

eBook

Pemrosesan Input/Output (I/O)

Minggu 8

Penyusun :

1. Imam Purwanto, S.Kom, MMSI
2. Ega Hegarini, S.Kom., MM
3. Rifki Amalia, S.Kom., MMSI
4. Arie Kusumawati, S.Kom

**Fakultas Teknologi Industri
Universitas Gunadarma
2013**

Input / Output (I/O)

Fungsi : Memindahkan informasi antara CPU atau memori utama dengan dunia luar

I/O terdiri :

- Piranti I/O (peripheral)
- Pengendali I/O (device controller)
- Perangkat lunak

Input / Output (I/O) (Lanj.)

Proses transfer informasi antara CPU dengan sebuah peripheral :

- Memilih I/O dan mengujinya.
- Menginisialisasi transfer & mengkoordinasikan pengaturan waktu operasi I/O.
- Mentransfer informasi.
- Menghentikan proses transfer.

Klasifikasi piranti I/O

Klasifikasi piranti I/O terdiri 3 kelompok:

- Kelompok yang memasukkan informasi (input), contoh : keyboard, ADC (analog to digital converter), scanner
- Kelompok yang menampilkan informasi (output), contoh : VDU/Video Display Unit (monitor), printer.
- Kelompok yang melayani input dan output, contoh : Floppy disk, harddisk.

Pengaksesan I/O terdiri dari 2 cara :

1. Memory mapped I/O

Piranti I/O dihubungkan sebagai lokasi memori virtual dimana port I/O tergantung memori utama

Karakteristik:

- Port I/O dihubungkan ke bus alamat.
- Piranti input sebagai bagian memori yang memberikan data ke bus data. Piranti output sebagai bagian memori yang memiliki data yang tersimpan di dalamnya.
- Port I/O menempati lokasi tertentu pada ruang alamat dan diakses seolah-olah adalah lokasi memori.

Pengaksesan I/O terdiri dari 2 cara : (Lanj.)

2. I/O mapped I/O (I/O isolated)

Piranti I/O dihubungkan sebagai lokasi terpisah dengan lokasi memori, dimana port I/O tidak tergantung pada memori utama.

Karakteristik:

- Port I/O tidak tergantung memori utama.
- Transfer informasi dilakukan di bawah kendali sinyal kontrol yang menggunakan instruksi INPUT dan OUTPUT.
- Operasi I/O tergantung sinyal kendali dari CPU.
- Instruksi I/O mengaktifkan baris kendali read/write pada port I/O, sedangkan instruksi memori akan mengaktifkan baris kendali read/write pada memori.
- Ruang memori dan ruang alamat I/O menyatu, sehingga dapat memiliki alamat yang sama.

Kelebihan dan kekurangan:

- I/O mapped I/O lebih cepat dan efisien, karena lokasi I/O terpisah dengan lokasi memori.
- I/O mapped I/O mempunyai keterbatasan jumlah instruksi yang dapat digunakan untuk operasi I/O.

Operasi I/O terbagi menjadi 3 metode :

1. I/O terprogram

Metode di mana CPU mengendalikan operasi I/O secara keseluruhan dengan menjalankan serangkaian instruksi I/O dengan sebuah program.

Karakteristik:

- Program tersebut digunakan untuk memulai, mengarahkan dan menghentikan operasi-operasi I/O.
- Membutuhkan sejumlah perangkat keras (register) yaitu:
 - Register status, berisi status piranti I/O dan data yang akan dikirimkan.
 - Register buffer, menyimpan data sementara sampai CPU siap menerimanya.
 - Pointer buffer, menunjuk ke lokasi memori di mana sebuah karakter harus ditulis atau di mana karakter tersebut harus dibaca.
 - Counter data, tempat penyimpanan jumlah karakter dan akan berkurang nilainya jika karakter ditransfer.
- Membutuhkan waktu proses yang lama dan tidak efisien dalam pemanfaatan CPU.

