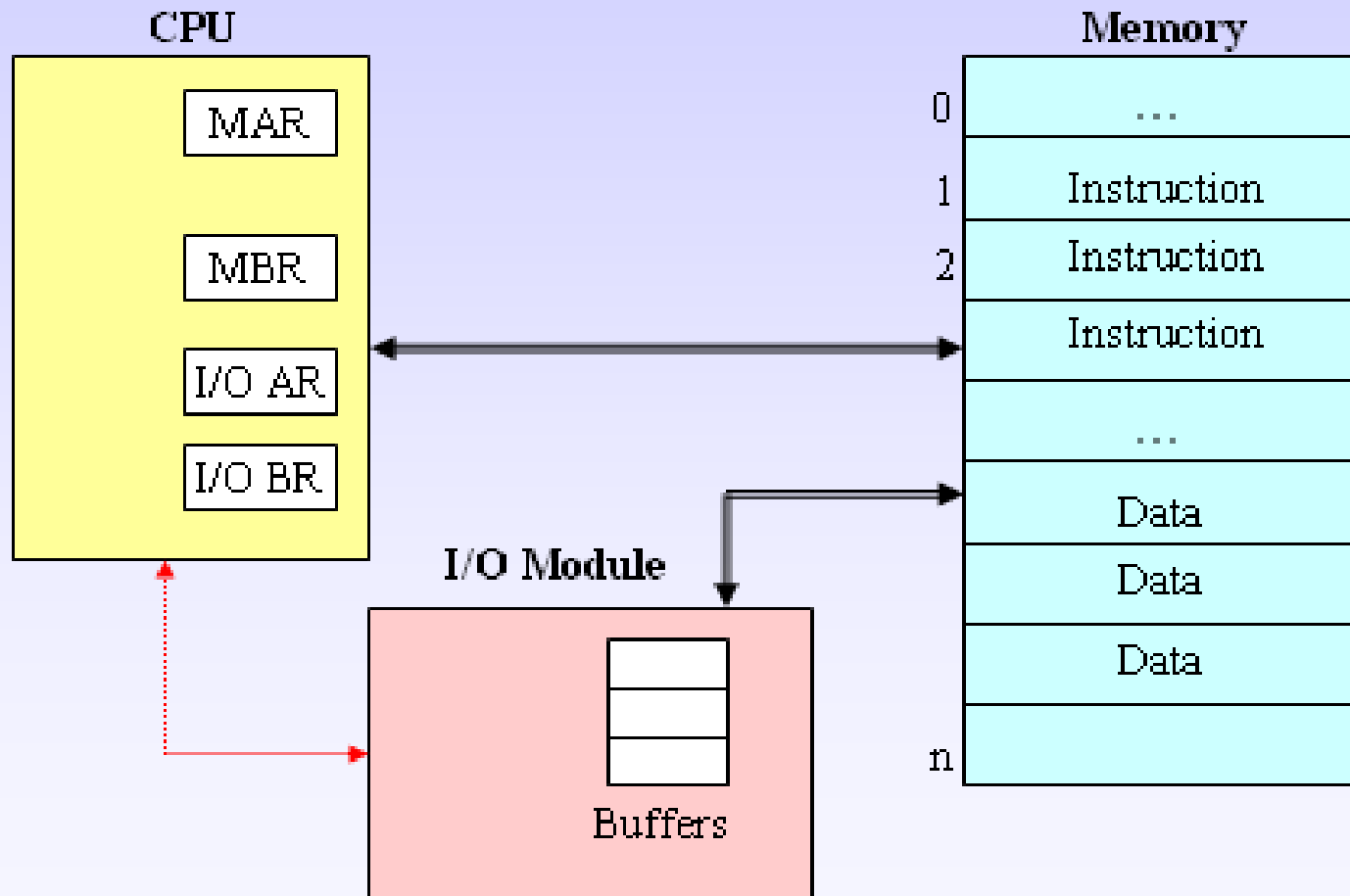


# BUS SISTEM PERTEMUAN

## BUS-BUS<sup>2</sup> SISTEM

# Komponen-komponen Komputer



# Komponen CPU#1

## Register

Register yang terdapat dalam CPU, yaitu :

**MAR** (Memory Address Register)

Menentukan alamat di dalam memori yang akan diakses untuk operasi Read/Write

**MBR** (Memory Buffer Register)

Berisi data yang akan di tuliskan ke dalam memori atau menerima data yang di baca dari memori

## Komponen CPU #2

### **PC (Program Counter)**

Mencatat alamat memori dimana instruksi di dalamnya akan dieksekusi

### **I/O AR (I/O Address Register)**

Menspesifikasikan perangkat I/O yang akan diakses

### **I/O BR (I/O Buffer Register)**

Menyimpan data yang akan dituliskan ke port atau data yang akan disalin dari port.

