarkan tabel korespondensi
 - c. Mengkonversi isi suatu word dari suatu bentuk ke bentuk lainnya
 - d. Menguji kondisi tertentu
 - e. Menterjemahkan perbandingan logika
7. Instruksi yang mengubah format instruksi yang beroperasi terhadap format data, disebut...
 - a. INPUT/ OUTPUT
 - b. CONVERSI
 - c. LOGICAL
 - d. ARITHMETIC
 - e. TRANSFER CONTROL
8. Memindahkan informasi dari sistem I/O ke tujuan merupakan fungsi dari instruksi INPUT/ OUTPUT, yaitu...
 - a. INPUT
 - b. OUTPUT
 - c. START I/O
 - d. TEST I/O
 - e. MOVE I/O
9. Tindakan CPU untuk transfer control, adalah...
 - a. Memindahkan data dari satu lokasi ke lokasi lain
 - b. Menset kode-kode kondisi dan flag
 - c. Mengupdate program counter untuk subrutin , call / return
 - d. Mengubah format instruksi yang beroperasi terhadap format data
 - e. Mengawali perintah ke modul I/O
10. Yang bukan merupakan operasi set instruksi untuk transfer control, yaitu...
 - a. NO OPERATION
 - b. SKIP BERSYARAT
 - c. JUMP SUBRUTIN
 - d. WAIT BERSYARAT
 - e. WAIT (HOLD)

2.5 Kegiatan belajar 5 – Instruksi percabangan

2.5.1. Tujuan Pembelajaran

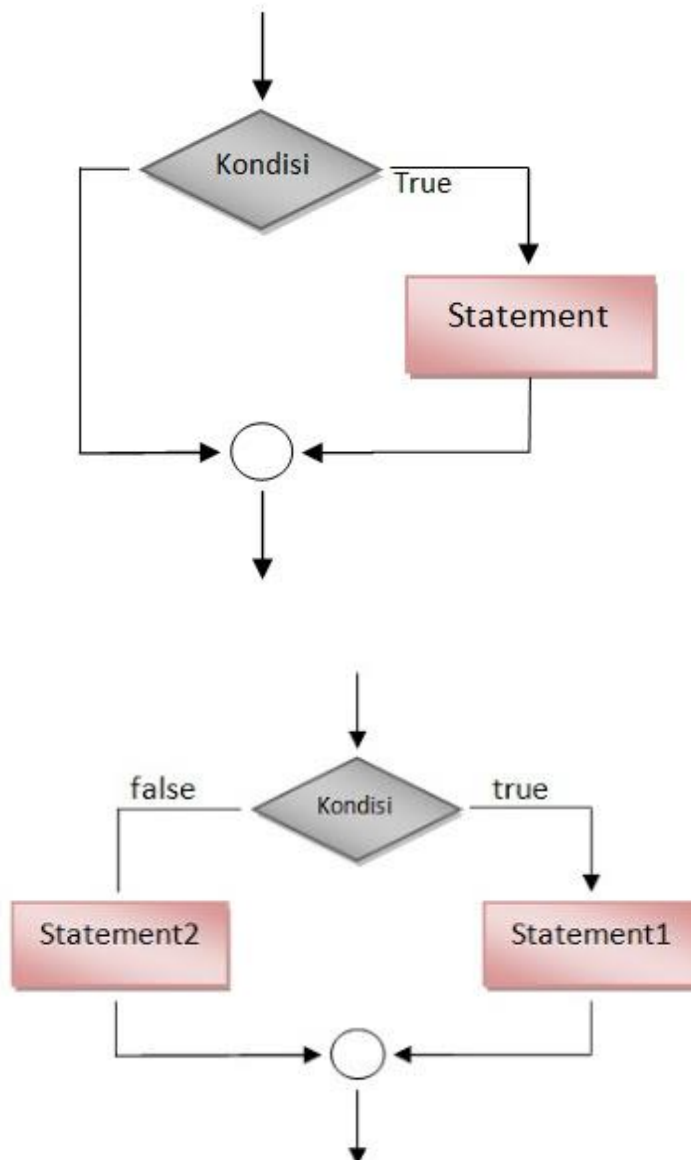
Setelah mengikuti pembelajaran, siswa mampu :

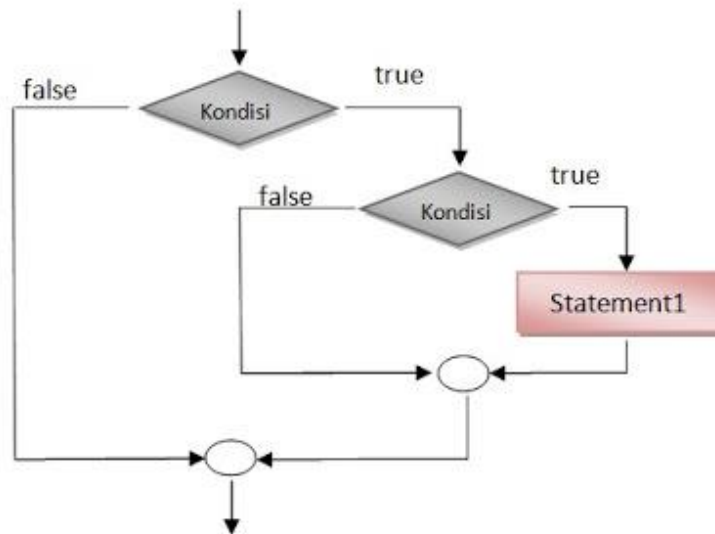
- Menyebutkan dan Memahami Instruksi percabangan

2.5.2. Aktivitas belajar siswa

2.5.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah gambar berikut ini





Sumber :<http://teknikinformatikaunm.blogspot.com>

2.5.2.2. Menanya

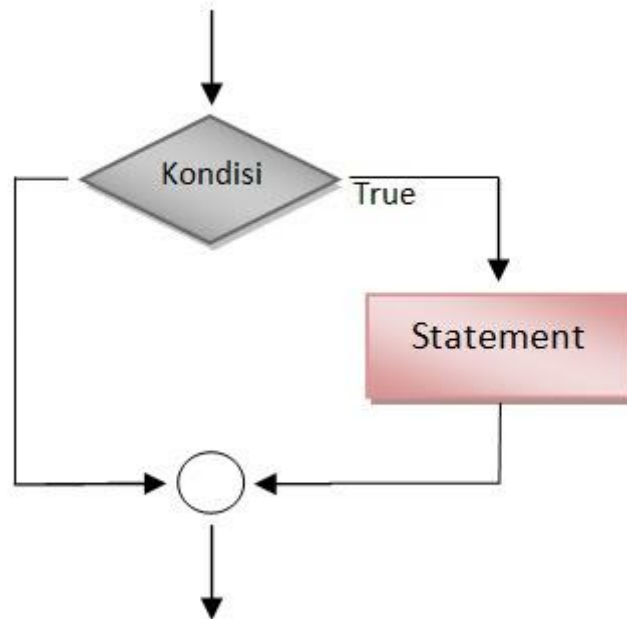
Buatlah pertanyaan kepada gurumu mengenai instruksi percabangan.

2.5.2.3. Mencoba/ Mengumpulkan informasi

Struktur Branching (Percabangan)

1. IF

Instruksi IF digunakan untuk memeriksa suatu kondisi dan melaksanakan instruksi lain jika kondisi tersebut terpenuhi atau bernilai true. Statement pada sintaks if boleh berupa satu instruksi tunggal atau beberapa instruksi (block statement) yang ditulis dalam {}. Jika evaluasi terhadap kondisi memberikan nilai true maka statement dikerjakan. Jika bernilai false maka statement tidak dikerjakan. Alur logika Instruksi IF dapat anda lihat pada gambar dibawah ini.



Sumber : <http://teknikinformatikaunm.blogspot.com>

Gambar 2.1. Alur logika Instruksi IF

Umumnya kondisi berupa sebuah ekspresi yang bernilai Boolean seperti $a == 5$ atau $b > 10$. Kondisi yang diuji dapat berupa kondisi tunggal atau kondisi gabungan.

Contoh pemakaian instruksi IF.

Dengan kondisi tunggal

```
if (nilai > 60) printf ("lulus") ;
```

Jika nilai lebih besar daripada 60 maka ditampilkan lulus

Dengan kondisi gabungan

```
if (( grade == 'D' ) || ( grade == 'E' ))
```

```
printf ("tidak lulus") ;
```

Jika grade berisi huruf D atau E maka ditampilkan tidak lulus

2. If Else

Adakalanya kita harus melaksanakan proses yang berbeda untuk kondisi yang berbeda, misalnya menampilkan kata lulus untuk nilai yang sama dengan atau lebih besar daripada 60 dan menampilkan kata tidak lulus untuk nilai yang kurang dari 60, selain dengan menggunakan dua instruksi if yang terpisah seperti berikut:

```
if (nilai >= 60) printf ("lulus") ;
```

```
if ( nilai < 60 ) printf ("tidak lulus") ;
```

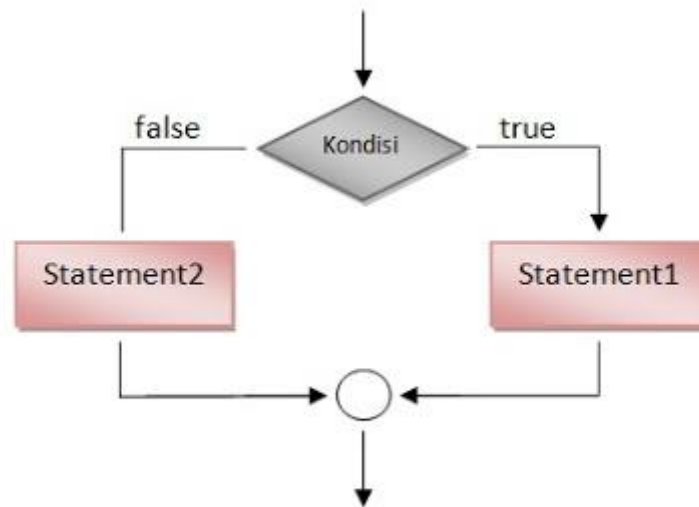
bahasa C menyediakan klausa else untuk digunakan bersama dengan instruksi if. Secara umum pola dari if else yaitu:

```
if (kondisi) statement1 ; else statement2;
```

Apabila evaluasi terhadap kondisi memberikan nilai true maka statement1 dikerjakan, sebaliknya jika false maka statement2 dikerjakan. Jadi yang dikerjakan selalu salah satu dari kedua statement. Statement dapat berupa

instruksi kosong, instruksi tunggal atau beberapa instruksi yang dilingkup dengan { }. Tidak diperbolehkan ada klausa else tanpa instruksi if.

Alur logika instruksi if else dapat andalihat pada gambar dibawah ini:



Sumber : <http://teknikinformatikaunm.blogspot.com>

Gambar 2.2. Logika instruksi if else

Contoh pemakaian instruksi if else

If (nilai >= 60) printf ("lulus") ;

Else printf ("tidak lulus")

Jika nilai lebih besar sama dengan 60 maka tampilkan lulus, selain dari itu maka tampilkan tidak lulus.

