***7***

|  |  |  |
| --- | --- | --- |
| **Mata Kuliah** | **:** | **Arsitektur dan Organisasi Komputer** |
| **Bobot Sks** | **:** | **3 sks** |
| **Dosen Pengembang** | **:** | **Catur Nugroho, S.Kom., M.Kom** |
| **Tutor** | **:** |  |
| **Capaian Pembelajaran Mata Kuliah** | **:** | **Mahasiswa mampu menggunakan dan menerapkan konsep**  **struktur CPU dan fungsi CPU** |
| **Kompetentsi Akhir Di Setiap Tahap (Sub-Cpmk)** | **:** | **Mahasiswa mampu menggunakan dan menerapkan konsep**  **struktur CPU dan fungsi CPU** |
| **Minggu Perkuliahan Online Ke-** | **:** | **Sesi 13** |

1. PENGERTIAN DAN ELEMEN CENTRAL PROCESSING UNIT (CPU)

Prosesor adalah komponen komputer sebagai otak yang menjalankan kerja komputer.

Kapasitas prosesor yang besar dan didukung oleh memori (RAM) yang juga besar akan membantu setiap pengerjaan instruksi dengan cepat, arti lain dari prosesor adalah chip yang sering disebut “microprocessor”, Prosesor merupakn ***integrated circuit*** (IC) yang mengontrol keseluruhan jalannya system komputer dan digunakan sebagai pusat atau otak dari komputer.

Satuan Kecepatan Prosesor.



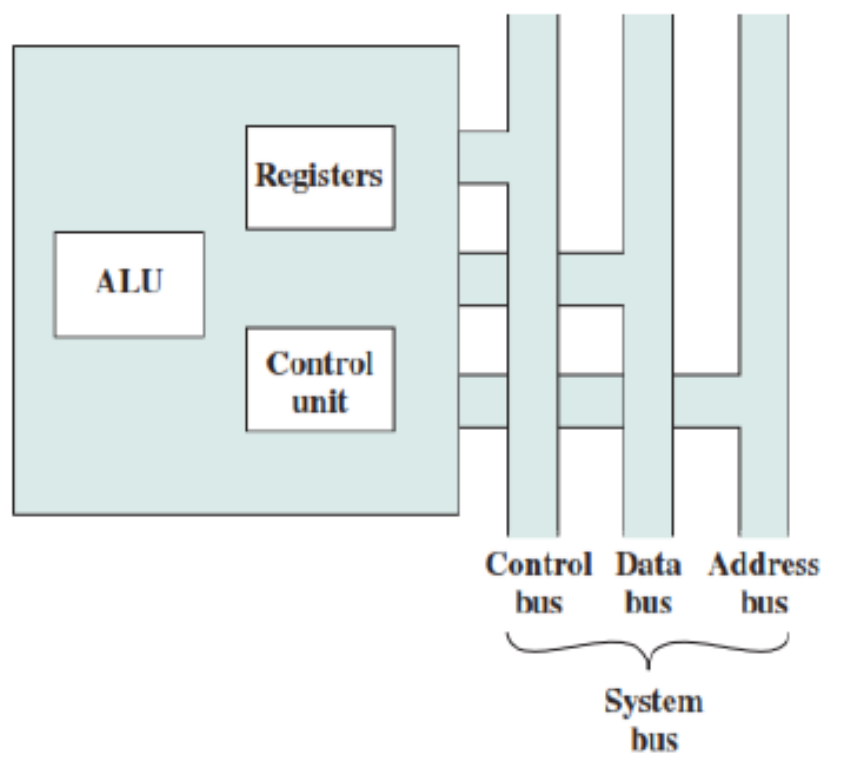
Semakin besar kecepatan prosesor, semakin cepat kinerja komputer.

