

Pertemuan ke - 15

Sistem Bus

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

Politeknik Elektronika Negeri Surabaya

Tujuan

- ⌘ Menjelaskan struktur antar hubungan
- ⌘ Menjelaskan *bus* antar hubungan
- ⌘ Menjelaskan elemen dari desain *bus*
- ⌘ Menjelaskan PCI, SCSI, Fire wire dan USB

Sistem Bus

- ⌘ Penghubung bagi keseluruhan komponen komputer dalam menjalankan tugasnya
- ⌘ Komponen komputer :
 - ☑ CPU
 - ☑ Memori
 - ☑ Perangkat I/O
- ⌘ Transfer data antar komponen komputer.
 - ☑ Data atau program yang tersimpan dalam memori dapat diakses dan dieksekusi CPU melalui perantara *bus*
 - ☑ Melihat hasil eksekusi melalui monitor juga menggunakan sistem *bus*
 - ☑ Kecepatan komponen penyusun komputer harus diimbangi kecepatan dan manajemen *bus* yang baik

Sistem Bus

⌘ Mikroprosesor

- ☑ Melakukan pekerjaan secara paralel
- ☑ Program dijalankan secara multitasking
- ☑ Sistem *bus* tidak hanya lebar tapi juga cepat

⌘ Interkoneksi komponen sistem komputer dalam menjalankan fungsinya

- ☑ Interkoneksi *bus*
- ☑ Pertimbangan–pertimbangan perancangan *bus*

Elemen Perancangan Bus

- ⌘ Parameter dasar perancangan bus dapat diklasifikasikan berdasarkan jenis
 - ☒ *Dedicated*
 - ☒ *Multiplexed*
- ⌘ Metode arbitrase
 - ☒ Tersentralisasi
 - ☒ Terdistribusi
- ⌘ Timing
 - ☒ Sinkron
 - ☒ Tak sinkron
- ⌘ Lebar bus
 - ☒ Lebar address
 - ☒ Lebar data
- ⌘ Jenis transfer data
 - ☒ *read*
 - ☒ *write*
 - ☒ *read-modify-write*
 - ☒ *read-alter-write, block*

Jenis Bus

⌘ *Dedicated bus*

- ☑ Bus dibedakan menjadi bus yang khusus menyalurkan data tertentu, misalnya paket data saja, atau alamat saja.

⌘ *Multiplexed bus*

- ☑ Bus dilalukan informasi yang berbeda baik data, alamat maupun sinyal kontrol dengan metode mulipleks data maka bus ini disebut
- ☑ Keuntungan adalah hanya memerlukan saluran sedikit sehingga dapat menghemat tempat
- ☑ Kerugiannya adalah kecepatan transfer data menurun dan diperlukan mekanisme yang kompleks untuk mengurai data yang telah dimulitipleks