Alamat port ditunjuk oleh I/O AR

# Komponen CPU #3

## **IR** (Instruction Register)

Menampung instruksi yang akan dilaksanakan

## **AC** (Accumulator)

Menyimpan data sementara baik data yang sedang diproses atau data yang dihasilkan

# Modul I/O

- ✚ Memindahkan data dari perangkat eksternal ke CPU dan sebaliknya
- ✚ Modul ini berisi buffer internal untuk menampung data ini sementara sampai data itu di kirimkan.

# Fungsi Komputer

- Fungsi dasar komputer adalah eksekusi program
- Program yang akan di eksekusi oleh CPU ada dalam memori

Ada 2 langkah pengambilan instruksi :

1. CPU membaca instruksi yang ada di memori (fetch)
2. CPU mengeksekusi setiap instruksi (execute)

# Siklus Fetch# 1

- ✚ Pada CPU yang umum, suatu register Program Counter (PC) di pakai untuk mengawasi instruksi yang akan di baca selanjutnya.
- ✚ Instruksi yang di baca akan di muatkan kedalam sebuah register (IR) Instructions Register



# Siklus Fetch# 2

Aksi-aksi yang dilakukan oleh CPU ketika menginterpretasikan instruksi di bagi menjadi 4 kategori :

## 1. CPU --- Memori

Data di pindahkan dari CPU ke memori atau sebaliknya

## 2. CPU --- I/O

Data dapat di pindahkan ke atau dari dunia luar dengan pemindahan antara CPU dan modul I/O.

# Siklus Fetch# 3

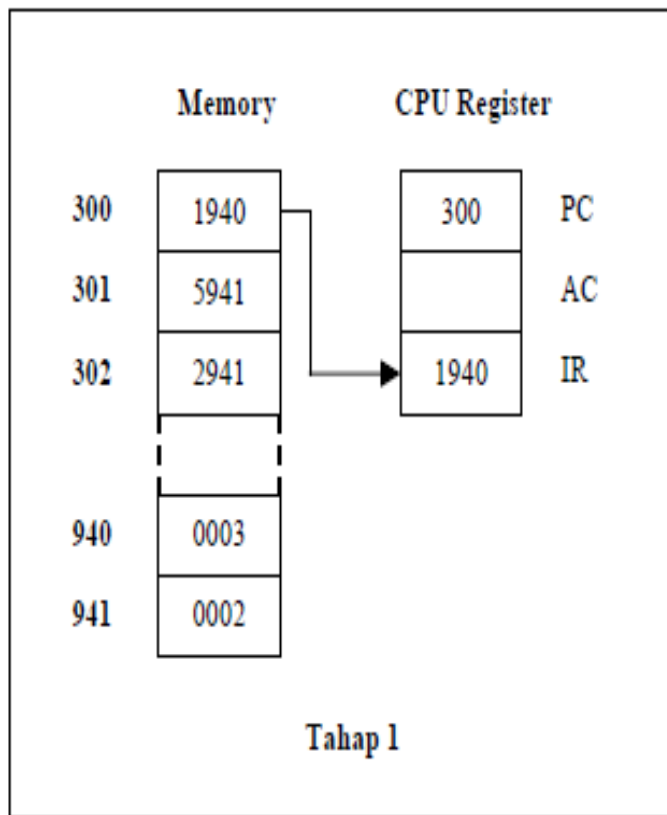
## 3. Pengolahan data

CPU dapat membentuk sejumlah operasi aritmatik /logik.

