

STRUKTUR FUNGSI CPU

Mata Kuliah Arsitektur Komputer
Program Studi Sistem Informasi 2013/2014
STMIK Dumai
-- Materi 03 --

Acknowledgement

- Main Material:
 - Stallings, William. “Computer Organization and Architecture 6th Edition”.
- Supplements:
 - Rizky N, Andes. “mas-andes.blogspot.com”. 2012.
 - Priguna, Gentur. “Arsitektur dan Organisasi Komputer”.
 - R, Rahajeng. “Struktur dan Fungsi CPU”. 2012

Tujuan Materi 03

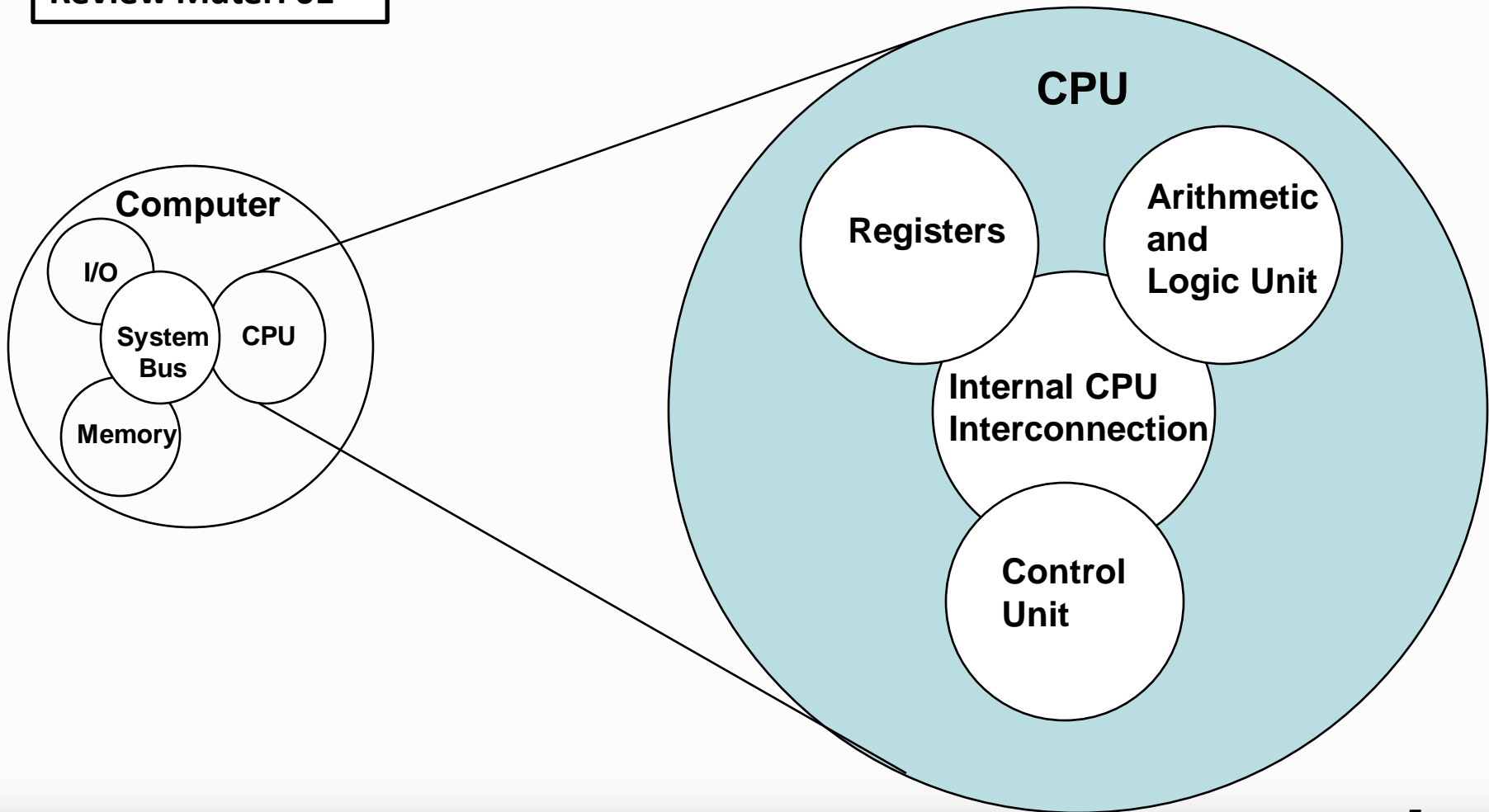
- Menjelaskan tentang komponen utama CPU dan Fungsi CPU
- Membahas struktur dan fungsi internal CPU, organisasi ALU, control unit dan register
- Menjelaskan fungsi CPU dalam menjalankan instruksi-instruksi mesin

CPU

- Central Processing Unit
- Merupakan komponen terpenting dari sistem komputer
- Komponen pengolah data berdasarkan instruksi yang diberikan kepadanya
- Dalam mewujudkan fungsi dan tugasnya, CPU tersusun atas beberapa komponen