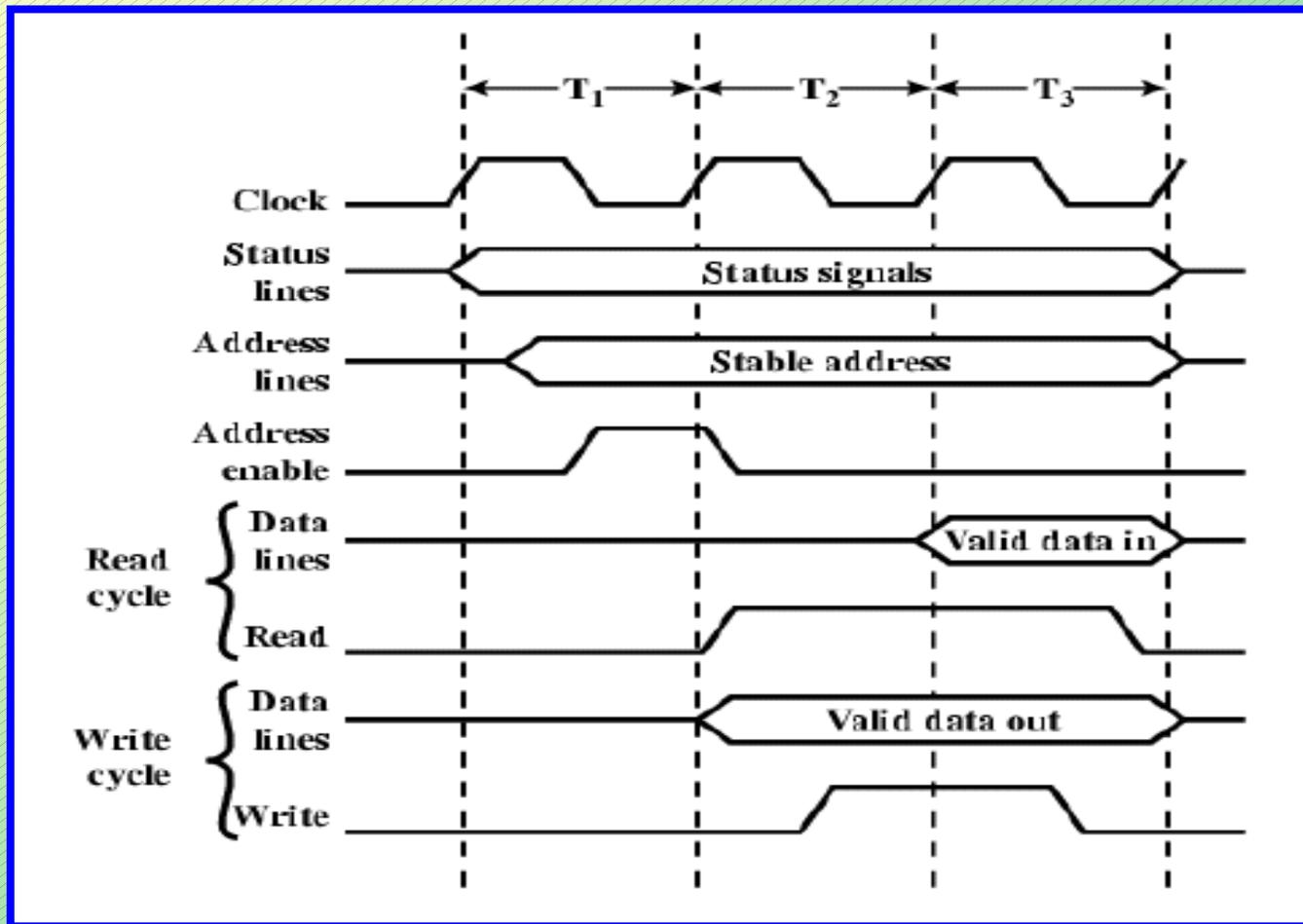
Metode Arbitrasi

- ⌘ Pada metode tersentral diperlukan pengontrol bus sentral atau *arbiter* yang bertugas mengatur penggunaan bus oleh modul. Arbiter bisa suatu modul atau bagian fungsi CPU.
- ⌘ Pada metode terdistribusi, setiap modul memiliki logika pengontrol akses (*access control logic*) yang berfungsi mengatur pertukaran data melalui bus.
- ⌘ Kedua metode arbitrasi intinya menugaskan suatu perangkat bisa modul I/O ataupun CPU bertindak sebagai master kontrol pertukaran

Timing-Sinkron

- ⌘ Metode pewaktuan *sinkron* terjadinya event pada bus ditentukan oleh sebuah pewaktu (*clock*).
- ⌘ Sebuah transmisi 1 – 0 disebut siklus waktu atau siklus bus dan menentukan besarnya slot waktu.
- ⌘ Semua perangkat modul pada bus dapat membaca atau mengetahui siklus clock.
- ⌘ Biasanya satu siklus untuk satu event.
- ⌘ Model ini mudah diimplementasikan dan cepat namun kurang fleksibel menangani peralatan yang beda kecepatan operasinya.
- ⌘ Biasanya digunakan untuk modul–modul tertentu yang sudah jelas karakteristiknya

