
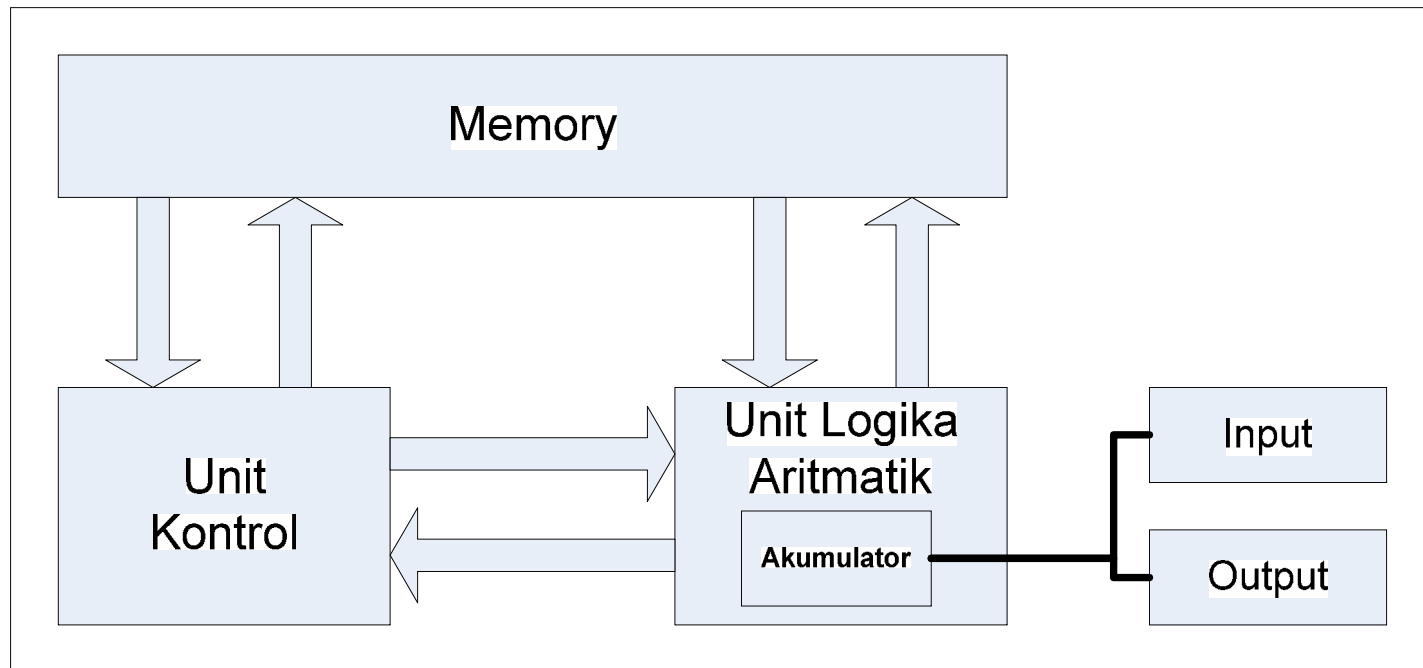


# ARSITEKTUR VON NEUMANN

Arsitektur Computer

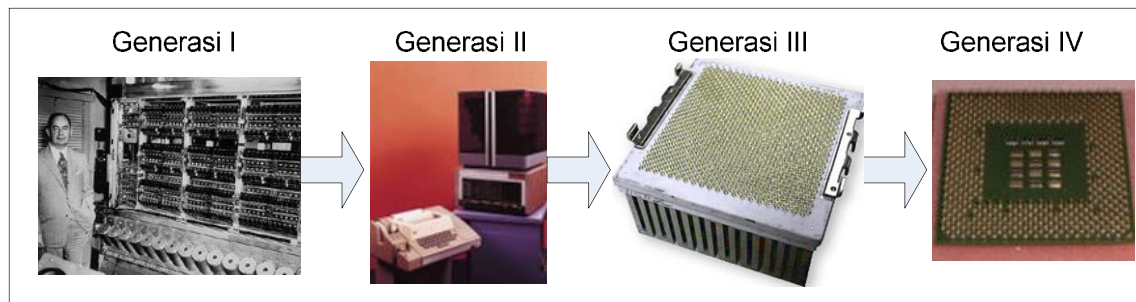
- 
- Pada jaman modern saat ini, hampir semua komputer mengadopsi arsitektur yang dibuat oleh John von Neumann (1903-1957).
  - Kunci utama arsitektur von Neumann adalah unit pemrosesan sentral (CPU), yang memungkinkan seluruh fungsi komputer untuk dikoordinasikan melalui satu sumber tunggal.



- Pada dasarnya komputer arsitektur Von Neumann adalah terdiri dari elemen sebagai berikut:
  - ▣ Prosesor, merupakan pusat dari kontrol dan pemrosesan instruksi pada komputer.
  - ▣ Memori, digunakan untuk menyimpan informasi baik program maupun data.
  - ▣ Perangkat input-output, berfungsi sebagai media yang menangkap respon dari luar serta menyajikan informasi keluar sistem komputer.

# Prosesor atau Central Processing Unit (CPU)

- CPU merupakan tempat untuk melakukan pemrosesan instruksi-instruksi dan pengendalian sistem komputer.
- Perkembangan perangkat CPU mengikuti generasi dari sistem komputer.
- Pada generasi pertama CPU terbuat dari rangkaian tabung vakum sehingga memiliki ukuran yang sangat besar.
- Pada generasi kedua telah diciptakan transistor sehingga ukuran CPU menjadi lebih kecil dari sebelumnya.
- Pada generasi ketiga CPU telah terbuat dari rangkaian IC sehingga ukurannya menjadi lebih kecil.
- Pada generasi keempat telah diciptakan teknologi VLSI dan ULSI sehingga memungkinkan ribuan sampai jutaan transistor tersimpan dalam satu chip.



