Control Unit Operation

William Stallings
Computer Organization
and Architecture
8th Edition

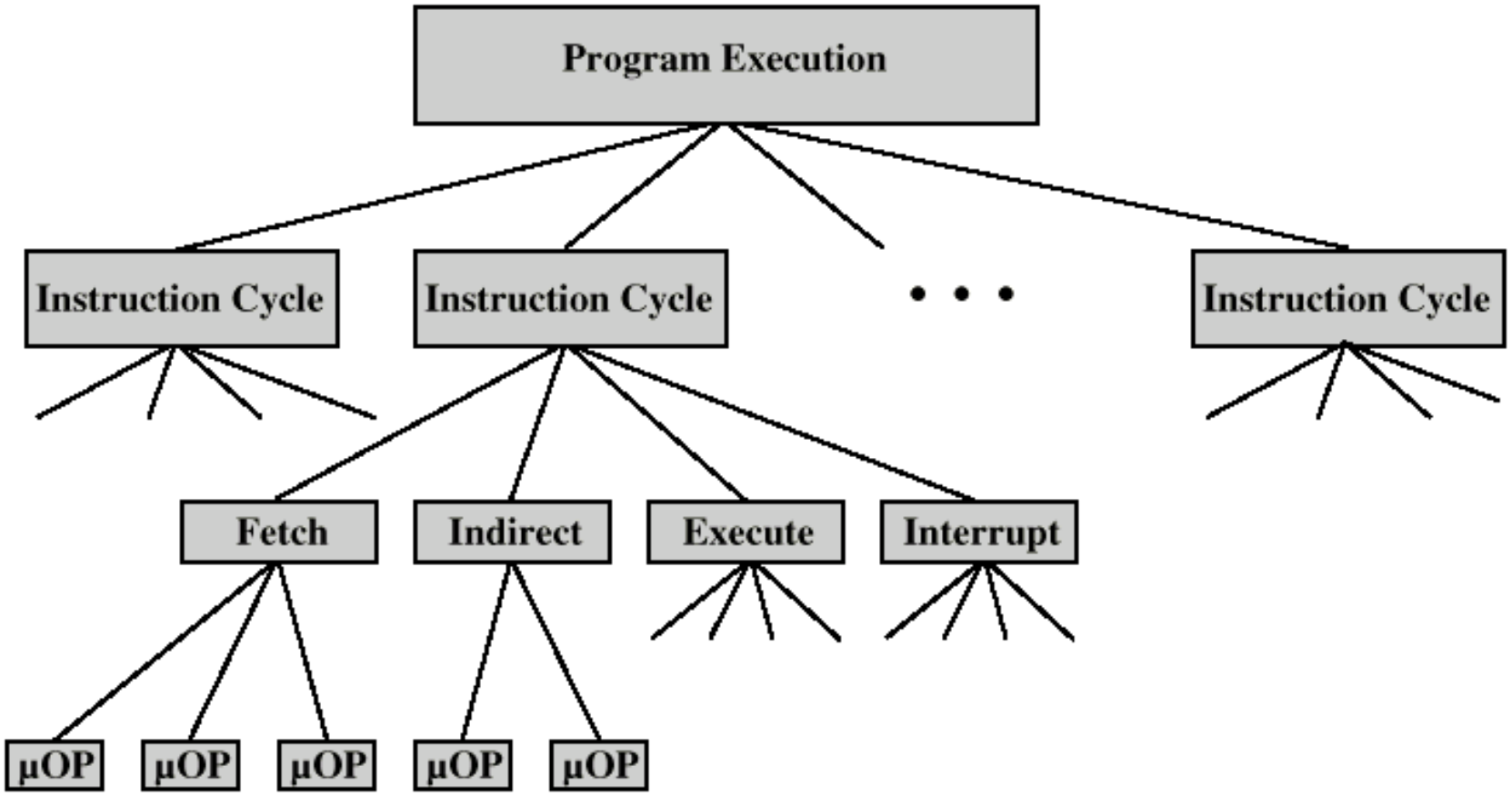
UNDIP | UNIVERSITAS DIPONEGORO
becomes an excellent research university

Micro-Operations

- Komputer menjalankan program sesuai intruksi-intruksi yang tertulis dalam program.
- Fetch/execute cycle
  - Proses eksekusi program melalui siklus fetch/eksekusi
- Setiap siklus terdiri dari beberapa langkah
- Langkah-Langkah tersebut disebut micro-operations
- Setiap micro-operation melakukan tugas sangat kecil dan spesifik
- Micro-operation adalah atomic operation dalam CPU (operasi terkecil, tidak dapat dibagi lagi)

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Constituent Elements of Program Execution



UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

4 Registers pada Fetch

- Memory Address Register (MAR)
  - MAR terhubung ke address bus untuk mengirimkan alamat memori yang akan diakses oleh CPU.
  - MAR menentukan alamat untuk operasi baca atau tulis
- Memory Buffer Register (MBR)
  - MBR terhubung ke data bus, untuk mengirimkan data antara CPU dan memori.
  - Menyimpan data untuk ditulis atau data terakhir dibaca
- Program Counter (PC)
  - Menyimpan alamat instruksi selanjutnya yang akan diambil
- Instruction Register (IR)
  - Menahan instruksi terakhir yang diambil