**A. 1 HAL-HAL YANG DILAKUKAN CPU**

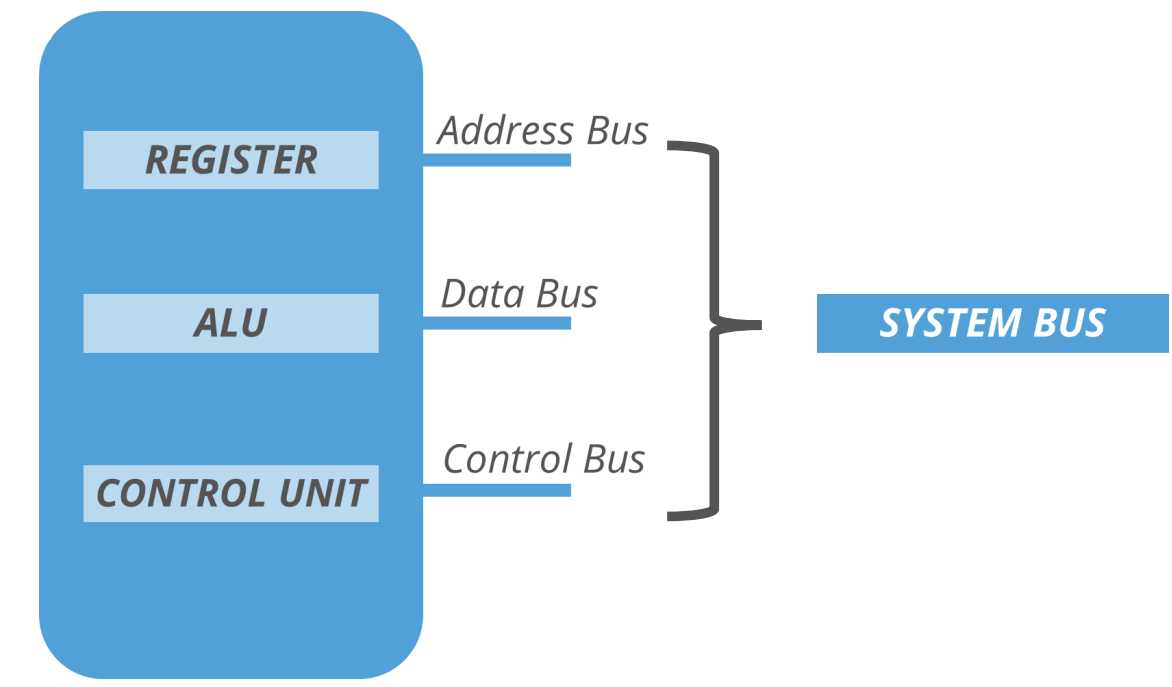
1. ***Fetch Instruction (*Mengambil Instruksi) :** CPU membaca instruksi dari memori.
2. ***Interpret Instruction* (Menerjemahkan Instruksi) :** CPU menerjemahkan instruksi untuk menentukan aksi yang diperlukan.
3. **Fetch Data (Mengambil Data) :** Pembacaan data dari memori atau modul I/O.
4. ***Process Data* (Mengolah Data)** : Eksekusi instruksi melalui operasi aritmetika dan logika.
5. ***Write Data* (Menulis Data) :** Penulisan data hasil eksekusi ke memori atau modul I/O.

**A.2 BAGIAN-BAGIAN ORGANISASI PROSESOR**

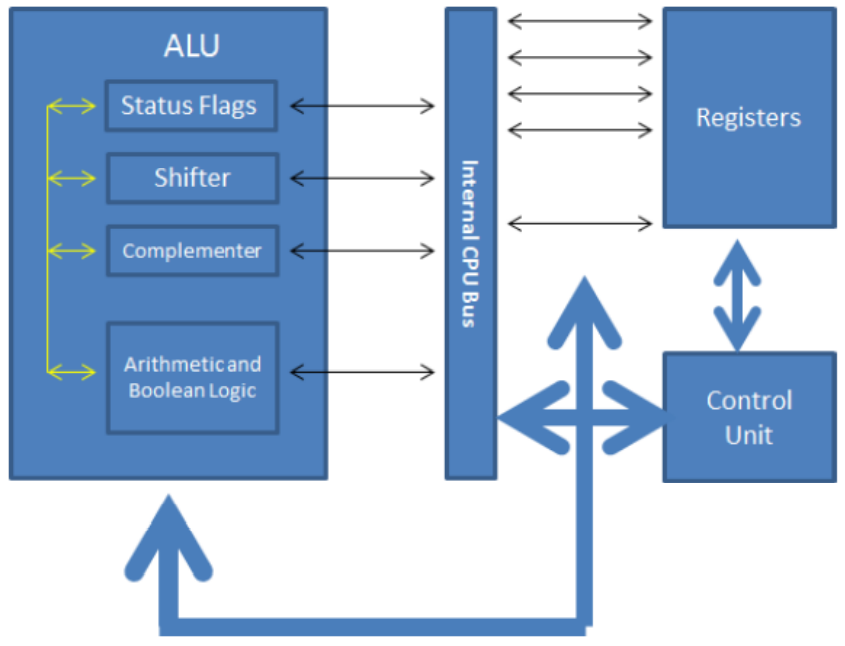
1. ALU (***Arithmetic Logic Unit*)** : Melakukan komputasi pengolahan data aktual.
2. CU (***Control Unit***) : Mengontrol perpindahan data dan instruksi ke atau dari CPU dan mengontrol operasi ALU.
3. ***Register*** : Sebagai memori internal dari CPU.