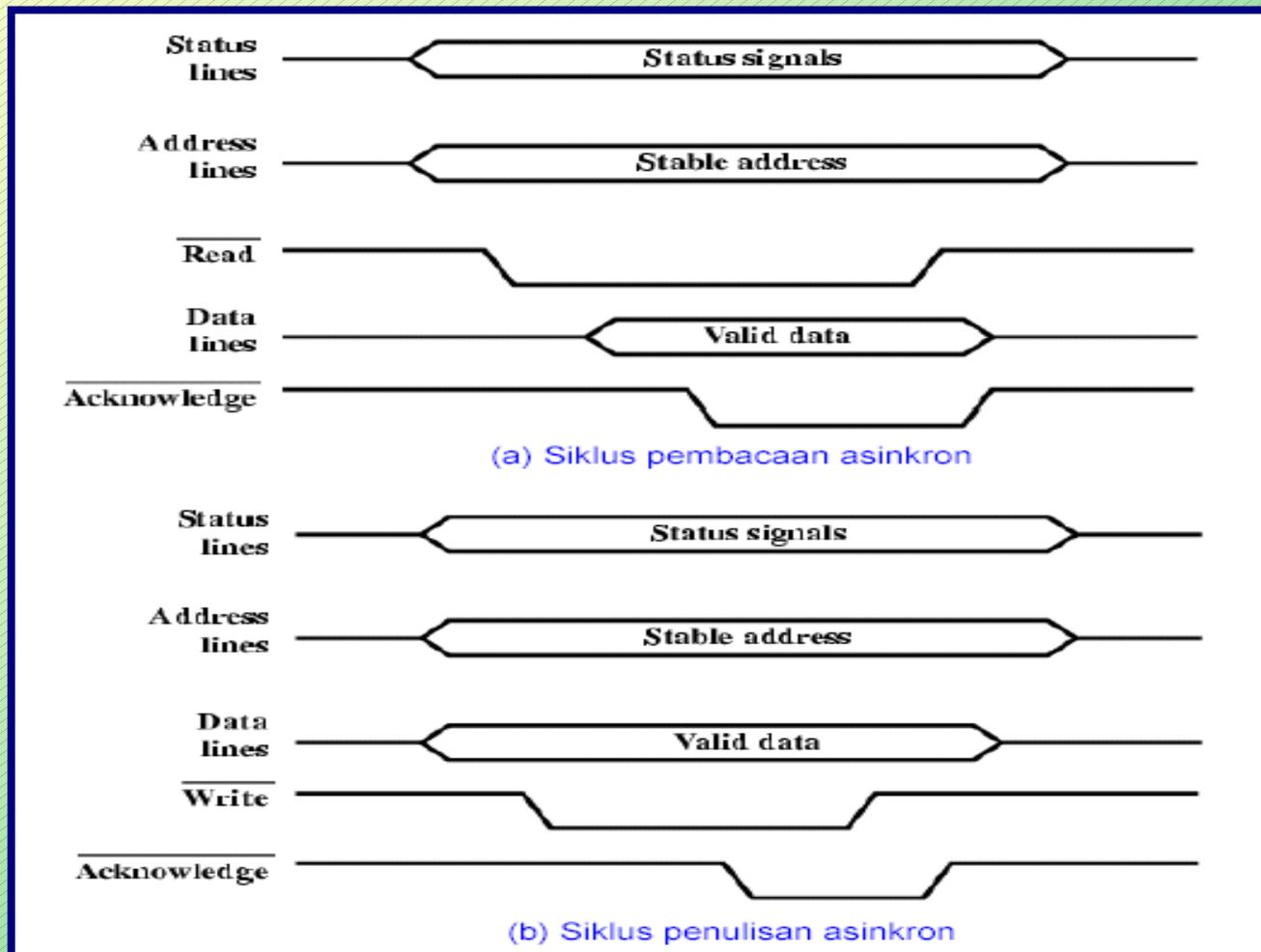
Contoh pewaktuan sinkron



Timing-Asinkron

- ⌘ Kerja modul yang tidak serempak kecepatannya.
- ⌘ Event yang terjadi pada bus tergantung event sebelumnya sehingga diperlukan sinyal – sinyal validasi untuk mengidentifikasi data yang ditransfer.
- ⌘ Sistem ini mampu menggabungkan kerja modul–modul yang berbeda kecepatan maupun teknologinya, asalkan aturan transfernya sama