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Fetch Sequence

- Alamat instruksi berikutnya yang akan diambil tersimpan dalam Program Counter (PC).
- Alamat (MAR) ditempatkan pada address bus
- Control unit (unit kontrol) mengeluarkan perintah READ untuk membaca data dari alamat memori yang ditunjukkan oleh MAR.
- Result (data from memory) appears on data bus
- Data from data bus copied into MBR
- PC incremented by 1 (in parallel with data fetch from memory)
- Data (instruction) moved from MBR to IR
- MBR is now free for further data fetches

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Fetch Sequence (symbolic)

- t1: MAR <- (PC) (isi dari PC disalin ke MAR)
- t2: MBR <- (memory) (data yang ada di alamat memori yang ditunjukkan oleh MAR disalin ke MBR)
- PC <- (PC) +1 (PC diinkrementasi untuk menunjuk ke alamat instruksi berikutnya)
- t3: IR <- (MBR) (data yang ada di salin ke IR)
- (tx = time unit/clock cycle)
- or
- t1: MAR <- (PC)
- t2: MBR <- (memory)
- t3: PC <- (PC) +1
- IR <- (MBR)

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Rules for Clock Cycle Grouping

- Urutan yang bernar harus diikuti
  - MAR <- (PC) harus mendahului MBR <- (memory)
- Konflik harus dihindari
  - Tidak boleh membaca dan menulis ke register yang sama pada waktu yang bersamaan.
  - MBR <- (memory) & IR <- (MBR) tidak boleh berada pada siklus yang sama.
- Also: PC <- (PC) +1 melibatkan operasi penambahan
  - Menggunakan ALU
  - Mungkin memerlukan micro-operations tambahan

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Indirect Cycle

- MAR <- ($IR_{address}$) - address field of IR
- MBR <- (memory)
- $IR_{address}$ <- ($MBR_{address}$)

- MBR berisi sebuah address
- IR sekarang berada dalam keadaan yang sama seolah-olah pengalamatan langsung telah digunakan.
- (What does this say about IR size?)

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Interrupt Cycle

- t1: MBR <-(PC)
- t2: MAR <- save-address
- PC <- routine-address
- t3: memory <- (MBR)
- This is a minimum
  - Mungkin ada operasi mikro tambahan yang diperlukan untuk mendapatkan atau menghitung alamat yang diperlukan selama siklus interupsi.
  - N.B. saving context is done by interrupt handler routine, not micro-ops

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Execute Cycle (ADD)

- Siklus eksekusi berbeda untuk setiap jenis intruksi.
- Contoh: ADD R1,X - Instruksi ini berarti menambahkan isi dari lokasi memori X ke Register 1 (R1) dan menyimpan hasilnya di R1.
  - t1: MAR <- ($IR_{address}$) (MAR diisi dengan alamat yang ada di IR)
  - t2: MBR <- (memory) (data dari alamat yang ditunjukkan oleh MAR dibaca dari memori dan disalin ke MBR)
  - t3: R1 <- R1 + (MBR) (Register 1 ditambahkan dengan nilai yang ada di MBR)
- Tidak ada tumpang tindih (overlap) dari operasi mikro dalam langkah-langkah ini.

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Execute Cycle (ISZ)

- ISZ X - increment and skip if zero
- **ISZ X** berfungsi untuk menambahkan 1 ke nilai di lokasi memori X dan melompat ke instruksi berikutnya jika hasil penjumlahan tersebut adalah nol
  - t1: MAR <- ($IR_{address}$)
  - t2: MBR <- (memory)
  - t3: MBR <- (MBR) + 1
  - t4: memory <- (MBR)
  - if (MBR) == 0 then PC <- (PC) + 1
- Notes:
  - jika nilai MBR adalah nol dianggap sebagai satu operasi mikro.
  - Operasi mikro dilakukan selama t4

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Execute Cycle (BSA)

- BSA X - Branch and save address
  - Alamat instruksi setelah BSA disimpan di X
  - Eksekusi berlanjut dari X+1
  - t1: MAR <- ($IR_{address}$)
  - MBR <- (PC)
  - t2: PC <- ($IR_{address}$)
  - memory <- (MBR)
  - t3: PC <- (PC) + 1

