

Interrupt

Pertemuan X

INTERRUPT

- Interrupt merupakan sinyal yang menyebabkan processor menghentikan suatu pekerjaan tertentu dan berpindah untuk mengerjakan pekerjaan yang lain.
- Jenis-jenis interrupt :
 - Software interrupt
merupakan interrupt yang disebabkan oleh software atau sering disebut sebagai System Call, misalnya suatu program ini mencetak hasil ke printer.
 - Hardware inteerrupt
merupakan interrupt yang disebabkan oleh hardware yang terjadi karena ada akse dari perangkat keras, seperti penekanan tombol keyboard dan penggerakan pointer mouse, flash disk, DVD Room.

Penyebab Interrupt (1)

- Program, sebagai akibat dari eksekusi instruksi, misalnya arithmetic Overflow, Devision by Zero.
- Interrupt yang disebabkan oleh I/O, yang mengindikasikan bahwa proses telah selesai atau terjadi error.
- Kegagalan hardware, disebabkan oleh kegagalan hardware seperti power failur, dll.

Penyebab Interrupt (2)

Table 1.1 Classes of Interrupts

Program	Generated by some condition that occurs as a result of an instruction execution, such as arithmetic overflow, division by zero, attempt to execute an illegal machine instruction, and reference outside a user's allowed memory space.
Timer	Generated by a timer within the processor. This allows the operating system to perform certain functions on a regular basis.
I/O	Generated by an I/O controller, to signal normal completion of an operation or to signal a variety of error conditions.
Hardware failure	Generated by a failure, such as power failure or memory parity error.

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 pewaktuan dalam prosesor. Sinyal ini memungkinkan sistem operasi menjalankan fungsi tertentu secara reguler.
- *I/O*, sinyal interupsi yang dibangkitkan 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 prosesor

Proses Interupsi

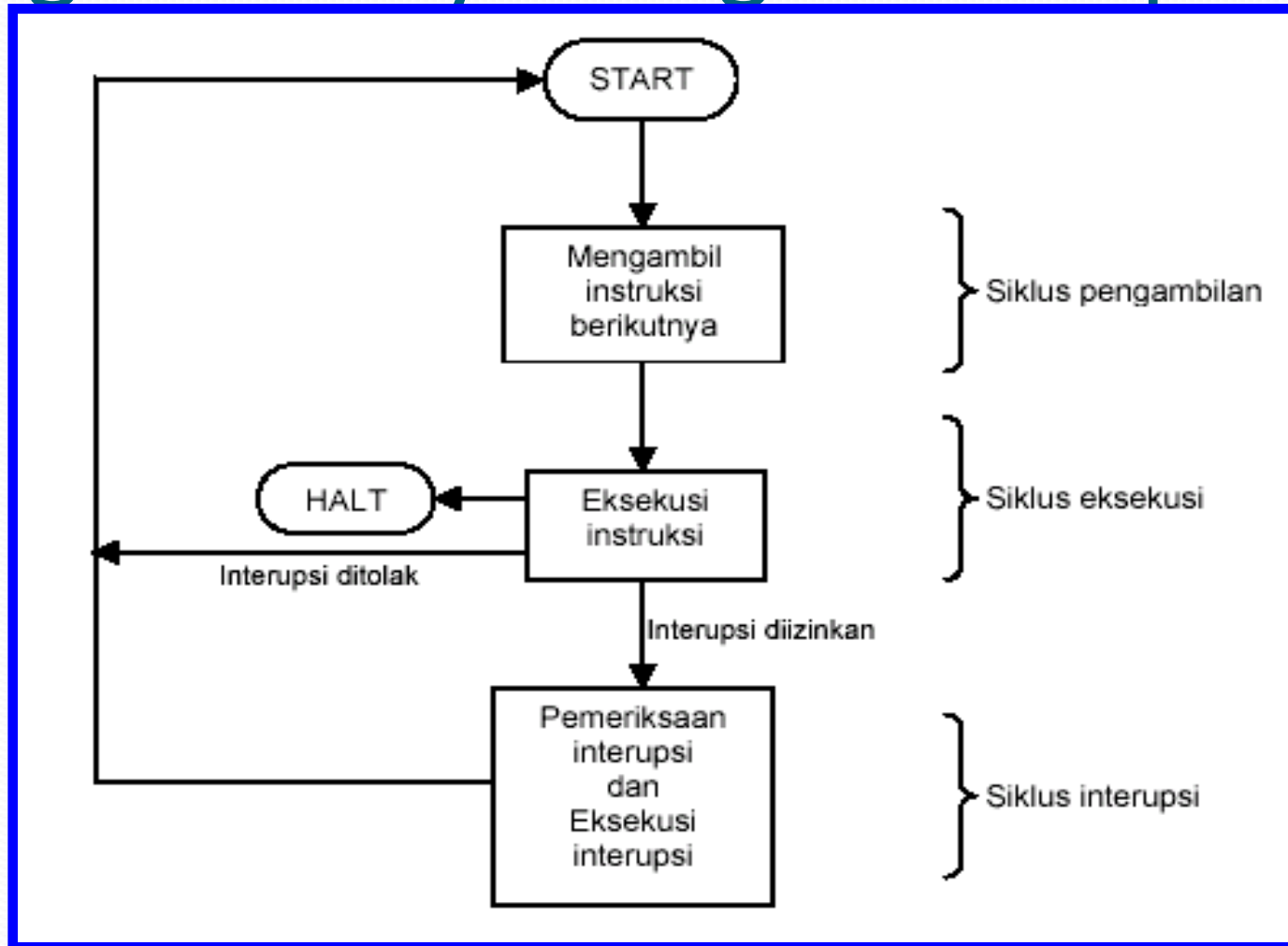
- Kemudian prosesor akan menghentikan eksekusi yang dijalankannya untuk menghandel routine interupsi.
- Setelah program interupsi selesai maka prosesor akan melanjutkan eksekusi programnya kembali.
- Saat sinyal interupsi diterima prosesor ada dua kemungkinan tindakan, yaitu interupsi diterima/ditanggguhkan 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 oleh prosesor dengan adanya fungsi interupsi