GAMBARAN SEDERHANA PROSESOR



GAMBARAN STRUKTUR INTERNAL CPU



* **ALU (ARITHMETIC LOGIC UNIT )**

Arithmetic and Logic Unit (ALU) adalah salah satu bagian/komponen di dalam sistem komputer yang berfungsi melakukan operasi/perhitungan aritmetika dan logika (seperti penjumlahan, pengurangan, dan beberapa logika lain). ALU berfungsi untuk melakukan operasi aritmetika ataupun logika berdasarkan instruksi yang diberikan, ALU dibagi dua bagian yaitu unit *aritmetika* dan *unit logika*, maka dari itu ALU sering disebut juga sebagai unit bahasa. ALU mengerjakan operasi perhitungan dengan menggunakan prinsip dasar penjumlahan - DISEBUT SEBAGAI ***ADDER***

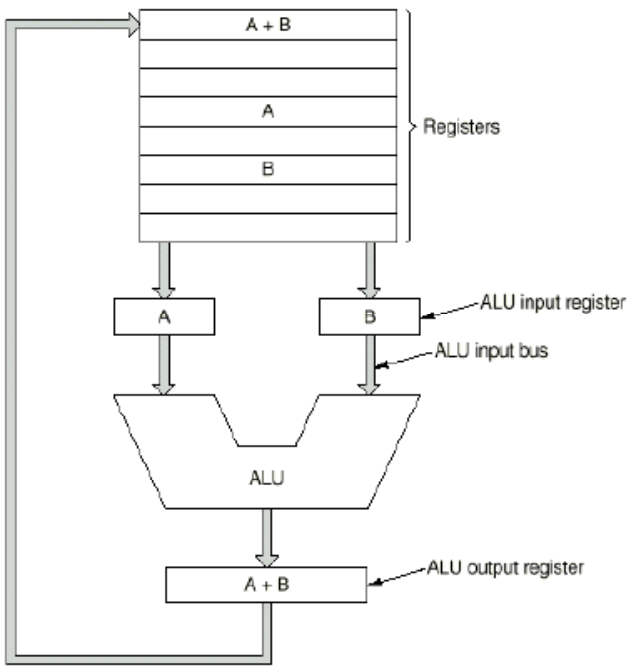
Tugas lain dari ALU adalah melakukan keputusan dari operasi sesuai instruksi program.

* **CARA KERJA ALU**

ALU bekerja sama dengan memori, di mana hasil dari perhitungan disimpan ke dalam memori. Perhitungan dalam ALU menggunakan kode biner, yang merepresentasikan instruksi yang akan dieksekusi (*opcode*) dan data yang diolah (*operand*).ALU merupakan bagian pengolah bilangan dari sebuah computer, Operasi aritmetika terdiri dari berbagai macam operasi, yaitu operasi penjumlahan, pengurangan, perkalian, dan pembagian,

Cara mendesain ALU hampir sama dengan mendesain enkoder, dekoder, multiplekser, dan demultiplekser, Rangkaian utama yang digunakan untuk melakukan perhitungan ALU adalah ***Adder.***

Berdasarkan gambar di bawah ini, berikut adalah contoh cara kerja ALU.



* A berisi 10
* B berisi 2
* ALU menghitung 10 + 2, lalu mengirimkan hasilnya kembali ke register.
* **OPERASI LOGIKA PADA ALU**

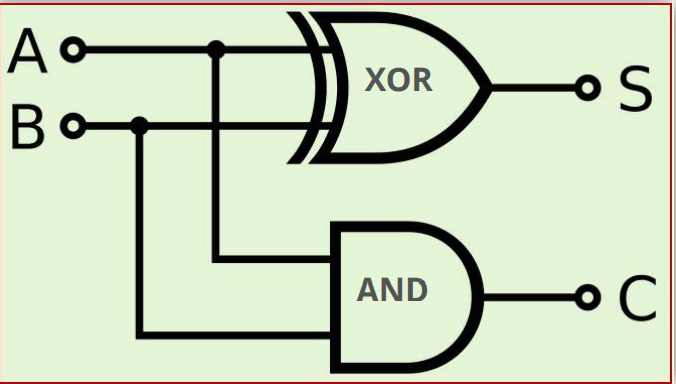
Tugas lain dari ALU adalah melakukan keputusan dari operasi logika sesuai dengan instruksi program. Operasi logika ini meliputi perbandingan dua buah elemen logika dengan menggunakan operator logika sebagai berikut.