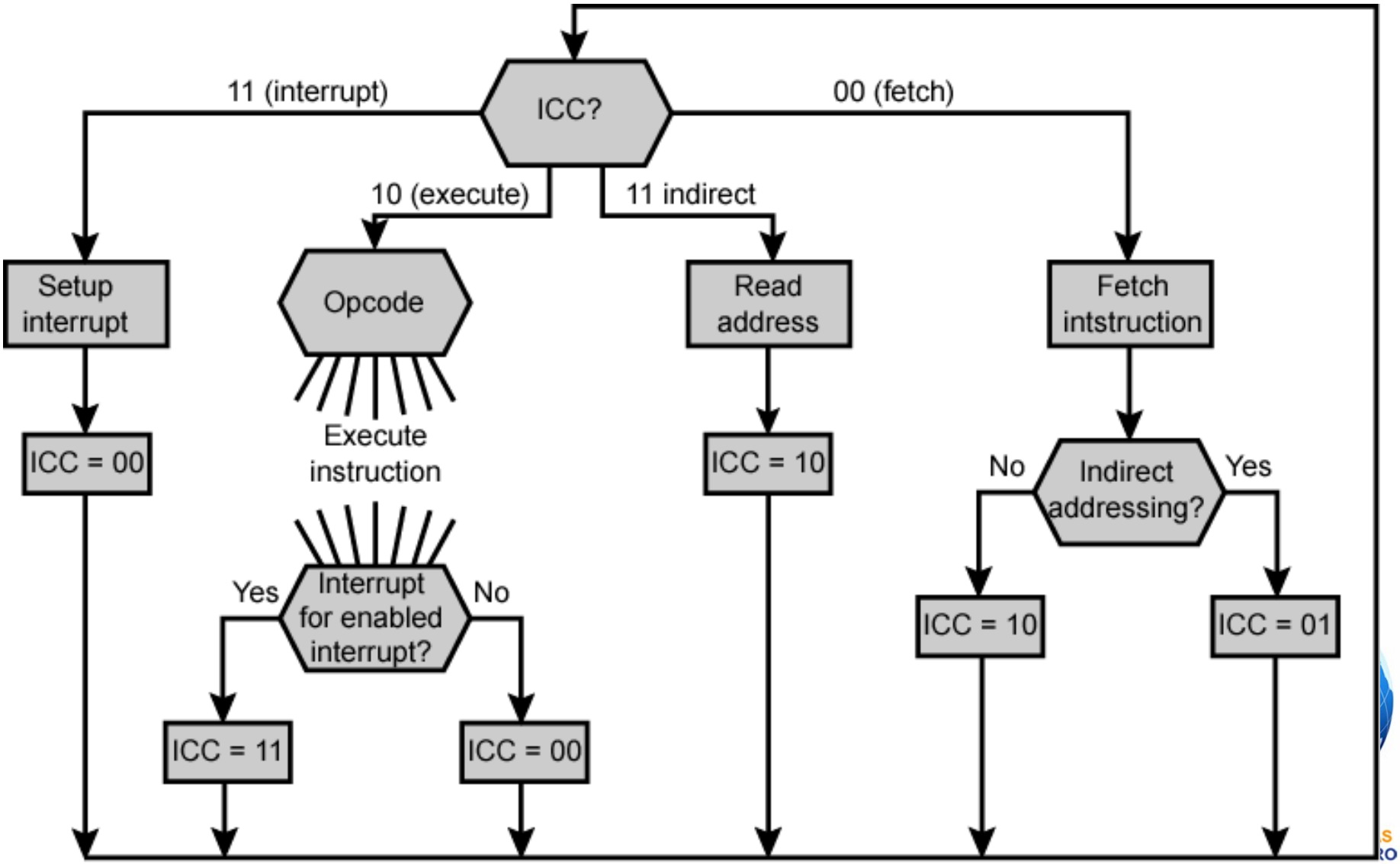
UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Instruction Cycle

- Setiap phase didekomposisi menjadi urutan micro operasi dasar.
- Misalnya: fetch, indirect, and interrupt cycles
- Execute cycle
  - Satu urutan micro-operations untuk setiap opcode
- Perlu menyatukan urutan
- Asumsikan 2-bit register baru
  - Instruction cycle code (ICC) menunjukkan bagian mana dari siklus prosesor yang berada
    - 00: Fetch
    - 01: Indirect
    - 10: Execute
    - 11: Interrupt

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Flowchart for Instruction Cycle



Functional Requirements sebuah processor

- Mendefinisikan elemen dasar dari prosesor
- Jelaskan operasi mikro yang dilakukan prosesor
- Tentukan fungsi yang harus dilakukan oleh control unit

UNDIP | UNIVERSITAS DIPONEGORO
becomes an excellent research university

Basic Elements of Processor

- ALU
- Registers
- Internal data paths
- External data paths
- Control Unit

UNDIP | UNIVERSITAS DIPONEGORO
becomes an excellent research university

Types of Micro-operation

- Transfer data between registers
- Transfer data from register to external
- Transfer data from external to register
- Melakukan arithmetic or logical ops

UNDIP | UNIVERSITAS DIPONEGORO
becomes an excellent research university

Functions of Control Unit

- Sequencing (pengurutan)
  - Menyebabkan CPU melewati serangkaian operasi mikro
- Execution
  - Menyebabkan kinerja setiap operasi mikro
- Dilakukan dengan menggunakan Control Signals