If ((bil % 2) == 0) printf ("bilangan genap") ;

Else printf ("bilangan ganjil")

Jika bilangan habis dibagi 2 maka tampilkan genap, selain dari itu maka tampilkan bilangan ganjil

If ((grade == 'D') || (grade == 'E'))

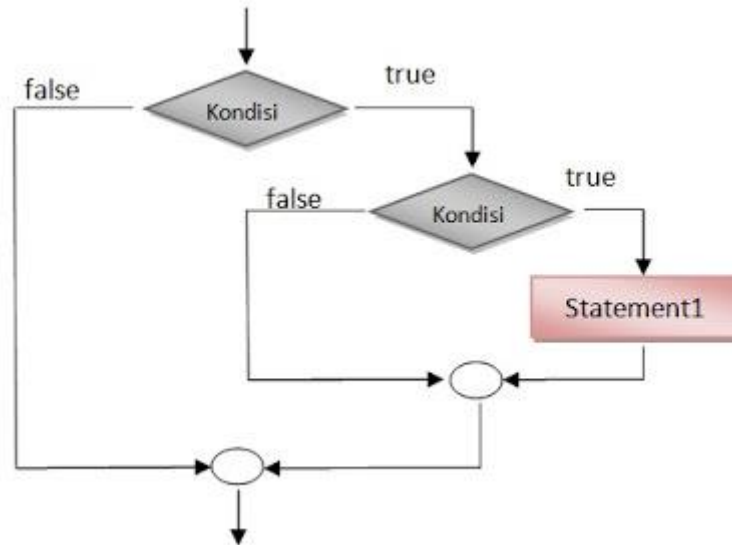
Printf ("tidak lulus")

Else printf ("lulus")

Jika grade berisi "D" atau "E" maka tampilkan tidak lulus, selain dari itu tampilkan lulus.

3. Nested If (Struktur If bersarang)

Pada suatu instruksi if, statement yang dikerjakan apabila kondisi bernilai true dapat berupa instruksi if yang lain. Struktur instruksi seperti ini disebut nested if (if bersarang). Hal yang sama juga berlaku untuk statement yang dikerjakan pada bagian klausa else. Berikut ini salah satu bentuk konfigurasi nested if beserta cara penulisannya :



Sumber : <http://teknikinformatikaunm.blogspot.com>

Gambar 2.3. Konfigurasi nested if

Berikut ini cara membaca dari flowchart Nested if diatas :

```

if (kondisi1)
if (kondisi2) {
}

```

Contoh soal

Terdapat tiga bilangan bulat yang berbeda. Tulis program untuk menampilkan bilangan bulat terbesar diantara bilangan bulat tersebut.

```

1  #include <studio.h>
2
3  Int main () {
4      Int a, b, c;
5
6      Scanf ("%d %d %d", &a, &b, &c);
7      If (a > b && a > c) printf ("terbesar : %d\n", a); else
8      If (b > a && b > c) printf ("terbesar : %d\n", b); else
9      If (c > a && c > b) printf ("terbesar : %d\n", c);
10     Retur 0;
11 }

```

20 30 10

Terbesar : 30

2.5.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang tipe tipe operasi pada tabel berikut ini :

Struktur percabangan	Penjelasan

Buatlah kesimpulan tentang struktur percabangan.

2.5.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri instruksi percabangan.

2.5.3. Rangkuman

Struktur Branching (Percabangan)

1. IF. Instruksi IF digunakan untuk memeriksa suatu kondisi dan melaksanakan instruksi lain jika kondisi tersebut terpenuhi atau bernilai true.
2. If Else. Apabila evaluasi terhadap kondisi memberikan nilai true maka statement dikerjakan, sebaliknya jika false maka statement tidak dikerjakan. Statement dapat berupa instruksi kosong, instruksi tunggal atau beberapa instruksi yang dilingkup dengan { }. Tidak diperbolehkan ada klausa else tanpa instruksi if.
3. Nested If (Struktur If bersarang), yaitu pada suatu instruksi if, statement yang dikerjakan apabila kondisi bernilai true dapat berupa instruksi if yang lain. Hal yang sama juga berlaku untuk statement yang dikerjakan pada bagian klausa else.

2.5.4. Tugas

1. Jelaskan struktur Branching itu!

.....

.....

.....

.....

.....

2. Apakah fungsi dari Instruksi IF?

.....

.....

.....

.....

.....

3. Jelaskan Alur logika Instruksi IF!

.....

4. Jelaskan alur logika Nested If (Struktur If bersarang)!

.....

5. Jelaskan alur logika If Else?

.....

2.5.5. Penilaian diri

Nama :

Nama-nama anggota kelompok :

Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....

▪ Mendengarkan orang lain	▪ Mengorganisasi kelompok
▪ Mengajukan pertanyaan	▪ Mengacaukan kegiatan
▪ Mengorganisasi ide-ide saya	▪ Melamun
6. Apa yang kamu lakukan selama kegiatan?

.....

.....
.....

2.5.6. Uji Kompetensi/Ulangan

1. Instruksi IF digunakan untuk...
 - a. Memeriksa suatu kondisi dan melaksanakan instruksi lain jika kondisi tersebut terpenuhi atau bernilai false
 - b. Memeriksa suatu kondisi dan melaksanakan instruksi lain jika kondisi tersebut terpenuhi atau bernilai true
 - c. Melaksanakan proses yang berbeda untuk kondisi yang berbeda
 - d. Melaksanakan proses yang berbeda untuk kondisi yang sama
 - e. Melaksanakan proses yang sama untuk kondisi yang berbeda
2. Statement pada sintaks if boleh berupa satu instruksi tunggal atau beberapa instruksi (block statement) yang ditulis dalam...
 - a. ()
 - b. <>
 - c. ><
 - d. {}
 - e. []
3. Yang bukan merupakan Struktur Branching (Percabangan), yaitu...
 - a. IF
 - b. IF Else
 - c. IF TRUE
 - d. Nested IF
 - e. Struktur IF bersarang

Bab 3 – Mode dan format pengalamatan

3.1 Kegiatan belajar 1 – Inherent

3.1.1. Tujuan Pembelajaran

Setelah mengikuti pembelajaran, siswa mampu :

- Memahami mode dan format pengalamatan inherent

3.1.2. Aktivitas belajar siswa

3.1.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah tabel berikut ini :

Instruksi	Mnemonic
Arithmetic Shift Left Arithmetic Shift Right Clear Carry Bit Clear Interrupt Mask Bit	ASLA, ASLX ASRA, ASRX CLC CLI
Clear Complement (invert all bits) Decrement Increment	CLRA, CLRX COMA, COMX DECA, DECX INCA, INCX
Logical Shift Left Logical Shift Right Multiply Negate (two's complement)	LSLA, LSLX LSRA, LSRX MUL NEGA, NEGX
No Operation Rotate Left thru Carry Rotate Right thru Carry Reset Stack Pointer	NOP ROLA, ROLX RORA, RORX RSP
Return from Interrupt Return from Subroutine Set Carry Bit Set Interrupt Mask Bit	RTI RTS SEC SEI
Enable IRQ, STOP Oscillator Software Interrupt Transfer Accumulator to Index Register Test for Negative Zero	STOP SWI TAX TSTA, TSTX
Transfer Index Register to Accumulator Wait for Interrupt	TXA WAIT

3.1.2.2. Menanya

Buatlah pertanyaan kepada gurumu tentang mode dan format pengalamatan inherent.

3.1.2.3. Mencoba/ Mengumpulkan informasi

Dalam mode pengalamatan *inherent*, semua informasi yang dibutuhkan untuk operasi telah diketahui otomatis oleh CPU, dan tidak dibutuhkan operan eksternal dari memori atau dari program. Operan yang digunakan hanyalah register internal dari CPU atau data dalam *stack*. Karena itu operasi ini hanyalah terdiri dari satu byte instruksi. Contoh: 0200 4C INCA ; increment akumulator

Pertama CPU membaca kode operasi \$4C yang menginstruksikan CPU untuk menambah harga dari isi akumulator. Kemudian CPU akan menyimpan harga baru ke dalam akumulator dan mengeset *bitflag* jika dibutuhkan. Tabel 5.2 menampilkan semua instruksi M68HC05 yang dapat menggunakan mode pengalamatan *inherent*.

Tabel 3.1. Instruksi-instruksi Mode Pengalamatan INHERENT

Instruksi	Mnemonic
Arithmetic Shift Left	ASLA, ASLX
Arithmetic Shift Right	ASRA, ASRX
Clear Carry Bit	CLC
Clear Interrupt Mask Bit	CLI
Clear	CLRA, CLRX
Complement (invert all bits)	COMA, COMX
Decrement	DECA, DECX
Increment	INCA, INCX
Logical Shift Left	LSLA, LSLX
Logical Shift Right	LSRA, LSRX
Multiply	MUL
Negate (two's complement)	NEGA, NEGX
No Operation	NOP
Rotate Left thru Carry	ROLA, ROLX
Rotate Right thru Carry	RORA, RORX
Reset Stack Pointer	RSP
Return from Interrupt	RTI
Return from Subroutine	RTS
Set Carry Bit	SEC
Set Interrupt Mask Bit	SEI
Enable IRQ, STOP Oscillator	STOP
Software Interrupt	SWI
Transfer Accumulator to Index Register	TAX
Test for Negative Zero	TSTA, TSTX
Transfer Index Register to Accumulator	TXA

Wait for Interrupt	WAIT
--------------------	------

3.1.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang mode pengalamatan inherent pada tabel berikut ini :

Instruksi	Penjelasan

Buatlah kesimpulan tentang mode pengalamatan inherent.

3.1.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri mengenai mode pengalamatan inherent.

3.1.3. Rangkuman

Dalam mode pengalamatan *inherent*, semua informasi yang dibutuhkan untuk operasi telah diketahui otomatis oleh CPU, dan tidak dibutuhkan operan eksternal dari memori atau dari program. Karena itu operasi ini hanyalah terdiri dari satu byte instruksi. Contoh: 0200 4C INCA ; increment akumulator.

Pertama CPU membaca kode operasi \$4C, kemudian CPU akan menyimpan harga baru ke dalam akumulator dan mengeset bit *flag* jika dibutuhkan.

Instruksi-instruksi Mode Pengalamatan *inherent*, antara lain: ASLA, ASRA, CLC, CLI; CLRA, COMA, DECA, INCA; LSLA, LSRA, MUL, NEGA; NOP, ROLA, RORA, RSP; RTI, RTS, SEC, SEI; STOP, SWI, TAX, TSTA; TXA, WAIT.

3.1.4. Tugas

1. Jelaskan yang dimaksud mode pengalamatan *inherent*?

.....

.....

.....

.....

.....

.....

2. Pada operasi *inherent* terdapat berapa instruksi? Sebutkan!

-

 3. Jelaskan instruksi-instruksi Mode Pengalamatan *inherent*!

-

 4. Mnemonic dari instruksi Transfer Index Register to Accumulator adalah?

-

 5. *Logical Shift Right* adalah?

3.1.5. Penilaian diri

Nama :
 Nama-nama anggota kelompok :
 Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....
 - Mendengarkan orang lain
 - Mengajukan pertanyaan

- Mengorganisasi ide-ide saya
 - Mengorganisasi kelompok
 - Mengacaukan kegiatan
 - Melamun
6. Apa yang kamu lakukan selama kegiatan?

.....

.....

.....

.....

.....

.....

