Pertemuan ke 4

Struktur CPU



Computer Organization

Dosen : Eko Budi Setiawan

Tujuan

- Menjelaskan tentang komponen utama CPU dan Fungsi CPU
- Membahas struktur dan fungsi internal prosesor, organisasi ALU, Control Unit dan Register
- Menjelaskan fungsi processor dalam menjalankan instruksi-instruksi mesin
- Kapasitas memori makin besar
- Bentuknya semakin kecil

Komponen Utama CPU

- Arithmetic and Logic Unit (ALU)
- Control Unit (CU)
- Registers
- CPU Interconections

Arithmetic and Logic Unit (ALU)

- Aritmetic and Logical Unit
  - Bertugas membentuk fungsi-fungsi pengolahan data komputer
  - *Arithmetic Logic Unit* sering disebut dengan bahasa mesin (machine language) karena bagian ini mengerjakan instruksi-instruksi bahasa mesin yang diberikan kepadanya.
  - *Arithmetic Logic Unit* terdiri dari dua bagian yaitu unti arithmetic dan unit logika boolean yang masing-masing memiliki spesifikasi tugas tersendiri

Control Unit [CU]

- Bertugas mengontrol operasi CPU dan secara keseluruhan mengontrol komputer sehingga terjadi sinkronisasi kerja antar komponen dalam menjalankan fungsi-fungsi operasinya

- Termasuk dalam tanggung jawab unit kontrol adalah mengambila instruksi-instruksi dari memori utama dan menentukan jenis instruksi tersebut