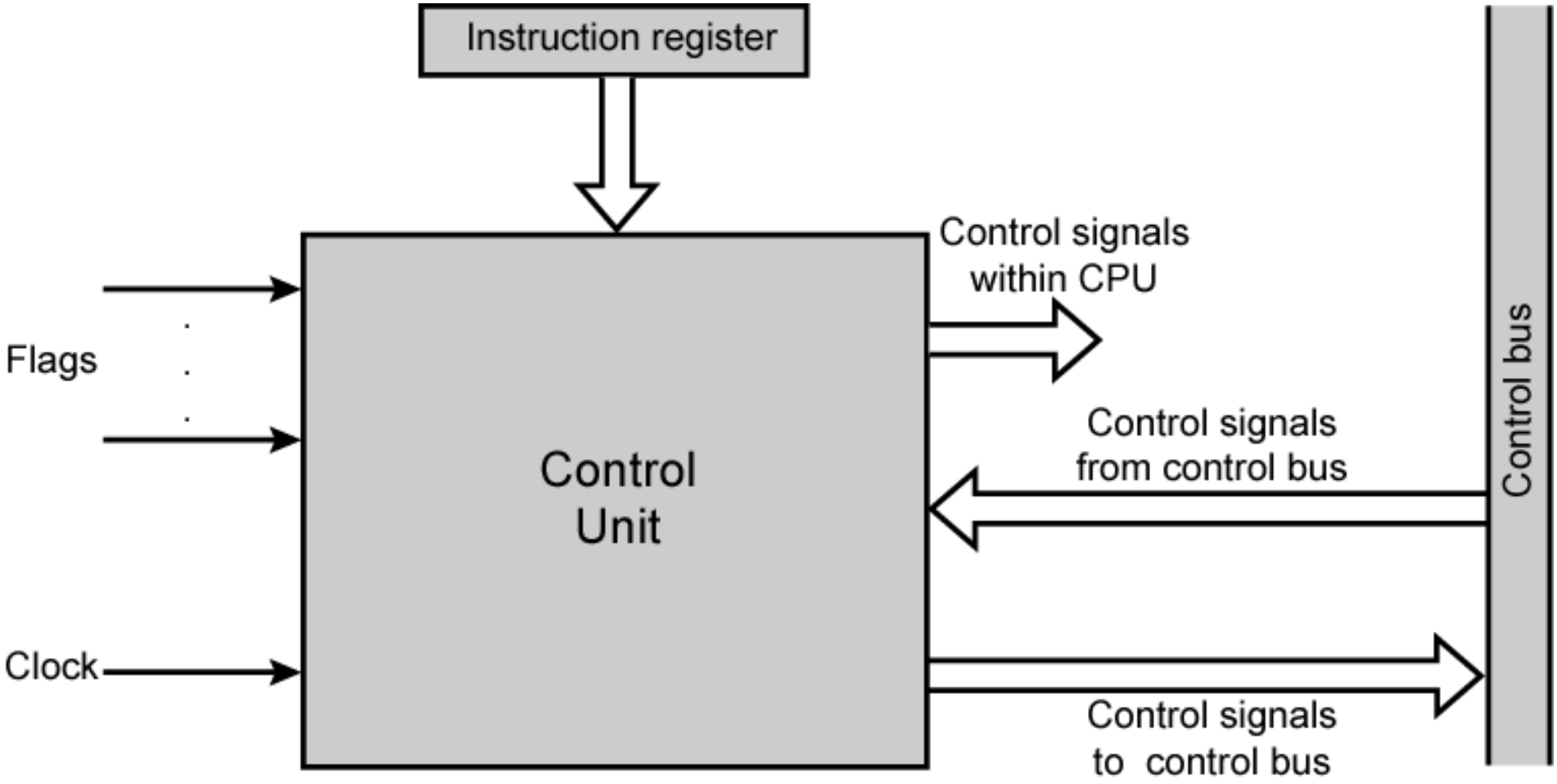
UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Control Signals

- Clock
  - One micro-instruction (or set of parallel micro-instructions) per clock cycle
- Instruction register
  - Op-code for current instruction
  - Menentukan intruksi micro-instructions mana yang dilakukan
- Flags
  - State of CPU
  - Results of previous operations
- From control bus
  - Interrupts
  - Acknowledgements (balasan)

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Model of Control Unit



UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Control Signals - output

- Within CPU (dalam)
  - Menyebabkan pergerakan data
  - Mengaktifkan fungsi tertentu
- Via control bus
  - To memory
  - To I/O modules

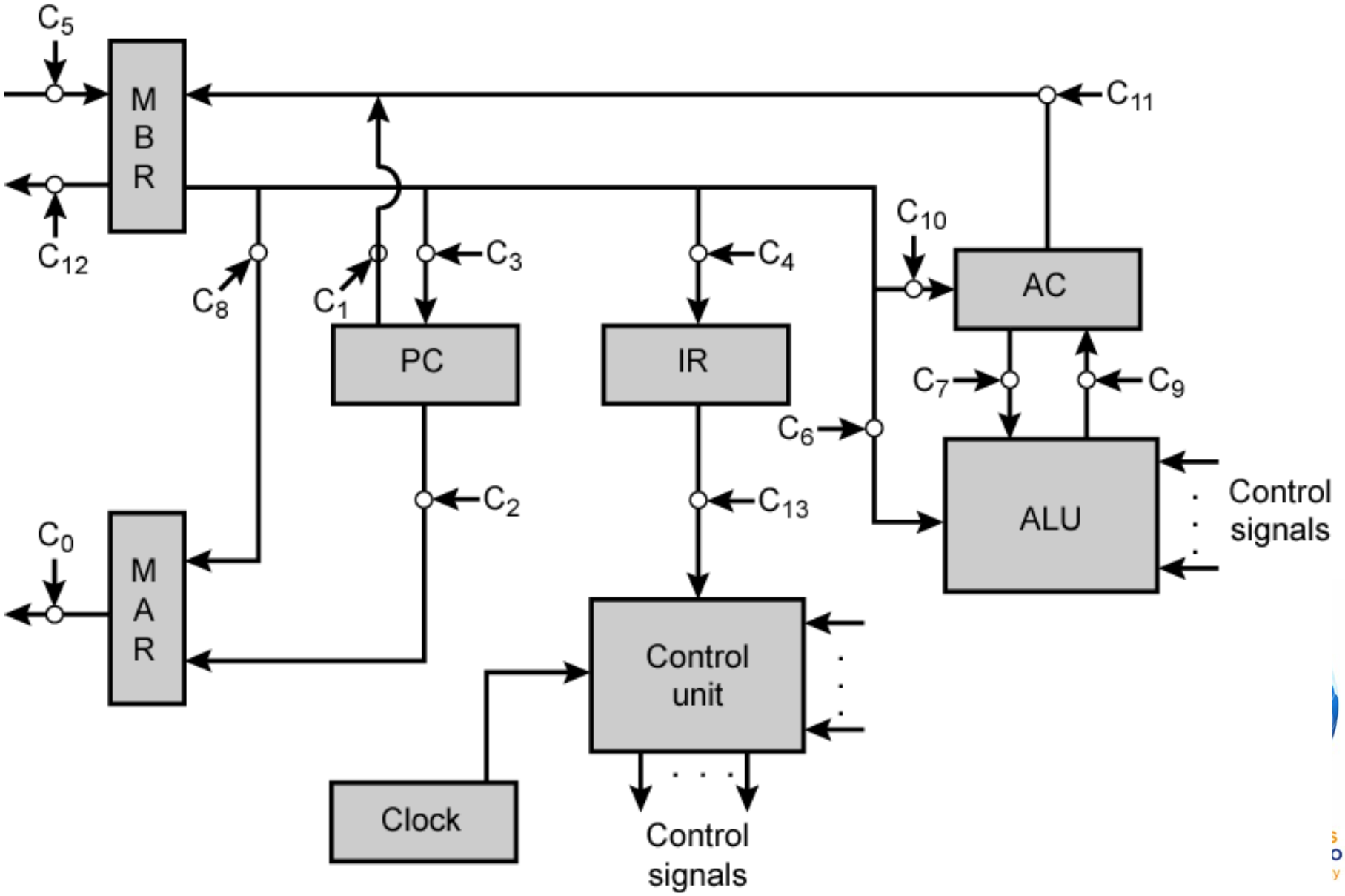
UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Example Control Signal Sequence - Fetch