3.1.6. Uji Kompetensi/Ulangan

1. Mnemonic dari instruksi Clear Carry Bit, yaitu...
 - a. CCB
 - b. CLB
 - c. CLC
 - d. CLE
 - e. CLI
2. Mnemonic dari instruksi Transfer Index Register to Accumulator, yaitu...
 - a. TAX
 - b. TSTX
 - c. TIRA
 - d. TIR
 - e. TXA
3. Mnemonic dari instruksi Software Interrupt, yaitu...
 - a. SWI
 - b. STI
 - c. SFW
 - d. SEI
 - e. SEC
4. ROLA, ROLX merupakan mnemonic dari instruksi...
 - a. Rotate Right thru Carry
 - b. Return from Interrupt
 - c. Rotate Left thru Carry
 - d. Reset Stack Pointer
 - e. Return from Subroutine
5. LSRA, LSRX merupakan mnemonic dari instruksi...
 - a. Logical Shift Left
 - b. Lexical Shift Right
 - c. Lexical Shift Left
 - d. Logaritma Shift Right
 - e. Logical Shift Right

3.2 Kegiatan belajar 2 – *Immediate*

3.2.1. Tujuan Pembelajaran

Setelah mengikuti pembelajaran, siswa mampu :

- Memahami mode dan format pengalamatan *Immediate*

3.2.2. Aktivitas belajar siswa

3.2.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah tabel berikut ini

Instruksi	Mnemonic
Add with Carry	ADC
Add (without carry)	ADD
Logical AND	AND
Bit Test Memory with Accumulator	BIT
Compare Accumulator with Memory	CMP
Compare Index Register with Memory	CPX
Exclusive OR Memory with Accumulator	EOR
Load Accumulator from Memory	LDA
Load Index Register from Memory	LDX
Inclusive OR	ORA
Substract with Carry	SBC
Substract (without borrow)	SUB

3.2.2.2. Menanya

Buatlah pertanyaan kepada gurumu tentang mode dan format pengalamatan *Immediate*.

3.2.2.3. Mencoba/ Mengumpulkan informasi

Dalam mode pengalamatan *immediate*, operan terkandung di dalam byte yang langsung mengikuti kode operasi. Mode ini digunakan saat suatu harga atau konstanta diketahui saat program dibuat dan tidak akan dirubah selama eksekusi program. Operasi dengan mode ini membutuhkan dua byte instruksi, satu untuk kode operasi dan satu lagi untuk data byte. Contoh: 0200 A6 02 LDA #\$02 ; Load konstanta ke akumulator

Pertama CPU akan membaca kode operasi \$A6 yang menginstruksikan CPU untuk me-*load* akumulator dengan nilai *immediate* yang mengikuti kode operasi. Kemudian CPU akan membaca data *immediate* \$02 dari lokasi memori dengan

alamat \$0201 ke dalam akumulator. Tabel 5.1 menampilkan semua instruksi keluarga M68HC05 yang dapat menggunakan mode pengalamatan *immediate* ini.

Tabel 3.2. Instruksi-instruksi Mode Pengalamatan IMMEDIATE

Instruksi	Mnemonic
Add with Carry	ADC
Add (without carry)	ADD
Logical AND	AND
Bit Test Memory with Accumulator	BIT
Compare Accumulator with Memory	CMP
Compare Index Register with Memory	CPX
Exclusive OR Memory with Accumulator	EOR
Load Accumulator from Memory	LDA
Load Index Register from Memory	LDX
Inclusive OR	ORA
Substract with Carry	SBC
Substract (without borrow)	SUB

3.2.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang mode dan format pengalamatan immediate pada tabel berikut ini :

Instruksi	Penjelasan

Buatlah kesimpulan tentang mode dan format pengalamatan immediate.

3.2.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri tentang mode dan format pengalamatan immediate.

3.2.3. Rangkuman

Dalam mode pengalamatan *immediate*, operan terkandung di dalam byte yang langsung mengikuti kode operasi. Operasi dengan mode ini membutuhkan dua byte instruksi, satu untuk kode operasi dan satu lagi untuk data byte. Contoh: 0200 A6 02 LDA #\$02 ; Load konstanta ke akumulator.

Pertama CPU akan membaca kode operasi \$A6, kemudian CPU akan membaca data *immediate* \$02 dari lokasi memori dengan alamat \$0201 ke dalam akumulator.

Instruksi-instruksi Mode Pengalamatan *immediate*, antara lain: ADC, ADD, AND, BIT; CMP, CPX, EOR, LDA; LDX, ORA, SBC, SUB.

3.2.4. Tugas

1. Mode pengalamatan *immediate* adalah?

.....

.....

.....

.....

.....

2. Operasi dengan mode *immediate* membutuhkan berapa byte instruksi?

.....

.....

.....

.....

.....

3. Jelaskan instruksi-instruksi Mode Pengalamatan *immediate*!

.....

.....

.....

.....

.....

4. Mnemonic dari instruksi Subtract (without borrow), yaitu?

.....

.....

.....

.....

.....

5. Mnemonic dari instruksi Exclusive OR Memory with Accumulator, yaitu?

.....

.....

.....

.....

.....

3.2.5. Penilaian diri

Nama :

Nama-nama anggota kelompok :

Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....
 - Mendengarkan orang lain
 - Mengajukan pertanyaan
 - Mengorganisasi ide-ide saya
 - Mengorganisasi kelompok
 - Mengacaukan kegiatan
 - Melamun
6. Apa yang kamu lakukan selama kegiatan?

.....

.....

.....

.....

.....

.....

3.2.6. Uji Kompetensi/Ulangan

1. Mode yang digunakan saat suatu harga atau konstanta diketahui saat program dibuat dan tidak akan dirubah selama eksekusi program adalah mode pengalamatan...
 - a. Extended
 - b. Direct
 - c. Immediate
 - d. Indexed
 - e. Inherent
2. Operasi dengan mode pengalamatan *immediate* membutuhkan dua byte instruksi, satu untuk ... dan satu lagi untuk ...
 - a. Kode operasi; data byte
 - b. Kode operasi; data base
 - c. Kode instruksi; data base
 - d. Kode instruksi; data byte
 - e. Kode pengalamatan; kode operasi
3. AND merupakan mnemonic dari instruksi...
 - a. Logical AND

- b. Inclusive AND
 - c. Add AND
 - d. Load AND
 - e. Compare AND
4. Mnemonic dari instruksi Exclusive OR Memory with Accumulator, yaitu...
- a. ORA
 - b. EOR
 - c. EMA
 - d. EOA
 - e. EOM
5. Mnemonic dari instruksi Substract (without borrow), yaitu...
- a. SBC
 - b. SUB
 - c. SWB
 - d. SBT
 - e. SBO

3.3 Kegiatan belajar 3 – Direction

3.3.1. Tujuan Pembelajaran

Setelah mengikuti pembelajaran, siswa mampu :

- Memahami mode dan format pengalamatan direction

3.3.2. Aktivitas belajar siswa

3.3.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah tabel berikut ini.

Instruksi	Mnemonic
Add with Carry	ADC
Add (without carry)	ADD
Logical AND	AND
Arithmetic Shift Left	ASL
Arithmetic Shift Right	ASR
Clear Bit in Memory	BCLR
Bit Test Memory with Accumulator	BIT
Branch if Bit n is Clear	BRCLR
Branch if Bit n is Set	BRSET
Set Bit in Memory	BSET
Clear	CLR
Compare Accumulator with Memory	CMP
Complement (invert all bits)	COM
Compare Index Register with Memory	CPX
Decrement	DEC
Exclusive OR Memory with Accumulator	EOR
Increment	INC
Jump	JMP
Jump to Subroutine	JSR
Load Accumulator from Memory	LDA
Load Index Register from Memory	LDX
Logical Shift Left	LSL
Logical Shift Right	LSR
Negate (two's complement)	NEG
Inclusive OR	ORA
Rotate Left thru Carry	ROL
Rotate Right thru Carry	ROR
Substract with Carry	SBC
Store Acuumulator in Memory	STA
Strore Index Register in Memory	STX

Subtract (without borrow)	SUB
Test for Negative or Zero	TST

3.3.2.2. Menanya

Buatlah pertanyaan kepada gurumu tentang mode dan format pengalamatan direction.

3.3.2.3. Mencoba/ Mengumpulkan informasi

Mode pengalamatan *direct* mirip dengan mode pengalamatan *extended* kecuali bahwa *upper byte* dari alamat operan selalu dianggap \$00. Karena itu, hanya *lower-byte* dari operan yang diperlukan untuk dimasukkan dalam instruksi. Pengalamatan *direct* menyebabkan efisiensi alamat dalam 256 byte pertama dalam memori. Area ini dinamakan dengan *direct page* dan mengandung *on-chip* RAM dan register I/O. Pengalamatan *direct* ini efisien bagi memori program dan waktu eksekusi. Dalam mode ini instruksi terdiri dari dua byte, satu untuk kode operasi dan satu lagi untuk alamat operan. Contoh: 0200 B6 E0 LDA \$E0

Pertama CPU akan membaca kode operasi \$B6 yang menginstruksikan CPU untuk menggunakan mode pengalamatan *direct*. Kemudian CPU membaca \$E0 dari lokasi memori dengan alamat \$0201. Harga \$E0 ini diterjemahkan sebagai *low-order* dari alamat dalam *direct page* (\$0000 sampai \$00FF). Setelah itu CPU menyusun alamat lengkap \$00E0 dengan menganggap *high-order* byte alamat sebagai \$00. Alamat lengkap \$00E0 ini kemudian diletakkan dalam bus alamat dan kemudian proses pembacaan data pun dikerjakan. Tabel 5.4 menampilkan semua instruksi yang bisa dikerjakan dengan mode pengalamatan *direct* ini.

Tabel 3.3. Instruksi-instruksi Mode Pengalamatan DIRECT

Instruksi	Mnemonic
Add with Carry	ADC
Add (without carry)	ADD
Logical AND	AND
Arithmetic Shift Left	ASL
Arithmetic Shift Right	ASR
Clear Bit in Memory	BCLR
Bit Test Memory with Accumulator	BIT
Branch if Bit n is Clear	BRCLR
Branch if Bit n is Set	BRSET
Set Bit in Memory	BSET
Clear	CLR
Compare Accumulator with Memory	CMP
Complement (invert all bits)	COM
Compare Index Register with Memory	CPX

Decrement	DEC
Exclusive OR Memory with Accumulator	EOR
Increment	INC
Jump	JMP
Jump to Subroutine	JSR
Load Accumulator from Memory	LDA
Load Index Register from Memory	LDX
Logical Shift Left	LSL
Logical Shift Right	LSR
Negate (two's complement)	NEG
Inclusive OR	ORA
Rotate Left thru Carry	ROL
Rotate Right thru Carry	ROR
Subtract with Carry	SBC
Store Accumulator in Memory	STA
Store Index Register in Memory	STX
Subtract (without borrow)	SUB
Test for Negative or Zero	TST

3.3.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang mode dan format pengalamatan direction pada tabel berikut ini :

Instruksi	Penjelasan

Buatlah kesimpulan tentang mode dan format pengalamatan direction.

3.3.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri tentang mode dan format pengalamatan direction.

3.3.3. Rangkuman

Mode pengalamatan *direct* menyebabkan efisiensi alamat dalam 256 byte pertama dalam memori. Pengalamatan direct ini efisien bagi memori program dan

waktu eksekusi. Dalam mode ini instruksi terdiri dari dua byte, satu untuk kode operasi dan satu lagi untuk alamat operan. Contoh: 0200 B6 E0 LDA \$E0

Pertama CPU akan membaca kode operasi \$B6, kemudian CPU membaca \$E0 dari lokasi memori dengan alamat \$0201. Harga \$E0 ini diterjemahkan sebagai *low-order* dari alamat dalam *direct page* (\$0000 sampai \$00FF). Setelah itu CPU menyusun alamat lengkap \$00E0 dengan menganggap *high-order* byte alamat sebagai \$00. Alamat lengkap \$00E0 ini kemudian diletakkan dalam bus alamat dan diproses.

