



PROGRAM STUDI
TEKNIK INFORMATIKA
FAKULTAS ILMU KOMPUTER
UNIVERSITAS DIAN NUSWANTORO

MATA KULIAH
ORGANISASI DAN ARSITEKTUR
KOMPUTER

Direct Memory Access

- ✓ Konsep DMA
- ✓ Buss

Tim pengampu

Sistem Komputer, Komunikasi dan Keamanan Data

T.A. 2020

Capaian Pembelajaran

Mahasiswa dapat memahami tentang teknik *direct memory access* (DMA), bus dan mengenal interface input/ output.



Review - Teknik operasi I/O

- **Programmed I/O**

Dimana *processor* mendapatkan akses untuk melakukan transaksi data dengan *memory* dan perangkat I/O melalui eksekusi instruksi didalam program dan setiap transaksi membutuhkan campur tangan *processor*.

- **Interrupt-driven I/O**

Saat *processor* membutuhkan akses kepada I/O ia memberikan perintah kepada *I/O module* lalu melanjutkan mengeksekusi instruksi lain hingga *I/O module* siap lalu menginterupsi *processor* untuk melanjutkan pertukaran data hingga menuliskan/ membaca data ke/ dari dalam *memory*.

	No Interrupts	Use of Interrupts
I/O-to-memory transfer through processor	Programmed I/O	Interrupt-driven I/O
Direct I/O-to-memory transfer		Direct memory access (DMA)

Tabel 12.1 – teknik I/O

Direct Memory Access (DMA) - Intro

Disatu sisi *programmed i/o* memiliki keunggulan berupa akses eksklusif atas *processor*, perbedaan kecepatan eksekusi *processor* dan *transfer rate* perangkat i/o yang terlampau jauh membuat *processor* harus membuang banyak waktu untuk menunggu perangkat i/o menyelesaikan sebuah operasi i/o.

Meskipun *interrupt-driven* lebih efisien dibanding *programmed i/o*, namun masih membutuhkan intervensi aktif *processor* untuk mengirimkan data antara *memory* dan *I/O module* dan setiap transmisi data harus melalui *processor*.

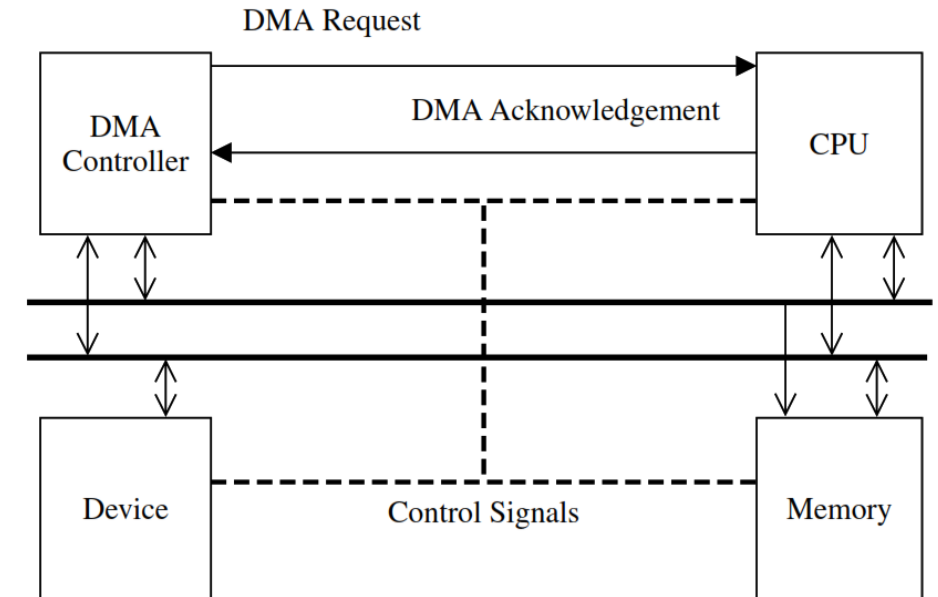
Dan saat sejumlah data yang besar akan dikirimkan kedua teknik tersebut akan membuang banyak waktu percuma yang dimiliki *processor* yang seharusnya bisa dimanfaatkan untuk melakukan banyak hal lain. Sehingga dibutuhkan alternatif teknik yang mampu menutup kelemahan tersebut.

Direct Memory Access (DMA)

Menambahkan sebuah *module* pada *system bus* yang mampu meniru kemampuan *processor* untuk mengelola sistem.

DMA memungkinkan perangkat i/o untuk mengakses *memory* secara langsung tanpa campur tangan *processor* secara aktif.