- MAR <- (PC)
  - Control unit activates signal to open gates between PC and MAR
- MBR <- (memory)
  - Open gates between MAR and address bus
  - Memory read control signal
  - Open gates between data bus and MBR

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Data Paths and Control Signals

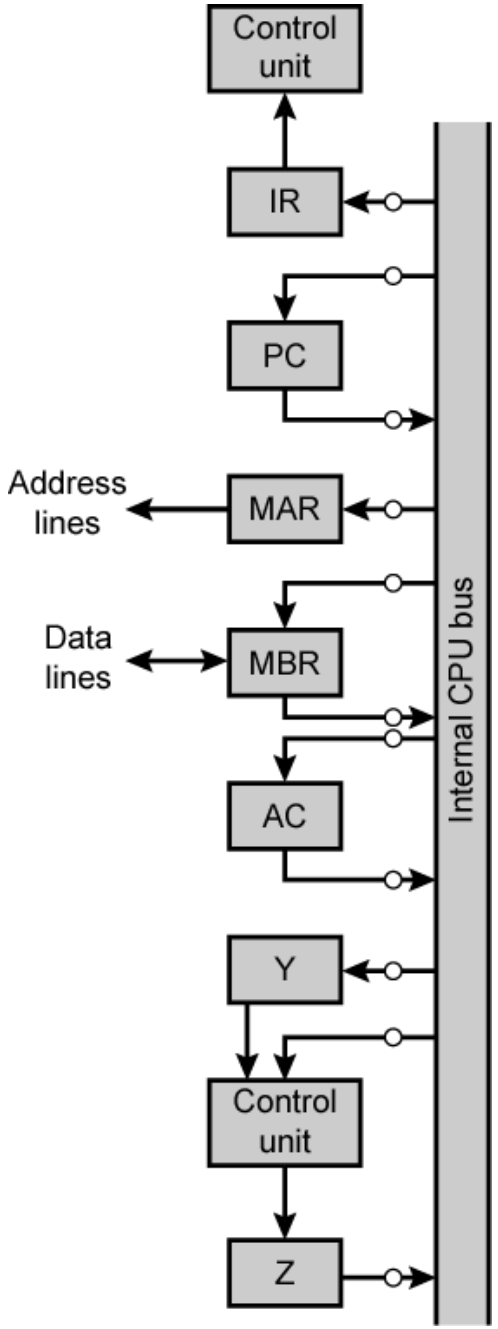


Internal Organization

- Kebanyakan CPU menggunakan satu bus internal untuk memindahkan data antara komponen internal.
- Gates mengontrol pergerakan data ke dan dari bus.
- Control signals mengontrol transfer data ke dan dari external systems bus
- Temporary registers diperlukan untuk operasi ALU yang tepat.