Operasi I/O terbagi menjadi 3 metode : (Lanj.)

2. I/O interupsi

Metode di mana CPU akan bereaksi ketika suatu piranti mengeluarkan permintaan untuk pelayanan.

Karakteristik:

- Lebih efisien dalam pemanfaatan CPU, karena tidak harus menguji status dari piranti.
- Interupsi dapat berasal dari piranti I/O, interupsi perangkat keras misalnya : timer, memori, power supply, dan Interupsi perangkat lunak misalnya : overflow, opcode / data yang ilegal, pembagian dengan nol

Ada 2 jenis Interupsi:

- Interupsi maskable

Interupsi yang dapat didisable (dimatikan) untuk sementara dengan sebuah instruksi disable interupsi khusus.

- Interupsi nonmaskable

Interupsi yang tidak dapat didisable dengan instruksi perangkat lunak.

Metode Interupsi (1)

Dalam sistem komputer terdapat lebih dari satu piranti yang memerlukan pelayanan interupsi, dapat digunakan metode:

- **Polling/polled interrupt**

Berdasarkan urutan prioritas yang telah ditentukan sebelum piranti memerlukan interupsi. Misal: piranti A dan B mempunyai urutan prioritas A lebih dulu dari B, maka jika A dan B secara bersamaan memerlukan pelayanan interupsi, maka piranti A akan didahulukan.

Metode Interupsi (2)

- **Vector Interrupt**

Peralatan yang berinterupsi diidentifikasi secara langsung dan dihubungkan routine pelayanan vector interrupt.

INTR = Sinyal yang dikeluarkan oleh peralatan.

INTA = Sinyal kendali yang digunakan CPU untuk menyiapkan pelayanan interrupt

Operasi I/O terbagi menjadi 3 metode : (Lanj.)

3. Direct Memory Access (DMA)

Metode transfer data secara langsung antara memori dengan piranti tanpa pengawasan dan pengendalian CPU.

- Skema transfer blok DMA dual port

CPU dan DMA controller mengakses memori utama melalui MAR dan MBR dengan menggunakan sebuah memori utama dual port (2 port).