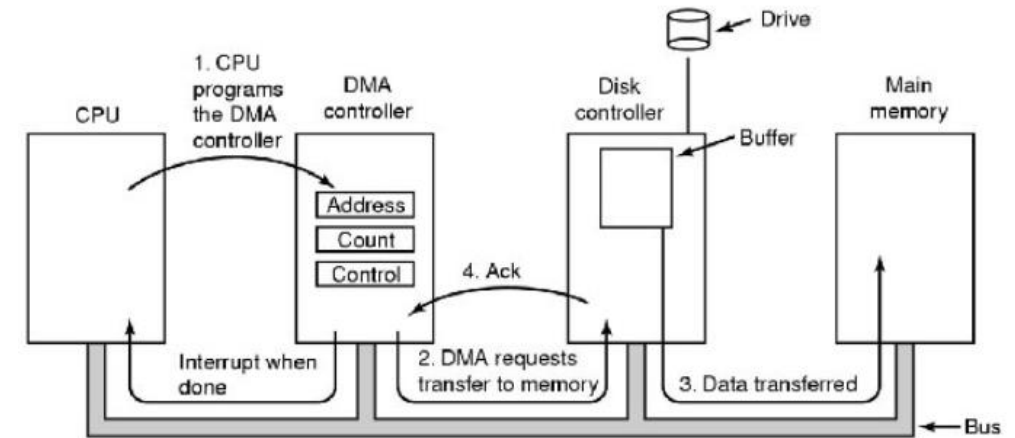
Dengan memanfaatkan bus untuk mengirimkan data dari dan ke *memory* hanya saat bus tidak digunakan oleh *processor*, atau bahkan DMA mampu memaksa *processor* untuk menunda sementara sebuah operasi . Teknik ini disebut *cycle stealing*.



Gambar 12.1 – DMA dan CPU berbagi *memory bus*

Direct Memory Access (DMA)

1. *Processor* memprogram DMA dengan men-setting *register- register* yang ada didalam DMA.
2. DMA meminta transfer ke *memory*, pada tahap ini data dari *drive* disimpan ke *buffer*;
3. Lalu data dikirimkan dari *buffer* ke *memory*;
4. Kemudian *disk controller* mengirimkan *acknowledge* ke DMA. Pada tahap ini langkah 2, 3, 4 akan terus berulang hingga seluruh data dari *drive* terkirim ke *memory*.
5. Dan saat pengiriman telah lengkap, maka DMA akan mengirimkan interrupt kepada *processor* menyatakan bahwa proses pengiriman telah selesai dikerjakan.



Gambar 12.2 – operasi DMA

Sumber : Stalling

Bus

Dalam terminologi komputer, *bus* koneksi fisik yang digunakan untuk membawa sinyal dari satu titik ke titik lain. Sinyal yang bawa ini bisa berupa alamat, data, sinyal kontrol atau daya. Biasanya bus terdiri dari sejumlah koneksi yang bekerja bersama, tiap koneksi disebut *bus line*.

Ada 4 jenis bus yang memiliki fungsi berbeda- beda :

address bus- yang membawa alamat lokasi *memory*,

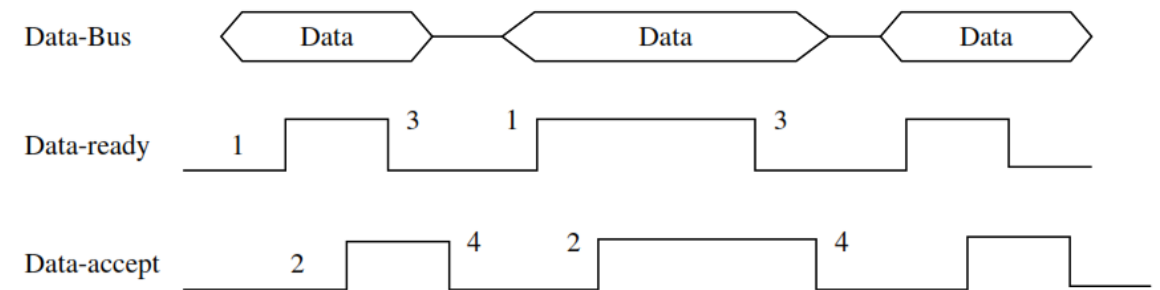
data bus- membawa data,

control bus- membawa sinyal kontrol dan

power bus- membawa tegangan catu daya.

Bus disebut *sinkron* bila pengiriman data diatur oleh *bus clock*, digunakan sebagai acuan untuk menentukan waktu yang tepat untuk mengirimkan data melalui bus.