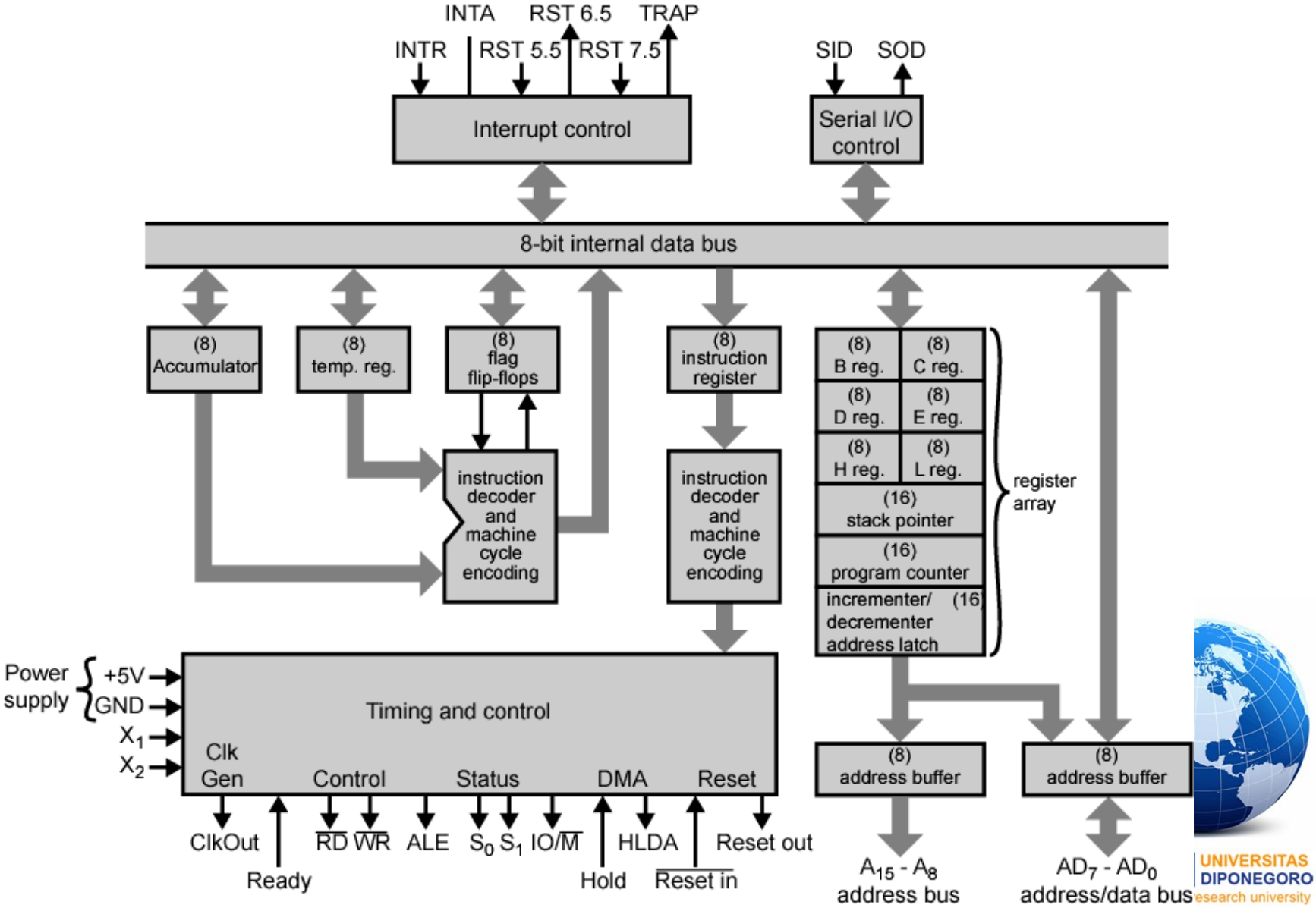
UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

CPU with Internal Bus



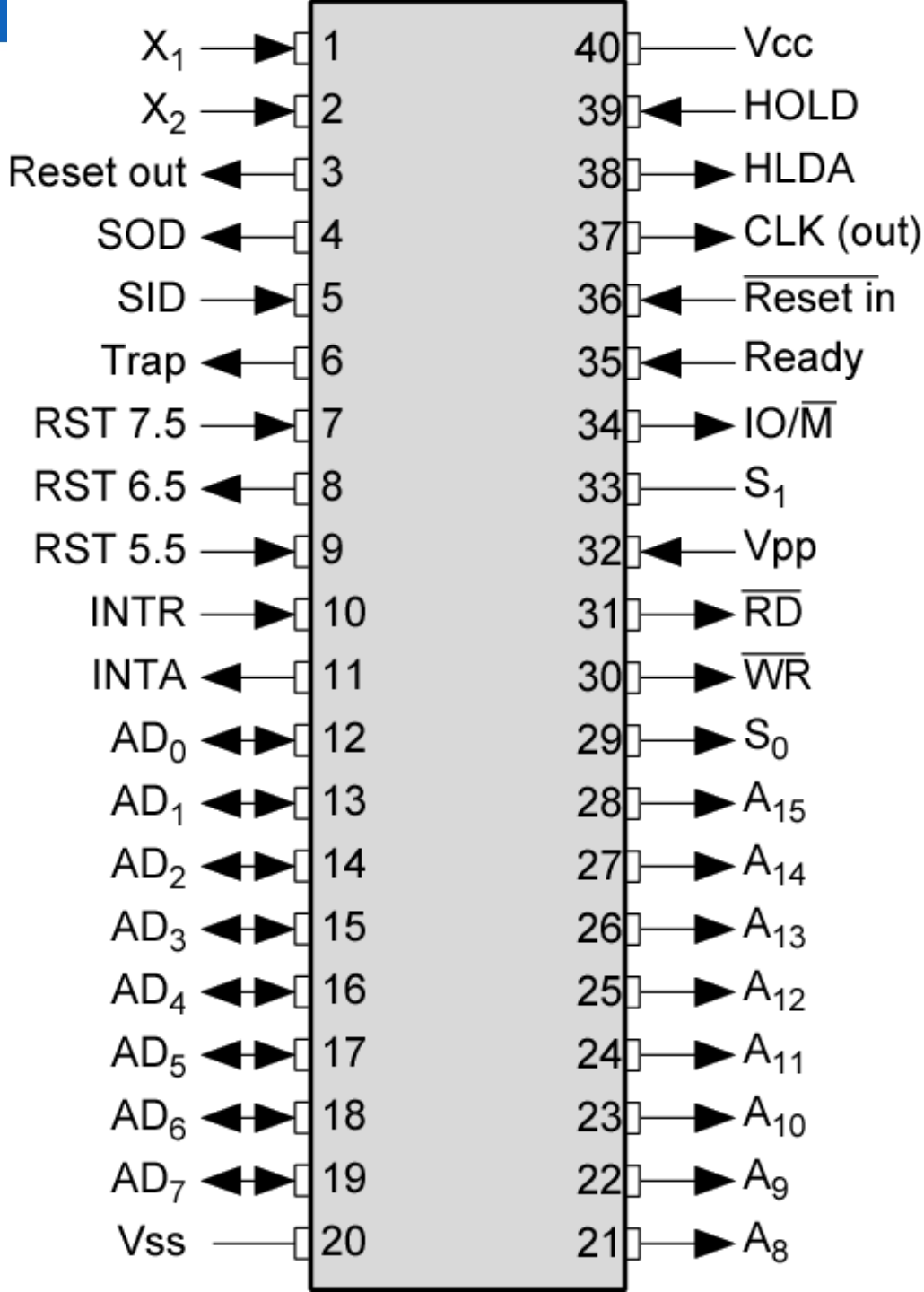
UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Intel 8085 CPU Block Diagram

