Pertemuan ke - 12
Unit Masukan dan Keluaran

Riyanto Sigit, ST.
Nur Rosyid, S.kom
Setiawardhana, ST
Hero Yudo M, ST

Politeknik Elektronika Negeri Surabaya

Tujuan

- Menjelaskan system komputer unit masukkan/keluaran
- Menjelaskan prinsip dan teknik unit masukkan/keluaran
- Menjelaskan peralatan luar (External device)

Sistem komputer

- Tiga komponen utama :
  - CPU,
  - Memori (primer dan sekunder)
  - Peralatan masukan/keluaran (*I/O devices*) seperti printer, monitor, keyboard, mouse, dan modem

Modul I/O

- Merupakan peralatan antarmuka (*interface*) bagi sistem *bus* atau switch sentral dan mengontrol satu atau lebih perangkat peripheral.

- Tidak hanya sekedar modul penghubung, tetapi sebuah piranti yang berisi logika dalam melakukan fungsi komunikasi antara peripheral dan *bus* komputer

Modul I/O

Piranti tidak tidak langsung dihubungkan dengan bus sistem komputer , Mengapa ?

- Bervariasinya metode operasi piranti peripheral, sehingga tidak praktis apabila sistem komputer harus menangani berbagai macam sisem operasi piranti peripheral tersebut.
- Kecepatan transfer data piranti peripheral umumnya lebih lambat dari pada laju transfer data pada CPU maupun memori.
- Format data dan panjang data pada piranti peripheral seringkali berbeda dengan CPU, sehingga perlu modul untuk menselaraskannya

Modul I/O

Dua fungsi utama :

- Sebagai piranti antarmuka ke CPU dan memori melalui *bus* sistem.
- Sebagai piranti antarmuka dengan peralatan peripheral lainnya dengan menggunakan link data tertentu

Sistem Masukan & Keluaran Komputer

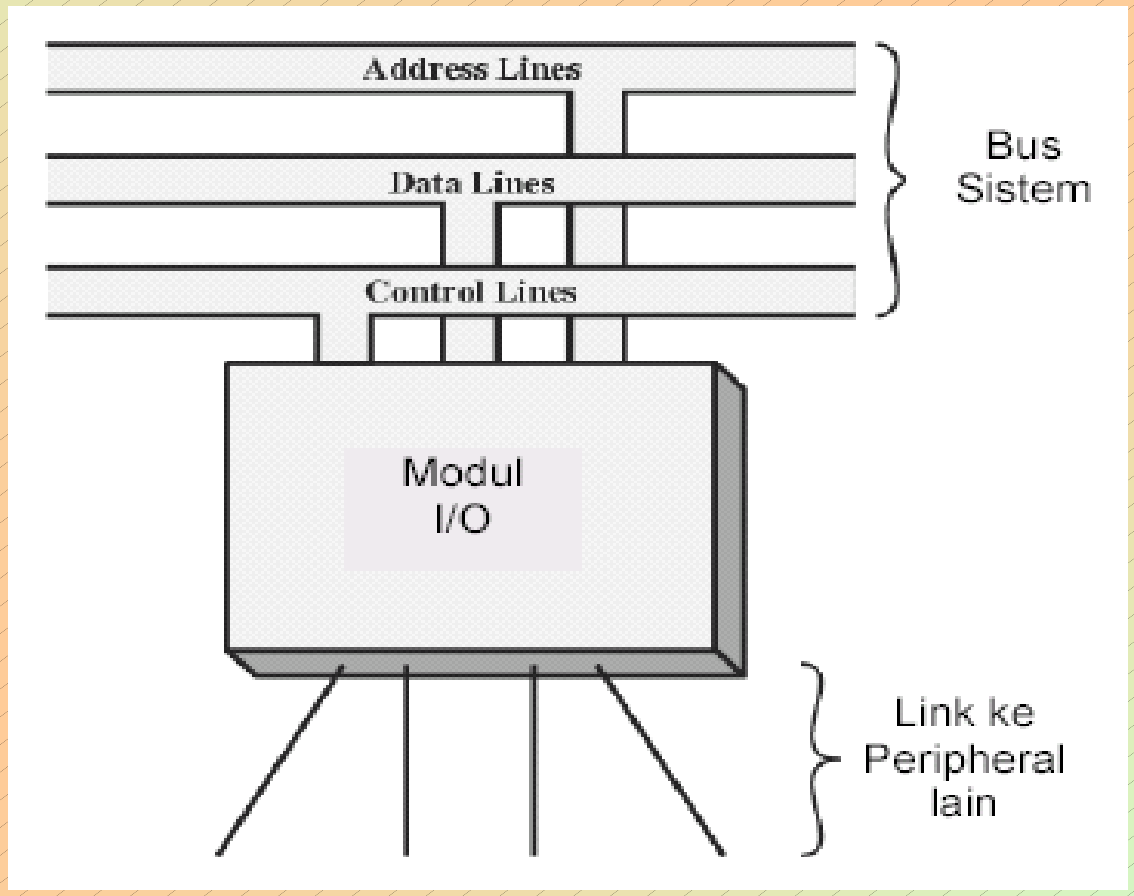
- Bagaimana modul I/O dapat menjalankan tugasnya ?
- Inti mempelajari sistem I/O suatu komputer ?