Instruksi-instruksi Mode Pengalamatan *direct*, antara lain: ADC, ADD, AND, ASL; ASR, BCLR, BIT, BRCLR; BRSET, BSET, CLR, CMP; COM, CPX, DEC, EOR; INC, JMP, JSR, LDA; LDX, LSL, LSR, NEG; ORA, ROL, ROR, SBC; STA, STX, SUB, TST.

3.3.4 Tugas

1. Mode pengalamatan *direct* adalah?

.....

.....

.....

.....

.....

2. Jelaskan instruksi dari mode *direct*!

.....

.....

.....

.....

.....

3. Mode pengalamatan *direct* dapat menyebabkan?

.....

.....

.....

.....

.....

4. Dalam mode pengalamatan *Direct*, instruksi terdiri dari dua byte, satu untuk ... dan satu lagi untuk?

.....

.....

.....

.....

.....

5. ADC merupakan mnemonic dari instruksi?

.....

.....

.....

3.3.5. Penilaian diri

Nama :
 Nama-nama anggota kelompok :
 Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....
 - Mendengarkan orang lain
 - Mengajukan pertanyaan
 - Mengorganisasi ide-ide saya
 - Mengorganisasi kelompok
 - Mengacaukan kegiatan
 - Melamun
6. Apa yang kamu lakukan selama kegiatan?

.....

3.3.6. Uji Kompetensi/Ulangan

1. *Upper byte* dari alamat operan pada mode pengalamatan *Direct* selalu dianggap...
 - a. \$00
 - b. \$00FF
 - c. #00
 - d. \$E0
 - e. \$00E0
2. Mnemonic dari instruksi Test for Negative or Zero, yaitu...

- a. TXT
 - b. TST
 - c. TNZ
 - d. TSZ
 - e. TNOR
3. BRCLR merupakan mnemonic dari instruksi...
- a. Clear Bit in Memory
 - b. Bit Test Memory with Accumulator
 - c. Branch if Bit n is Clear
 - d. Branch if Bit n is Set
 - e. Set Bit in Memory
4. Dalam mode pengalamatan *Direct*, instruksi terdiri dari dua byte, satu untuk ... dan satu lagi untuk...
- a. Kode petunjuk; alamat instruksi
 - b. Kode instruksi; alamat operan
 - c. Kode operasi; alamat instruksi
 - d. Kode operasi; alamat operan
 - e. Kode operan; alamat instruksi
5. ADC merupakan mnemonic dari instruksi...
- a. Add (without Carry)
 - b. Logical add
 - c. Load Index Register from Memory
 - d. Load Accumulator from Memory
 - e. Add with Carry

3.4 Kegiatan belajar 4 – Extended

3.4.1. Tujuan Pembelajaran

Setelah mengikuti pembelajaran, siswa mampu :

- Memahami mode dan format pengalamatan extended

3.4.2. Aktivitas belajar siswa

3.4.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah tabel berikut ini :

Instruksi	Mnemonic
Add with Carry	ADC
Add (without carry)	ADD
Logical AND	AND
Bit Test Memory with Accumulator	BIT
Compare Accumulator with Memory	CMP
Compare Index Register with Memory	CPX
Exclusive OR Memory with Accumulator	EOR
Jump	JMP
Jump to Subroutine	JSR
Load Accumulator from Memory	LDA
Exclusive OR Memory with Accumulator	LDX
Jump	ORA
Substract with Carry	SBC
Store Accumulator in Memory	STA
Store Index Register in Memory	STX
Substract (without borrow)	SUB

3.4.2.2. Menanya

Buatlah pertanyaan kepada gurumu tentang mode dan format pengalamatan extended.

3.4.2.3. Mencoba/ Mengumpulkan informasi

Dalam mode pengalamatan *extended*, alamat dari operan terkandung dalam dua byte yang mengikuti kode operasi. Pengalamatan extended ini dapat digunakan untuk mengakses semua lokasi dalam memori mikrokontroler termasuk I/O, RAM,

ROM, dan EPROM. Karena itu operasi ini membutuhkan tiga byte, satu untuk kode operasi, dan dua untuk alamat dari operan. Contoh : 0200 C6 03 65 LDA \$0365

Pertama CPU akan membaca kode operasi C6 yang menginstruksikan akumulator untuk menggunakan mode pengalamatan extended. Kemudian CPU akan membaca \$03 dari lokasi memori \$0201 dan diterjemahkan sebagai alamat *high-order*. Setelah itu CPU membaca \$65 dari lokasi \$0202 dan diterjemahkan sebagai *low-order* dari alamat. Terakhir, CPU membangun alamat lengkap \$0365 dan meletakkannya dalam bus alamat dan kemudian melakukan operasi read sehingga isi memori dengan alamat \$0365 akan tersedia dalam bus data. Tabel 5.3 di bawah ini akan memperlihatkan semua instruksi yang dapat digunakan dalam mode pengalamatan extended.

Tabel 3.4. Instruksi-instruksi Mode Pengalamatan EXTENDED

Instruksi	Mnemonic
Add with Carry	ADC
Add (without carry)	ADD
Logical AND	AND
Bit Test Memory with Accumulator	BIT
Compare Accumulator with Memory	CMP
Compare Index Register with Memory	CPX
Exclusive OR Memory with Accumulator	EOR
Jump	JMP
Jump to Subroutine	JSR
Load Accumulator from Memory	LDA
Exclusive OR Memory with Accumulator	LDX
Jump	ORA
Subtract with Carry	SBC
Store Accumulator in Memory	STA
Store Index Register in Memory	STX
Subtract (without borrow)	SUB

3.4.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang mode dan format pengalamatan extended pada tabel berikut ini :

Instruksi	Penjelasan

Buatlah kesimpulan tentang mode dan format pengalamatan extended.

3.4.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri tentang mode dan format pengalamatan *extended*..

3.4.3. Rangkuman

Pengalamatan *extended* ini dapat digunakan untuk mengakses semua lokasi dalam memori mikrokontroler termasuk I/O, RAM, ROM, dan EPROM. Karena itu operasi ini membutuhkan tiga byte, satu untuk kode operasi, dan dua untuk alamat dari operan. Contoh : 0200 C6 03 65 LDA \$0365

Pertama CPU akan membaca kode operasi C6, kemudian CPU akan membaca \$03 dari lokasi memori \$0201 dan diterjemahkan sebagai alamat *high-order*. Setelah itu CPU membaca \$65 dari lokasi \$0202 dan diterjemahkan sebagai *low-order* dari alamat. Terakhir, CPU membangun alamat lengkap \$0365 dan meletakkannya dalam bus alamat dan kemudian melakukan operasi read.

Instruksi-instruksi Mode Pengalamatan *extended*, antara lain: ADC, ADD, AND, BIT; CMP, CPX, EOR, JMP; JSR, LDA, LDX, ORA; SBC, STA, STX, SUB.

3.4.4. Tugas

1. Mode pengalamatan *extended* adalah?

.....

.....

.....

.....

.....

2. Pengalamatan *extended* ini dapat digunakan untuk apa?

.....

.....

.....

.....

.....

3. Jelaskan Instruksi-instruksi Mode Pengalamatan *extended*!

.....

.....

.....

.....

.....

4. JSR merupakan mnemonic dari instruksi?

.....

.....

.....

-

 5. Yang bukan merupakan mnemonic dari instruksi-instruksi mode pengalamatan *extended*, yaitu?

.....

3.4.5. Penilaian diri

Nama :
 Nama-nama anggota kelompok :
 Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....
 - Mendengarkan orang lain
 - Mengajukan pertanyaan
 - Mengorganisasi ide-ide saya
 - Mengorganisasi kelompok
 - Mengacaukan kegiatan
 - Melamun
6. Apa yang kamu lakukan selama kegiatan?

.....

3.4.6. Uji Kompetensi/Ulangan

1. Mode pengalamatan *extended* ini dapat digunakan untuk mengakses semua lokasi dalam memori mikrokontroler di bawah ini, kecuali...
 - a. I/O
 - b. RAM
 - c. EPROM
 - d. ROM
 - e. EPROM
2. Mnemonic dari instruksi Compare Accumulator with Memory, yaitu...
 - a. CPM
 - b. CMP
 - c. CPX
 - d. CAM
 - e. CPA
3. JSR merupakan mnemonic dari instruksi...
 - a. Jump to Subroutine
 - b. Jump
 - c. Jump to Subtract
 - d. Subtract with Carry
 - e. Store Accumulator in Memory
4. Yang bukan merupakan mnemonic dari instruksi-instruksi mode pengalamatan *extended*, yaitu...
 - a. EOR
 - b. LDX
 - c. LDA
 - d. ADC
 - e. DEC
5. Di bawah ini merupakan instruksi-instruksi mode pengalamatan *extended*, kecuali...
 - a. Jump
 - b. Store Index Register in Memory
 - c. Store Accumulator Register in Memory
 - d. Negate (two's complement)
 - e. Load Accumulator from Memory

3.5 Kegiatan belajar 5 – Indexed

3.5.1. Tujuan Pembelajaran

Setelah mengikuti pembelajaran, siswa mampu :

- Memahami Mode dan pengalamatan indexed

3.5.2. Aktivitas belajar siswa

3.5.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah tabel berikut ini.

Instruksi	Mnemonic
Add with Carry	ADC
Add (without carry)	ADD
Logical AND	AND
Bit Test Memory with Accumulator	BIT
Compare Accumulator with Memory	CMP
Compare Index Register with Memory	CPX
Exclusive OR Memory with Accumulator	EOR
Jump	JMP
Jump to Subroutine	JSR
Load Accumulator from Memory	LDA
Exclusive OR Memory with Accumulator	LDX
Jump	ORA
Substract with Carry	SBC
Store Accumulator in Memory	STA
Store Index Register in Memory	STX
Substract (without borrow)	SUB

3.5.2.2. Menanya

Buatlah pertanyaan kepada gurumu tentang Mode dan pengalamatan indexed.

3.5.2.3. Mencoba/ Mengumpulkan informasi

Dalam mode pengalamatan *indexed*, alamat efektif adalah variabel dan tergantung pada dua faktor: 1) isi index register saat itu dan 2) nilai offset yang terkandung dari byte yang mengikuti kode operasi. Terdapat tiga jenis pengalamatan indexed yang didukung oleh CPU keluarga M68HC05, yaitu: no-offset, 8-bit offset, dan 16-bit offset. Dalam mode pengalamatan ***indexed-no offset***, alamat efektif dari operan terkandung dalam index register 8-bit. Karena itu, mode pengalamatan ini

dapat mengakses 256 lokasi memori (dari \$0000 sampai \$00FF). Instruksi mode ini membutuhkan satu byte instruksi. Contoh: 0200 F6 LDA 0,X

Pertama CPU akan membaca kode operasi \$F6 yang menginstruksikan CPU untuk menggunakan mode pengalamatan *indexed-no offset*. Kemudian CPU menyusun alamat lengkap dengan menjumlahkan \$0000 ke isi dari index register 8-bit (X). Alamat ini kemudian diletakkan dalam bus alamat dan setelah itu dilakukan proses pembacaan data. Tabel 5.5 menampilkan semua instruksi yang dapat menggunakan mode pengalamatan ini.