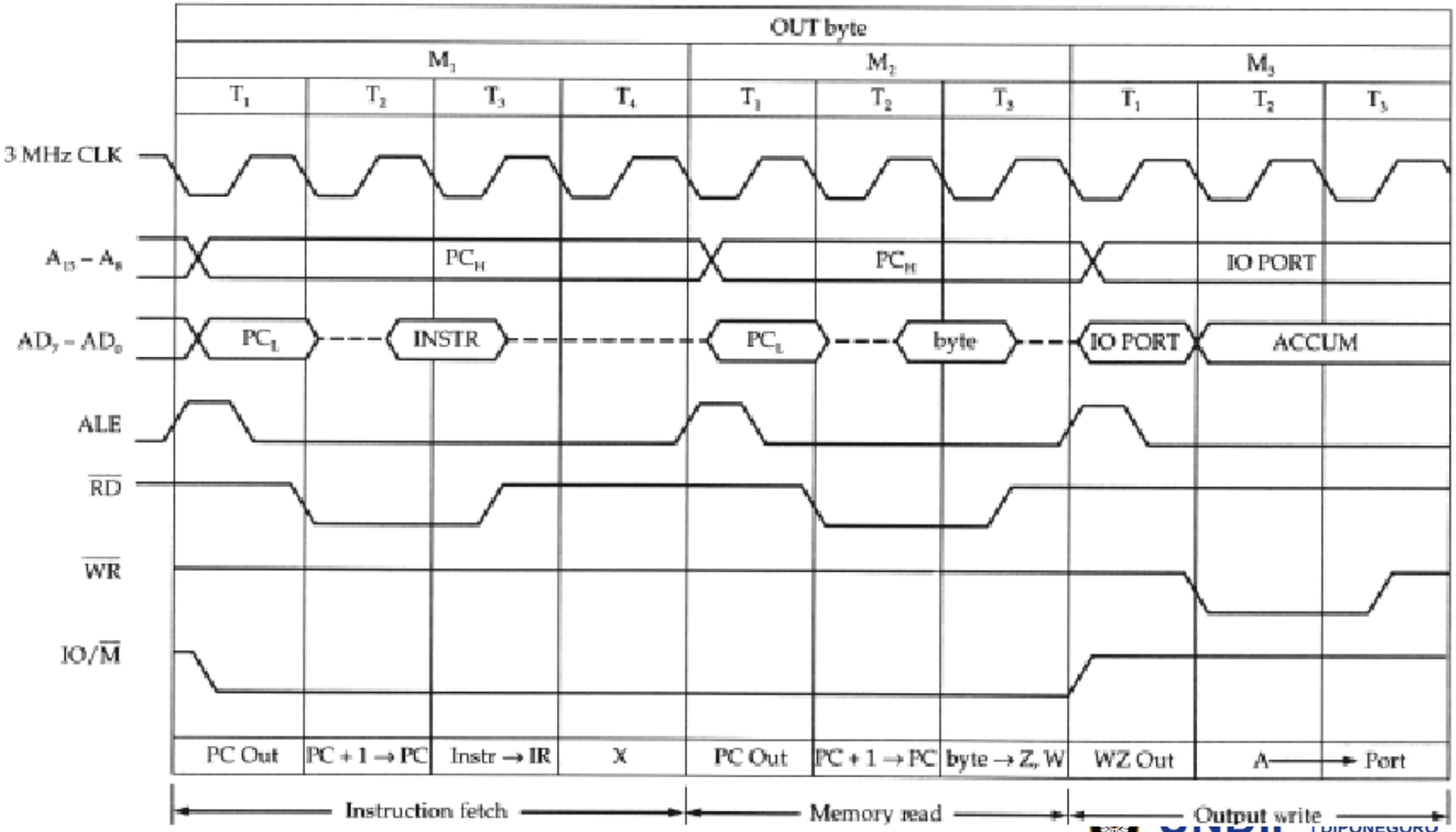


UNIVERSITAS DIPONEGORO
esearch university

Intel 8085 Pin Configuration



Intel 8085 OUT Instruction Timing Diagram



becomes an excellent research university

Hardwired Implementation (1)