Struktur CPU

Review Materi 01



Kesalahan Persepsi

- Apa bedanya CPU dengan Prosesor?
- Prosesor adalah nama lain untuk CPU dalam kaitannya dengan komputer
- CPU hanya merujuk pada unit central (utama) di komputer
- Jadi, CPU bukanlah unit komplet dari sebuah komputer (tanpa monitor) yang dibungkus oleh casingnya. Persepsi ini salah salah karena yang disebut CPU adalah satu komponen di dalamnya yaitu prosesor itu sendiri 😊

Komponen Utama CPU

- Arithmetic and Logic Unit (ALU)
- Control Unit
- Registers
- CPU Interconnections

Arithmetic and Logic Unit (ALU)

- Bertugas membentuk fungsi – fungsi pengolahan data komputer.
- ALU merupakan komponen CPU yang mengerjakan instruksi – instruksi bahasa mesin (machine language) yang diberikan padanya.
- ALU terdiri dari dua bagian, yaitu unit aritmetika dan unit logika boolean, yang masing – masing memiliki spesifikasi tugas tersendiri.
- Tugas lain dari ALU adalah melakukan keputusan dari suatu operasi logika sesuai dengan instruksi program. Operasi logika meliputi perbandingan dua operand dengan menggunakan operator logika tertentu, misal: sama dengan ($=$), tidak sama dengan (\neq), kurang dari ($<$), kurang atau sama dengan (\leq), lebih besar dari ($>$), atau lebih besar atau sama dengan (\geq).

Control Unit

- Mengatur dan mengendalikan masukan (*input*) dan keluaran (*output*) instruksi ke CPU.
- Mengambil instruksi-instruksi dari memori utama.
- Mengambil data dari memori utama (RAM) untuk diproses.
- Mengirim instruksi ke ALU bila ada perhitungan aritmatika atau perbandingan logika serta mengawasi kerja dari ALU.
- Menyimpan hasil proses ke memori utama.

Control Unit

- PC: Program Counter
Berisi alamat memory untuk mengambil instruksi selanjutnya
- IR: Instruction Register
Menampung instruksi yang diambil dari memory
- AC: Accumulator
Penampung sementara untuk data hasil eksekusi instruksi

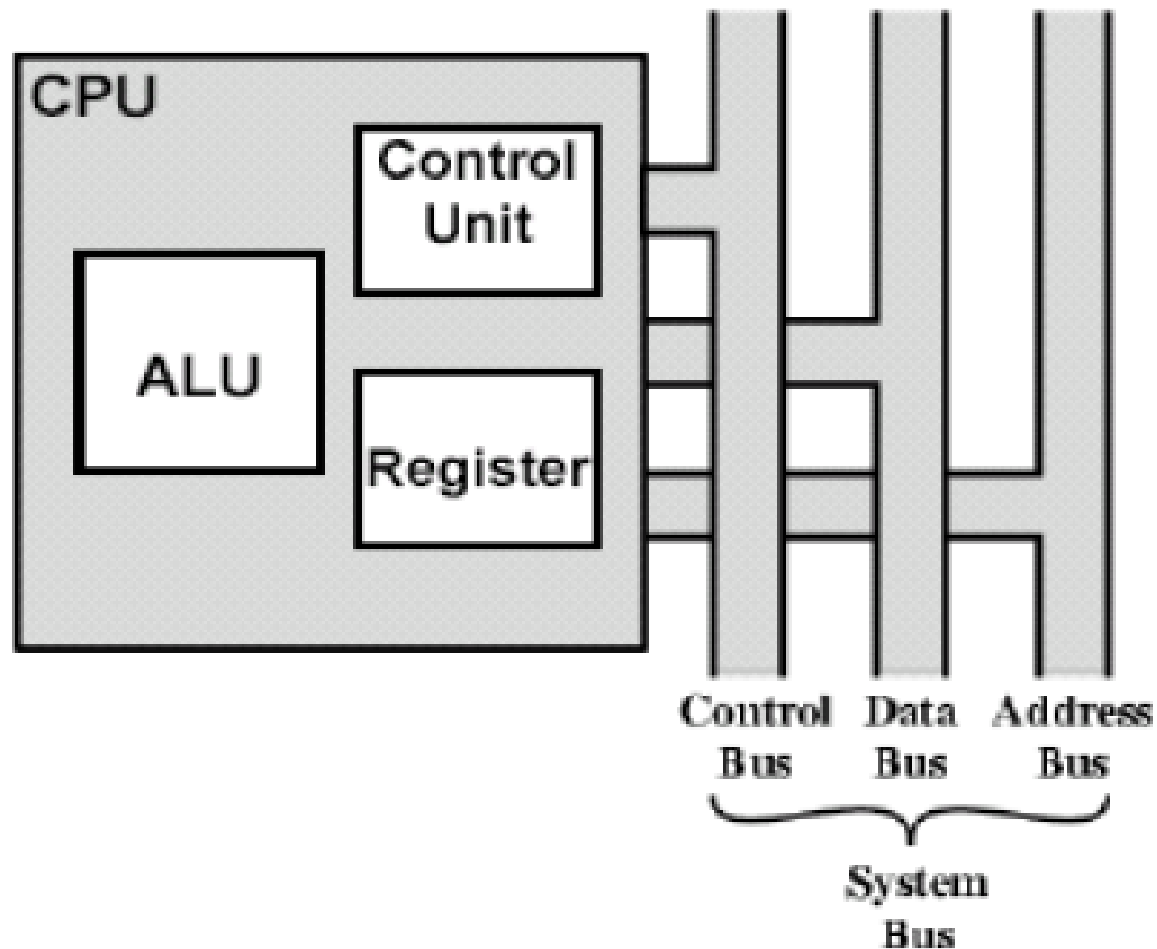
Registers

- 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.