Tabel 3.5. INDEXED-NO OFFSET dan INDEXED-8 BIT OFFSET

Instruksi	Mnemonic
Add with Carry	ADC
Add (without carry)	ADD
Logical AND	AND
Arithmetic Shift Left	ASL
Arithmetic Shift Right	ASR
Bit Test Memory with Accumulator	BIT
Clear	CLR
Compare Accumulator with Memory	CMP
Complement (invert all bits)	COM
Compare Index Register with Memory	CPX
Decrement	DEC
Exclusive OR Memory with Accumulator	EOR
Increment	INC
Jump	JMP
Jump to Subroutine	JSR
Load Accumulator from Memory	LDA
Load Index Register from Memory	LDX
Logical Shift Left	LSL
Logical Shift Right	LSR
Negate (two's complement)	NEG
Inclusive OR	ORA
Rotate Left thru Carry	ROL
Rotate Right thru Carry	ROR
Substract with Carry	SBC
Store Acuumulator in Memory	STA
Strore Index Register in Memory	STX
Substract (without borrow)	SUB
Test for Negative or Zero	TST

Dalam mode pengalamatan *indexed-8 bit offset*, alamat efektif dicapai dengan menambahkan data byte yang mengikuti kode operasi dengan isi dari index register. Harga byte offset yang disediakan dalam instruksi adalah integer tak bertanda 8-bit. Karena itu operasi ini adalah dua byte instruksi di mana offset

terkandung dalam byte yang mengikuti kode operasi. Isi dari index register tidak akan dirubah. Contoh: 0200 E6 05 LDA 5,X

Pertama CPU akan membaca kode operasi \$E6 yang menginstruksikan CPU menggunakan mode pengalamatan *indexed 8-bit offset*. Kemudian CPU akan membaca 8-bit offset (\$05) dari alamat \$0201. CPU akan menyusun alamat lengkap dengan menambahkan nilai offset \$05 ke isi dari index register 8-bit (X). Setelah itu, alamat diletakkan dalam bus alamat dan operasi pembacaan data ke akumulator dikerjakan. Tabel 5.5 menunjukkan semua instruksi yang mampu menggunakan mode pengalamatan ini. Dalam mode pengalamatan ***indexed-16 bit offset***, alamat efektif dari operan suatu instruksi adalah hasil penjumlahan antara isi dari index register 8-bit dengan dua byte alamat yang mengikuti kode operasi. Isi dari kode operasi tidak dirubah. Instruksi ini memakan tiga byte, satu untuk kode operasi dan dua byte untuk offset 16-bit. Contoh: 0200 D6 03 77 LDA \$377,X

Pertama CPU akan membaca kode operasi \$D6 yang menginstruksikan CPU untuk menggunakan mode pengalamatan *indexed -16 bit offset*. Kemudian CPU akan membaca *high order* alamat (\$03) dari alamat memori \$0201 dan membaca *low order* alamat (\$77) dari alamat \$0202. CPU akan menyusun alamat lengkap dengan menambahkan isi dari index register 8-bit dengan 16-bit alamat yang baru saja dibaca. Alamat ini diletakkan dalam bus alamat dan kemudian operasi pembacaan data dilakukan.

Tabel 3.6. Instruksi-instruksi Mode Pengalamatan INDEXED-16 BIT OFFSET

Instruksi	Mnemonic
Add with Carry	ADC
Add (without carry)	ADD
Logical AND	AND
Bit Test Memory with Accumulator	BIT
Compare Accumulator with Memory	CMP
Compare Index Register with Memory	CPX
Exclusive OR Memory with Accumulator	EOR
Jump	JMP
Jump to Subroutine	JSR
Load Accumulator from Memory	LDA
Exclusive OR Memory with Accumulator	LDX
Jump	ORA
Substract with Carry	SBC
Store Accumulator in Memory	STA
Store Index Register in Memory	STX
Substract (without borrow)	SUB

3.5.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang mode dan format pengalamatan Indexed pada tabel berikut ini :

Instruksi	Penjelasan

Buatlah kesimpulan tentang mode dan format pengalamatan Indexed.

3.5.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri tentang mode dan format pengalamatan Indexed.

3.5.3. Rangkuman

Terdapat tiga jenis pengalamatan indexed yang didukung oleh CPU keluarga M68HC05, yaitu:

1. Mode pengalamatan ***indexed-no offset***. Dalam mode pengalamatan ini, alamat efektif dari operan terkandung dalam index register 8-bit. Karena itu, mode pengalamatan ini dapat mengakses 256 lokasi memori (dari \$0000 sampai \$00FF). Instruksi mode ini membutuhkan satu byte instruksi. Contoh: 0200 F6 LDA 0,X

Pertama CPU akan membaca kode operasi \$F6, kemudian CPU menyusun alamat lengkap dengan menjumlahkan \$0000 ke isi dari index register 8-bit (X). Alamat ini kemudian diletakkan dalam bus alamat dan setelah itu dilakukan proses pembacaan data.

Indexed-no offset dan Indexed-8 bit offset, antara lain: ADC, ADD, AND, ASL; ASR, BIT, CLR, CMP; COM, CPX, DEC, EOR; INC, JMP, JSR, LDA; LDX, LSL, LSR, NEG; ORA, ROL, ROR, SBC; STA, STX, SUB, TST.

2. Mode pengalamatan ***indexed-8 bit offset***. Dalam mode pengalamatan ini, alamat efektif dicapai dengan menambahkan data byte yang mengikuti kode operasi dengan isi dari index register. Operasi ini adalah dua byte instruksi di mana offset terkandung dalam byte yang mengikuti kode operasi. Isi dari index register tidak akan dirubah. Contoh: 0200 E6 05 LDA 5,X

Pertama CPU akan membaca kode operasi \$E6, kemudian CPU akan membaca 8-bit offset (\$05) dari alamat \$0201, lalu menyusun alamat lengkap dengan menambahkan nilai offset \$05 ke isi dari index register 8-bit (X). Setelah itu, alamat diletakkan dalam bus alamat dan diproses.

3. Mode pengalamatan ***indexed-16 bit offset***. Dalam mode pengalamatan ini, alamat efektif dari operan suatu instruksi adalah hasil penjumlahan antara isi dari index register 8-bit dengan dua byte alamat yang mengikuti kode operasi. Instruksi ini memakan tiga byte, satu untuk kode operasi dan dua byte untuk offset 16-bit. Contoh: 0200 D6 03 77 LDA \$377,X

Pertama CPU akan membaca kode operasi \$D6, kemudian CPU akan membaca *high order* alamat (\$03) dari alamat memori \$0201 dan membaca *low order* alamat (\$77) dari alamat \$0202. CPU akan menyusun alamat lengkap dengan menambahkan isi dari index register 8-bit dengan 16-bit alamat yang baru saja dibaca. Alamat ini diletakkan dalam bus alamat dan diproses.

Instruksi-instruksi Mode Pengalamatan ***indexed-16 bit offset***, antara lain: ADC, ADD, AND, BIT; CMP, CPX, EOR, JMP; JSR, LDA, LDX, ORA; SBC, STA, STX, SUB.

3.5.4. Tugas

1. Mode pengalamatan ***indexed-no offset*** adalah?
.....
.....
.....
.....
.....
2. Mode pengalamatan ***indexed-8 bit offset*** adalah!
.....
.....
.....
.....
.....
3. Terdapat tiga jenis pengalamatan indexed yang didukung oleh CPU keluarga M68HC05, yaitu?
.....
.....
.....
.....
.....
4. Instruksi indexed-no offset dan Indexed-8 bit offset adalah?
.....
.....
.....
.....
.....
5. Instruksi-instruksi Mode Pengalamatan ***indexed-16 bit offset*** adalah?
.....
.....
.....
.....
.....

3.5.5. Penilaian diri

Nama :
 Nama-nama anggota kelompok :
 Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.
 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....
 - Mendengarkan orang lain
 - Mengajukan pertanyaan
 - Mengorganisasi ide-ide saya
 - Mengorganisasi kelompok
 - Mengacaukan kegiatan
 - Melamun
6. Apa yang kamu lakukan selama kegiatan?

.....

3.5.6. Uji Kompetensi/Ulangan

1. Alamat efektif terdiri dari dua factor, antara lain...
 - a. Isi index register; nilai onset byte
 - b. Letak register; isi index
 - c. Isi index register; nilai offset byte
 - d. Letak register; nilai onset
 - e. Isi index register; nilai offset byte
2. Terdapat tiga jenis pengalamatan indexed yang didukung oleh CPU keluarga M68HC05, yaitu ..., ..., dan 16-bit offset.
 - a. 0-bit offset; 4-bit offset
 - b. No-offset; 8-bit offset
 - c. 0-bit offset; 8-bit offset
 - d. No-offset; 4-bit offset

- e. 4-bit offset; 8-bit offset
- 3. ROR merupakan mnemonic dari instruksi...
 - a. Rotate Right thru Carry
 - b. Register with Memory
 - c. Register OR
 - d. Register from Memory
 - e. Register in Memory
- 4. Mnemonic dari instruksi Store Accumulator in Memory pada 16-bit offset, yaitu...
 - a. SAM
 - b. SMO
 - c. STM
 - d. STA
 - e. SAC
- 5. Pada mode pengalamatan ***indexed-16 bit offset***, alamat efektif dari operan suatu instruksi adalah hasil penjumlahan antara isi dari index register ... bit dengan dua byte alamat yang mengikuti kode operasi.
 - a. 0
 - b. 2
 - c. 4
 - d. 6
 - e. 8

3.6 Kegiatan belajar 6 – Relative

3.6.1. Tujuan Pembelajaran

Setelah mengikuti pembelajaran, siswa mampu :

- Memahami Mode dan pengalamatan relative

3.6.2. Aktivitas belajar siswa

3.6.2.1. Mengamati/ observasi

1. Buatlah kelompok dengan anggota 4-5 orang,
2. Amatilah tabel berikut ini :

Instruksi	Mnemonic
Branch if Carry Clear	BCC
Branch if Carry Set	BCS
Branch if Equal	BEQ
Branch if Half-Carry Clear	BHCC
Branch if Half-Carry Set	BHCS
Branch if Higher	BHI
Branch if Higher or Same	BHS
Branch if Interrupt Line is High	BIH
Branch if Interrupt Line is Low	BIL
Branch if Lower	BLO
Branch if Lower or Same	BLS
Branch if Interrupt Mask is Clear	BMC
Branch if Minus	BMI
Branch if Interrupt Mask is Set	BMS
Branch if Not Equal	BNE
Branch if Plus	BPL
Branch Always	BRA
Branch if Bit n is Clear	BRCLR
Branch if Bit n is Set	BRSET
Branch Never	BRN
Branch to Subroutine	BSR

3.6.2.2. Menanya

Buatlah pertanyaan kepada gurumu tentang Mode dan pengalamatan relative.