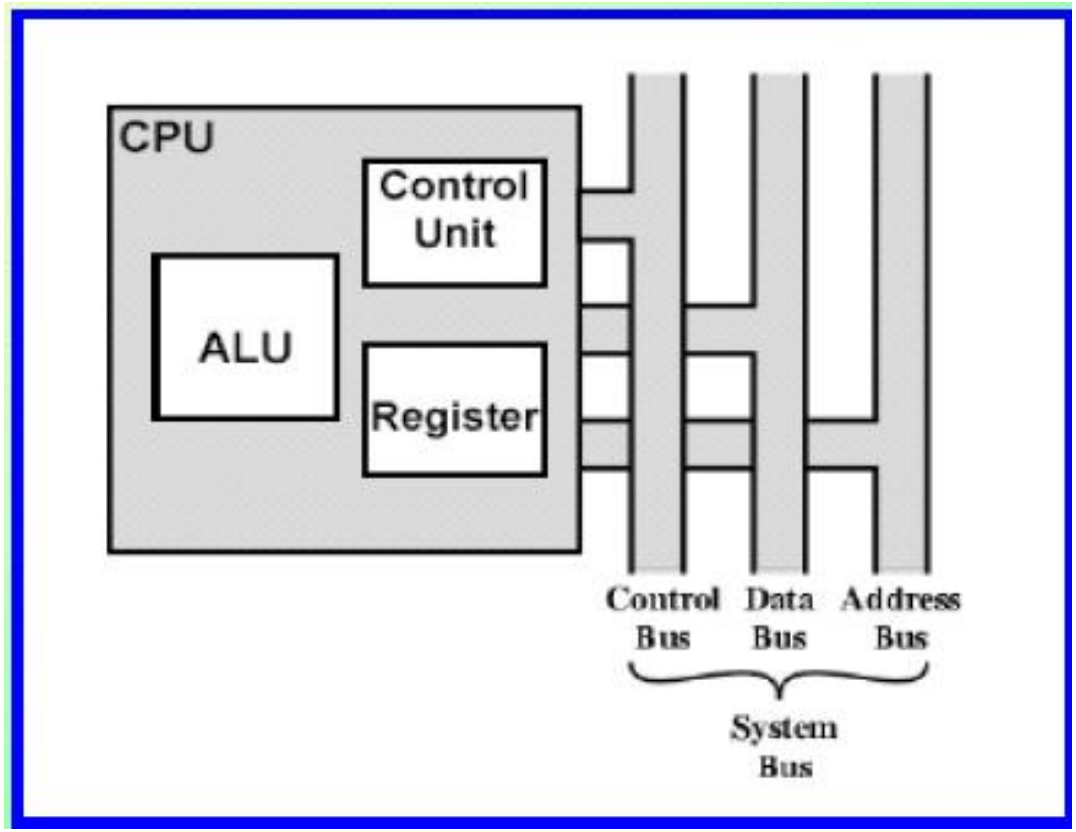
Registers

- Media penyimpan internal CPU yang digunakan saat proses pengolahan data
- Memori ini bersifat sementara, biasanya digunakan untuk menyimpan data saat diolah ataupun data untuk pengolahan selanjutnya

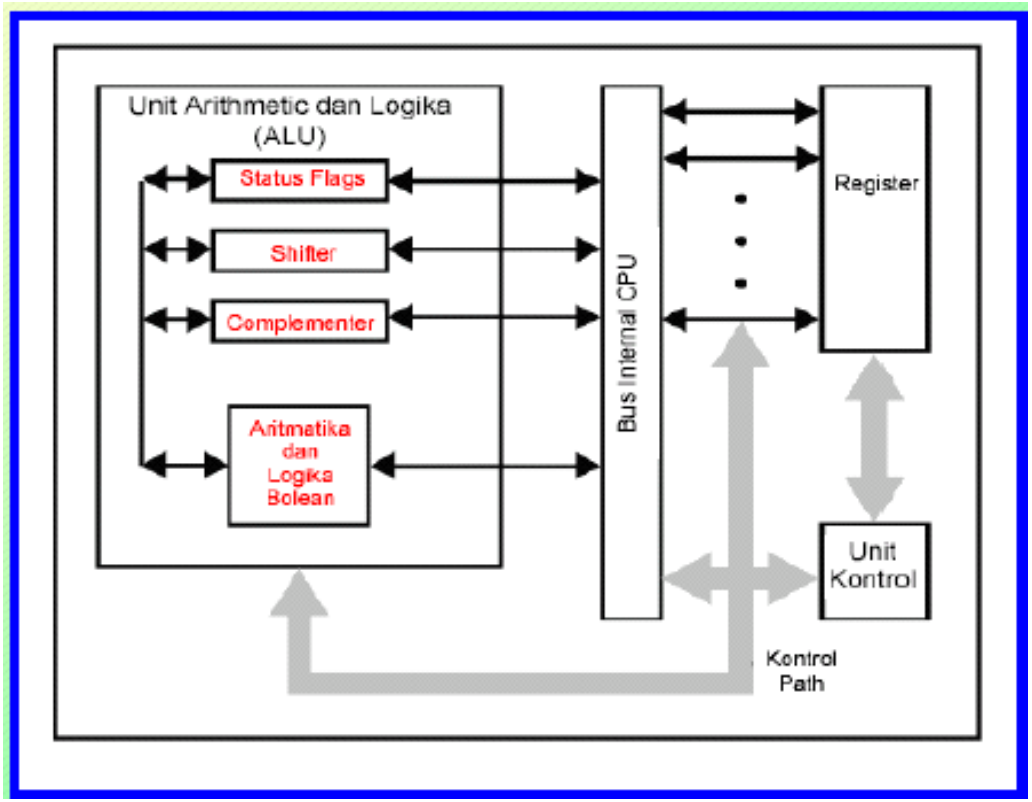
CPU Interconnections

- Sistem koneksi dan bus yang menghubungkan komponen internal dan bus-bus eksternal CPU
- Komponen internal CPU yaitu ALU, unit kontrol dan register-register
- Komponen eksternal CPU : sistem lainnya, seperti memori utama, piranti masukan dan keluaran