Sistem Masukan & Keluaran Komputer

- Menjembatani CPU dan memori dengan dunia luar merupakan hal yang terpenting untuk kita ketahui

- Mengetahui fungsi dan struktur modul I/O

Model generik dari suatu modul I/O



Modul I/O

- Modul I/O adalah suatu komponen dalam sistem komputer
  - Bertanggung jawab atas pengontrolan sebuah perangkat luar atau lebih
  - Bertanggung jawab pula dalam pertukaran data antara perangkat luar tersebut dengan memori utama ataupun dengan register – register CPU.
- Antarmuka internal dengan komputer (CPU dan memori utama)
- Antarmuka dengan perangkat eksternalnya untuk menjalankan fungsi – fungsi pengontrolan

Fungsi Modul I/O

- Kontrol dan pewaktuan.
- Komunikasi CPU.
- Komunikasi perangkat eksternal.
- Pem-buffer-an data.
- Deteksi kesalahan

Kontrol dan Pewaktuan

- Fungsi kontrol dan pewaktuan (*control & timing*) merupakan hal yang penting untuk mensinkronkan kerja masing – masing komponen penyusun komputer.
- Dalam sekali waktu CPU berkomunikasi dengan satu atau lebih perangkat dengan pola tidak menentu dan kecepatan transfer komunikasi data yang beragam, baik dengan perangkat internal seperti register – register, memori utama, memori sekunder, perangkat peripheral.
- Proses tersebut bisa berjalan apabila ada fungsi kontrol dan pewaktuan yang mengatur sistem secara keseluruhan
- Transfer data tidak akan lepas dari penggunaan sistem *bus*, maka interaksi CPU dan modul I/O akan melibatkan kontrol dan pewaktuan sebuah arbitrasi *bus* atau lebih