3.6.2.3. Mencoba/ Mengumpulkan informasi

Mode pengalamatan *relative* ini digunakan hanya dalam instruksi percabangan. Instruksi percabangan, selain percabangan instruksi manipulasi bit, membangkitkan dua byte kode mesin: satu untuk kode operasi dan satu untuk offset relatifnya. Karena kemampuannya untuk bercabang ke dua arah, byte offset adalah bilangan bertanda dengan jangkauan -128 sampai $+127$. Jika kondisi percabangan TRUE, isi dari byte bertanda 8-bit yang mengikuti kode operasi akan ditambahkan dengan isi dari PC untuk membentuk alamat efektif percabangan; jika FALSE maka kontrol program akan terus ke instruksi di bawah instruksi percabangan. Programmer akan menspesifikasikan tujuan dari percabangan sebagai alamat *absolute* (dengan label atau alamat langsung). Kemudian *assembler* akan mengkalkulasi offset relatif 8-bit yang akan diletakkan di belakang kode memori dalam memori. Contoh: 0200 27 rr BEQ DEST

Pertama CPU akan membaca kode operasi \$27. Bit CCR Z akan set jika hasil dari operasi aritmatika atau logika sebelumnya adalah nol. CPU kemudian akan membaca harga offset \$rr dari alamat \$0201. Setelah siklus ini, PC akan menunjukkan ke byte pertama dari instruksi berikutnya (\$0202). Jika bit Z nol, maka tidak ada aksi apa-apa. Program akan terus ke instruksi berikutnya pada alamat \$0202. Jika bit Z satu, CPU akan menambahkan nilai offset \$rr ke isi dari PC sekarang untuk mendapatkan alamat tujuan dari percabangan. Hal ini menyebabkan eksekusi program berlanjut ke alamat baru (DEST). Tabel 5.7 menunjukkan instruksi yang dapat menggunakan mode pengalamatan *relative*.

Instruksi-instruksi Mode Pengalamatan RELATIVE

Instruksi	Mnemonic
Branch if Carry Clear	BCC
Branch if Carry Set	BCS
Branch if Equal	BEQ
Branch if Half-Carry Clear	BHCC
Branch if Half-Carry Set	BHCS
Branch if Higher	BHI
Branch if Higher or Same	BHS
Branch if Interrupt Line is High	BIH
Branch if Interrupt Line is Low	BIL
Branch if Lower	BLO
Branch if Lower or Same	BLS
Branch if Interrupt Mask is Clear	BMC
Branch if Minus	BMI
Branch if Interrupt Mask is Set	BMS
Branch if Not Equal	BNE
Branch if Plus	BPL
Branch Always	BRA
Branch if Bit n is Clear	BRCLR
Branch if Bit n is Set	BRSET
Branch Never	BRN
Branch to Subroutine	BSR

3.6.2.4. Mengasosiasi/ menalar

Tuliskan hasil pengamatanmu tentang mode dan format pengalamatan relative pada tabel berikut ini :

Instruksi	Penjelasan

Buatlah kesimpulan tentang mode dan format pengalamatan relative.

3.6.2.5. Mengkomunikasikan

Presentasikanlah hasil kerja kelompokmu didepan kelas dengan penuh rasa percaya diri tentang mode dan format pengalamatan relative.

3.6.3. Rangkuman

Mode pengalamatan *relative* ini digunakan hanya dalam instruksi percabangan. Karena kemampuannya untuk bercabang ke dua arah, byte offset adalah bilangan bertanda dengan jangkauan -128 sampai $+127$. Jika kondisi percabangan TRUE, isi dari byte bertanda 8-bit; jika FALSE maka kontrol program akan terus ke instruksi di bawah instruksi percabangan. Contoh: 0200 27 rr BEQ DEST

Pertama CPU akan membaca kode operasi \$27. Bit CCR Z akan set jika hasil dari operasi aritmatika atau logika sebelumnya adalah nol. CPU kemudian akan membaca harga offset \$rr dari alamat \$0201. Setelah siklus ini, PC akan menunjukkan ke byte pertama dari instruksi berikutnya (\$0202). Jika bit Z nol, maka tidak ada aksi apa-apa. Program akan terus ke instruksi berikutnya pada alamat \$0202. Jika bit Z satu, CPU akan menambahkan nilai offset \$rr ke isi dari PC sekarang untuk mendapatkan alamat tujuan dari percabangan. Hal ini menyebabkan eksekusi program berlanjut ke alamat baru (DEST).

Instruksi-instruksi Mode Pengalamatan relative, antara lain: BCC, BCS, BEQ, BHCC; BHCS, BHI, BHS, BIH; BIL, BLO, BLS, BMC; BMI, BMS, BNE, BPL; BRA, BRCLR, BRSET, BRN; BSR.

3.6.4. Tugas

1. Mode pengalamatan *relative* ini digunakan untuk?

.....

-
-
-
2. Jelaskan kemampuan mode pengalamatan *relative*!

-
-
-
-
-
3. Jika kondisi percabangan FALSE maka kontrol program akan?

-
-
-
-
-
4. BNE merupakan mnemonic dari instruksi?

-
-
-
-
-
5. Sebutkan instruksi-instruksi Mode Pengalamatan *relative*?

3.6.5. Penilaian diri

Nama :

Nama-nama anggota kelompok :

Kegiatan kelompok :

Isilah pernyataan berikut dengan jujur. Untuk No. 1 s.d. 4, isilah dengan cara melingkari jawaban dibawah pertanyaan.

1. Selama diskusi saya mengusulkan ide kepada kelompok untuk didiskusikan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
2. Ketika kami berdiskusi, tiap orang diberi kesempatan mengusulkan sesuatu.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
3. Semua anggota kelompok kami melakukan sesuatu selama kegiatan.
4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
4. Tiap orang sibuk dengan yang dilakukannya dalam kelompok saya.

- 4 : Selalu 3 : Sering 2 : Kadang-kadang 1 : Tidak pernah
5. Selama kerja kelompok, saya....
- Mendengarkan orang lain
 - Mengajukan pertanyaan
 - Mengorganisasi ide-ide saya
 - Mengorganisasi kelompok
 - Mengacaukan kegiatan
 - Melamun
6. Apa yang kamu lakukan selama kegiatan?

.....

.....

.....

.....

.....

.....

3.6.6. Uji Kompetensi/Ulangan

1. Bilangan bertanda dengan jangkauan -128 sampai $+127$, disebut...
 - a. bar offset
 - b. byte offset
 - c. column offset
 - d. sign offset
 - e. bit offset
2. Jika kondisi percabangan TRUE, isi dari byte bertanda ... bit yang mengikuti kode operasi akan ditambahkan dengan isi dari PC untuk membentuk alamat efektif percabangan.
 - a. 0
 - b. 4
 - c. 6
 - d. 8
 - e. 16
3. Jika kondisi percabangan FALSE maka kontrol program akan...
 - a. Berhenti
 - b. Mengubah perintah
 - c. Terus ke instruksi di bawah instruksi percabangan
 - d. Terus ke instruksi di atas instruksi percabangan
 - e. Menunggu sampai kondisi percabangan menjadi TRUE
4. Mnemonic dari instruksi Branch if Interrupt Line is High, yaitu...
 - a. BIH
 - b. BHI
 - c. BIL
 - d. BRL
 - e. BRH
5. BNE merupakan mnemonic dari instruksi...
 - a. Branch if Enterrupt
 - b. Branch if Interrupt

- c. Branch if Plus
- d. Branch if Equal
- e. Branch if Not Equal

PENUTUP

RANGKUMAN

1. Bab 1 – Memahami Struktur dan fungsi CPU

1.1 Kegiatan belajar 1 – Struktur dan fungsi CPU

Rangkuman

CPU adalah komponen pengolah data berdasarkan instruksi – instruksi yang diberikan kepadanya, yang tersusun atas beberapa komponen, yaitu:

1. Arithmetic and Logic Unit (ALU) atau language machine, bertugas membentuk fungsi – fungsi pengolahan data computer, yang terdiri dari dua bagian, yaitu unit aritmetika dan unit logika Boolean.
2. Control Unit, bertugas mengontrol operasi CPU dan secara keseluruhan mengontrol computer, termasuk mengambil instruksi – instruksi dari memori utama dan menentukan jenis instruksi tersebut.
3. Registers, adalah media penyimpanan internal CPU yang digunakan saat proses pengolahan data, yang bersifat sementara.
4. CPU Interconnections, adalah sistem koneksi dan bus yang menghubungkan komponen internal CPU, yaitu ALU, unit kontrol dan register – register dan juga dengan bus – bus eksternal CPU.

Hal-hal yang dilakukan CPU adalah sebagai berikut :

1. Fetch instruction (mengambil instruksi).
2. Interpret instruction (menterjemahkan instruksi).
3. Fetch data (mengambil data).
4. Process data (mengolah data).
5. Write data (menulis data).

1.2 Kegiatan belajar 2 – Control Unit

Rangkuman

Tugas dari CU : Mengatur dan mengendalikan alat-alat input dan output, Mengambil instruksi-instruksi dari memori utama, Mengambil data dari memori utama kalau diperlukan oleh proses, Mengirim instruksi ke ALU bila ada perhitungan aritmatika atau perbandingan logika serta mengawasi kerja serta Menyimpan hasil proses ke memori utama.

Masukan-masukan unit control: Clock / pewaktu, Register instruksi, Flag, Sinyal control untuk mengontrol bus

1.3 Kegiatan belajar 3 – Register

Rangkuman

Sistem Komputer menggunakan hirarki memori pada tingkatan yang atas, memori lebih cepat, lebih kecil, lebih mahal. Di dalam CPU, terdapat sekumpulan register yang tingkatan memorinya berada di atas hirarki memori utama dan cache.

Register dalam CPU memiliki dua fungsi:

1. User-visible Register, yaitu register yang dapat direferensikan dengan menggunakan bahasa mesin yang dieksekusi CPU, User-visible terdiri dari: General Purpose Register; Register Alamat; Register Data; dan Register Kode Status Kondisi (Flag)

2. Control and Register, yaitu register-register yang digunakan oleh unit kontrol untuk mengontrol operasi CPU dan oleh program sistem operasi untuk mengontrol eksekusi program. Terdapat empat register yang penting, yaitu: Program Counter (PC); Instruction Register (IR); Memory Address Register (MAR); Memori Buffer Register (MBR); Memori Data Register (MDR)

Fungsi CPU, yaitu Menjalankan program – program yang disimpan dalam memori utama dengan cara mengambil instruksi – instruksi, menguji instruksi tersebut dan mengeksekusinya satu persatu sesuai alur perintah.

2. Bab 2 – Memahami Karakteristik set Instruksi

2.1 Kegiatan belajar 1 – Elemen elemen instruksi

Rangkuman

Elemen-elemen instruksi yang diperlukan oleh CPU, antara lain; Operation Code/Kode Operasi; Operasi dispesifikasikan oleh kode biner; Source Operand Reference/Referensi Operand Sumber; Result Operand Reference/Referensi Operand Hasil; Next Instruction Reference/Referensi Operand Selanjutnya.

Sumber dan hasil operand dapat berada di salah satu dari tiga daerah, yaitu: Memori utama atau memori virtual; Register CPU; Perangkat I/O.

Dalam komputer, instruksi direpresentasikan oleh sekumpulan bit. Mnemonic adalah singkatan-singkatan yang mengindikasikan suatu operasi yang merupakan representasi dari opcode. Contoh: ADD; SUB; LOAD; INC.

2.1 Kegiatan belajar 2 – Tipe – tipe Instruksi

Rangkuman

Tipe atau jenis-jenis instruksi, antara lain:

1. Data processing: Arithmetic dan Logic Instructions

Data processing adalah jenis pemrosesan yang dapat mengubah data menjadi informasi atau pengetahuan. Setelah diolah, data ini biasanya mempunyai nilai yang informative, maka istilah pemrosesan data sering dikatakan sebagai sistem informasi.