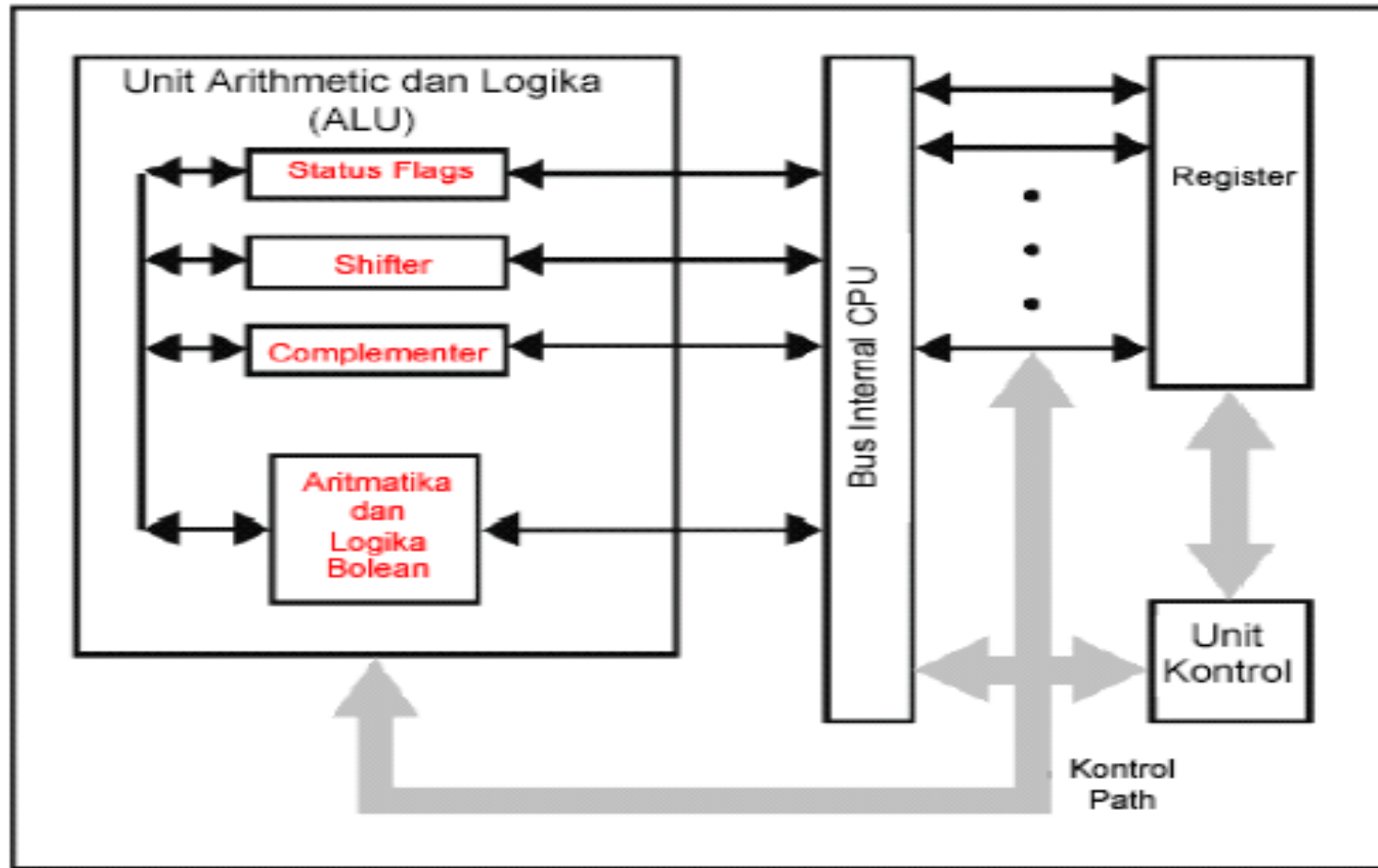
CPU Interconnections

- Sistem koneksi (bus) yang menghubungkan komponen internal dan bus –bus eksternal CPU
- Komponen internal CPU: yaitu ALU, unit kontrol dan register – register.
- Komponen eksternal CPU: memori utama, piranti masukan/keluaran (I/O)

Komponen Internal CPU



Struktur Detail Internal CPU



Apakah Program itu?

- Serangkaian langkah-langkah (*a sequence of steps*)
- Pada tiap langkah, dilakukan sebuah operasi logika atau aritmetik
- Untuk tiap-tiap operasi, disediakan sebuah kode instruksi yang unik.
- Satu kode instruksi mengandung satu set **control signal** yang dapat dimengerti oleh hardware

Eksekusi Program, Siapa yang Terlibat?