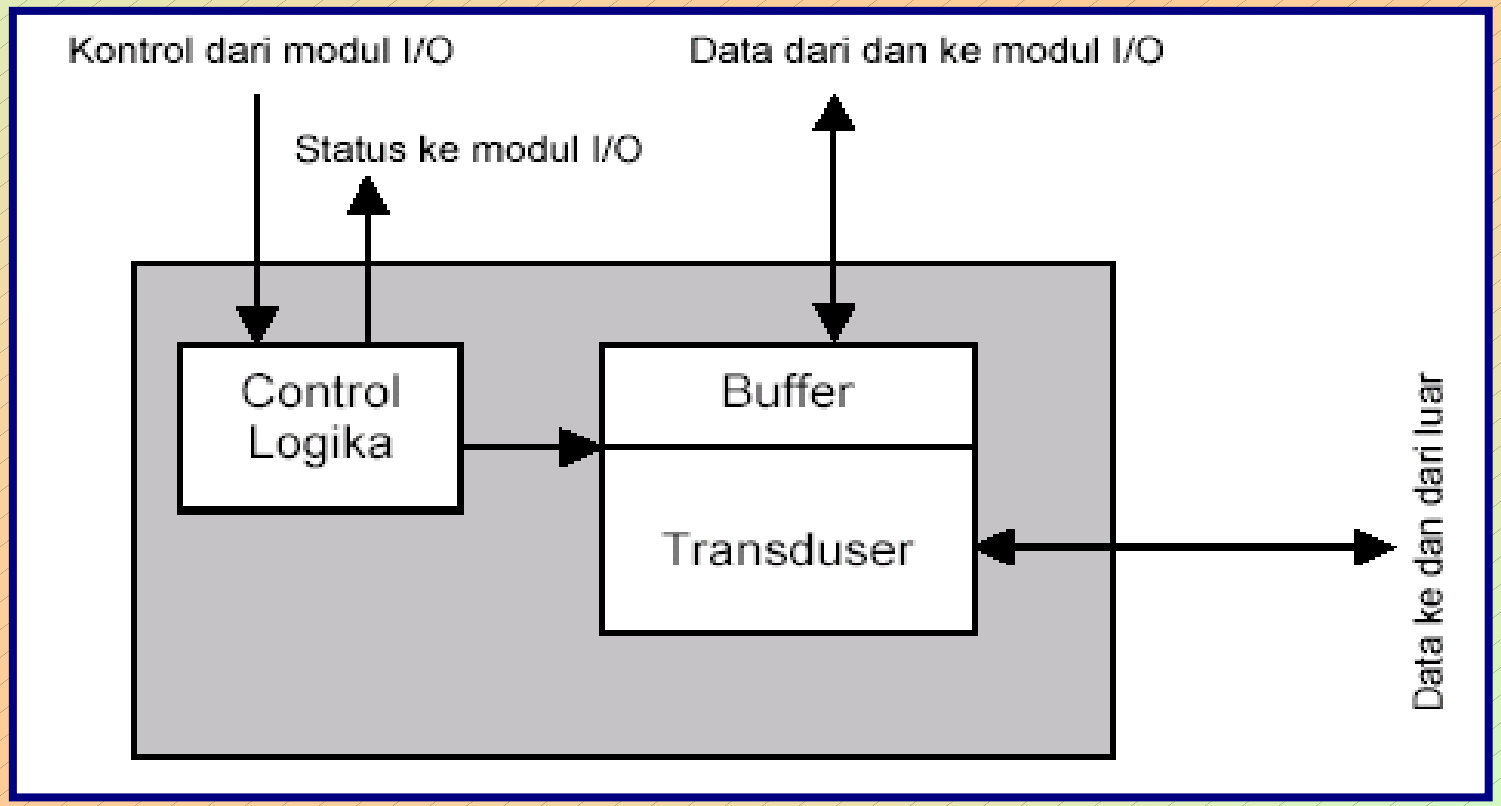
Langkah-langkah pemindahan data dari peripheral ke CPU melalui sebuah modul I/O

- Permintaan dan pemeriksaan status perangkat dari CPU ke modul I/O.
- Modul I/O memberi jawaban atas permintaan CPU.
- Apabila perangkat eksternal telah siap untuk transfer data, maka CPU akan mengirimkan perintah ke modul I/O.
- Modul I/O akan menerima paket data dengan panjang tertentu dari peripheral.
- Selanjutnya data dikirim ke CPU setelah diadakan sinkronisasi panjang data dan kecepatan transfer oleh modul I/O sehingga paket – paket data dapat diterima CPU dengan baik

Proses fungsi komunikasi antara CPU dan modul I/O

- *Command Decoding*, yaitu modul I/O menerima perintah – perintah dari CPU yang dikirimkan sebagai sinyal bagi *bus* kontrol. Misalnya, sebuah modul I/O untuk disk dapat menerima perintah: Read sector, Scan record ID, Format disk.
- *Data,* pertukaran data antara CPU dan modul I/O melalui *bus* data.
- *Status Reporting,* yaitu pelaporan kondisi status modul I/O maupun perangkat peripheral, umumnya berupa status kondisi *Busy* atau *Ready*. Juga status bermacam – macam kondisi kesalahan (*error*).
- *Address Recognition,* bahwa peralatan atau komponen penyusun komputer dapat dihubungi atau dipanggil maka harus memiliki alamat yang unik, begitu pula pada perangkat peripheral, sehingga setiap modul I/O harus mengetahui alamat peripheral yang dikontrolnya

Skema suatu perangkat peripheral



Buffering

- Tujuan utama adalah mendapatkan penyesuaian data sehubungan perbedaan laju transfer data dari perangkat peripheral dengan kecepatan pengolahan pada CPU.