# Elemen CPU

- Pada perkembangan komputer modern, setiap prosesor terdiri atas:
  - ▣ Arithmetic and Logic Unit (ALU).
  - ▣ Register.
  - ▣ Control Unit (CU).
  - ▣ Bus.

# Arithmetic and Logic Unit (ALU).

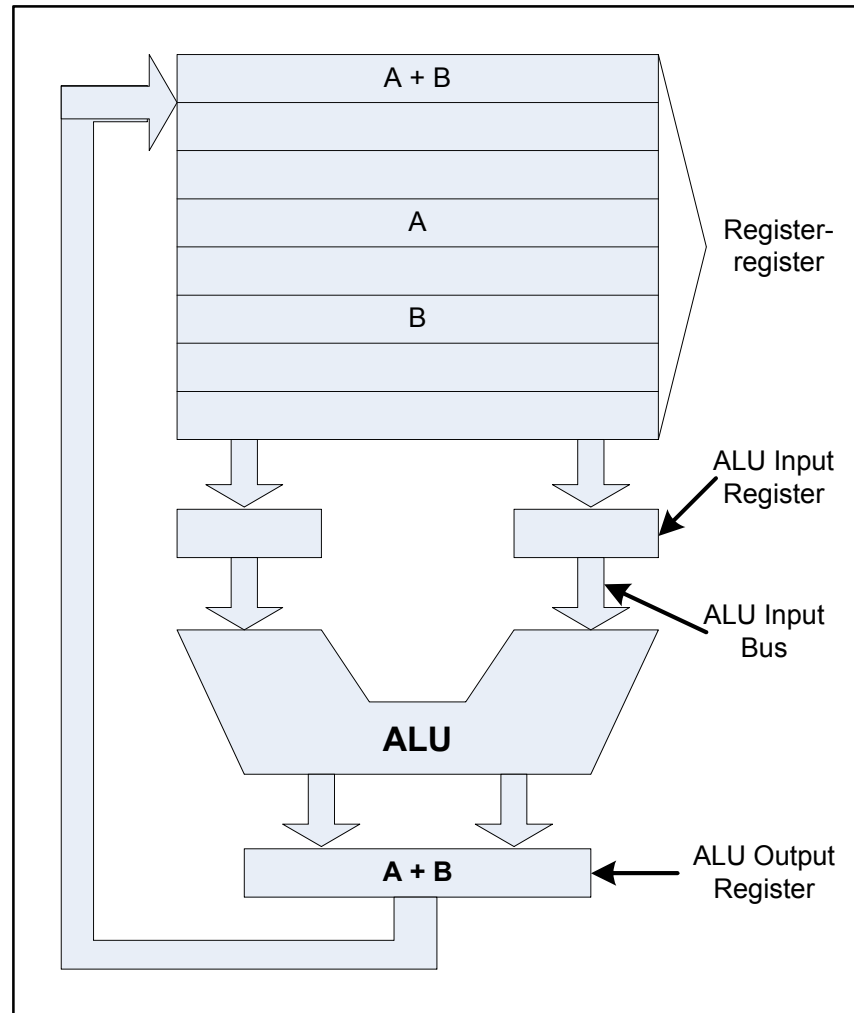


- Arithmetic and Logic Unit atau Unit Aritmetika dan Logika berfungsi untuk melakukan semua perhitungan aritmatika (matematika) dan logika yang terjadi sesuai dengan instruksi program.
- ALU menjalankan operasi penambahan, pengurangan, dan operasi-operasi sederhana lainnya pada input-inputnya dan memberikan hasilnya pada register output.