Sistem operasi kompleks

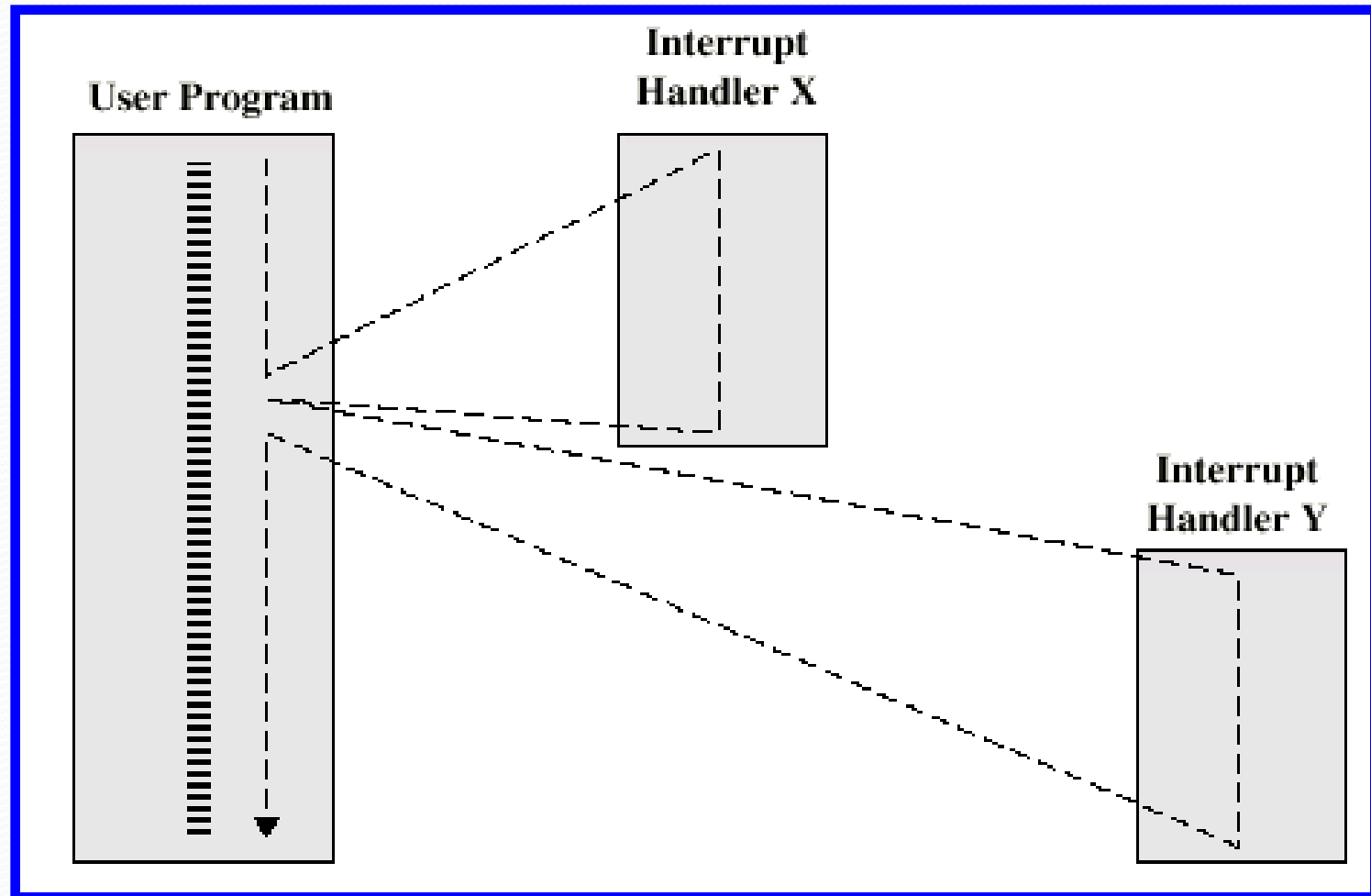
- Interupsi ganda (*multiple interrupt*).
 - Misalnya suatu komputer akan menerima permintaan interupsi saat proses pencetakan dengan printer selesai, disamping itu dimungkinkan dari saluran komunikasi akan mengirimkan permintaan interupsi setiap kali data tiba.
- Dapat diambil dua buah pendekatan untuk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