- Control Unit di CPU → mengatur aliran program
- Memory → menyimpan baris-baris program yang akan dieksekusi dan hasil outputnya
- Arithmetic and Logic Unit di CPU → untuk melakukan operasi aritmetik dan logika sesuai yang diperintahkan oleh program

Fungsi CPU

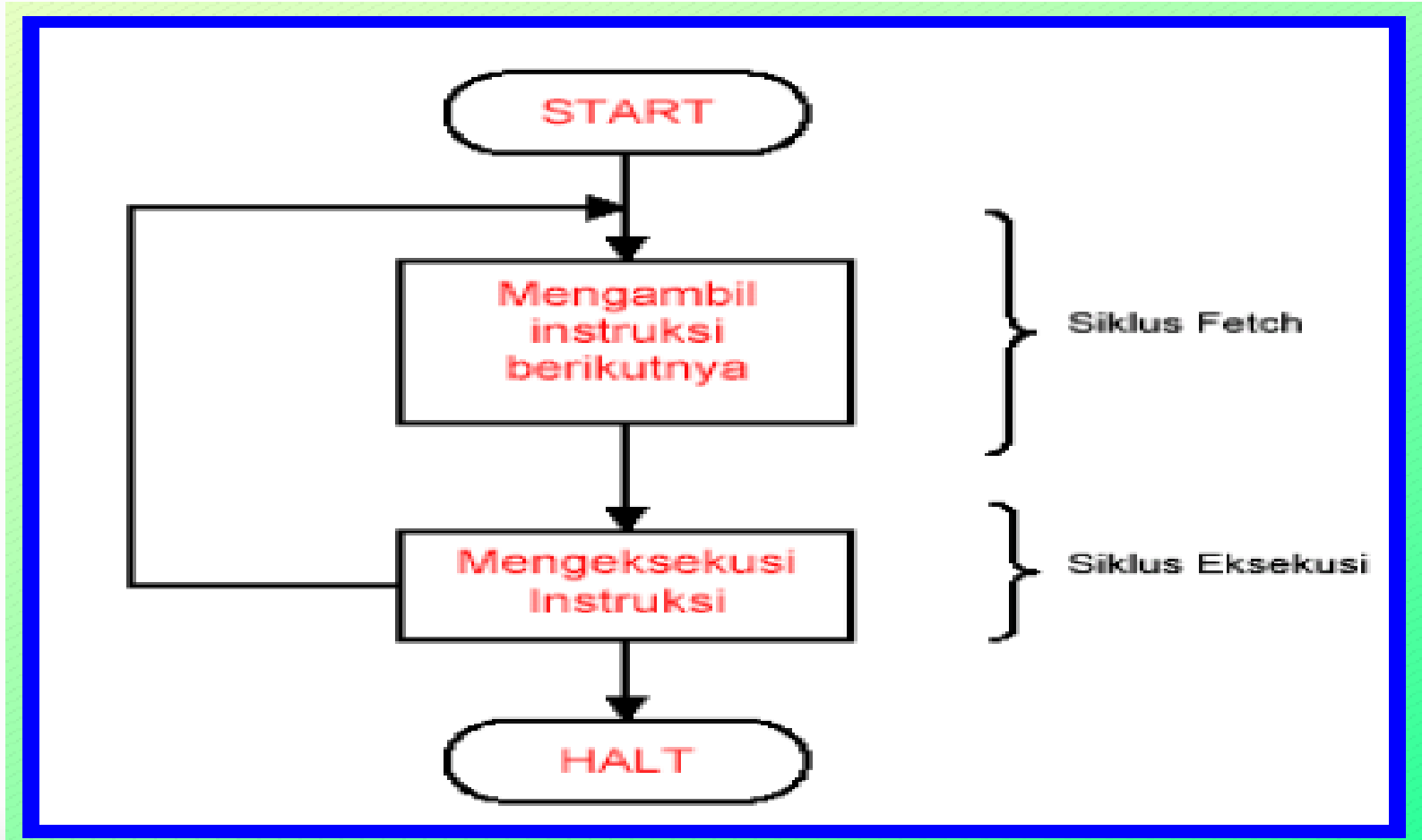
- Menjalankan program – program yang disimpan dalam memori utama dengan cara mengambil instruksi – instruksi, menguji instruksi tersebut dan mengeksekusinya satu persatu sesuai alur perintah.
- Pemrosesan instruksi dalam CPU dibagi atas dua tahap, Tahap-I disebut **Instruction Fetch** (operasi pembacaan instruksi), sedangkan Tahap-II disebut **Instruction Execute** (operasi pelaksanaan instruksi)

Pemrosesan Instruksi

- Tahap-I → pemrosesan CPU dimana Control Unit mengambil data dan/atau instruksi dari main-memory ke register
- Tahap-II → pemrosesan CPU dimana Control Unit menghantarkan data dan/atau instruksi dari register ke main-memory, setelah Instruction Fetch dilakukan.
- Waktu pada tahap-I ditambah tahap-II disebut waktu siklus mesin (machine cycles time), atau disebut juga **satu siklus instruksi**.