## 4. Control

Sebuah instruksi yang dapat mengubah urutan eksekusi

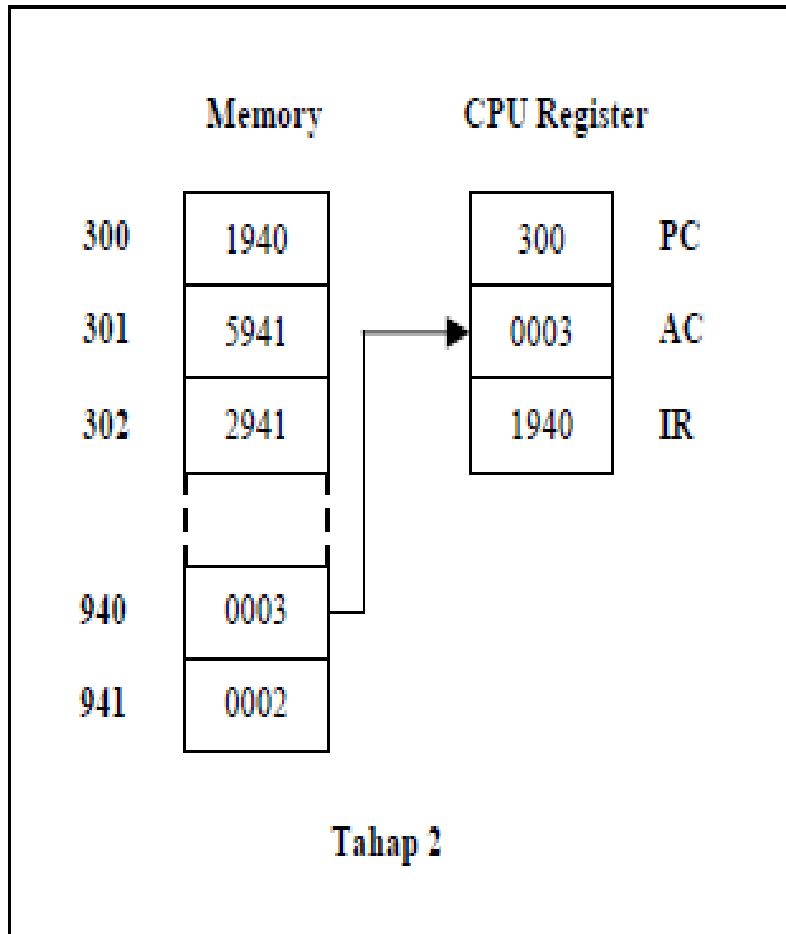
# Contoh Eksekusi Program Dalam CPU #1



berisi 300 alamat instruksi

dan dalam Instruction Register (IR)