2. Data storage: Memory instructions

Sering disebut sebagai memori komputer, merujuk kepada komponen komputer, perangkat komputer, dan media perekaman yang mempertahankan data digital yang digunakan untuk beberapa interval waktu.

Dalam penggunaan kontemporer, memori komputer merujuk kepada bentuk media penyimpanan berbahan semikonduktor, yang dikenal dengan sebutan Random Access Memory (RAM). Akan tetapi, istilah "computer storage" sekarang secara umum merujuk kepada media penyimpanan massal seperti halnya hard disk.

3. Data Movement: I/O instructions

Proses data movement ini adalah memindahkan (dapat dikatakan membackup juga) data – data dari database yang berupa data, indeks, grand, schema, dan lain – lain ketempat baru.

Data movement terdiri dari 2 bagian besar yaitu: Load & Upload dan Export & Import. Load berfungsi untuk memasukan data / transaksi ke sebuah table. Sedangkan upload berfungsi untuk membuat dari data table ke fisik / file. Upload Parameter, antara lain: limit, sample, dan when.

Bulk Data Movement (Software Pendukung), antara lain: ETL [Extrat Transform Load] dan Replication and Propagation. Hak akses dalam load & unload, import & export minimal adalah akses select.

Dalam distribution database terdapat 3 istilah yaitu: Autonomi [idependent]; Isolation [stand alone]; dan Transparency [all user]. Lawan dari database terdistribusi adalah database terpusat.

4. Control: Test and branch instructions

CU diimplementasikan sebagai sebuah microprogram yang disimpan di dalam tempat penyimpanan kontrol (control store). Control Unit - CU) adalah salah satu bagian dari CPU yang bertugas untuk memberikan arahan/kendali/ kontrol

terhadap operasi yang dilakukan di bagian ALU (Arithmetic Logical Unit) di dalam CPU tersebut.

2.2 Kegiatan belajar 3 – Tipe – tipe operand

Rangkuman

Operand adalah sebuah objek yang ada pada operasi matematika yang dapat digunakan untuk melakukan operasi. Operand atau operator dalam bahasa C berbentuk symbol yang berupa karakter yang ada pada keyboard seperti = , * dan sebagainya.

Tipe Operand, antara lain:

- Tipe Data, dapat berupa angka bulat (integer), angka biasa (real), atau berupa karakter (char), dan sebagainya. Ada dua kategori dari tipe data yaitu tipe dasar (berupa Bilangan bulat Integer, Long, Byte; Bilangan pecahan double dan float; dan karakter) dan tipe bentukan (berupa Array (larik) dan string).
- Variabel, yaitu nama yang mewakili suatu elemen data seperti : jenkel untuk jenis kelamin, t4lahir untuk tempat lahir. Aturan yang wajib diikuti dalam pemberian nama variabel, antara lain: Harus dimulai dengan huruf abjad; Tidak boleh ada spasi diantaranya; tidak menggunakan simbol-simbol; Sebaiknya memiliki arti yang sesuai dengan elemen data; Sebaiknya tidak terlalu panjang
- Operator dan Operan. Operand adalah data, tetapan, perubah atau hasil dari suatu fungsi sedangkan Operator merupakan simbol-simbol yang memiliki fungsi untuk menghubungkan operand sehingga menjadi transformasi. Jenis-jenis operator adalah: Operator Aritmetika; Operator relational; Operator Logik.

2.3 Kegiatan belajar 4 – Tipe – tipe operasi data

Rangkuman

Transfer data, meliputi kegiatan: Menetapkan lokasi operand sumber dan operand tujuan; Lokasi-lokasi tersebut dapat berupa memori, register atau bagian paling atas daripada stack; Menetapkan panjang data yang dipindahkan; Menetapkan mode pengalamatan. Tindakan CPU untuk melakukan transfer data adalah: Memindahkan data dari satu lokasi ke lokasi lain; Menetapkan alamat memori, Menjalankan transformasi alamat memori virtual ke alamat memori actual, Mengawasi pembacaan / penulisan memori (Apabila memori dilibatkan). Operasi set instruksi untuk transfer data, antara lain: MOVE, STORE, LOAD, EXCHANGE, CLEAR / RESET, SET, PUSH, POP.

ARITHMETIC. Tindakan CPU untuk melakukan operasi arithmetic, antara lain: Transfer data sebelum atau sesudah; Melakukan fungsi dalam ALU; dan Menset kode-kode kondisi dan flag. Operasi set instruksi untuk arithmetic, yaitu: ADD, SUBTRACT, DIVIDE, ABSOLUTE, NEGATIVE, DECREMENT, INCREMENT.

LOGICAL. Tindakan CPU untuk melakukan operasi logical, sama dengan arithmetic. Operasi set instruksi untuk operasi logical, yaitu: AND, OR, NOT, EXOR; COMPARE; TEST; SHIFT; ROTATE.

CONVERSI. Tindakan CPU untuk melakukan operasi Conversi, sama dengan arithmetic dan logical. Operasi set instruksi untuk conversi, yaitu: TRANSLATE dan CONVERT.

I/O. Tindakan CPU untuk melakukan INPUT /OUTPUT, yaitu: Apabila memory mapped I/O maka menentukan alamat memory mapped , dan Mengawasi perintah ke modul I/O. Operasi set instruksi Input / Output, antara lain: INPUT, OUTPUT, START I/O, TEST I/O.

TRANSFER CONTROL. Tindakan CPU untuk transfer control, yaitu Mengupdate program counter untuk subrutin , call / return. Operasi set instruksi untuk transfer control, meliputi: JUMP (cabang); JUMP BERSYARAT; JUMP SUBRUTIN; RETURN; EXECUTE; SKIP; SKIP BERSYARAT; HALT; WAIT (HOLD); NO OPERATION.

CONTROL SYSTEM. Hanya dapat dieksekusi ketika prosesor berada dalam keadaan khusus tertentu atau sedang mengeksekusi suatu program yang berada dalam area khusus, biasanya digunakan dalam sistem operasi. Contoh : membaca atau mengubah register kontrol.

2.4 Kegiatan belajar 5 – Instruksi percabangan

Rangkuman

Struktur Branching (Percabangan)

1. IF. Instruksi IF digunakan untuk memeriksa suatu kondisi dan melaksanakan instruksi lain jika kondisi tersebut terpenuhi atau bernilai true.
2. If Else. Apabila evaluasi terhadap kondisi memberikan nilai true maka statement dikerjakan, sebaliknya jika false maka statement tidak dikerjakan. Statement dapat berupa instruksi kosong, instruksi tunggal atau beberapa instruksi yang dilingkup dengan { }. Tidak diperbolehkan ada klausa else tanpa instruksi if.
3. Nested If (Struktur If bersarang), yaitu pada suatu instruksi if, statement yang dikerjakan apabila kondisi bernilai true dapat berupa instruksi if yang lain. Hal yang sama juga berlaku untuk statement yang dikerjakan pada bagian klausa else.

3. Bab 3 – Mode dan format pengalamatan

3.1 Kegiatan belajar 1 – Inherent

Rangkuman

Dalam mode pengalamatan *inherent*, semua informasi yang dibutuhkan untuk operasi telah diketahui otomatis oleh CPU, dan tidak dibutuhkan operan eksternal dari memori atau dari program. Karena itu operasi ini hanyalah terdiri dari satu byte instruksi. Contoh: 0200 4C INCA ; increment akumulator.

Pertama CPU membaca kode operasi \$4C, kemudian CPU akan menyimpan harga baru ke dalam akumulator dan mengeset bitflag jika dibutuhkan.

Instruksi-instruksi Mode Pengalamatan *inherent*, antara lain: ASLA, ASRA, CLC, CLI; CLRA, COMA, DECA, INCA; LSLA, LSRA, MUL, NEGA; NOP, ROLA, RORA, RSP; RTI, RTS, SEC, SEI; STOP, SWI, TAX, TSTA; TXA, WAIT.

3.2 Kegiatan belajar 2 – Immediate

Rangkuman

Dalam mode pengalamatan *immediate*, operan terkandung di dalam byte yang langsung mengikuti kode operasi. Operasi dengan mode ini membutuhkan dua byte instruksi, satu untuk kode operasi dan satu lagi untuk data byte. Contoh: 0200 A6 02 LDA #\$02 ; Load konstanta ke akumulator.

Pertama CPU akan membaca kode operasi \$A6, kemudian CPU akan membaca data *immediate* \$02 dari lokasi memori dengan alamat \$0201 ke dalam akumulator.

Instruksi-instruksi Mode Pengalamatan *immediate*, antara lain: ADC, ADD, AND, BIT; CMP, CPX, EOR, LDA; LDX, ORA, SBC, SUB.

3.3 Kegiatan belajar 3 – Direction

Rangkuman

Mode pengalamatan *direct* menyebabkan efisiensi alamat dalam 256 byte pertama dalam memori. Pengalamatan *direct* ini efisien bagi memori program dan waktu eksekusi. Dalam mode ini instruksi terdiri dari dua byte, satu untuk kode operasi dan satu lagi untuk alamat operan. Contoh: 0200 B6 E0 LDA \$E0

Pertama CPU akan membaca kode operasi \$B6, kemudian CPU membaca \$E0 dari lokasi memori dengan alamat \$0201. Harga \$E0 ini diterjemahkan sebagai *low-order* dari alamat dalam *direct page* (\$0000 sampai \$00FF). Setelah itu CPU menyusun alamat lengkap \$00E0 dengan menganggap *high-order* byte alamat sebagai \$00. Alamat lengkap \$00E0 ini kemudian diletakkan dalam bus alamat dan diproses.

Instruksi-instruksi Mode Pengalamatan *direct*, antara lain: ADC, ADD, AND, ASL; ASR, BCLR, BIT, BRCLR; BRSET, BSET, CLR, CMP; COM, CPX, DEC, EOR; INC, JMP, JSR, LDA; LDX, LSL, LSR, NEG; ORA, ROL, ROR, SBC; STA, STX, SUB, TST.

3.4 Kegiatan belajar 4 – Extended

Rangkuman

Pengalamatan *extended* ini dapat digunakan untuk mengakses semua lokasi dalam memori mikrokontroler termasuk I/O, RAM, ROM, dan EPROM. Karena itu operasi ini membutuhkan tiga byte, satu untuk kode operasi, dan dua untuk alamat dari operan. Contoh : 0200 C6 03 65 LDA \$0365

Pertama CPU akan membaca kode operasi C6, kemudian CPU akan membaca \$03 dari lokasi memori \$0201 dan diterjemahkan sebagai alamat *high-order*. Setelah itu CPU membaca \$65 dari lokasi \$0202 dan diterjemahkan sebagai *low-order* dari

alamat. Terakhir, CPU membangun alamat lengkap \$0365 dan meletakkannya dalam bus alamat dan kemudian melakukan operasi read.

Instruksi-instruksi Mode Pengalamatan *extended*, antara lain: ADC, ADD, AND, BIT; CMP, CPX, EOR, JMP; JSR, LDA, LDX, ORA; SBC, STA, STX, SUB.