Siklus Instruksi

Terdiri dari siklus fetch dan siklus eksekusi



Siklus Fetch

- Program Counter (PC) yang berisi alamat instruksi akan mulai dibaca
- CPU membaca instruksi dari alamat memory yang ditunjuk PC
- Increment PC ($PC = PC + 1$) - kecuali ada perintah lain
- Instruksi di-*load* ke *Instruction Register* (IR)
- CPU menterjemahkan isi instruksi dan memerintahkan aksi yang sesuai dengan isi instruksi

Siklus Eksekusi

Yang termasuk Siklus Eksekusi adalah:

- Transfer data antara CPU dengan main memory
- Transfer data antara CPU dengan modul I/O
- Data processing (operasi aritmetik / logika terhadap data oleh CPU)
- Control
 - Perubahan urutan program (merubah isi PC) dengan instruksi JUMP
 - Menghentikan program di tengah jalan dengan insruksi HALT
- Kombinasi dari hal-hal diatas

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 Program Counter (PC)
- PC akan menambah satu hitungannya setiap kali CPU membaca instruksi

Siklus Fetch - Eksekusi

- Instruksi – instruksi yang dibaca akan dimuat dalam register instruksi (IR).
- Instruksi – instruksi ini sudah dalam bentuk kode – kode biner yang dapat diinterpretasikan oleh CPU kemudian dilakukan aksi yang diperlukan

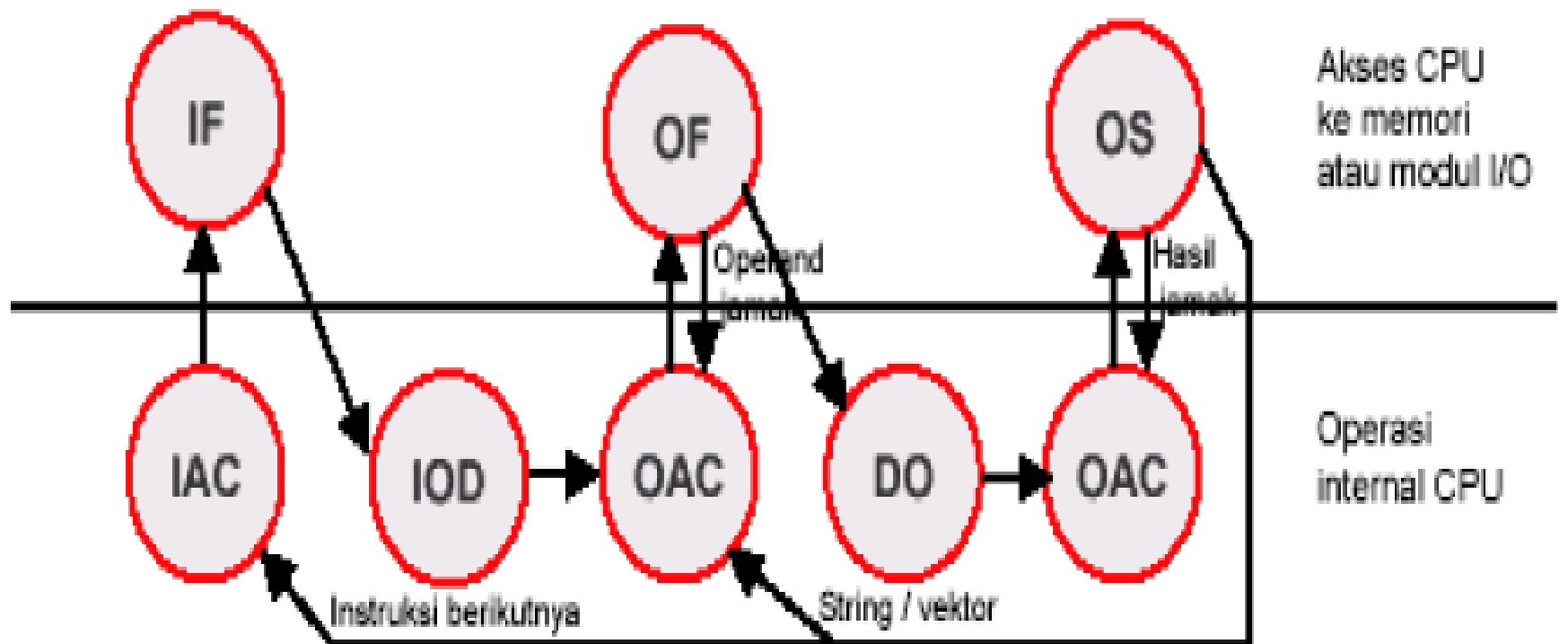
Siklus Instruksi

- *Instruction Address Calculation (IAC)*, yaitu mengkalkulasi atau menentukan alamat instruksi berikutnya yang akan dieksekusi. Biasanya melibatkan penambahan bilangan tetap ke alamat instruksi sebelumnya. Misalnya, bila panjang setiap instruksi 16 bit padahal memori memiliki panjang 8 bit, maka tambahkan 2 ke alamat sebelumnya.
- *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.

Siklus Instruksi

- *Operand Address Calculation (OAC)*, yaitu menentukan alamat operand, hal ini dilakukan apabila melibatkan referensi operand pada memori.
- *Operand Fetch (OF)*, adalah 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 ke dalam memori

Diagram Siklus Instruksi



Interupsi

Interupsi – Apakah itu?

- Suatu instruksi dalam program kadang hasilnya tidak sesuai yang diinginkan
- Ada hambatan yang berasal dari:
 - Kekeliruan matematis (overflow, division by zero)
 - Proses I/O (menulis, membaca data)
 - Kegagalan hardware
 - Timer CPU → Penjadwalan

Interupsi – No Problem?