# Register



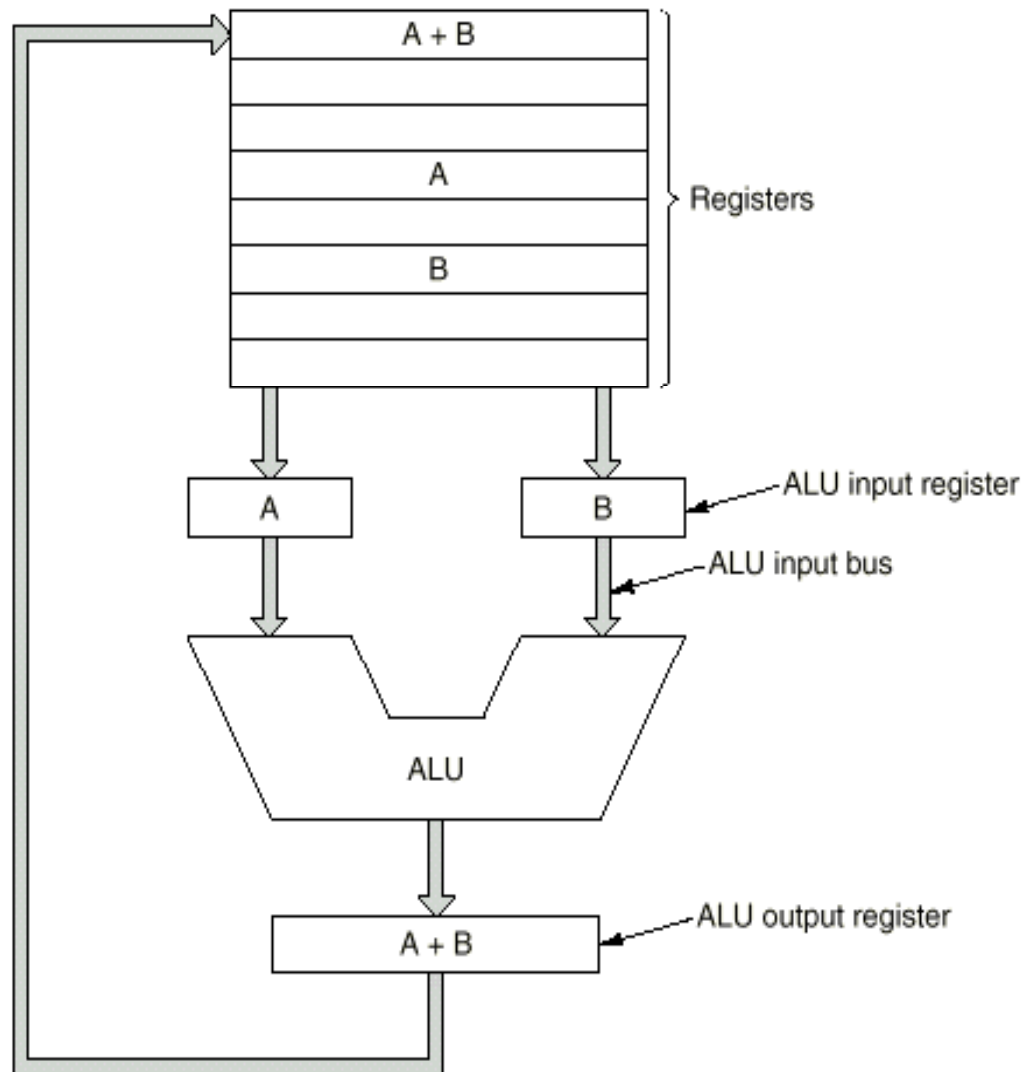
- ❑ Register merupakan alat penyimpanan kecil yang mempunyai kecepatan akses cukup tinggi, yang digunakan untuk menyimpan data dan instruksi yang sedang diproses, sementara data dan instruksi lainnya yang menunggu giliran untuk diproses masih disimpan di dalam memori utama. Setiap register dapat menyimpan satu bilangan hingga mencapai jumlah maksimum tertentu tergantung pada ukurannya.
- ❑ Register-register dapat dibaca dan ditulis dengan kecepatan tinggi karena berada pada CPU.





# How ALU works

9



## Example:

- A contains 10
- B contains 2
- ALU calculates  $10 + 2$ , sends result back to register

# Beberapa jenis register adalah:

- Program Counter (PC), merupakan register yang menunjuk ke instruksi berikutnya yang harus diambil dan dijalankan.
- Instruction Register (IR), merupakan register yang menyimpan instruksi yang sedang dijalankan.
- General Purpose Register, merupakan register yang memiliki kegunaan umum yang berhubungan dengan data yang diproses.
- Memory Data Register (MDR), merupakan register yang digunakan untuk menampung data atau instruksi hasil pengiriman dari memori utama ke CPU atau menampung data yang akan direkam ke memori utama dari hasil pengolahan oleh CPU.
- Memory address register (MAR), merupakan register yang digunakan untuk menampung alamat data atau instruksi pada memori utama yang akan diambil atau yang akan diletakkan.
- Sebagian besar komputer memiliki beberapa register lain, sebagian digunakan untuk tujuan umum, dan sebagian lainnya untuk tujuan khusus.

# Control Unit (CU)

- Control Unit atau Unit Kontrol berfungsi untuk mengatur dan mengendalikan semua peralatan yang ada pada sistem komputer. Unit kendali akan mengatur kapan alat input menerima data dan kapan data diolah serta kapan ditampilkan pada alat output.
- Unit ini juga mengartikan instruksi-instruksi dari program komputer, membawa data dari alat input ke memori utama, dan mengambil data dari memori utama untuk diolah.
- Bila ada instruksi untuk perhitungan aritmatika atau perbandingan logika, maka unit kendali akan mengirim instruksi tersebut ke ALU. Hasil dari pengolahan data dibawa oleh unit kendali ke memori utama lagi untuk disimpan, dan pada saatnya akan disajikan ke alat output.

# Bus



- Bus merupakan penghubung antara semua komponen CPU.
- Bus berupa sekumpulan kabel-kabel paralel untuk mentransmisikan alamat (address), data, dan sinyal-sinyal kontrol.



# **Klasifikasi Prosesor**

# Berdasarkan jenis mikroprosesor:



- Tipe Intel untuk Personal Computer (PC), diproduksi oleh Intel Corp., Advanced Micro Devices (AMD), Cyrix, DEC, dll.
- Tipe Motorola untuk komputer Macintosh, diproduksi oleh Motorola.

# Ukuran kecepatan prosesor adalah:

- Hertz, yaitu jumlah clock atau ketukan prosesor tiap satu detik. Untuk prosesor modern memakai satuan Megahertz atau Gigahertz.
- MIPS, singkatan dari Million Instruction Per Second, yaitu jumlah instruksi dalam juta tiap satu detik.
- Fops, singkatan dari Floating Point per Second, yaitu jumlah perhitungan floating point tiap satu detik. Floating point adalah metode untuk menuliskan bilangan dengan mantisa, contoh:  $3 \times 10^{-5}$ .
- Fractions of a second, yaitu waktu eksekusi relatif dari suatu instruksi pada sistem komputer.

# Dalam desain mikroprosesor, terdapat dua jenis desain, yaitu:



- ❑ CISC (Complex instruction set computing chips), dapat menampung banyak instruksi yang kompleks.
- ❑ RISC (Reduced instruction set computing chips), dapat meringkas beberapa instruksi sehingga dapat mempercepat kerja prosesor.