Komponen Internal CPU



Stuktur detail internal CPU

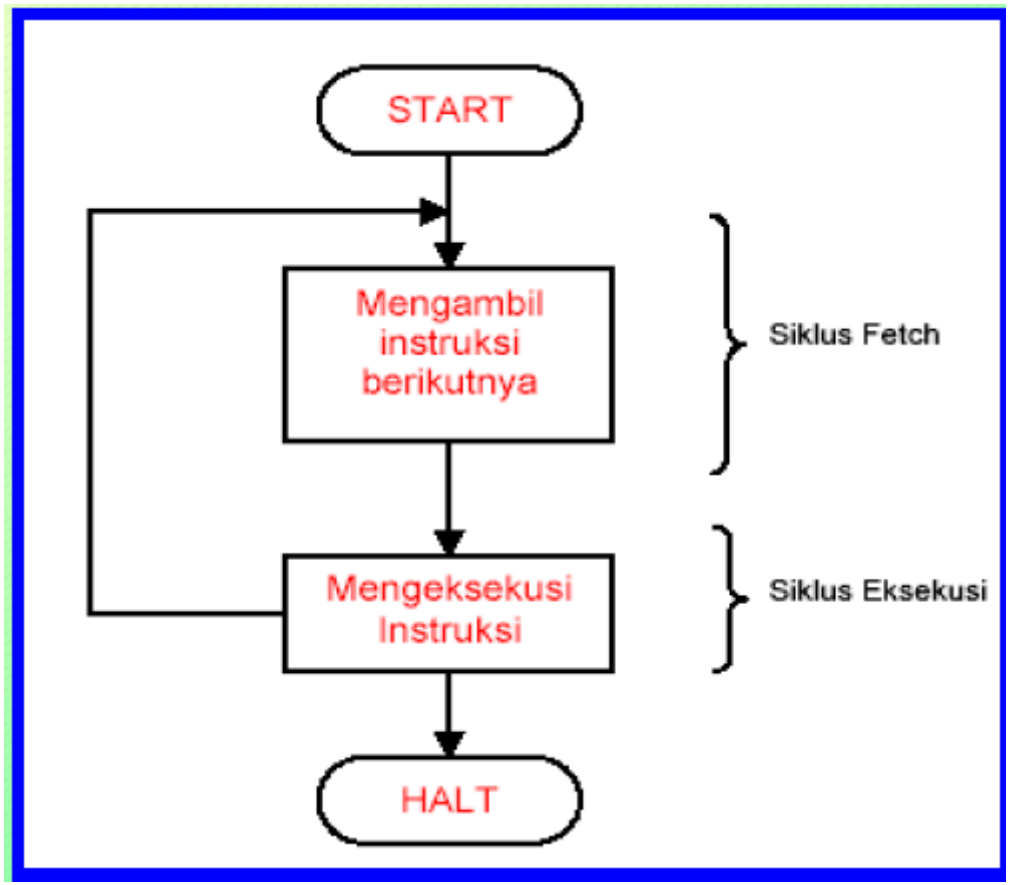


Fungsi CPU

- Menjalankan program-program yang disimpan dalam memori utama dengan cara mengambil instruksi-instruksi, menuji instruksi tersebut dan mengeksekusinya satu persatu sesuai alur perintah
- Pandangan paling sederhana proses eksekusi program adalah dengan mengambil pengolahan instruksi yang terdiri dari dua langkah, yaitu : operasi pembacaan instruksi (fetch) dan operasi pelaksanaan instruksi (execute)

Siklus Instruksi

- Terdiri dari siklus fetch dan siklus eksekusi



Siklus Fetch - Eksekusi

- Pada setiap siklus instruksi, CPU awalnya akan membaca instruksi dari memori
- Terdapat register dalam CPU yang berfungsi mengawasi dan menghitung instruksi selanjutnya, yang disebut dengan Program Counter (PC)
- PC akan menambah satu hitungannya setiap kali CPU membaca instruksi
- Instruksi – instruksi yang dibaca akan dibuat dalam refister instruksi (IR)
- Instruksi – instruksi ini dalam bentuk kode-kode biner yang dapat di interpretasikan oleh CPU kemudian dilakukan aksi yang diperlukan