Dan disebut *asinkron* jika data dikirimkan melalui bus berdasarkan 'ketersediaan' data, teknik yang digunakan dalam pengiriman data melalui bus asinkron disebut *handshaking*.



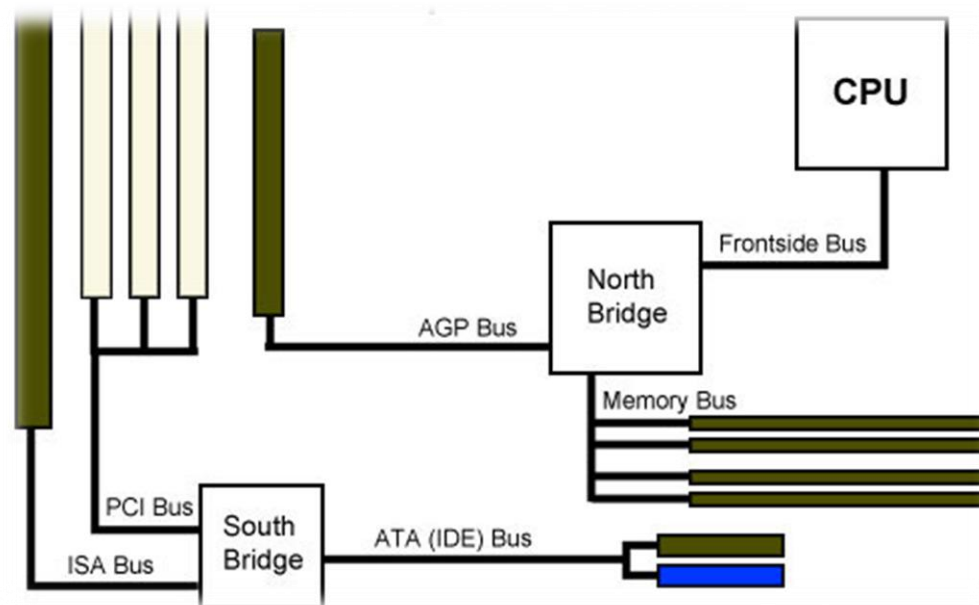
Gambar 12.3 – Asynchronous bus timing using *handshaking protocol*

Primary Buses

Terdapat 2 bus utama pada komputer :

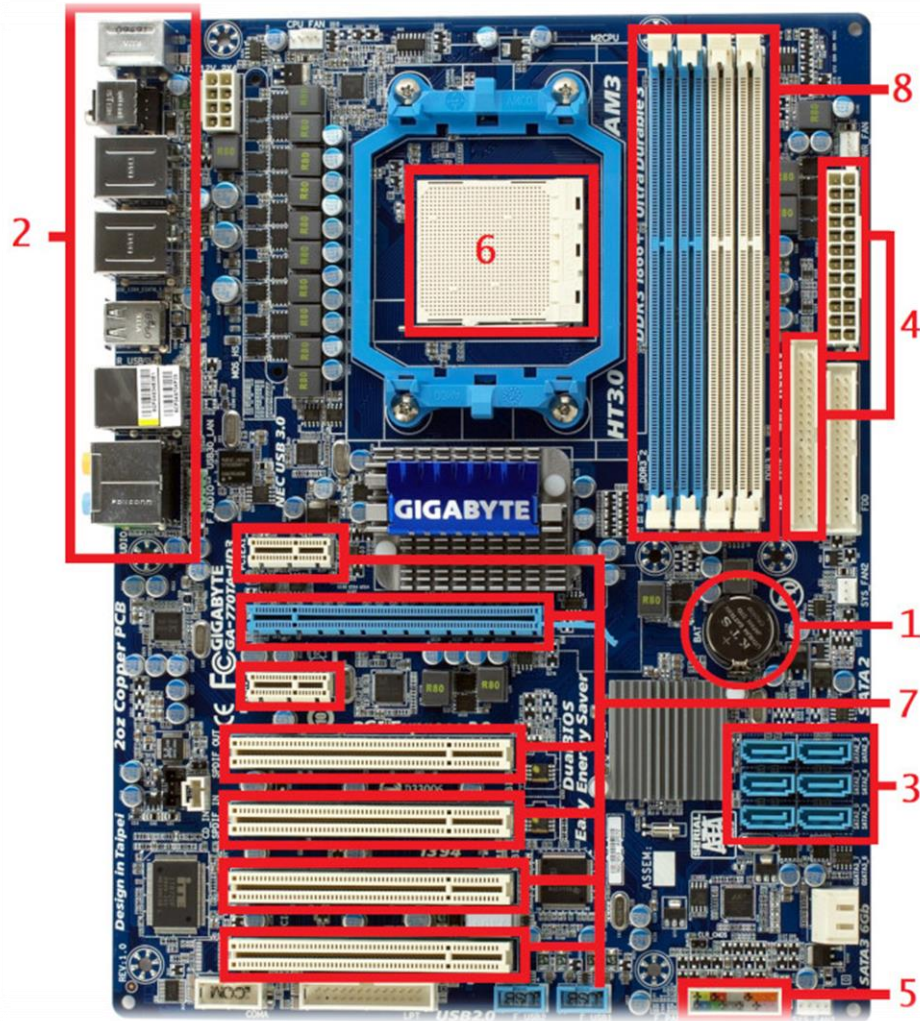
Internal bus (sering disebut FSB – *front side bus*) – menghubungkan *processor* dan komponen internal (RAM dan *graphic card*).