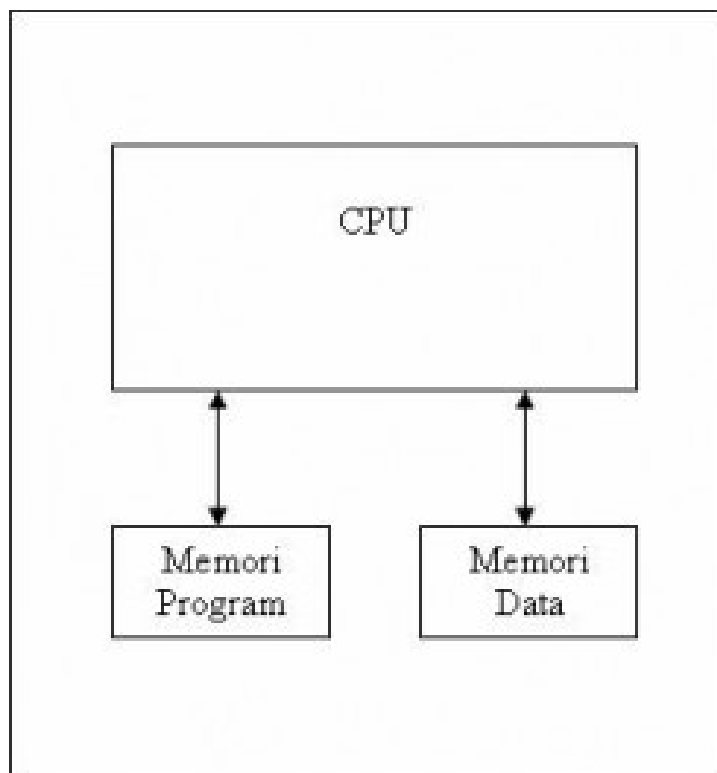
# Jenis bus yang telah didukung oleh sistem komputer saat ini adalah:

- ISA, singkatan dari Industry Standard Architecture, merupakan jenis bus standar pertama yang digunakan industri. Bus ISA beroperasi pada kecepatan 8.33 MHz. Versi perkembangan dari ISA adalah EISA (Extended ISA).
- PCI, singkatan dari Peripheral Component Interconnect bus, merupakan jenis bus yang dikembangkan dan dipatenkan oleh Intel pada tahun 1990. Versi pertama PCI beroperasi pada kecepatan 33 MHz dengan bandwidth 133 MB/dtk. PCI 2.0 diperkenalkan tahun 1993 dan PCI 2.1 tahun 1995 dengan bandwidth 528 MB/dtk.
- AGP, singkatan dari Accelerator Graphic Port, merupakan bus hasil perkembangan dari PCI yang dikhususkan untuk pemrosesan data grafik dan video.
- USB, singkatan dari Universal Serial Bus, pada awalnya dikembangkan secara bersama-sama oleh tujuh perusahaan, yaitu Compaq, DEC, IBM, Intel, Microsoft, NEC, dan Northern Telecom. Saat ini USB telah menjadi standar yang digunakan secara luas dalam Personal Computer.

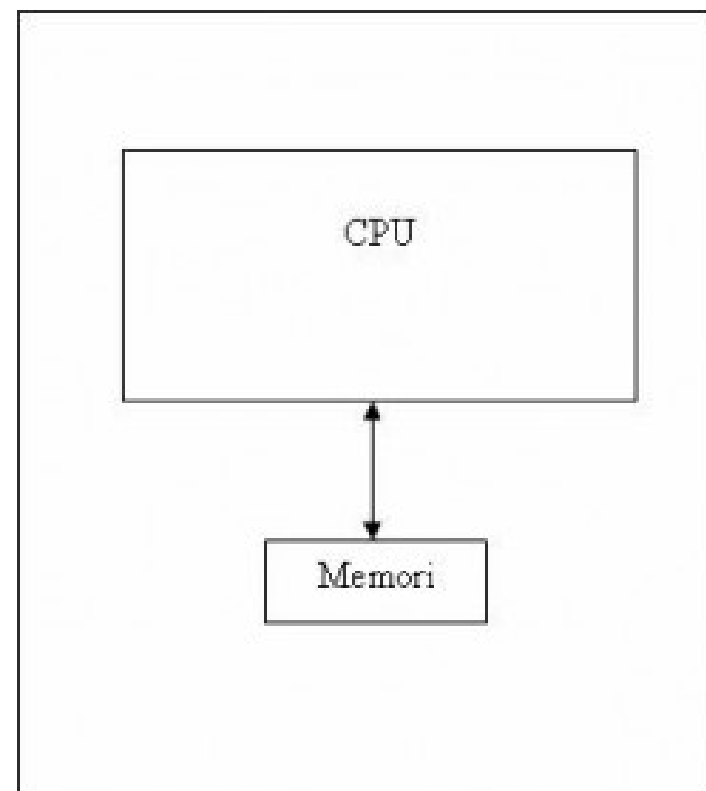


# Von Neumann vs Harvard

- 
- Ada dua kelas utama dari arsitektur komputer, yaitu 'arsitektur Havard' dan 'arsitektur Von Neumann (atau Princeton).
  - Banyak desain khusus mikrokontroler dan DSP (Digital Signal Processor) menggunakan arsitektur Havard.