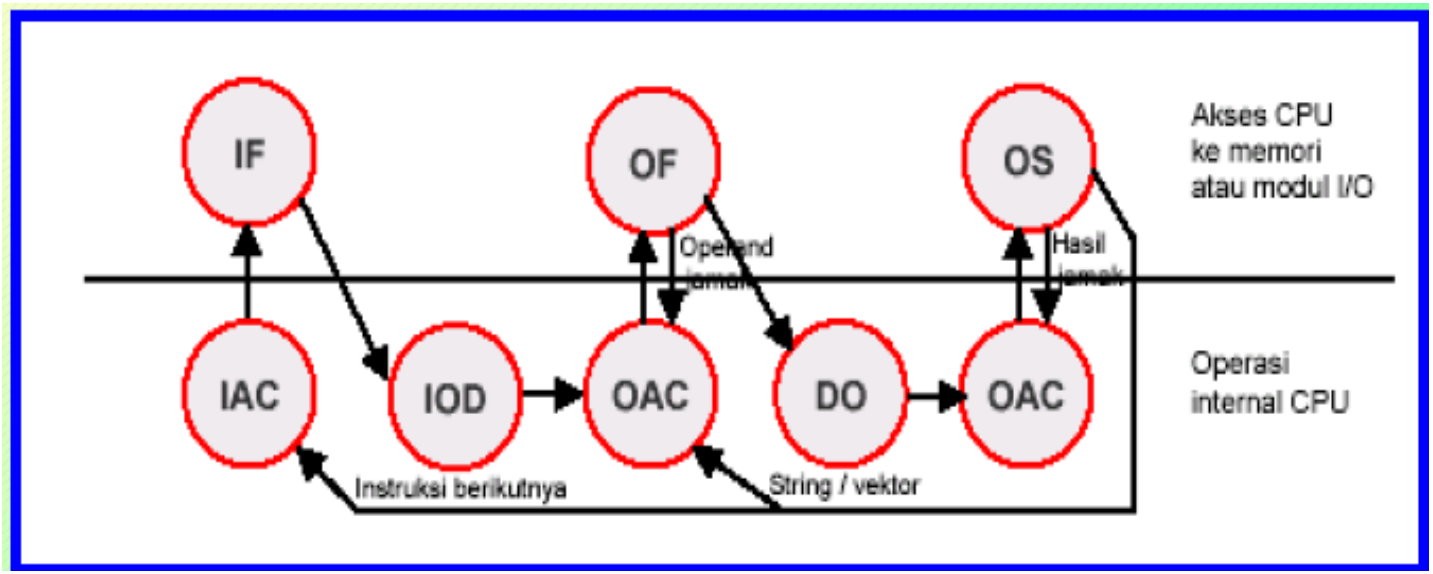
Aksi CPU

- CPU – Memori, perpindahan data dari CPU ke memori dan sebaliknya
- CPU – I/O, perpindahan data dari CPU ke modul I/O dan sebaliknya
- Pengolahan Data, CPU membentuk sejumlah operasi aritmatika dan logika terhadap data
- Kontrol, merupakan instruksi untuk pengontrolan fungsi atau kerja. Misalnya instruksi pengubahan urusan eksekusi

Siklus Eksekusi

- Instruction Address Calculation (IAC), yaitu mengkalkulasi atau menentukan alamat instruksi berikutnya yang akan dieksekusi.
- Instruction Fetch (IF), yaitu membaca atau mengambil instruksi dari lokasi memorinya ke CPU.
- Instruction Operation Decoding (IOD), yaitu menganalisa instruksi untuk menentukan jenis operasi yang akan dibentuk dan operand yang akan digunakan.
- Operand Address Calculation (OAC), yaitu menentukan alamat operand, hal ini dilakukan apabila melibatkan referensi operand pada memori.
- Operand Fetch (OF), mengambil operand dari memori atau dari modul I/O.
- Data Operation (DO), yaitu membentuk operasi yang diperintahkan dalam instruksi
- Operand Store (OS), yaitu menyimpan hasil eksekusi kedalam memori