Contoh pewaktuan asinkron



Lebar Bus

- ⌘ Semakin lebar *bus* maka semakin besar data yang dapat ditransfer sekali waktu.
- ⌘ Semakin besar *bus* alamat, akan semakin banyak range lokasi yang dapat direfensikan

Jenis Transfer Data

- ⌘ Operasi transfer data adalah pertukaran data antar modul sebagai tindak lanjut atau pendukung operasi yang sedang dilakukan.
- ⌘ Saat operasi baca (*read*), terjadi pengambilan data dari memori ke CPU, begitu juga sebaliknya pada operasi penulisan maupun operasi – operasi kombinasi.
- ⌘ *Bus* harus mampu menyediakan layanan saluran bagi semua operasi komputer

Contoh Bus - Bus ISA

- ⌘ *Industry Standar Architecture*
- ⌘ *Bus* PC/AT yang beroperasi pada 8,33 MHz
- ⌘ Keuntungannya adalah bahwa pendekatan ini tetap mempertahankan kompatibilitas dengan mesin-mesin dan kartu-kartu yang ada.
- ⌘ Pendekatan ini juga didasarkan pada sebuah *bus* yang telah dilisensikan secara bebas oleh IBM kepada banyak perusahaan dalam rangka untuk menjamin bahwa sebanyak mungkin pihak ketiga dapat memproduksi kartu-kartu untuk PC pertama, sesuatu yang kembali menghantui IBM.

Contoh Bus - Bus PCI

- ⌘ *Peripheral Component Interconnect (PCI)*
- ⌘ Bus yang tidak tergantung prosesor dan berfungsi sebagai bus mezzanine atau bus peripheral
- ⌘ PCI memiliki kinerja tinggi untuk sistem I/O berkecepatan tinggi seperti : video adaptor, NIC, disk controller, sound card, dan lain-lain.
- ⌘ Standard PCI adalah 64 saluran data pada kecepatan 33 MHz, laju transfer data 264 MB per detik atau 2,112 Gbps.
- ⌘ Keunggulan PCI tidak hanya pada kecepatannya saja tetapi murah dengan keping yang sedikit
- ⌘ Intel mulai menerapkan PCI pada tahun 1990 untuk sistem pentiumnya.
- ⌘ Untuk mempercepat penggunaan PCI, Intel mempatenkan PCI bagi domain publik sehingga vendor dapat mengeluarkan produk dengan PCI tanpa royalti

Contoh Bus - Bus USB

- ⌘ Semua perangkat peripheral tidak efektif apabila dipasang pada bus berkecepatan tinggi PCI
- ⌘ Banyak peralatan yang memiliki kecepatan rendah seperti keyboard, mouse, dan printer.
- ⌘ Solusi : tujuh vendor komputer (Compaq, DEC, IBM, Intel, Microsoft, NEC, dan Northern Telecom) bersama-sama merancang bus untuk peralatan I/O berkecepatan rendah.
- ⌘ Standard yang dihasilkan dinamakan Universal Standard Bus (USB).

Keuntungan USB

- ⌘ Pemakai tidak harus memasang tombol atau jumper pada PCB atau peralatan
- ⌘ Pemakai tidak harus membuka casing untuk memasang peralatan I/O baru
- ⌘ Hanya satu jenis kabel yang diperlukan sebagai penghubung
- ⌘ Dapat mensuplai daya pada peralatan-peralatan I/O
- ⌘ Memudahkan pemasangan peralatan-peralatan yang hanya sementara dipasang pada komputer
- ⌘ Tidak diperlukan reboot pada pemasangan peralatan baru dengan USB
- ⌘ Murah

Pengkabelan USB

- ⌘ Bandwidth total USB adalah 1,5 MB per detik.
- ⌘ Bandwidth itu sudah mencukupi peralatan I/O berkecepatan rendah seperti keyboard, mouse, scanner, telepon *digital*, printer, dan sebagainya.
- ⌘ Kabel pada bus terdiri dari 4 kawat, 2 untuk data, 1 untuk power (+5 volt), dan 1 untuk ground.
- ⌘ Sistem pensinyalan mentransmisikan sebuah bilangan nol sebagai transisi tegangan dan sebuah bilangan satu bila tidak ada transmisi tegangan