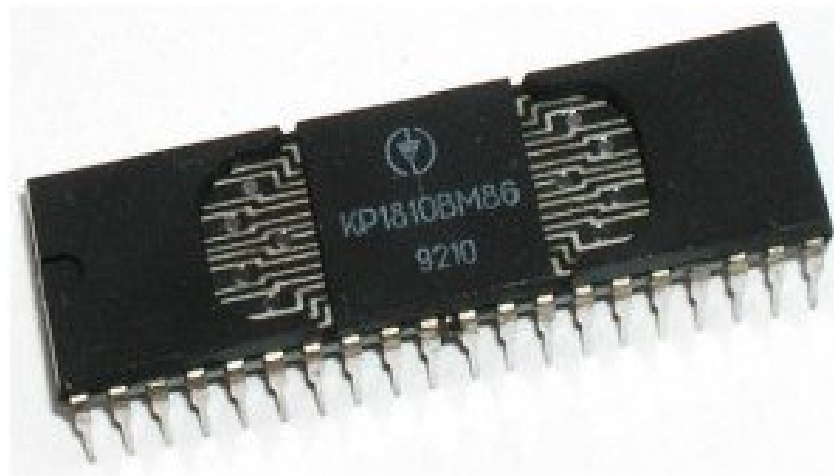
Arsitektur Havard



Arsitektur Von Neumann

# Arsitektur Harvard

- Arsitektur Harvard menggunakan memori terpisah untuk program dan data dengan alamat dan bus data yang berdiri sendiri.
- Karena dua perbedaan aliran data dan alamat, maka tidak diperlukan multiplexing alamat dan bus data.
- Arsitektur ini tidak hanya didukung dengan bus paralel untuk alamat dan data, tetapi juga menyediakan organisasi internal yang berbeda sedemikian rupa instruksi dapat diambil dan dikodekan ketika berbagai data sedang diambil dan dioperasikan.
- Lebih lanjut lagi, bus data bisa saja memiliki ukuran yang berbeda dari bus alamat. Hal ini memungkinkan pengoptimalan bus data dan bus alamat dalam pengeksekusian instruksi yang cepat.
- Sebagai contoh, mikrokontroler Intel keluarga MCS-51 menggunakan arsitektur Harvard karena ada perbedaan kapasitas memori untuk program dan data, dan bus terpisah (internal) untuk alamat dan data. Begitu juga dengan keluarga PIC dari Microchip yang menggunakan arsitektur Harvard.



# Arsitektur Von Neumann

- Pada arsitektur Von Neumann, program dan data dibagi pada ruang memori yang sama.
- Arsitektur Von Neumann menyediakan fitur penyimpanan dan modifikasi program secara mudah. Bagaimanapun, penyimpanan program tidak mungkin optimal dan membutuhkan berbagai pengumpulan program dan data untuk membentuk instruksi. Pengumpulan program dan data diselesaikan menggunakan time division multiplexing yang akan berpengaruh pada performa mikrokontroler itu sendiri.
- Salah satu contoh mikrokontroler yang menggunakan arsitektur Von Neumann (princeton) adalah Motorola 68HC11.

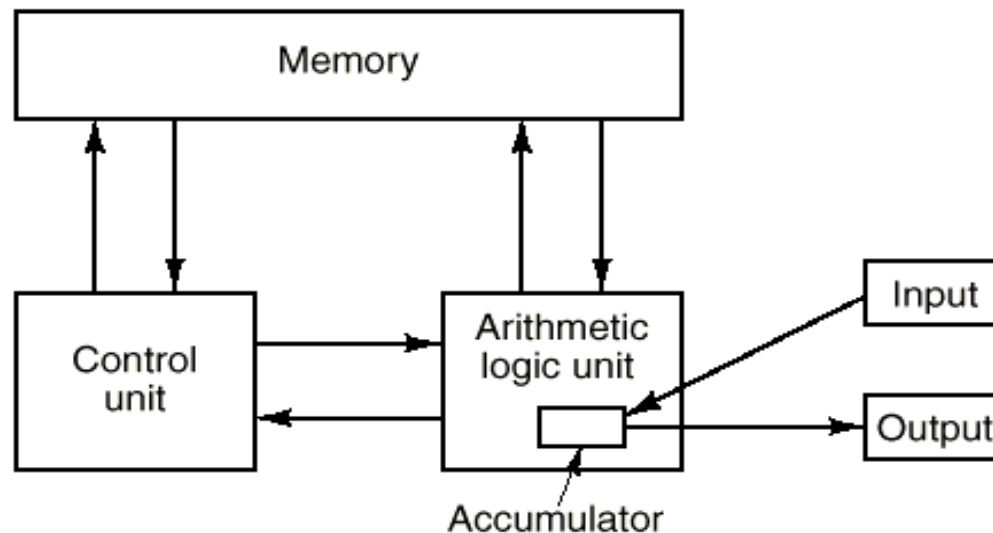






# Cara kerja prosesor

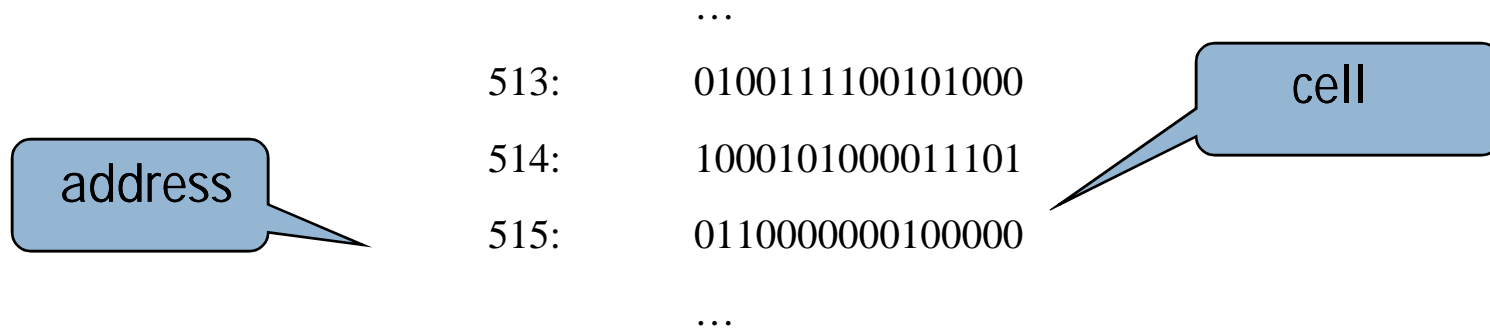
# The Von Neumann Computer



- Designed by John von Neumann, Princeton, in 1940s.
- Used in EDSAC, by M Wilkes, built in Cambridge in 1949.
- **Breakthrough:** digital (bits, 0 or 1) and stored program.
- Still prevailing today.