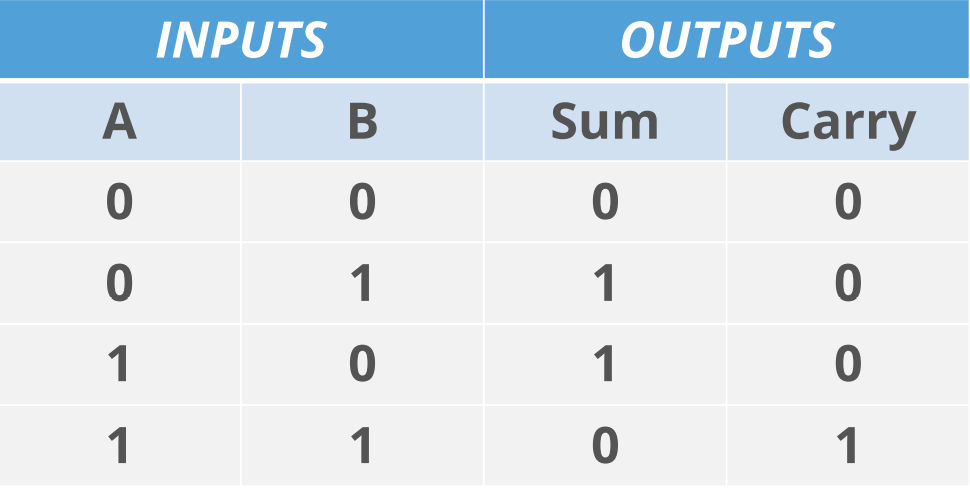
* 1. Sama dengan (=)
  2. Tidak sama dengan (<>)
  3. Kurang dari (<)
  4. Kurang dari atau sama dengan (<=)
  5. Lebih besar dari (>)
  6. Lebih besar dari atau sama dengan (>=)
* **PENGERTIAN ADDER**
* Rangkaian ALU (Arithmetic and Logic Unit) yang digunakan untuk menjumlahkan bilangan dinamakan dengan Adder.
* Adder digunakan untuk memproses operasi aritmetika, sehingga *Adder* juga sering disebut rangkaian kombinasional aritmetika.

Jenis-jenis Adder:

1. Half adder
2. Full adder
3. Parallel adder

* **HALF ADDER :** Rangkaian ini adalah rangkaian yang hanya menjumlahkan dua bit





Rangkaian half adder merupakan dasar penjumlahan

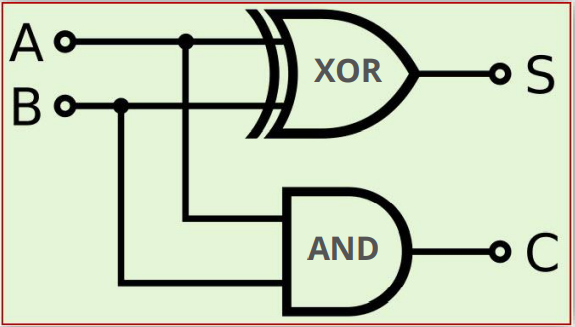
* bilangan biner yang masing-masing hanya terdiri dari satu
* bit, oleh karena itu dinamakan penjumlah tak lengkap.
* Jika A=0 dan B=0 dijumlah, hasilnya S (Sum)=0.
* Jika A=0 dan B=1 dijumlah, hasilnya S (Sum)=1.
* Jika A=1 dan B=1 dijumlah, hasilnya S (Sum)=0
* dengan nilai pindahan C (Carry Out)=1.

Dengan demikian, half adder memiliki 2 masukan (A dan B) dan dua keluaran (S dan C).

XOR: TRUE when only 1 element is TRUE.