Expansion bus – menghubungkan berbagai komponen didalam komputer (USB, SCSI, dsb.).



Gambar 12.4 – Computer Buses
Sumber : ComputerHope.com

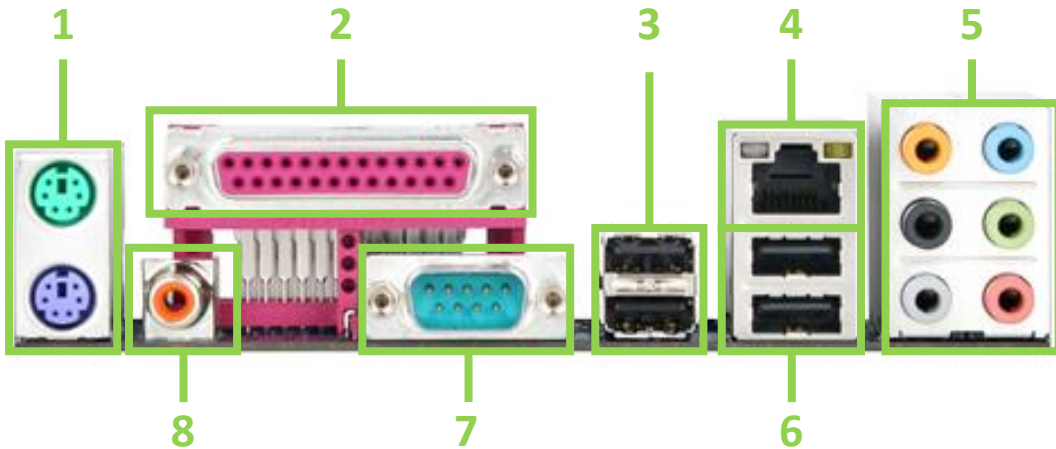
I/O Interfaces



No	Keterangan
1	BIOS (Basic Input Output System) dan baterai CMOS (Complementary Metal Oxide Semi-Conductor)
2	Back Panel (Halaman selanjutnya)
3	SATA Connector
4	Power connector (atas) dan IDE connector (bawah)
5	Front panel connector
6	Processor socket
7	Expansion Slot (atas ke bawah : PCI Express x1, PCI Express x16, PCI Express x1, PCI Slot)
8	4x RAM DDR3 DIMM Slot

Gambar 12.5 – Computer Buses (sumber : gigabyte.com)

I/O Interfaces



Gambar 12.6 – Back Panel
Sumber : Gigabyte.com

No	Keterangan
1	PS/2 port (Compatible device : Keyboard/ Mouse)
2	Parallel port (Compatible device :printer)
3	2x USB 1.1
4	RJ45 port
5	6x Audio jack
6	2x USB 2.0
7	Serial port (Compatible device :mouse)
8	Coaxial S/PDIF Out (Audio output)

Tugas

1. Sebutkan min. 4 jenis *socket* dan *processor* yang sesuai!
2. Sebutkan min. 3 jenis konektor pada motherboard yang bisa digunakan untuk memasang *storage*!
3. Sebutkan min. 3 jenis *expansion slot* beserta jenis *expansion card* yang compatible!
4. Sebutkan min. 4 jenis port pada back panel dan peripheral yang dapat dipasangkan!
5. Sebutkan min. 3 jenis *display port* yang umum digunakan!

SUMBER PUSTAKA

- Mostafa dan Hesham.2005.Fundamentals Of Computer Organization And Architecture.New Jersey : Wiley Interscience
- W. Stallings.2016. Computer Organization and Architecture.Hoboken:Pearson Education
- A.S. Tanenbaum..Structured Computer Organization.New Jersey : Pearson Prentice Hall



THANKS

ANY QUESTIONS?