# Memory

- Has **cells (words)**
- Each cell has **address** (number) and contains **n bits**



- In von Neumann computer
  - 4096 word memory, 40 bit word
- Typical computer has 16 or 32 bit word and lots more memory!

# Von Neumann's Main Ideas

- Represent **data** as sequences of bits (0 or 1)  
integer 1 as 000000000001 (on 12 bits)

no reals!

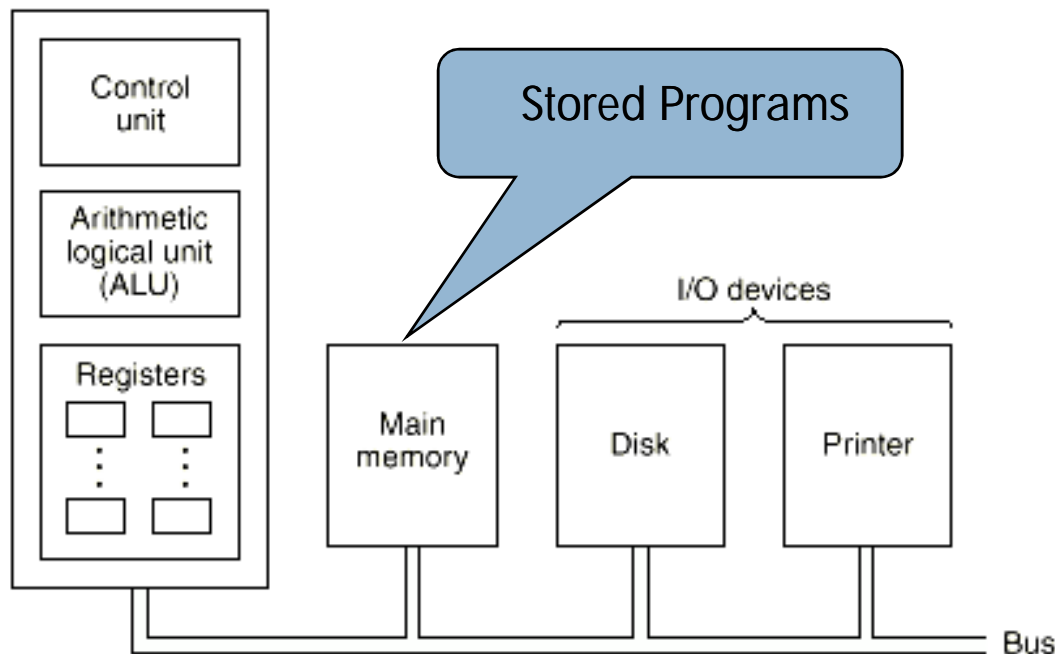
- Memory has 4096 cells, with 40 bits word
- Represent **instructions** as sequences of bits  
**code** (8 bits), **address** of argument (12 bits)

01100000000000000001

- Store both data and program instructions in memory
  - ▣ 2 instructions per word
- Instructions
  - ▣ use argument from memory, add to **accumulator**

# Simple Modern Computer

Central processing unit (CPU)



- Just two I/O devices.
- PC (Program Counter): address of next instruction.
- IR (Instruction Register): instruction being executed.

- Essentially von Neumann design.
- Difference: connected by a **bus** (parallel wires for data transfer).
- Several registers.