Diagram siklus instruksi



Fungsi Interupsi

- Mekanisme penghentian atau pengalihan pengolahan instruksi dalam CPU kepada routine interupsi
- Hampir semua modul (memori dan I/O) memiliki mekanisme yang dapat menginterupsi kerja CPU

Tujuan Interupsi

- Secara umum untuk manajemen pengeksekusian routine instruksi agar efektif dan efisien antar CPU dan modul-modul I/O maupun memori.

- Setiap komponen komputer dapat menjalankan tugasnya secara bersamaan, tetapi kendali terletak pada CPU disamping itu kecepatan eksekusi masing-masing modul berbeda

- Dapat sebagai sinkronisasi kerja antar modul

Kelas sinyal Interupsi

- Program, yaitu interupsi yang dibangkitkan dengan beberapa kondisi yang terjadi pada hasil eksekusi program. Contohnya : aritmatika overflow, pembagian nol, operasi ilegal.
- Timer, adalah interupsi yang dibangkitkan perwaktuan dalam processor. Sinyal ini memungkinkan sistem operasi menjalankan fungsi tertentu secara reguler
- I/O, sinyal interupsi yang dbangkitkan oleh modul I/O sehubungan pemberitahuan kondisi error dan penyelesaian suatu operasi
- Hardware failure, adalah interupsi yang dibangkitkan oleh kegagalan daya atau kesalahan paritas memori

Proses Interupsi

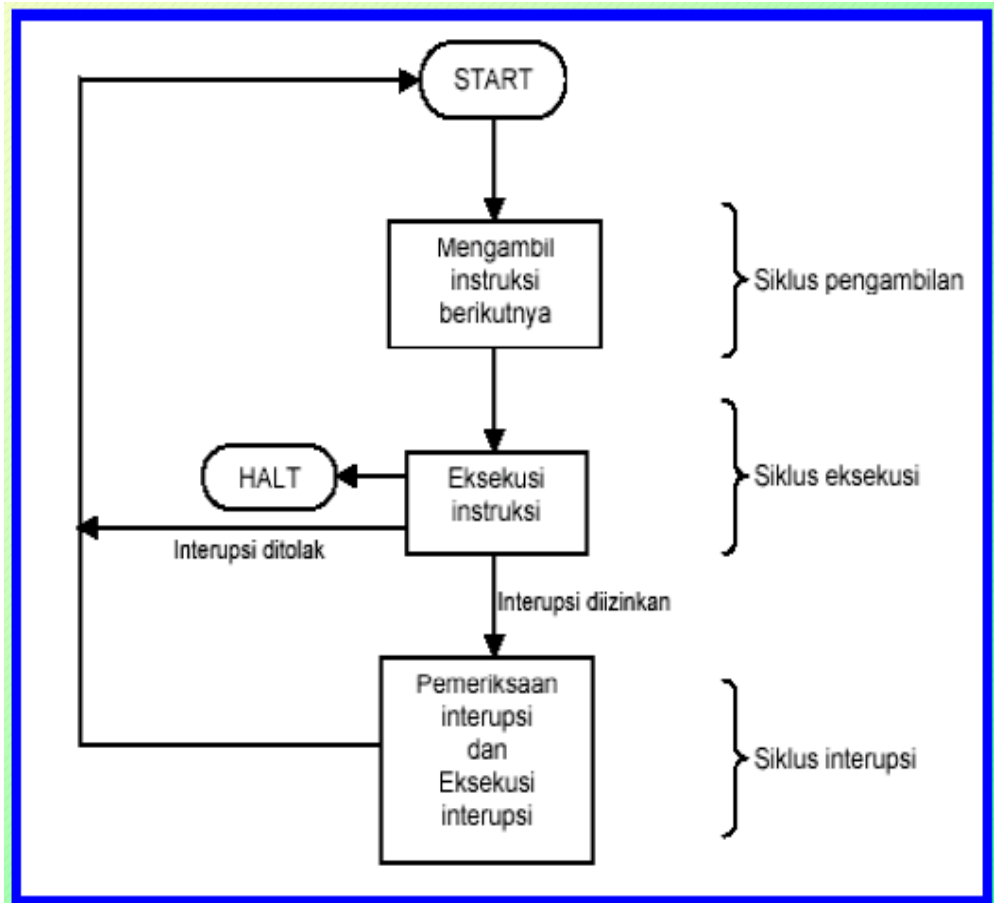
- Dengan adanya mekanisme interupsi, prosesor dapat digunakan untuk mengeksekusi instruksi – instruksi lain
- Saat suatu modul telah selesai menjalankan tugasnya dan siap menerima tugas berikutnya, maka modul ini akan mengirimkan permintaan interupsi ke prosessor
- Kemudian prosessor akan menghentikan eksekusi yang dijalankannya untuk menghandle routine interupsi
- Setelah program interupsi selesai, maka prosesor akan melanjutkan eksekusi programmnya
- Saat sinyal interupsi diterima prosesor ada dua kemungkinan tindakan, yaitu interupsi diterima/ditolak dan interupsi ditolak