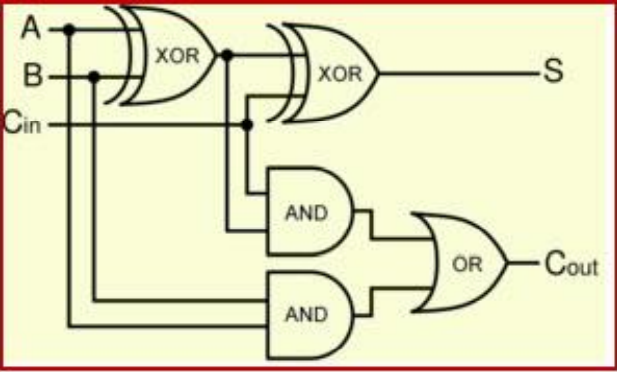
AND: TRUE when both elements are TRUE

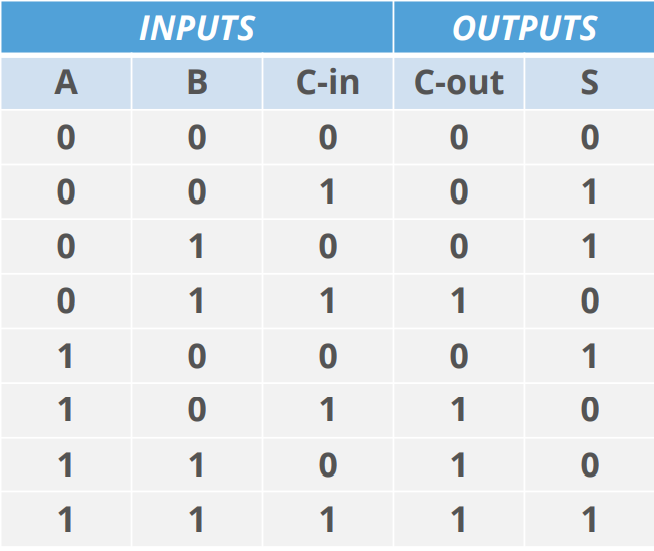
Terlihat bahwa nilai logika dari SUM sama dengan nilai logika dari gerbang XOR, sedangkan nilai logika C sama dengan nilai dari gerbang logika AND. Sehingga rangkaian half adder dapat dilihat pada gambar rangkaian di bawah.



* **FULL ADDER**

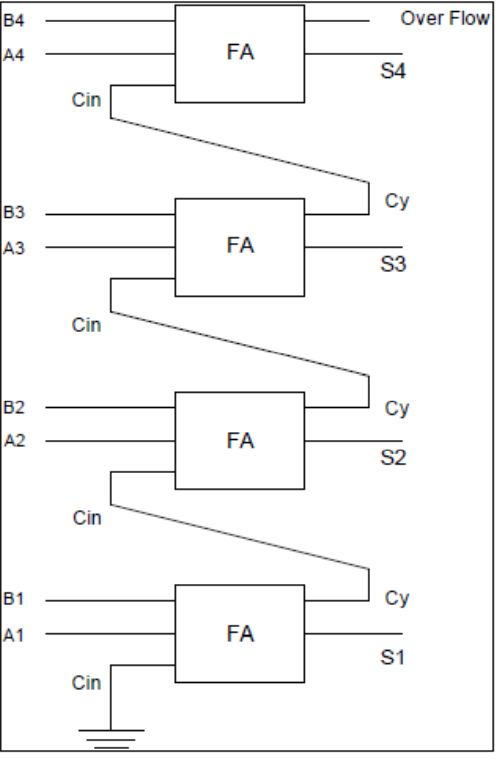
Rangkaian ini mengolah penjumlahan untuk 3 bit bilangan atau lebih (bit tidak terbatas), oleh karena itu dinamakan rangkaian penjumlah lengkap.





* **PARALLEL ADDER**

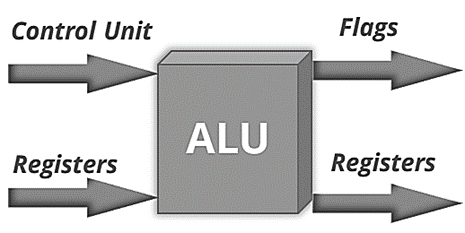
Parallel adder adalah rangkaian full adder yang disusun paralel dan berfungsi untuk menjumlah bilangan biner berapapun bitnya, tergantung jumlah full adder yang diparalelkan, Gambar menunjukkan parallel adder terdiri dari 4 buah full adder tersusun paralel sehingga berbentuk penjumlah 4 bit.



Aritmetika komputer dibentuk dari dua jenis bilangan yang sangat berbeda, yaitu:Integer : **1, 2, 3**

Floating point : **16, 32, 64, 80, 128 -Bit**

**BLOK DIAGRAM ALU**



Data diberikan ke ALU dari dalam register dan hasil operasi disimpan

dalam register, ALU dapat mengatur flag sebagai hasil dari operasi. Misalnya, flag overflow = 1 jika hasil perhitungan melebihi panjang dari register, nilai-nilai flag juga disimpan dalam register pada control unit processor.

Control unit memberikan sinyal yang mengontrol pengoperasian ALU dan pergerakan data ke dalam dan keluar dari ALU.

* **CONTOH OPERASI ARITMETIKA ALU**

Contoh operasi aritmetika adalah operasi penjumlahan dan pengurangan, sedangkan contoh operasi logika adalah AND dan OR, Tugas utama dari ALU adalah melakukan semua perhitungan aritmetika atau matematika sesuai

dengan instruksi program.

ALU melakukan operasi aritmetika dengan dasar pertambahan, Sedang operasi aritmetika yang lainnya, seperti pengurangan, perkalian, dan

pembagian dilakukan dengan dasar penjumlahan.

Sirkuit elektronik di ALU yang digunakan untuk melaksanakan operasi aritmetika ini disebut adder.