- Control unit inputs
- Flags and control bus
  - Setiap bit mempunyai makna tertentu
- Instruction register
  - Kode operasi (op-code) dari sebuah instruksi menyebabkan sinyal kontrol yang berbeda-beda untuk setiap instruksi yang berbeda.
  - Unique logic for each op-code
  - Decoder mengambil input yang di encode and menghasilkan single output
  - *n* binary inputs and 2^n outputs

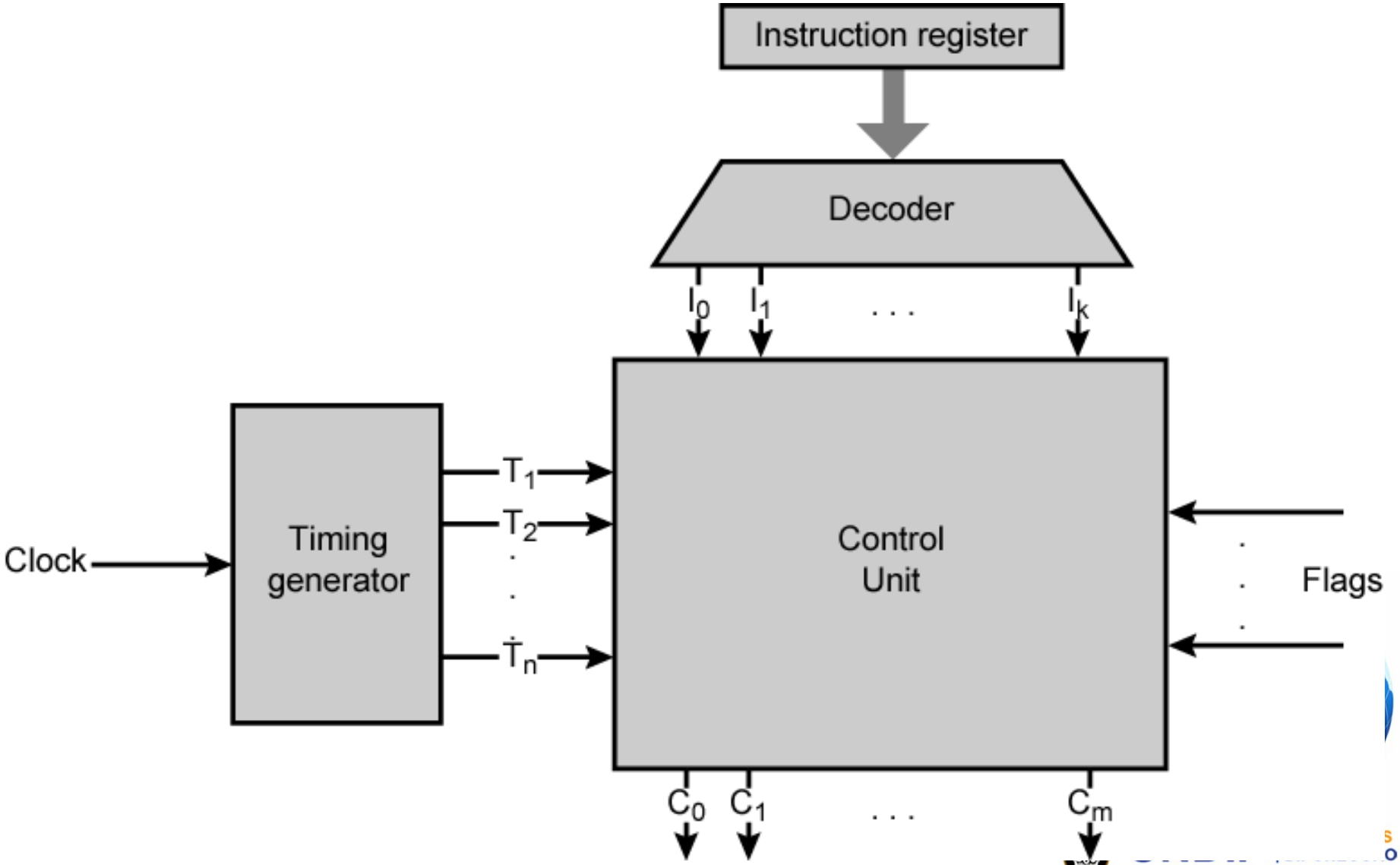
UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Hardwired Implementation (2)

- Clock
  - Repetitive sequence of pulses
  - Useful for measuring duration of micro-ops
  - Must be long enough to allow signal propagation
  - Different control signals at different times within instruction cycle
  - Need a counter with different control signals for t1, t2 etc.

UNDIP UNIVERSITAS DIPONEGORO
becomes an excellent research university

Control Unit with Decoded Inputs



becomes an excellent research university

Problems With Hard Wired Designs

- Complex sequencing & micro-operation logic
- Difficult to design and test
- Inflexible design
- Difficult to add new instructions

UNDIP | UNIVERSITAS DIPONEGORO
becomes an excellent research university

Required Reading

- Stallings chapter 15

UNDIP | UNIVERSITAS DIPONEGORO
becomes an excellent research university

TERIMAKASIH

UNDIP | UNIVERSITAS DIPONEGORO
becomes an excellent research university