- Kalau hambatan muncul seketika, bisa langsung dihandle lewat program itu sendiri (ada *handling error code*), misal dengan menyiapkan percabangan berdasarkan kondisi
- Bagaimana kalau eksekusi sebuah instruksi itu butuh waktu? Apakah harus menunggu hasilnya, error atau tidak?

Interupsi – terus bagaimana ya?

- CPU membiarkan sebuah instruksi dieksekusi
- CPU melanjutkan membaca instruksi berikutnya
- TETAPI sebelum membaca instruksi berikutnya, CPU memeriksa, **adakah sinyal interupsi yang masuk** dari instruksi yang tadi?

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.

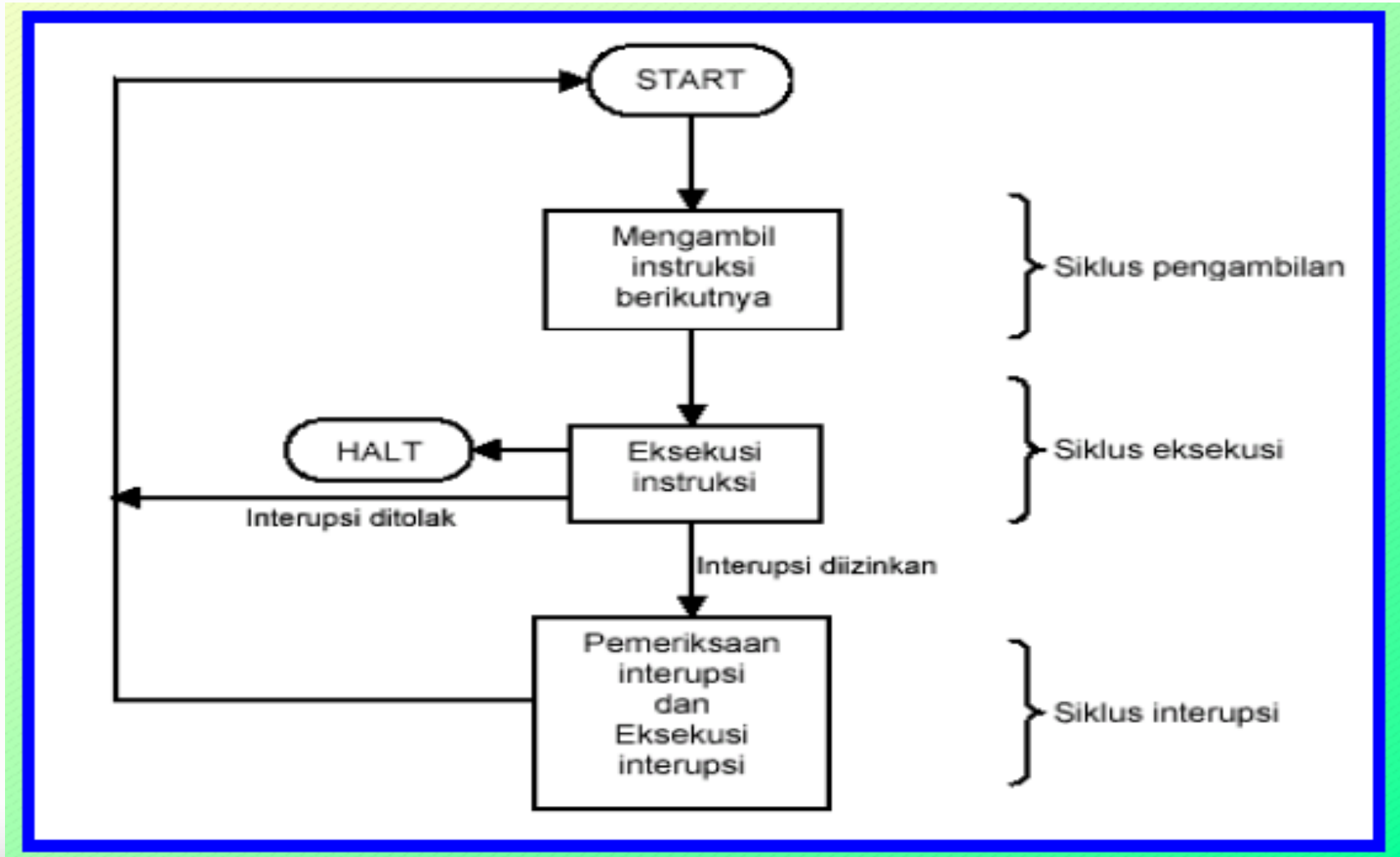
Tujuan Interupsi

- Secara umum, interupsi ditujukan untuk manajemen pengeksekusian 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

Jenis Sinyal Interupsi

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

Siklus Eksekusi Oleh Prosesor Dengan Adanya Fungsi Interupsi



Handle Interupsi

- Sebelum membaca instruksi berikutnya, CPU memeriksa, ada **sinyal interupsi** atau tidak
- Jika tidak ada interupsi, baca instruksi berikutnya
- Jika ada interupsi, arahkan PC (*program counter*) ke alamat instruksi Interrupt Handler, dan hentikan program yg sedang berjalan untuk sementara.
- Jika handling sudah selesai, arahkan kembali PC ke alamat instruksi berikutnya yang seharusnya dibaca.