* **KEBUTUHAN SISTEM ALU**
* **Semua hal direpresentasikan dalam digit 0 & 1.**
* **Bilangan-bilangan positif disimpan dalam bentuk biner.**
* **Contoh penulisan angka dalam bilangan biner.**

**41 = 00101001**

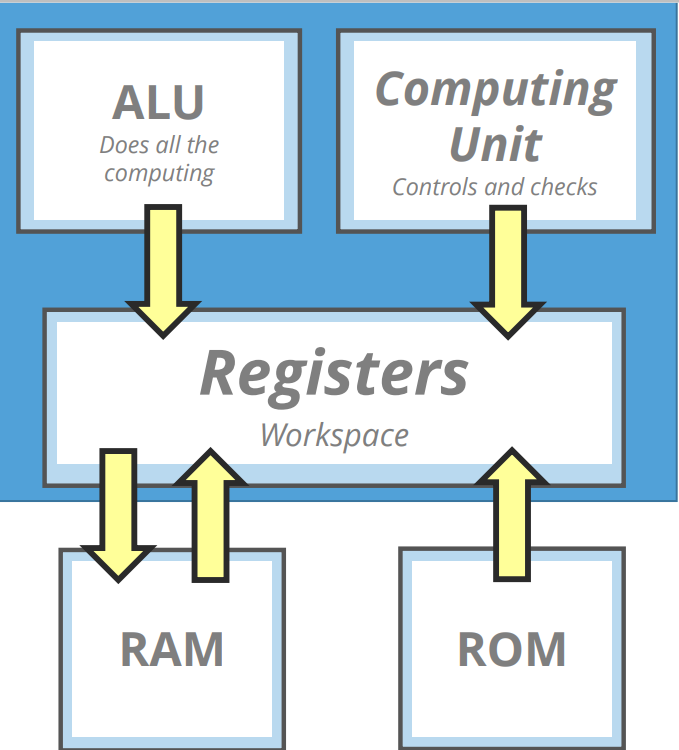
* **CONTROL UNIT**

**Fungi control unit**

* **Mengatur dan mengendalikan alat I/O.**
* Mengambil instruksi dari memori utama.
* Mengambil data dari memori utama.
* Mengirim instruksi ke ALU dan mengawasi kerja ALU.
* **REGISTER**

Register adalah alat penyimpanan dengan kapasitas kecil tetapi memiliki kecepatan yang tinggi, Register digunakan untuk menyimpan data ataupun instruksi yang sedang diproses, hampir sama dengan RAM, bersifat sementara untuk menyimpan data yang sedang diolah atau yang akan diolah selanjutnya.

Register pada CPU dapat dilihat pada gambar di bawah :



**FUNGSI REGISTER**

**Adapun fungsi register adalah sebagai berikut:**

1. Mode pengalamatan dan data
2. Akumulator (*arithmetic, shift, rotate*)
3. *Base registers* (*rotate, shift, arithmetic*)
4. *Counter registers* (*looping*)
5. *Data registers* (menyimpan alamat I/O device)

**JENIS-JENIS REGISTER**

1. ***Program Counter :*** Register yang menunjuk ke instruksi berikutnya untuk diambil dan dijalankan.
2. ***Instruction Register :*** Register yang menyimpan instruksi yang sedang dijalankan.
3. ***General Purpose Register* :** Register yang memiliki kegunaan umum yang berhubungan dengan data yang diproses.
4. ***Memory Data Register :***Register yang digunakan untuk menampung data maupun instruksi yang dikirim oleh RAM ke CPU maupun dari CPU ke RAM.
5. ***Memory Address Register*** : Register yang digunakan untuk menampung alamat data atau instruksi pada memori utama.