# How It Works



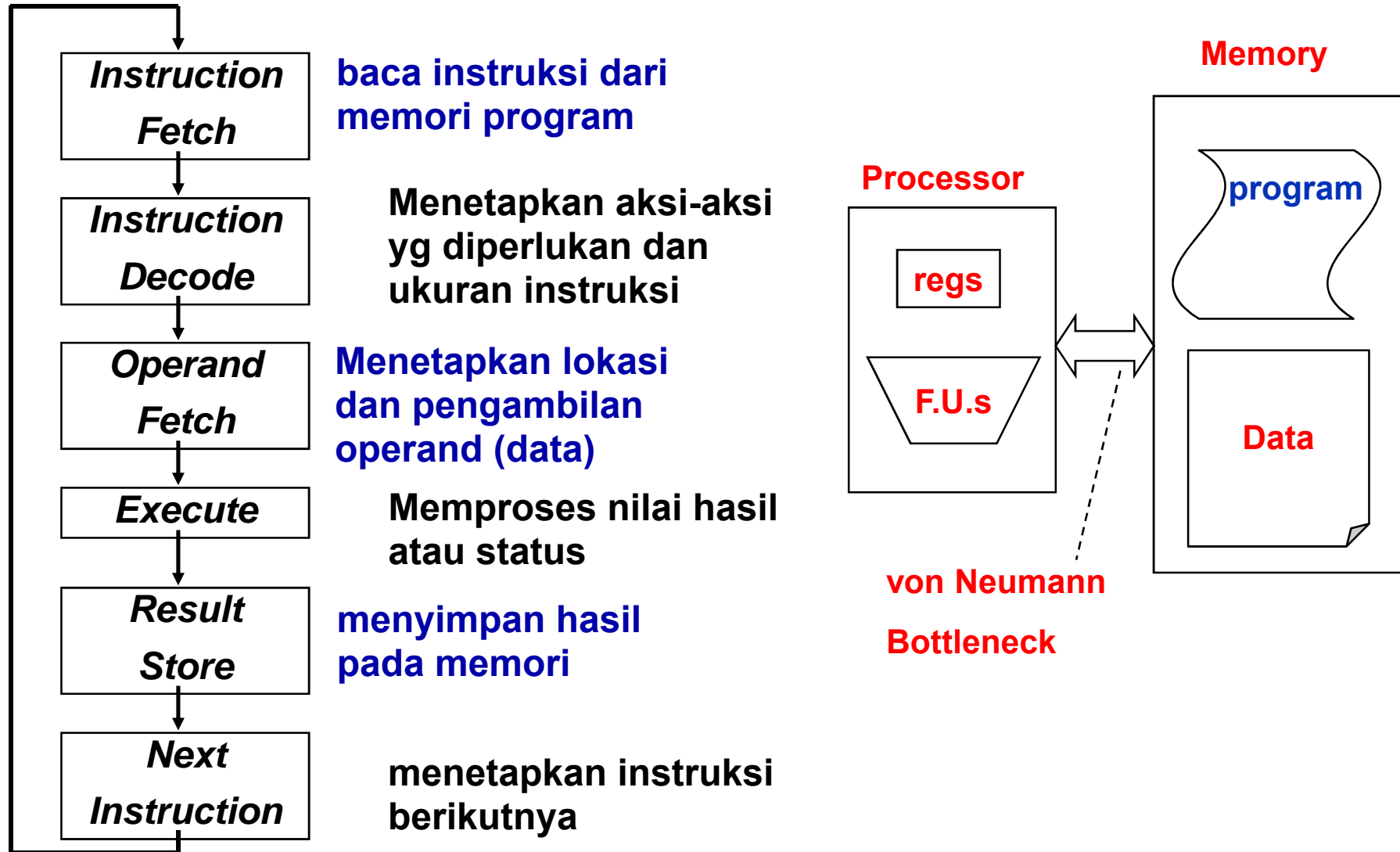
- **Main memory**: stores data and program
- **Bus**: transfer of data, addresses and control signals from/to memory and devices
- **CPU(Central Processing Unit)**:
  - ▣ **control unit**: fetches instructions and executes them
  - ▣ **ALU (Arithmetic Logic Unit)**: performs operations (add, subtract, etc)
  - ▣ **registers** (fast memory): store temporary results and control information (address of next instruction)
- **I/O devices (Input/Output)**: user communication

# How the Computer Executes





- **Fetch** instruction into **IR**
- **Increment PC** to point to **next** instruction
- Determine **type** of instruction
- If contains address, **fetch contents** into register
- **Execute** instruction
- Repeat from the above procedures

# Fundamental Execution Cycle





- 
- ▣ A **single data path** between the CPU and main memory.
    - This single path is known as the von Neumann bottleneck.