Contoh Bus - Bus SCSI

- ⌘ *Small Computer System Interface* (SCSI)
- ⌘ Perangkat peripheral eksternal yang dipopulerkan oleh macintosh pada tahun 1984.
- ⌘ SCSI merupakan interface standard untuk drive CD-ROM, peralatan audio, hard disk, dan perangkat penyimpanan eksternal berukuran besar.
- ⌘ SCSI menggunakan interface paralel dengan 8, 16, atau 32 saluran data
- ⌘ Perangkat SCSI memiliki dua buah konektor
 - ⊠ Konektor input
 - ⊠ Konektor output.
- ⌘ Seluruh perangkat berfungsi secara independen dan dapat saling bertukar data
 - ⊠ misalnya *hard disk* dapat mem-back up diri ke tape drive tanpa melibatkan prosesor

Contoh Bus - Bus SCSI

⌘ Beberapa macam versi SCSI.

- ☑ SCSI-1 dibuat tahun 1980 memiliki 8 saluran data, dan beroperasi pada kecepatan 5 MHz. Versi ini memungkinkan sampai 7 perangkat dihubungkan secara daisy-chain.
- ☑ SCSI-2 diperkenalkan tahun 1992 dengan spesifikasi 16 atau 32 saluran data pada kecepatan 10 MHz.
- ☑ SCSI-3 yang mendukung kecepatan yang lebih tinggi sampai saat ini masih dalam tahap penelitian

Contoh Bus - Bus P1394 / Fire Wire

- ⌘ Kebutuhan bus I/O berkecepatan tinggi dan
- ⌘ Semakin cepatnya prosesor saat ini yang mencapai 1 GHz

>Perlu diimbangi dengan bus berkecepatan tinggi<

Dikembangkan bus performance tinggi yang dikenal dengan *Fire Wire* (P1394 standard IEEE)

Contoh Bus - Bus P1394 / Fire Wire

- ⌘ P1394 memiliki kelebihan dibandingkan dengan interface I/O lainnya
 - ☑ Sangat cepat
 - ☑ Murah
 - ☑ Mudah untuk diimplementasikan.
- ⌘ P1394 tidak hanya populer pada sistem komputer, namun juga pada peralatan elektronik
 - ☑ kamera *digital*
 - ☑ VCR
 - ☑ Televisi.
- ⌘ Kelebihan lain adalah penggunaan transmisi serial sehingga tidak memerlukan banyak kabel

Kesimpulan

1. Secara umum fungsi saluran bus dikategorikan dalam tiga bagian, yaitu saluran data, saluran alamat dan saluran kontrol.
2. Saat ini terdapat banyak implementasi sistem bus, tetapi parameter dasar perancangan bus dapat diklasifikasikan berdasarkan jenis (*dedicated* dan *multiplexed*), metode arbitrase (tersentralisasi dan terdistribusi), timing (sinkron dan tak sinkron), lebar bus (lebar address dan lebar data) dan jenis transfer datanya (*read*, *write*, *read-modify-write*, *read-alter-write*, *block*).
3. Diantara jenis bus yang beredar di pasaran saat ini adalah PCI, ISA, USB, SCSI, *FuturaBus+*, *FireWire*, dll

Soal-soal

Sebuah sistem memori terdiri dari sejumlah modul memori yang seluruhnya terhubung dengan sebuah memori. Ketika write request dilakukan, bus berisi dalam 100 ns oleh data, alamat, dan signal-signal kontrol. Selama 100 ns dan 500 ns berikutnya, modul memori beralamat mengeksekusi 1 siklus yang menerima dan menyimpan data. Operasi modul-modul memori dapat *overlap*, namun hanya akan terdapat sebuah *request* yang berada pada bus pada suatu saat.

- ☒ a) Asumsikan bahwa terdapat delapan buah modul seperti itu yang terhubung ke bus. Pada kecepatan maksimum berapakah (dalam *word/detik*) dimana data dapat masih disimpan?
- ☒ b) Buatlah sketsa diagram kecepatan write maksimum sebagai fungsi waktu siklus modul, dengan mengasumsikan delapan modul memori dan waktu sibuk bus 100 ns.