**JENIS ORGANISASI REGISTER**

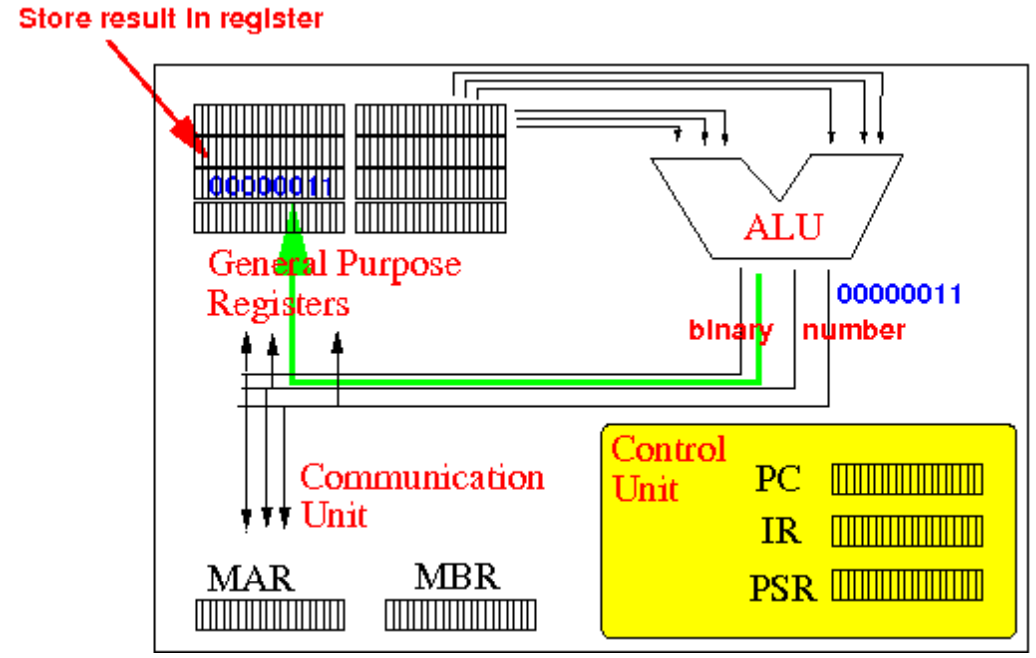
**Organisasi register meliputi jenis-jenis register di bawah ini:**

* **General:** terdapat 8 buah register general-purpose 32-bit.

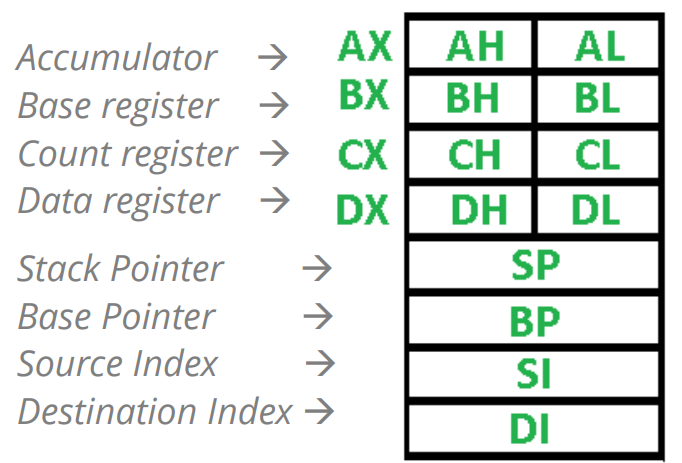
Register-register ini juga dapat menampung berbagai operands untuk keperluan kalkulasi alamat.

* **Segment:** terdapat enam segment registers 16-bit berisi lokasi spesifik segmen-segmen tertentu (data segment, code segment, dsb), yang diindeks ke dalam tabel segmen.

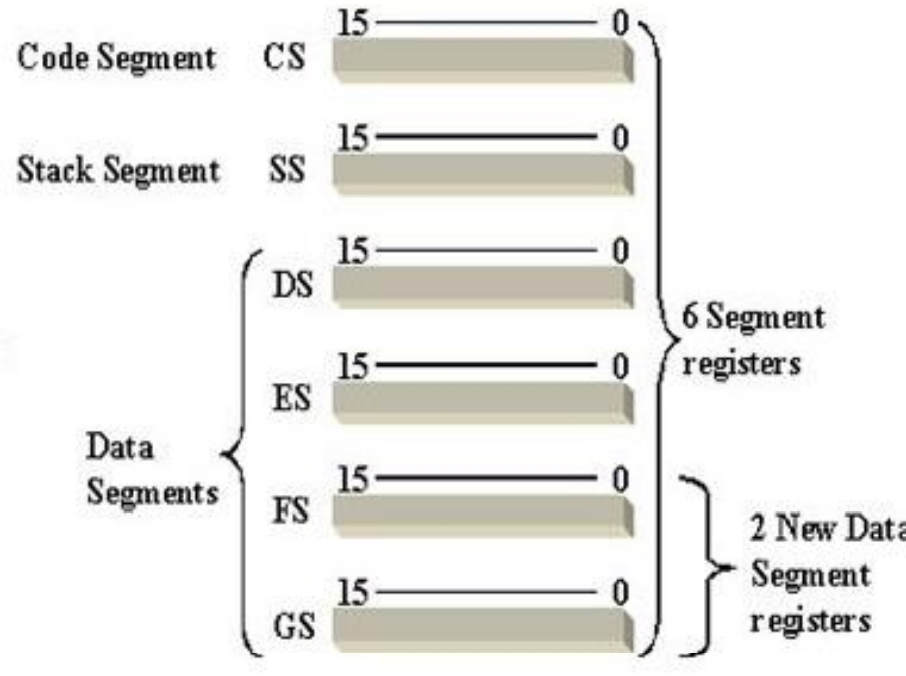
**VISUALISASI GENERAL-PURPOSE REGISTER**