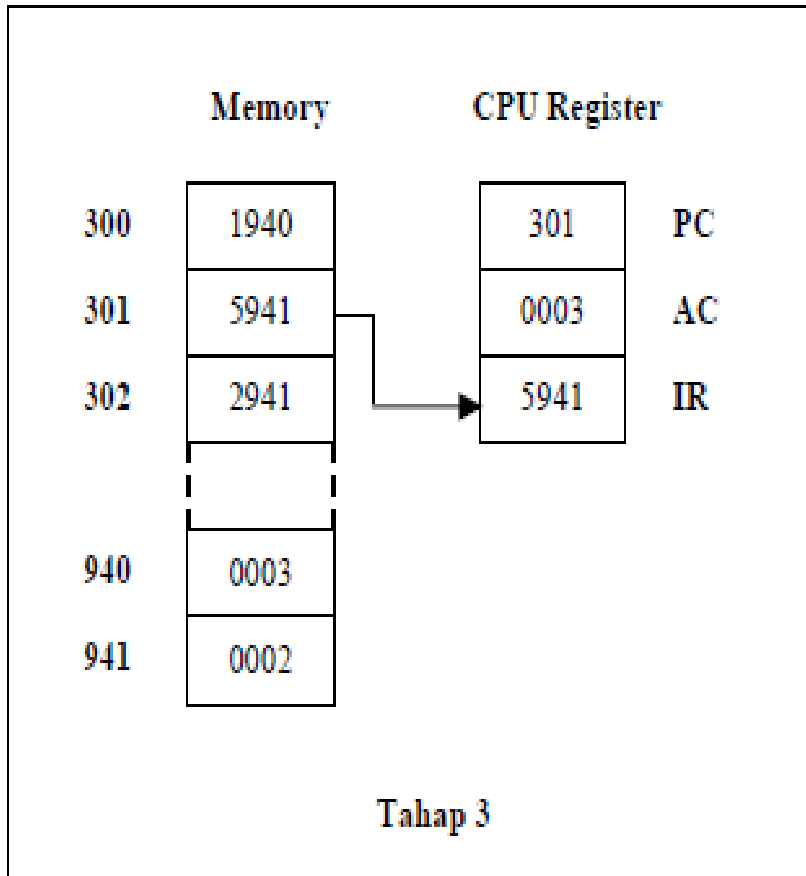
# Contoh Eksekusi Program Dalam CPU #2



mengindikasikan bahwa  
nuatkan

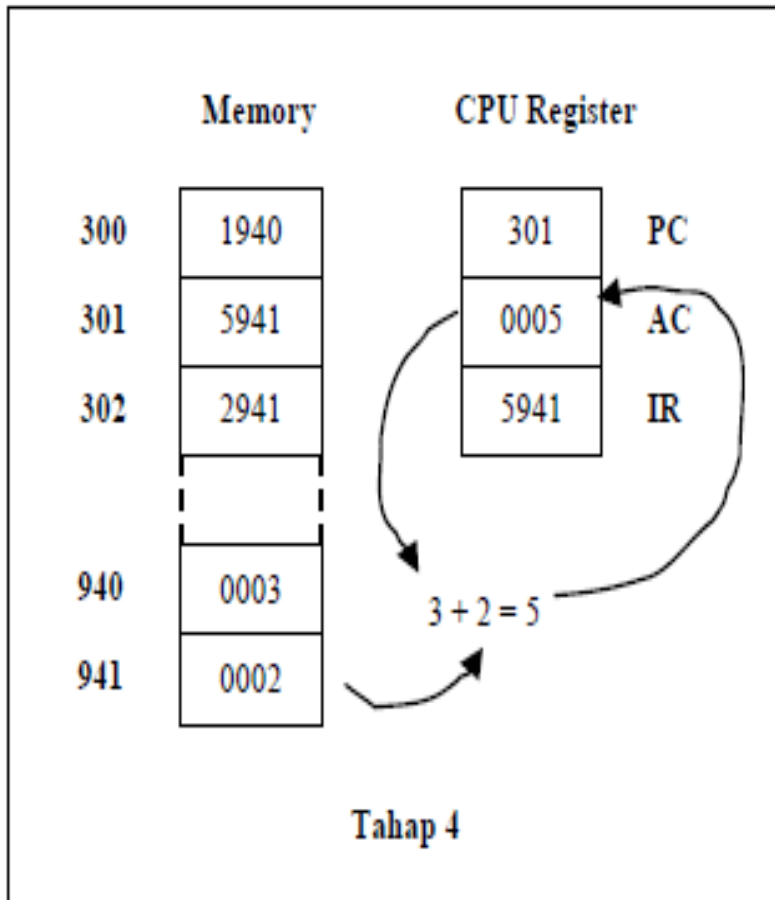
in alamat, yaitu 940

# Contoh Eksekusi Program Dalam CPU #3



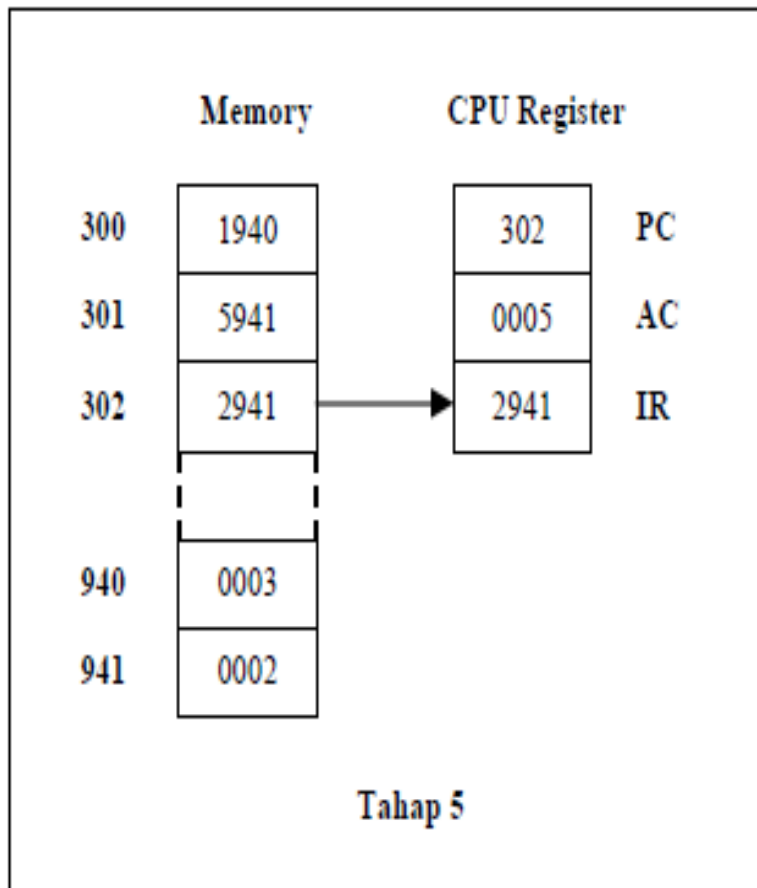
nstruksi berikutnya akan

# Contoh Eksekusi Program Dalam CPU #4



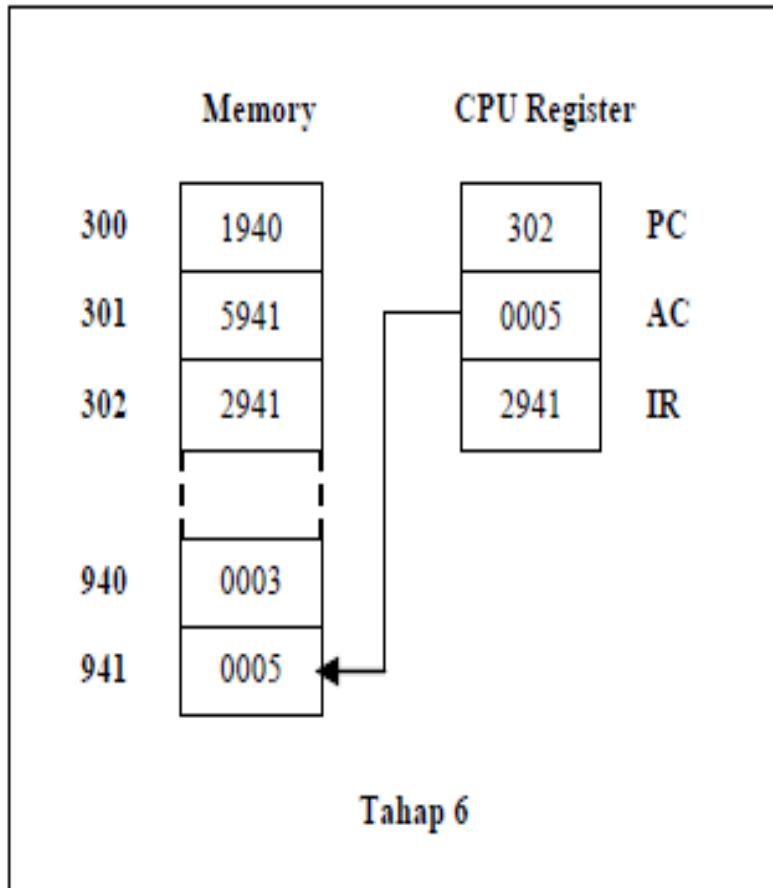
okasi 941 ditambahkan dan  
n AC

# Contoh Eksekusi Program Dalam CPU #5



instruksi berikutnya akan

# Contoh Eksekusi Program Dalam CPU #6



da lokasi 941



# Interrupts #1

## Kelas-kelas Interrupt :

### 1. Program

dibangkitkan dengan beberapa kondisi yang terjadi sebagai hasil dari suatu eksekusi instruksi

### 2. Timer

dibangkitkan oleh timer di dalam processor

# Interrupts #2

## 3. I/O

di bangkitkan oleh I/O kontroller untuk memberi signal penyelesaian normal atau memberikan signal bergagai kondisi error