Interupsi Ditangguhkan

Apa yang dilakukan Processor ?

- Prosesor menangguhkan eksekusi program yang dijalankan dan menyimpan konteksnya. Tindakan ini adalah menyimpan alamat instruksi berikutnya yang akan dieksekusi dan data lain yang relevan
- Prosesor menyetel program counter (PC) ke alamat awal routine interrupt handler

Siklus Eksekusi Instruksi Processor



Sistem Operasi Kompleks

- Interupsi Ganda (multiple interrupt)
  - Misalnya suatu komputer akan menerima permintaan interupsi saat proses pencetakan dengan printer selesai, disamping itu dimungkinkan dari saluran komunikasi akan mengirimkan permintaan interupsi setiap kali data tiba
- Dapat diambil dua buah pendekatan untuk menangani interupsi ganda ini

Pendekatan Interupsi Ganda

Ada 2 pendekatan :

- Pendekatan ini disebut pengolahan interupsi berurutan / sekuensial
  - Menolak atau tidak mengizinkan interupsi lain saat suatu interupsi ditangani processor
  - Setelah prosesor selesai menangani suatu interupsi maka interupsi lain baru ditangani
- Pengolahan interupsi bersarang yaitu mendefinisikan prioritas bagi interupsi
  - Interrupr handler mengizinkan interupsi berprioritas lebih tinggi ditangani terlebih dahulu

Contoh Kasus

Suatu sistem memiliki tiga perangkat I/O : printer, disk dan saluran komunikasi, masing – masing prioritasnya 2,4 dan 5. Bagaimana proses interupsinya ?

Contoh Kasus

- Pada awal sistem melakukan pencetakan dengan printer, saat itu terdapat pengiriman data pada saluran komunikasi sehingga modul komunikasi meminta interupsi
- Proses selanjutnya adalah oengalihan eksekusi interupsi modul komunikasi, sedangkan interupsi printer ditangguhkan
- Saat pengeksekusian modul komunikasi terjadi interupsi disk, namun karena prioritasnya lebih rendah maka interupsi disk ditangguhkan
- Setelah interupsi modul komunikasi selesai akan dilanjutkan interupsi yang memiliki prioritas lebih tinggi, yaitu disk
- Bila interupsi disk selesai dilanjutkan eksekusi interupsi printer. Selanjutnya dilanjutkan eksekusi program utama