- Laju transfer data dari perangkat peripheral lebih lambat dari kecepatan CPU maupun media penyimpan

Deteksi Kesalahan

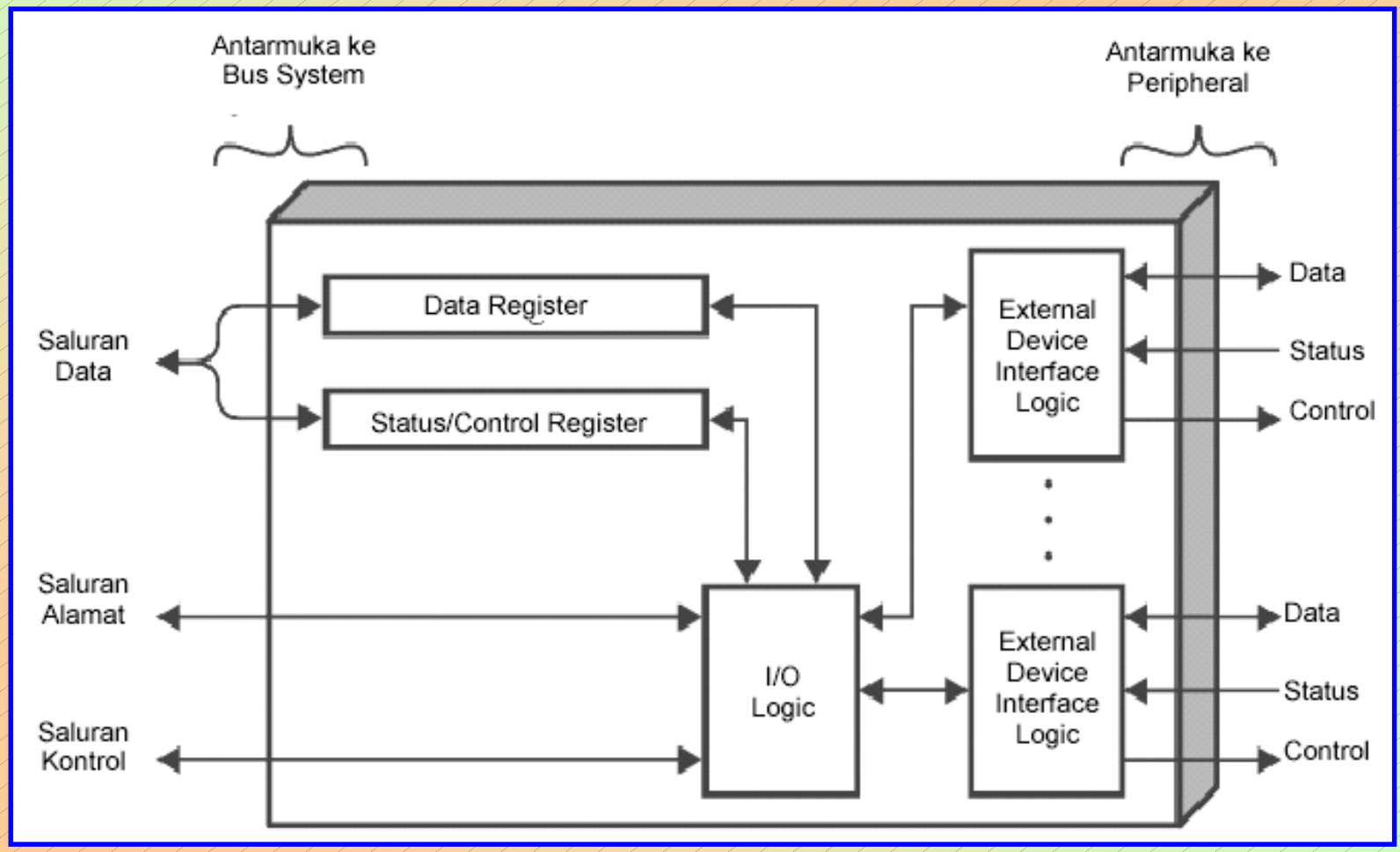
- Bila perangkat peripheral terdapat masalah sehingga proses tidak dapat dijalankan, maka modul I/O akan melaporkan kesalahan tersebut.
  - Misal informasi kesalahan pada peripheral printer seperti: kertas tergulung, pinta habis, kertas habis.

- Teknik yang umum untuk deteksi kesalahan adalah penggunaan bit paritas

Struktur Modul I/O

- Berbagai macam modul I/O seiring perkembangan komputer.
  - Intel 8255A yang sering disebut PPI (*Programmable Peripheral Interface*).
- Bagaimanapun kompleksitas suatu modul I/O, terdapat kemiripan struktur.