**Ada 8 general purpose registers pada mikroprosesor 8086.**



**VISUALISASI SEGMENT REGISTER**



**FUNGSI REGISTER PADA CPU  
Terdapat 2 fungsi register pada CPU, yaitu:**

1. ***User-visible registers :*** Pemrogram bahasa mesin dan bahasa assembler meminimalkan referensi main memory dengan cara mengoptimasi penggunaan register.
2. ***Control and 1 status registers :*** Digunakan oleh unit kontrol untuk mengontrol operasi CPU dan oleh program sistem operasi untuk mengontrol eksekusi program.

**MACAM-MACAM REGISTER**

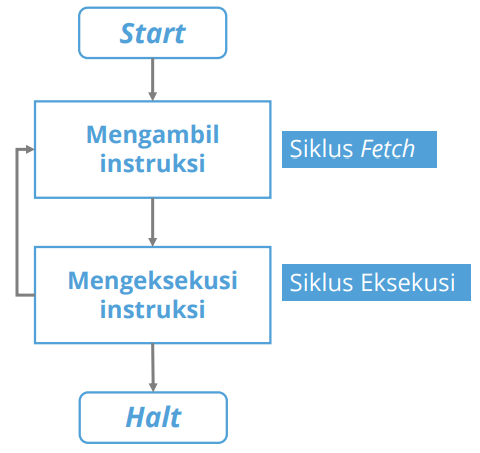
Terdapat 4 macam register yang penting bagi eksekusi instruksi, yaitu:

1. ***Program Counter (PC)*** atau Pencacah Program: berisi alamat instruksi yang akan diambil.
2. ***Instruction Register (IR):*** berisi instruksi yang terakhir diambil.
3. ***Memory Address Register (MAR):*** berisi alamat sebuah lokasi di dalam memori.
4. ***Memori Buffer Register (MBR):*** berisi sebuah word data yang akan dituliskan ke dalam memori atau word yang terakhir dibaca.
5. SIKLUS INSTRUKSI

Dalam memahami fungsi CPU dan proses interaksinya dengan komponen lain, perlu ditinjau lebih jauh proses kerja eksekusi program, proses eksekusi program paling sederhana adalah dengan mengambil pengolahan instruksi yang terdiri dari 2 langkah, yaitu:

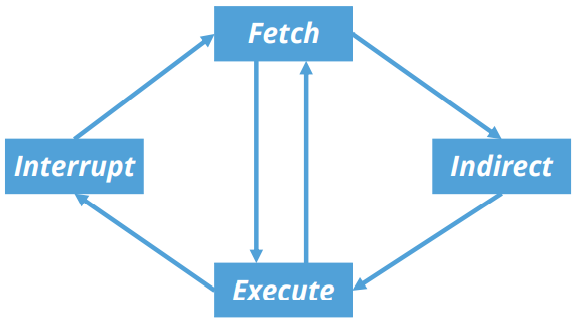
1. Operasi pelaksanaan instruksi (*execute*)
2. Operasi pembacaan instruksi (*fetch*)

**FLOWCHART SIKLUS INSTRUKSI**



1. **Mengambil instruksi (*instruction fetch*)** dari memori Fetch instruksi adalah operasi umum bagi setiap instruksi, dan terdiri dari pembacaan instruksi dari suatu lokasi di dalam memori.
2. **Mengeksekusi instruksi (*instruction execution*)** Eksekusi instruksi dapat melibatkan sejumlah operasi dan tergantung pada sifat-sifat instruksi.
3. ***Interrupt*** Suatu mekanisme yang disediakan bagi modul-modul lain (seperti I/O) untuk dapat menginterupsi operasi normal CPU.

Berikut ini merupakan konsep dari siklus instruksi :



**PROSES SIKLUS INSTRUKSI**

1. Mengambil/membaca instruksi dan mengeksekusi instruksi.
2. Instruksi diperiksa untuk menentukan apakah diperlukan pengalamatan tidak langsung (***indirect***) atau tidak.
3. Interrupt dapat diproses sebelum membaca/mengambil instruksi berikutnya.
4. Instruksi yang dibaca dimuatkan ke dalam sebuah register di dalam CPU.
5. Instruksi berbentuk kode biner yang menentukan apa yang perlu dilakukan oleh CPU.
6. CPU menginterprestasikan instruksi dan melakukan aksi yang diperlukan.

**REFERENSI/DAFTAR PUSTAKA**

1. Indrajani & Martin, Pemograman Berbasis objek dengan java. 2007,