Apabila Interupsi Ditanggguhkan...

1. CPU menanggguhkan eksekusi program yang dijalankan dan menyimpan konteksnya. Tindakan ini adalah menyimpan alamat instruksi berikutnya yang akan dieksekusi dan data lain yang relevan.
2. CPU menyetel program counter (PC) ke alamat awal *interrupt handler*.

Interupsi Ganda

- Terbagi atas dua jenis:
 - Interupsi Ganda - Sekuensial
 - Interupsi Ganda - Nested (Bersarang)

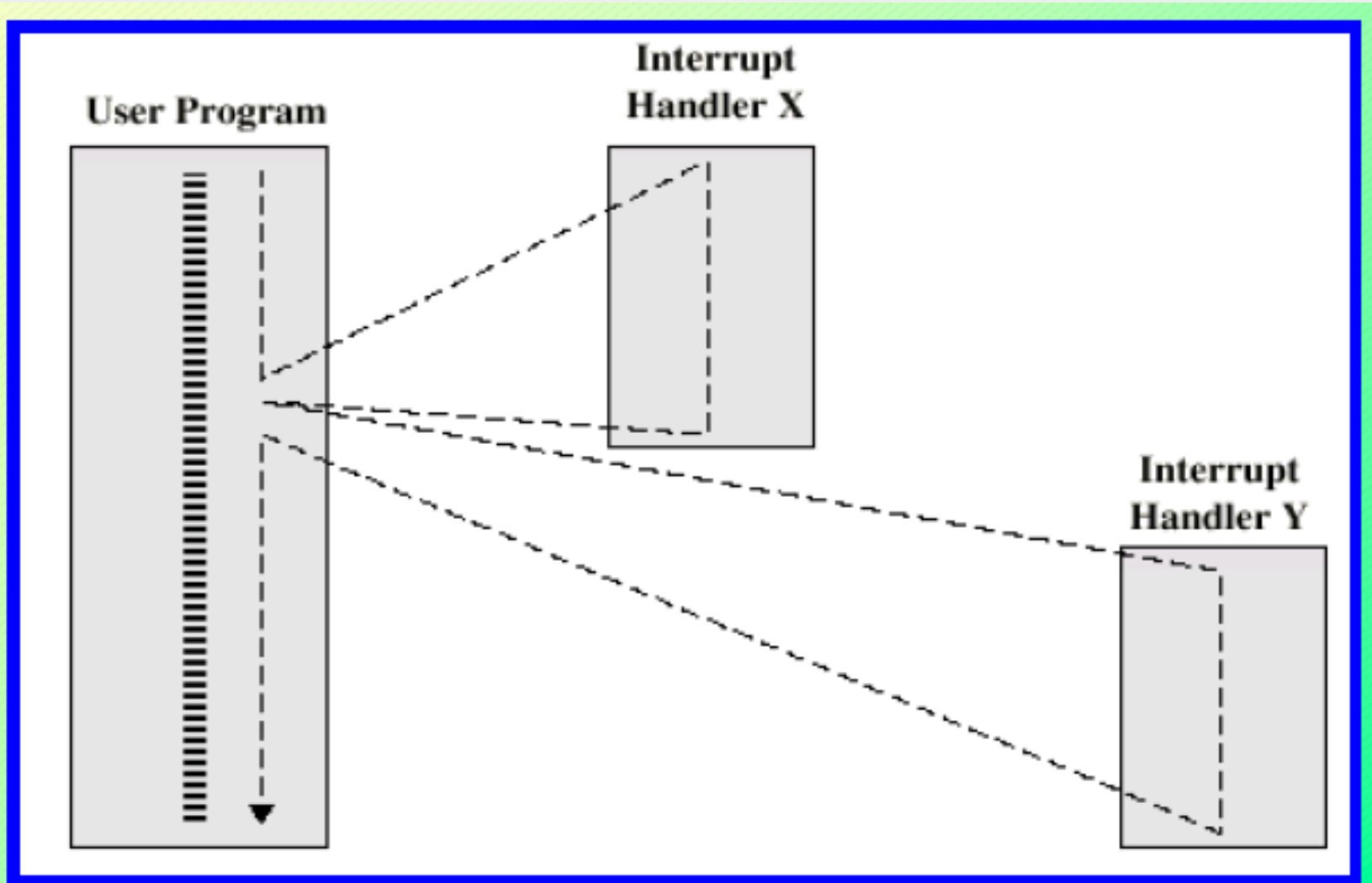
Interupsi Ganda - Sekuensial

1. Menolak atau tidak mengizinkan interupsi lain saat suatu interupsi ditangani prosesor. Kemudian setelah CPU selesai menangani suatu interupsi maka interupsi lain baru di tangani.

Disebut *pengolahan interupsi berurutan/sekuensial*. Pendekatan ini cukup baik dan sederhana karena interupsi ditangani dalam urutan yang cukup ketat.

Kelemahan → tidak memperhitungkan prioritas interupsi.

Interupsi Ganda - Sekuensial



Interupsi Ganda - Nested

2. CPU mendefinisikan tingkat prioritas bagi setiap interupsi dan *interrupt handler* mengizinkan interupsi berprioritas lebih tinggi ditangani terlebih dahulu. Disebut *pengolahan interupsi bersarang (nested)*.