## 4. H/W Failure

di bangkitkan oleh kegagalan seperti kegagalan daya atau memori parity error

# Pengertian Interrupts

Interrupt disediakan terutama sebagai cara untuk meningkatkan efisiensi pengolahan, karena sebagian besar perangkat eksternal jauh lebih lambat dibandingkan prosesor

# Interrupt & Siklus Instruksi #1

-  Dengan memakai interrupt, processor dapat dipakai dalam mengeksekusi instruksi-instruksi lain operasi I/O yang sedang di laksanakan
-  Ketika perangkat eksternal telah siap untuk dilayani, maka modul I/O untuk perangkat eksternal itu mengirimkan signal interrupt request ke prosessor.

# Interrupt & Siklus Instruksi #2



Processor menanggapi dengan menahan operasi program yang sedang dilakukannya, mencabangkannya ke suatu program untuk melayani perangkat I/O itu, yang di kenal dengan Interrupt Handler, dan kembali melaksanakan eksekusi mula-mula, setelah perangkat itu di layani.

# Multiple Interrupt #1

1. Dengan tidak mengijinkan terjadinya interrupt lain pada saat suatu interrupt sedang di proses. (interrupt lain di tangguhkan)

## Keuntungannya :

Pendekatan tersebut cukup baik dan sederhana karena interrupt di tangani dalam urutan yang cukup ketat.

## Kekurangannya :

Pendekatan ini tidak memperhitungkan prioritas relatif atau kebutuhan waktu kritis

# Multiple Interrupt #2

2. Dengan mendefinisikan prioritas bagi interrupt dan mengijinkan interrupt berprioritas tinggi menyebabkan interrupt handler yang berprioritas lebih rendah untuk menginterupsi diri sendiri.

# Fungsi I/O

Sebuah I/O dapat bertukar data secara langsung dengan CPU.

Sebuah I/O juga dapat bertukar data langsung dengan memori.



# Pertukaran Data Antara I/O dan Memori

- Dalam kasus ini CPU memberikan otoritas kepada modul I/O untuk membaca dari atau menulis ke memori, sehingga perpindahan data terjadi tanpa terpaut dengan CPU
- Selama perpindahan seperti itu, modul I/O mengeluarkan perintah baca/ tulis ke memori, yang membebaskan CPU dari tanggung jawab pertukaran data
- Operasi seperti ini di kenal dengan DMA (Dirrect Memory Access)

# Struktur Interkoneksi

- + Komputer terdiri dari CPU – Memori – I/O
- + Komponen bus/ lintasan yang menghubungkan berbagai modul disebut dengan struktur interkoneksi

# Struktur interkoneksi harus mendukung jenis perpindahan berikut ini: #1

## 1. Memori ke CPU

CPU membaca sebuah instruksi atau satuan data dari memori

## 2. CPU ke Memori

CPU menuliskan sebuah satuan data ke memori

Struktur interkoneksi harus mendukung jenis perpindahan berikut ini: #2

### 3. I/O ke CPU

CPU membaca data dari perangkat I/O melalui sebuah modul I/O

### 4. CPU ke I/O

CPU mengirimkan data ke perangkat I/O

### 5. I/O ke memori atau memori ke I/O

Pada kedua kasus ini sebuah modul I/O di izinkan untuk dapat bertukar data secara langsung tanpa melalui CPU dengan menggunakan DMA

# Interkoneksi Bus

Bus adalah media transmisi yang dapat di gunakan bersama

Bila dua buah perangkat melakukan transmisi dalam waktu yang bersamaan, maka signal-signalnya akan bertumpang tindih dan menjadi rusak. Berarti harus hanya satu buah perangkat saja yang akan berhasil melakukan transmisi pada suatu saat tertentu.

# Struktur Bus

Bus-bus yang menghubungkan komponen utama sistem (CPU, Memori, I/O) disebut dengan BUS SISTEM

Biasanya bus sistem terdiri dari 50 – 100 saluran yang terpisah

# Klasifikasi Bus

## Saluran Data :

memberikan lintasan bagi perpindahan data antara 2 modul sistem.