3.5 Kegiatan belajar 5 – Indexed

Rangkuman

Terdapat tiga jenis pengalamatan indexed yang didukung oleh CPU keluarga M68HC05, yaitu:

1. Mode pengalamatan ***indexed-no offset***. Dalam mode pengalamatan ini, alamat efektif dari operan terkandung dalam index register 8-bit. Karena itu, mode pengalamatan ini dapat mengakses 256 lokasi memori (dari \$0000 sampai \$00FF). Instruksi mode ini membutuhkan satu byte instruksi. Contoh: 0200 F6 LDA 0,X

Pertama CPU akan membaca kode operasi \$F6, kemudian CPU menyusun alamat lengkap dengan menjumlahkan \$0000 ke isi dari index register 8-bit (X). Alamat ini kemudian diletakkan dalam bus alamat dan setelah itu dilakukan proses pembacaan data.

Indexed-no offset dan Indexed-8 bit offset, antara lain: ADC, ADD, AND, ASL; ASR, BIT, CLR, CMP; COM, CPX, DEC, EOR; INC, JMP, JSR, LDA; LDX, LSL, LSR, NEG; ORA, ROL, ROR, SBC; STA, STX, SUB, TST.

2. Mode pengalamatan ***indexed-8 bit offset***. Dalam mode pengalamatan ini, alamat efektif dicapai dengan menambahkan data byte yang mengikuti kode operasi dengan isi dari index register. Operasi ini adalah dua byte instruksi di mana offset terkandung dalam byte yang mengikuti kode operasi. Isi dari index register tidak akan dirubah. Contoh: 0200 E6 05 LDA 5,X

Pertama CPU akan membaca kode operasi \$E6, kemudian CPU akan membaca 8-bit offset (\$05) dari alamat \$0201, lalu menyusun alamat lengkap dengan menambahkan nilai offset \$05 ke isi dari index register 8-bit (X). Setelah itu, alamat diletakkan dalam bus alamat dan diproses.

3. Mode pengalamatan ***indexed-16 bit offset***. Dalam mode pengalamatan ini, alamat efektif dari operan suatu instruksi adalah hasil penjumlahan antara isi dari index register 8-bit dengan dua byte alamat yang mengikuti kode operasi. Instruksi ini memakan tiga byte, satu untuk kode operasi dan dua byte untuk offset 16-bit. Contoh: 0200 D6 03 77 LDA \$377,X

Pertama CPU akan membaca kode operasi \$D6, kemudian CPU akan membaca *high order* alamat (\$03) dari alamat memori \$0201 dan membaca *low order* alamat (\$77) dari alamat \$0202. CPU akan menyusun alamat lengkap dengan menambahkan isi dari index register 8-bit dengan 16-bit alamat yang baru saja dibaca. Alamat ini diletakkan dalam bus alamat dan diproses.

Instruksi-instruksi Mode Pengalamatan ***indexed-16 bit offset***, antara lain: ADC, ADD, AND, BIT; CMP, CPX, EOR, JMP; JSR, LDA, LDX, ORA; SBC, STA, STX, SUB.

3.6 Kegiatan belajar 6 – Relative

Rangkuman

Mode pengalamatan *relative* ini digunakan hanya dalam instruksi percabangan. Karena kemampuannya untuk bercabang ke dua arah, byte offset adalah bilangan bertanda dengan jangkauan -128 sampai +127. Jika kondisi percabangan TRUE, isi dari

byte bertanda 8-bit; jika FALSE maka kontrol program akan terus ke instruksi di bawah instruksi percabangan. Contoh: 0200 27 rr BEQ DEST

Pertama CPU akan membaca kode operasi \$27. Bit CCR Z akan set jika hasil dari operasi aritmatika atau logika sebelumnya adalah nol. CPU kemudian akan membaca harga offset \$rr dari alamat \$0201. Setelah siklus ini, PC akan menunjukkan ke byte pertama dari instruksi berikutnya (\$0202). Jika bit Z nol, maka tidak ada aksi apa-apa. Program akan terus ke instruksi berikutnya pada alamat \$0202. Jika bit Z satu, CPU akan menambahkan nilai offset \$rr ke isi dari PC sekarang untuk mendapatkan alamat tujuan dari percabangan. Hal ini menyebabkan eksekusi program berlanjut ke alamat baru (DEST).

Instruksi-instruksi Mode Pengalamatan relative, antara lain: BCC, BCS, BEQ, BHCC; BHCS, BHI, BHS, BIH; BIL, BLO, BLS, BMC; BMI, BMS, BNE, BPL; BRA, BRCLR, BRSET, BRN; BSR.

DAFTAR PUSTAKA

- Jogiyanto. (2005). Pengenalan Komputer. Yogyakarta: Andi Offset.
- Poerwadarminta, W. (1991). Kamun Umum Bahasa Indonesia. Jakarta: Balai Pustaka.
- Stalling, W. (1997). Organisasi dan Arsitektur Komputer, Perancangan dan kinerja Edisi Bahasa Indonesia. PT Prenhallindo.
- Supriyanto, A. (2005). Merakt, Mengupgrade dan Mengatasi Masalah PC. Yogyakarta: Graha Ilmu.
- Jogiyanto. (2005). Pengenalan Komputer. Yogyakarta: Andi Offset.
- Poerwadarminta, W. (1991). Kamun Umum Bahasa Indonesia. Jakarta: Balai Pustaka.
- Stalling, W. (1997). Organisasi dan Arsitektur Komputer, Perancangan dan kinerja Edisi Bahasa Indonesia. PT Prenhallindo.
- Supriyanto, A. (2005). Merakit, Mengupgrade dan Mengatasi Masalah PC. Yogyakarta: Graha Ilmu.
- Drs. H. Suparmin, M.Pd; Rismawan, S.Kom.2014 Sistem Komputer. Surakarta. Mediatama.
- Jogiyanto. (2005). Pengenalan Komputer. Yogyakarta: Andi Offset.
- Poerwadarminta, W. (1991). Kamun Umum Bahasa Indonesia. Jakarta: Balai Pustaka.
- Stalling, W. (1997). Organisasi dan Arsitektur Komputer, Perancangan dan kinerja Edisi Bahasa Indonesia. PT Prenhallindo.
- Supriyanto, A. (2005). Merakit, Mengupgrade dan Mengatasi Masalah PC. Yogyakarta: Graha Ilmu.
- Firman Adi Hindarsyah, 'Elemen elemen instruksi mesin' 19 Nopember 2014.
<http://firmanadihindarsyah.blogspot.com/2010/05/elemen-elemen-instruksi-mesin.html>
- Danang Yuditya Haryono, 'Instruksi Dan Mode Pengalamatan Pada Komputer' 19 Nopember 2014.
<http://danang-dancil.blogspot.com/2012/11/instruksi-dan-mode-pengalamatan-pada.html>
- Suharno Wicaksono, 'Karakteristik Instruksi Mesin' 19 Nopember 2014.
<http://harnowicaksono.blogspot.com/2013/11/karakteristik-instruksi-mesin.html>
- Ilma Qodri, 'Pengertian Organisasi & Arsitektur KOMputer' 20 Nopember 2014.
http://ilmaqadri03.blogspot.com/2014_11_01_archive.html
- Ona Fitri, 'Jenis jenis Instruksi' 20 Nopember 2014.
<http://donafitri23.blogspot.com/2013/05/jenis-jenis-instruksi.html>
- Rezaky Aisyah 'Tipe tipe operand dan operasi' 20 Nopember 2014.
<http://kikireisyah.wordpress.com/tipe-tipe-operand-dan-operasi/>
- Adam Aris Rizaldy 'Mode format pengalamatan' 19 Nopember 2014.
<https://adamrizaldy09.wordpress.com/2014/11/11/mode-format-pengalamatan/>
- Sutanta, Hermawan. 'Inherent' 24 Nopember 2014. <http://mikrokontroler.tripod.com/6805/im-in.htm>

- Sutanta, Hermawan. 'Extended' 24 Nopember 2014.
<http://mikrokontroler.tripod.com/6805/ext-dir.htm>
- Sutanta, Hermawan. 'Indexed' 24 Nopember 2014. <http://mikrokontroler.tripod.com/6805/idx-rel.htm>
- Sutanta, Hermawan. 'File Kode Objek' 24 Nopember 2014.
<http://mikrokontroler.tripod.com/6805/as-file.htm>
- Niki Niku Niko Radino, 'apa itu control unit' 23 Nopember 2014. <http://kj-jogja.blogspot.com/2013/09/apa-itu-cu-control-unit.html>
- Yudhi Pri, 'Set instruksi dan teknik pengalamatan dalam komputer' 24 Nopember 2014. <http://jovanangga.blogspot.com/2012/11/set-instruksi-dan-teknik-pengalamatan.html>
- Serdiwansyah N. A. Set Instruksi dan Teknik Pengalamatan Teknik Elektro Universitas Negri Makasar
- Eka Pramudita Kusumaari, 'Pengertian Control Unit (CU) dan Aritmetic Logical Unit (ALU)' 21 Nopember 2014. <http://ekapka.blogspot.com/2013/09/pengertian-control-unit-cu-dan.html>
- Destu Wijayanto, 'Pengertian Monitor dan Fungsinya' 23 Nopember 2014. <http://destuwas.blogspot.com/2013/02/pengertian-monitor-dan-fungsinya.html>
- Anggi43ani, 'unit keluaran/masukan' 23 Nopember 2014. <http://penawikara.wordpress.com/2013/05/31/unit-keluaranmasukan/>
- Muhammad Arif Abdurrahman, 'storage device dan contohnya' 24 Nopember 2014
<http://arifdba.blogspot.com/2013/07/storage-device-dan-contohnya.html>
- Anonim/MSaiful, 'Siklus Mesin' 24 Nopember 2014. <http://msaiful391.blogspot.com/>
- Fatimah, 'Pengertian Plotter' 20 Nopember 2014. <http://fatimah9a.blogspot.com/2011/11/pengertian-plotter-plotter-merupakan.html>
- Dini Maulita, 'Siklus Organisasi Prosesor' 21 Nopember 2014. <http://dinimaulita.blogspot.com/2011/12/siklus-organisasi-prosesor.html>
- Anonim, Siklus Pengolah Mesin' 20 Nopember 2014. <http://blog.student.uny.ac.id/rheza/2011/06/03/siklus-pengolah-intruksi/>
- Satria Adhi Kharisma, 'Pengertian Modem dan Jenis-Jenisnya' 20 Nopember 2014. http://satriaadhikharisma.blogspot.com/2013/07/pengertian-modem-dan-jenis-jenisnya_17.html
- Ari Setiawan, 'Pengertian Network Interface Card (NIC) Beserta Jenisnya' 20 Nopember 2014. <http://freesoft41.blogspot.com/2014/11/pengertian-network-interface-card-nic.html>
- PC Solution, 'Pengertian dan Fungsi Flash disk' 21 Nopember 2014.
<http://solusikompi.blogspot.com/2014/11/pengertian-dan-fungsi-flash-disk.html>
- Khunsul Marlia, 'Pengertian dan Fungsi Flash disk' 21 Nopember 2014.
<http://solusikompi.blogspot.com/2014/11/pengertian-dan-fungsi-flash-disk.html>
- Khunsul Marlia, 'Pengertian dan Fungsi CD/DVD Rom' 21 Nopember 2014. <http://solusikompi.blogspot.com/2014/10/pengertian-dan-fungsi-cddvd-rom.html>

Juhriah. 'Organisasi prosesor dan register' 21 Nopember 2014.

<http://juhriah011.wordpress.com/2013/11/06/78/>

Ahmad Amirudin. 'Register pada CPU' 25 Nopember

2014.<http://orkomp.blogspot.com/2012/06/register.html>

Milik Negara
Tidak Diperdagangkan

SISTEM KOMPUTER