Port I ----> melayani CPU

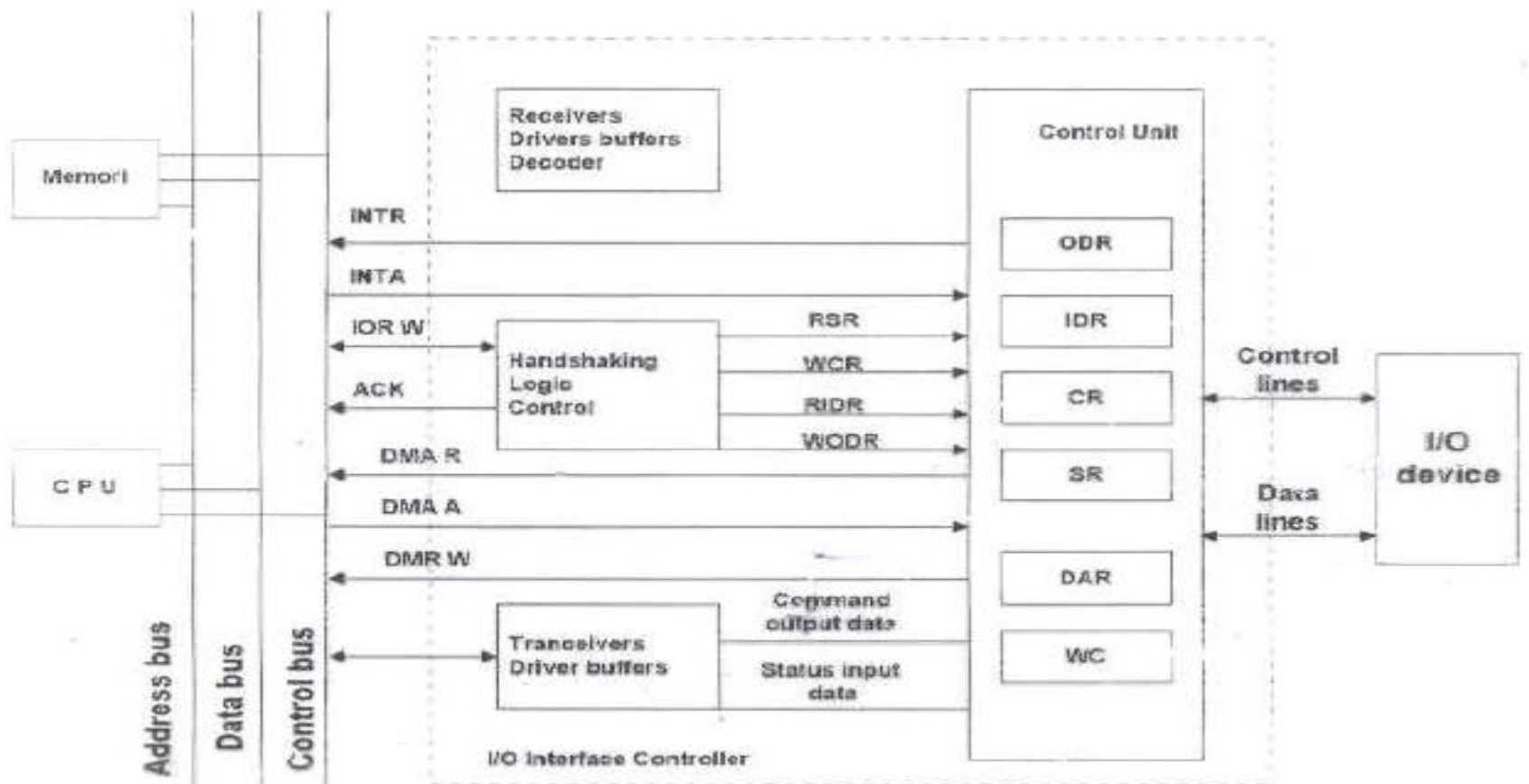
Port II ---> melayani DMA controller

- Skema transfer blok DMA cycle stealing (pencurian siklus)

Hanya memerlukan sebuah memori port tunggal di mana CPU dan piranti I/O berada cepat pada basis asinkron, prioritas utama akan diberikan pada piranti I/O.

Interfacing

Adalah peralatan yang digunakan untuk menghubungkan suatu piranti dengan CPU melalui bus.



Gambar 1. diagram blok sebuah unit interface I/O secara skematis

Keterangan :

- Register kendali (CR) digunakan untuk mencatat berbagai perintah dan informasi lainnya dalam peripheral.
- Register status (SR) digunakan untuk menyimpan status piranti dan memberitahukan pesan-pesan kesalahan .
- Register data input (IDR) dan register data output(ODR) masing-masing berfungsi sebagai bufer data untuk operasi input dan output.
- Urutan operasi interface:
 - Unit logika handshaking memasok unit kendali dengan empat sinyal.
 - Dua sinyal, register kendali penulisan (WCR atau write control register) dan register status pembacaan (RSR atau read status register), masing-masing berhubungan dengan CR dan SR.
 - Sedangkan dua sinyal lainnya adalah register pembacaan data input (RIDR atau read input data register) dan register penulisan data output (WODR atau write output data register), masing-masing mengendalikan IDP dan ODR.

Latihan Soal :

1. Macam-macam input/output?
2. Jelaskan mengapa oprasi input/output tergantung signal kendali dari CPU?
3. Jelaskan pengertian register status, buffer dan pointer buffer?
4. Jelaskan intrupsi yang berasal dari piranti input/output dan interupsi perangkat keras? Berikan contoh!
5. Jelaskan urutan prioritas yang telah ditentukan sebelum piranti memerlukan interupsi?