Beberapa **contoh** metode pemilihan prioritas:

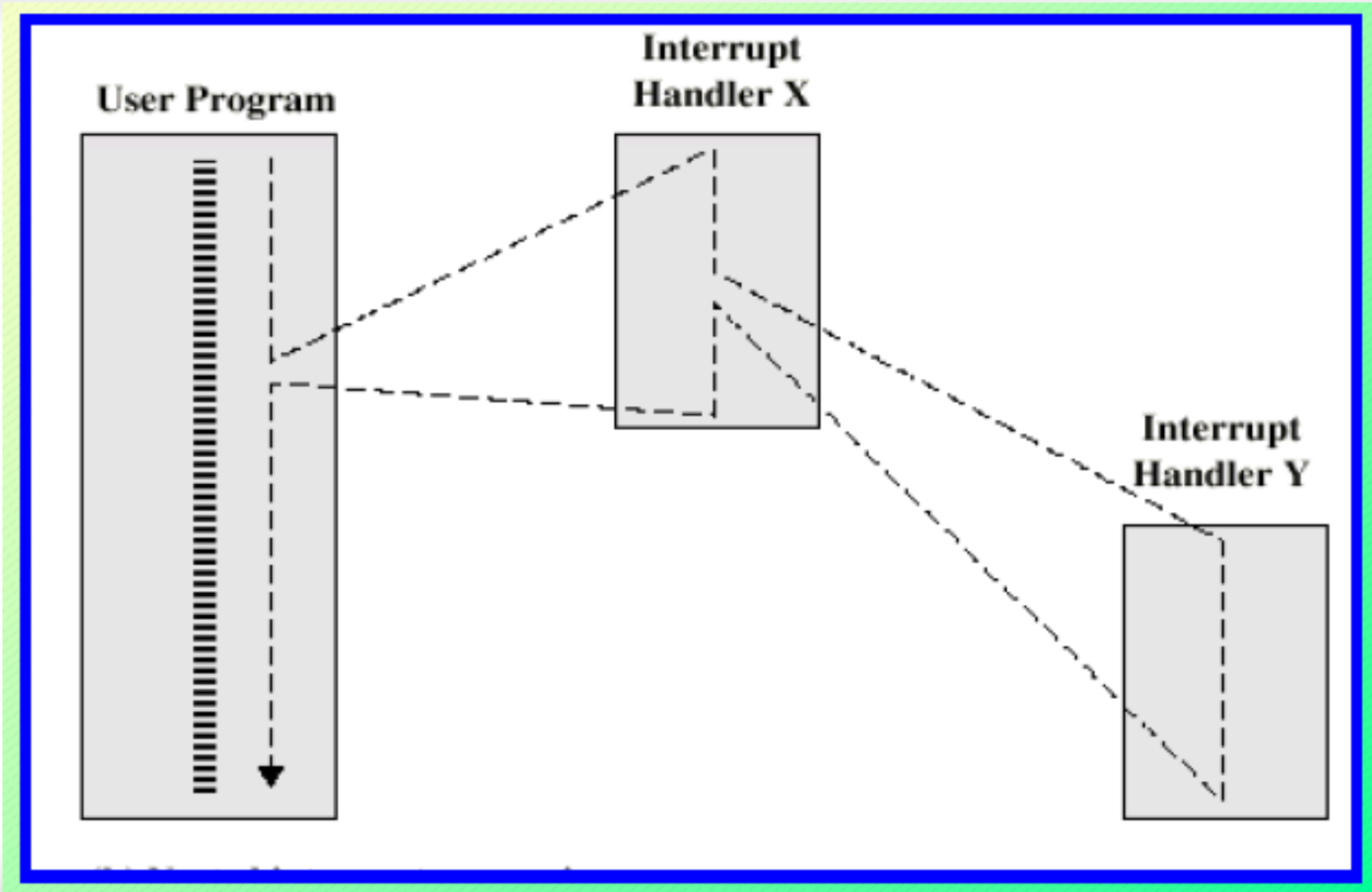
Teknik daisy chain

Teknik daisy chain adalah sebuah teknik menghubungkan peralatan tambahan secara seri ke sebuah computer. Alat yang dipasang pertama kali selalu menjadi prioritas utama pelaksanaan tugas. Sedangkan peralatan yang terletak di akhir rangkaian akan menjadi prioritas terendah.

Menggunakan programmable interrupt controller (Intel 8259A)

Chips ini dapat diprogram untuk menentukan prioritas modul I/O yang lebih dulu ditangani CPU apabila ada permintaan interupsi yang bersamaan.

Interupsi Ganda - Nested



Contoh Kasus

Untuk Interupsi Ganda - Nested

- Suatu sistem memiliki tiga perangkat I/O: printer, disk, dan saluran komunikasi, dengan prioritas masing-masing 2, 4 dan 5 (dalam kasus ini semakin tinggi angka semakin tinggi nilai prioritas).
- Pada awal sistem melakukan pencetakan dengan printer, saat itu terdapat pengiriman data pada saluran komunikasi sehingga modul komunikasi meminta interupsi.
- Proses selanjutnya adalah pengalihan 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.