Struktur Modul I/O



Blok diagram struktur modul I/O

Struktur Modul I/O

- Antarmuka modul I/O ke CPU melalui *bus* sistem komputer terdapat tiga saluran
  - Saluran data
  - Saluran alamat
  - Saluran kontrol.

- Bagian terpenting adalah blok logika I/O yang berhubungan dengan semua peralatan antarmuka peripheral, terdapat fungsi pengaturan dan switching pada blok ini

I/O Terprogram

- Data saling dipertukarkan antara CPU dan modul I/O.
- CPU mengeksekusi program yang memberikan operasi I/O kepada CPU secara langsung
  - Pemindahan data
  - Pengiriman perintah baca maupun tulis
  - Monitoring perangkat

I/O Terprogram

Kelemahan :

- CPU akan menunggu sampai operasi I/O selesai dilakukan modul I/O sehingga akan membuang waktu, CPU lebih cepat proses operasinya.
- Dalam teknik ini, modul I/O tidak dapat melakukan interupsi kepada CPU terhadap proses – proses yang diinteruksikan padanya.
- Seluruh proses merupakan tanggung jawab CPU sampai operasi lengkap dilaksanakan

Klasifikasi perintah I/O

1. Perintah *control.*

   Perintah ini digunkan untuk mengaktivasi perangkat peripheral dan memberitahukan tugas yang diperintahkan padanya.

2. Perintah *test.*

   Perintah ini digunakan CPU untuk menguji berbagai kondisi status modul I/O dan peripheralnya. CPU perlu mengetahui perangkat peripheralnya dalam keadaan aktif dan siap digunakan, juga untuk mengetahui operasi – operasi I/O yang dijalankan serta mendeteksi kesalahannya.

3. Perintah *read.*

   Perintah pada modul I/O untuk mengambil suatu paket data kemudian menaruh dalam buffer internal. Proses selanjutnya paket data dikirim melalui *bus* data setelah terjadi *sinkronisasi* data maupun kecepatan transfernya.

4. Perintah *write.*

   Perintah ini kebalikan dari *read*. CPU memerintahkan modul I/O untuk mengambil data dari *bus* data untuk diberikan pada perangkat peripheral tujuan data tersebut.

I/O terprogram

Implementasi perintah dalam instruksi I/O :

- *Memory-mapped I/O*
- *Isolated I/O*

Memory-mapped I/O

- Terdapat ruang tunggal untuk lokasi memori dan perangkat I/O.
- CPU memperlakukan register status dan register data modul I/O sebagai lokasi memori dan menggunakan instruksi mesin yang sama untuk mengakses baik memori maupun perangkat I/O.
- Konskuensinya adalah diperlukan saluran tunggal untuk pembacaan dan saluran tunggal untuk penulisan.
- Keuntungan *memory-mapped I/O* adalah efisien dalam pemrograman, namun memakan banyak ruang memori alamat

Isolated I/O

- Dilakukan pemisahan ruang pengalamatan bagi memori dan ruang pengalamatan bagi I/O.
- Dengan teknik ini diperlukan *bus* yang dilengkapi dengan saluran pembacaan dan penulisan memori ditambah saluran perintah output.
- Keuntungan *isolated I/O* adalah sedikitnya instruksi I/O

Interrupt – Driven I/O

- Proses tidak membuang – buang waktu
- Prosesnya :
  - CPU mengeluarkan perintah I/O pada modul I/O, bersamaan perintah I/O dijalankan modul I/O maka CPU akan melakukan eksekusi perintah – perintah lainnya.
  - Apabila modul I/O telah selesai menjalankan instruksi yang diberikan padanya akan melakukan interupsi pada CPU bahwa tugasnya telah selesai

Interrupt – Driven I/O

- Kendali perintah masih menjadi tanggung jawab CPU, baik pengambilan perintah dari memori maupun pelaksanaan isi perintah tersebut.