Pendekatan Interupsi ganda

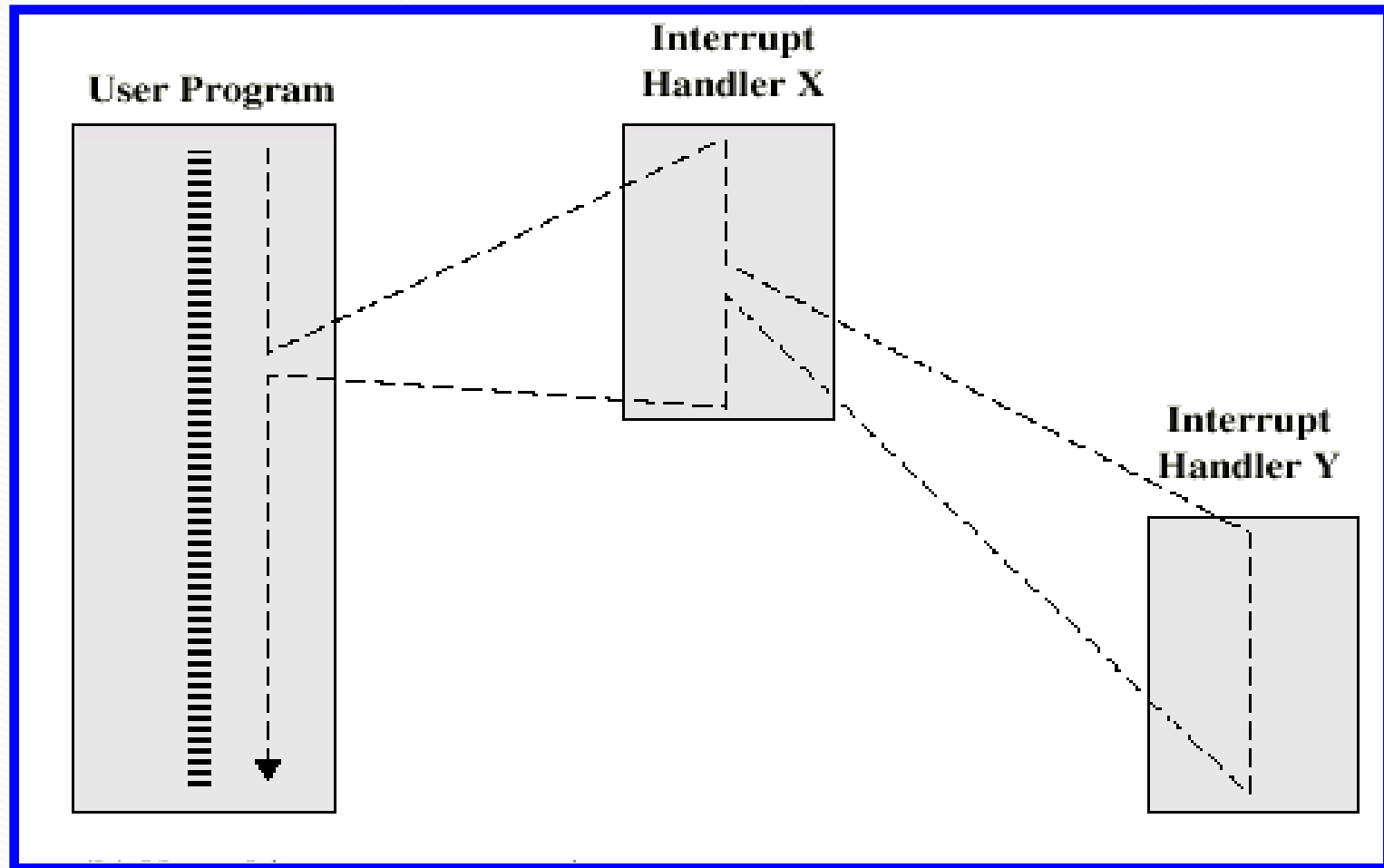
Ada 2 Pendekatan :

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

Multiple Interrupts - Sequential



Multiple Interrupts - Nested



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 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