Saluran ini secara kolektif disebut BUS DATA

## Saluran Alamat :

digunakan untuk menandakan sumber atau tujuan data pada bus data.

## Saluran Kontrol :

di gunakan untuk mengontrol akses ke saluran alamat dan penggunaan data dan saluran alamat.

# Elemen-elemen Rancangan BUS

## A. Jenis

### 1. Dedicated

Suatu saluran bus di dedicated secara permanen diberi sebuah fungsi atau subset fisik komponen-komponen komputer

### 2. Multiplexed

Metode penggunaan saluran yang sama untuk berbagai keperluan



# Keuntungan :

1. **Dedicated :**  
**Throughput yang tinggi, karena terjadi kemacetan yang kecil**
2. **Multiplexed :**  
**Memerlukan saluran yang sedikit menghemat ruang dan biaya**

# Kerugian :

1. Dedicated :  
Meningkatnya ukuran dan biaya sistem
2. Multiplexed :  
Rangkaian lebih kompleks, terjadi penurunan kinerja, karena event-event tertentu yang menggunakan saluran bersama-sama tidak dapat berfungsi secara paralel

## **B. Metode Arbitrasi**

### **1. Tersentralisasi :**

sebuah perangkat H/W (pengontrol bus arbiter) bertanggung jawab atas alokasi waktu pada bus

### **2. Terdistribusi :**

tidak terdapat pengontrol sentral, tapi setiap modul terdiri dari access control logic dan modul-modul berkerja sama untuk memakai bus bersama-sama

## C. Timing

### 1. Synchronous

terjadinya event pada bus ditentukan oleh clock

### 2. Asynchronous

terjadinya event pada bus ditentukan oleh event sebelumnya

## D. Lebar Bus

- ⌚ Semakin lebar bus data, semakin besar bit yang dapat di transfer pada suatu saat
- ⌚ Semakin lebar bus alamat, semakin besar range lokasi yang dapat di referensi

# Jenis Transfer Data

- ✚ **Read**
- ✚ **Write**
- ✚ **Read modify write**
- ✚ **Read after write**
- ✚ **Blok**

# **Bus PCI**

## **(Peripheral Component Interconnect)**

### **(1990)**

- **Merupakan bus yang tidak tergantung processor dan bandwidth tinggi yang dapat berfungsi sebagai bus mezzanine/bus berkecepatan tinggi**
- **Mezzanine adalah bus berkecepatan tinggi yang sangat terintegrasi dengan sistem**

# Future Bus+

Future Bus+ adalah standard bus asinkron yang berkinerja tinggi



# Syarat-syarat Future Bus #1

- ❖ Tidak tergantung pada arsitektur, processor dan teknologi tertentu
- ❖ Memiliki protokol transfer asinkron dasar
- ❖ mengizinkan protokol tersinkronisasi pada sumber untuk kebutuhan optional
- ❖ tidak berdasarkan pada teknologi terancang

# Syarat-syarat Future Bus #2

- terdiri dari protokol-protokol paralel terdistribusi penuh dan arbitrase yang mendukung baik protokol circuit switched maupun protokol split transactions
- Menyediakan dukungan bagi sistem-sistem yang fault-tolerant dan yang memiliki reliabilitas tinggi
- menawarkan dukungan langsung terhadap memori berbasis cache yang dapat digunakan bersama
- memberikan definisi transportasi pesan yang compatible

# Syarat-syarat Future Bus #3

- Future bus+ mendukung bus-bus data 32,64, 128,256 bit
- Future bus+ mendukung baik model terdistribusi maupun tersentralisasi
- Future bus+ merupakan salah satu standar bus yang secara teknis paling kompleks
- Future bus+ merupakan spesifikasi bus yang dapat di gunakan untuk bus prosessor –memori atau yang dapat di gunakan dengan PCI untuk mendukung peripheral-peripheral berkecepatan tinggi.

# Perbedaan PCI dan Future bus+

- ✗ PCI di tujukan bagi implementasi murah yang membutuhkan bidang fisik secara minimal
- ✗ Future bus+ dimaksudkan untuk memberikan fleksibilitas yang tinggi dan fungsionalitas yang luas untuk memenuhi kebutuhan berbagai sistem yang berkinerja tinggi terutama sistem-sistem yang mahal.

*The End*