- Terdapat selangkah kemajuan dari teknik sebelumnya
  - CPU melakukan *multitasking* beberapa perintah sekaligus
  - Tidak ada waktu tunggu bagi CPU = Proses cepat

Interrupt – Driven I/O

- Cara kerja teknik interupsi di sisi modul I/O
  - Modul I/O menerima perintah, misal *read*.
  - Modul I/O melaksanakan perintah pembacaan dari peripheral dan meletakkan paket data ke register data modul I/O
  - Modul mengeluarkan sinyal interupsi ke CPU melalui saluran kontrol.
  - Modul menunggu datanya diminta CPU. Saat permintaan terjadi
  - Modul meletakkan data pada *bus* data
  - Modul siap menerima perintah selanjutnya

Interrupt

- Pengolahan interupsi saat perangkat I/O telah menyelesaikan sebuah operasi I/O :
  - Perangkat I/O akan mengirimkan sinyal interupsi ke CPU.
  - CPU menyelesaikan operasi yang sedang dijalankannya kemudian merespon interupsi.
  - CPU memeriksa interupsi tersebut, kalau valid maka CPU akan mengirimkan sinyal *acknowledgment* ke perangkat I/O untuk menghentikan interupsinya.
  - CPU mempersiapkan pengontrolan transfer ke routine interupsi. Hal yang dilakukan adalah menyimpan informasi yang diperlukan untuk melanjutkan operasi yang tadi dijalankan sebelum adanya interupsi. Informasi yang diperlukan berupa:
    - Status prosesor, berisi register yang dipanggil PSW (*program status word*).
    - Lokasi intruksi berikutnya yang akan dieksekusi.

    Informasi tersebut kemudian disimpan dalam stack pengontrol sistem.

Interrupt

- Pengolahan interupsi saat perangkat I/O telah menyelesaikan sebuah operasi I/O :
  - CPU akan menyimpan PC (*program counter*) eksekusi sebelum interupsi ke stack pengontrol bersama informasi PSW.
  - Mempersiapkan PC untuk penanganan interupsi.
  - CPU memproses interupsi sempai selesai
  - Bila pengolahan interupsi selasai, CPU akan memanggil kembali informasi yang telah disimpan pada stack pengontrol untuk meneruskan operasi sebelum interupsi .

Interrupt

- Teknik yang digunakan CPU dalam menangani program interupsi
  - *Multiple Interrupt Lines*.
  - *Software poll*.
  - *Daisy Chain*.
  - *Arbitrasi bus*

Multiple Interrupt Lines

- Teknik yang paling sederhana
- Menggunakan saluran interupsi berjumlah banyak
- Tidak praktis untuk menggunakan sejumlah saluran *bus* atau pin CPU ke seluruh saluran interupsi modul – modul I/O

Software poll

- CPU mengetahui adanya sebuah interupsi, maka CPU akan menuju ke routine layanan interupsi yang tugasnya melakukan poll seluruh modul I/O untuk menentukan modul yang melakukan interupsi
- Kerugian *software poll*
  - memerlukan waktu yang lama karena harus mengidentifikasi seluruh modul untuk mengetahui modul I/O yang melakukan interupsi

Daisy chain

- Teknik yang lebih efisien
- Menggunakan *hardware poll*
- Seluruh modul I/O tersambung dalam saluran interupsi CPU secara melingkar (*chain*)
- Apabila ada permintaan interupsi, maka CPU akan menjalankan sinyal *acknowledge* yang berjalan pada saluran interupsi sampai menjumpai modul I/O yang mengirimkan interupsi

Arbitrasi bus

- Modul I/O memperoleh kontrol *bus* sebelum modul ini menggunakan saluran permintaan interupsi
- Hanya akan terdapat sebuah modul I/O yang dapat melakukan interupsi

Pengontrol Interrupt Intel 8259A

- Intel mengeluarkan chips 8259A
- Sebagai *interrupt arbiter* pada mikroprosesor Intel 8086
- Manajemen interupsi modul - modul I/O
- Chips ini dapat diprogram untuk menentukan prioritas modul I/O yang lebih dulu ditangani CPU apabila ada permintaan interupsi yang bersamaan
- Mode – mode interupsi ?

Mode pada Interrupt Intel 8259A

- *Fully Nested*

  Permintaan interupsi dengan prioritas mulai 0 (IR0) hingga 7(IR7).

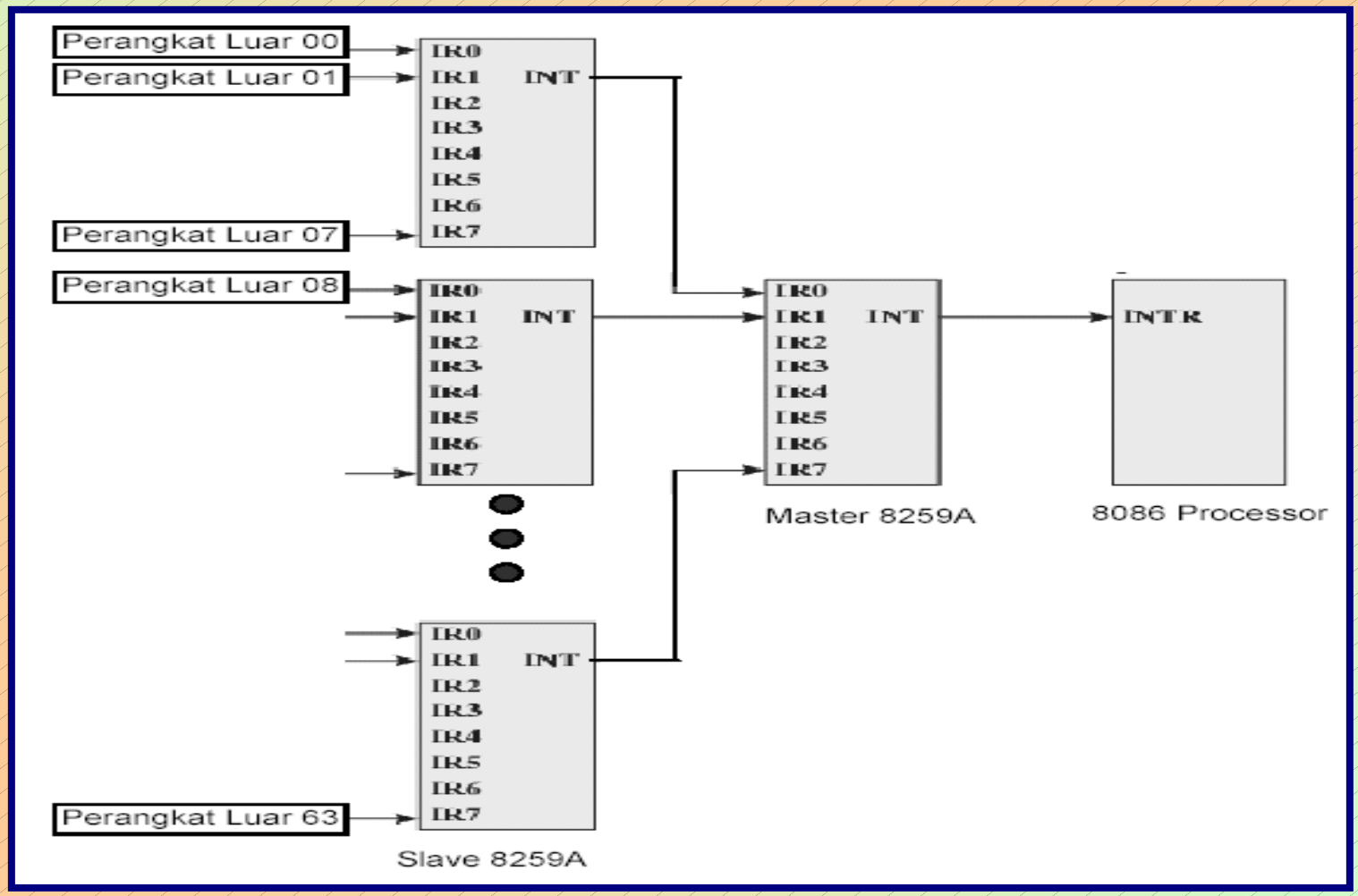
- *Rotating*

  Bila sebuah modul telah dilayani interupsinya akan menempati prioritas terendah.

- *Special Mask*

  Prioritas diprogram untuk modul I/O tertentu secara spesial.

Pemakaian pengontrol interupsi 8559A pada 8086



Kesimpulan

1. Modul I/O merupakan peralatan antarmuka (*interface*) bagi sistem *bus* atau switch sentral dan mengontrol satu atau lebih perangkat peripheral.
2. Modul I/O adalah suatu komponen dalam sistem komputer yang bertanggung jawab atas pengontrolan sebuah perangkat luar atau lebih dan bertanggung jawab pula dalam pertukaran data antara perangkat luar tersebut dengan memori utama ataupun dengan register – register CPU.

Soal-soal

1. Jelaskan sistem komputer unit masukkan/keluaran?
2. Jelaskan prinsip dan teknik unit masukkan/keluaran?
3. Pada vectored interrupts, sebutkan alas an kenapa modul I/O menempatkan vector pada saluran data dan bukannya pada saluran alamat.