A short animation  
of simple processor routines

Instructions in Memory

3

2

1

Old non-  
pipelined  
processor

Fetch

Decode

Execute

Instructions in Memory

3

2

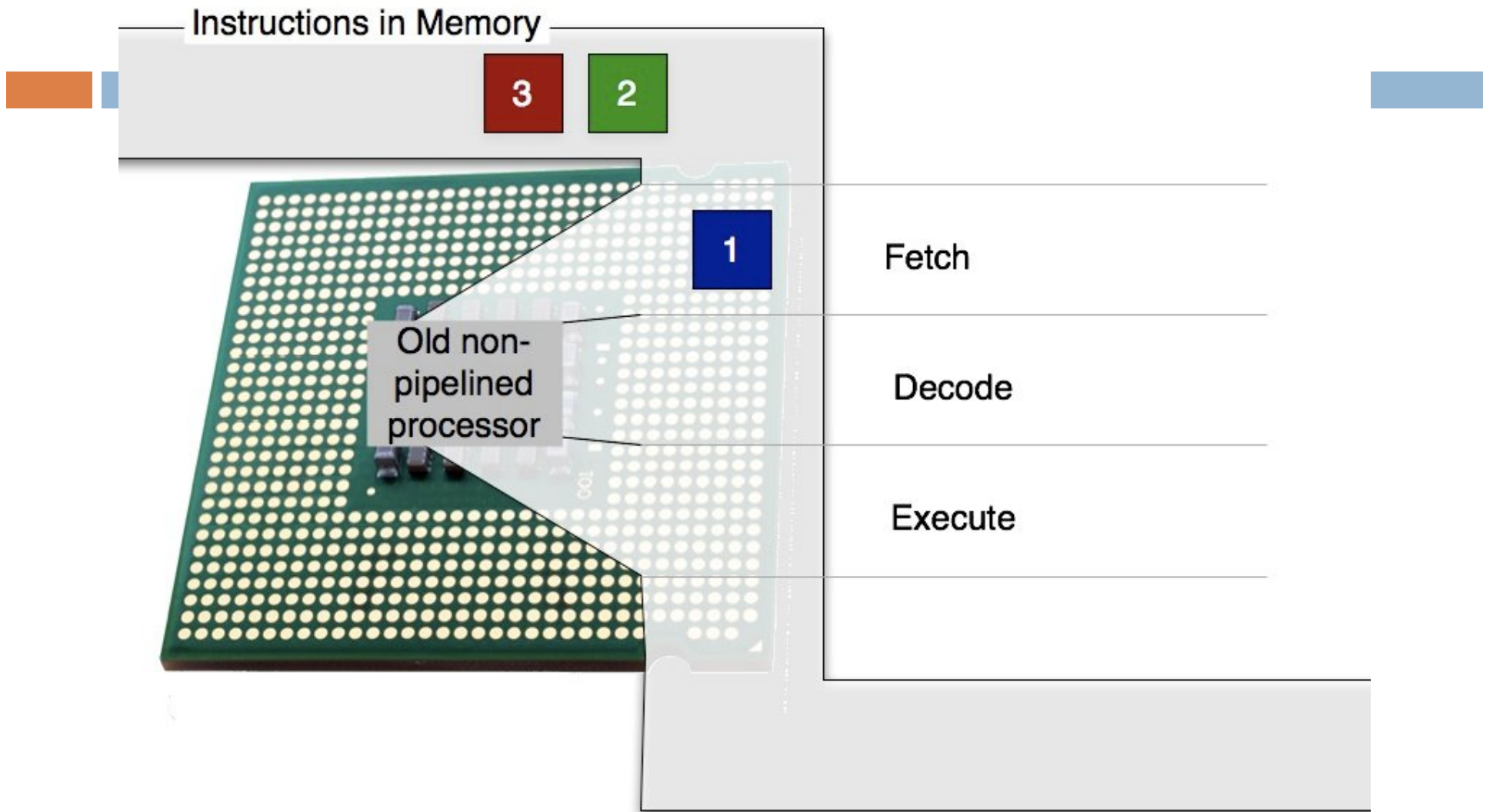
1

Old non-  
pipelined  
processor

Fetch

Decode

Execute



Instructions in Memory

3

2

Old non-  
pipelined  
processor

1

Fetch

Decode

Execute

Instructions in Memory

3

2

Old non-  
pipelined  
processor

1

Fetch

Decode

Execute



Instructions in Memory

3

2

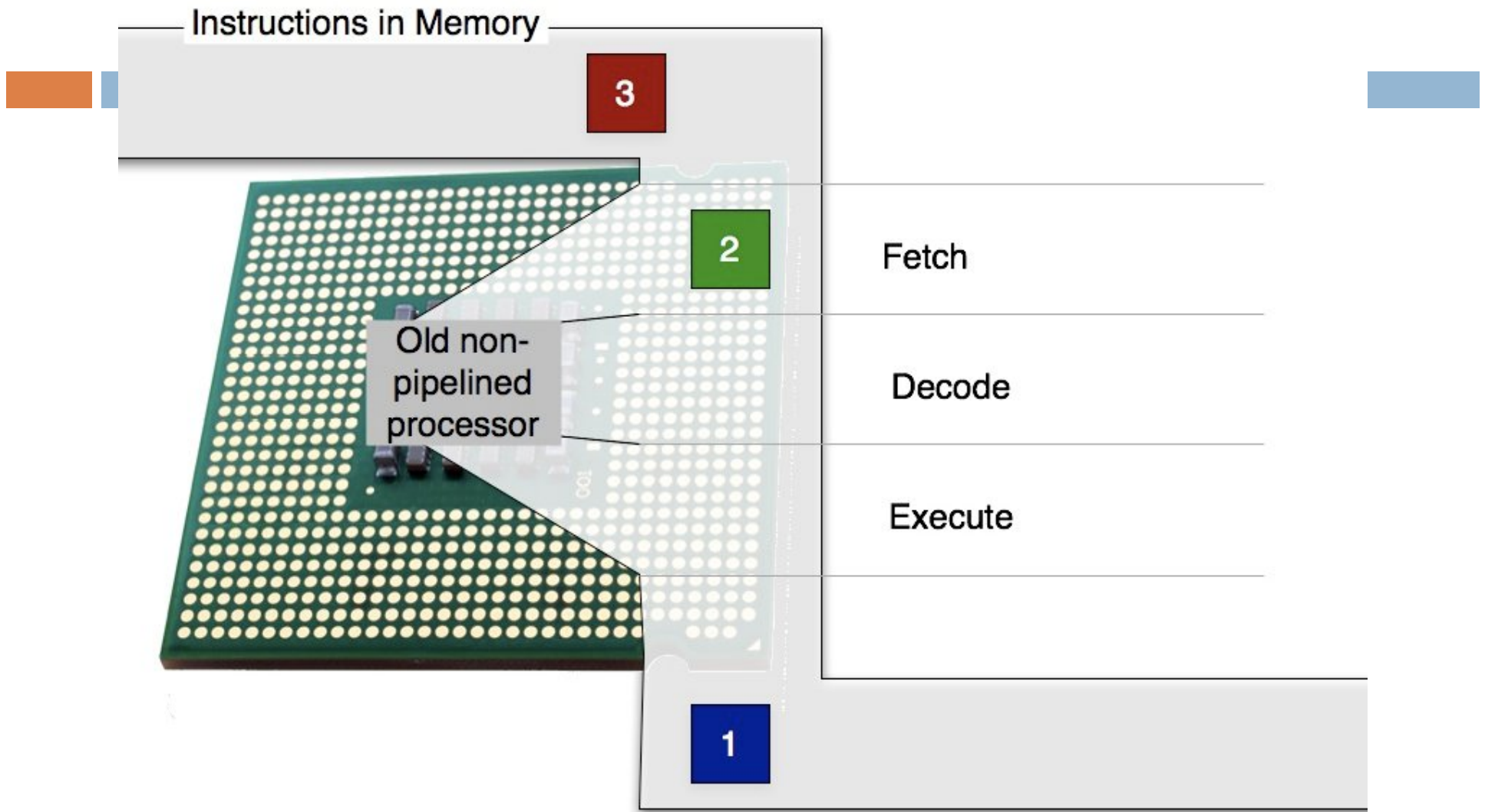
Old non-  
pipelined  
processor

Fetch

Decode

Execute

1



Instructions in Memory

3

Old non-  
pipelined  
processor

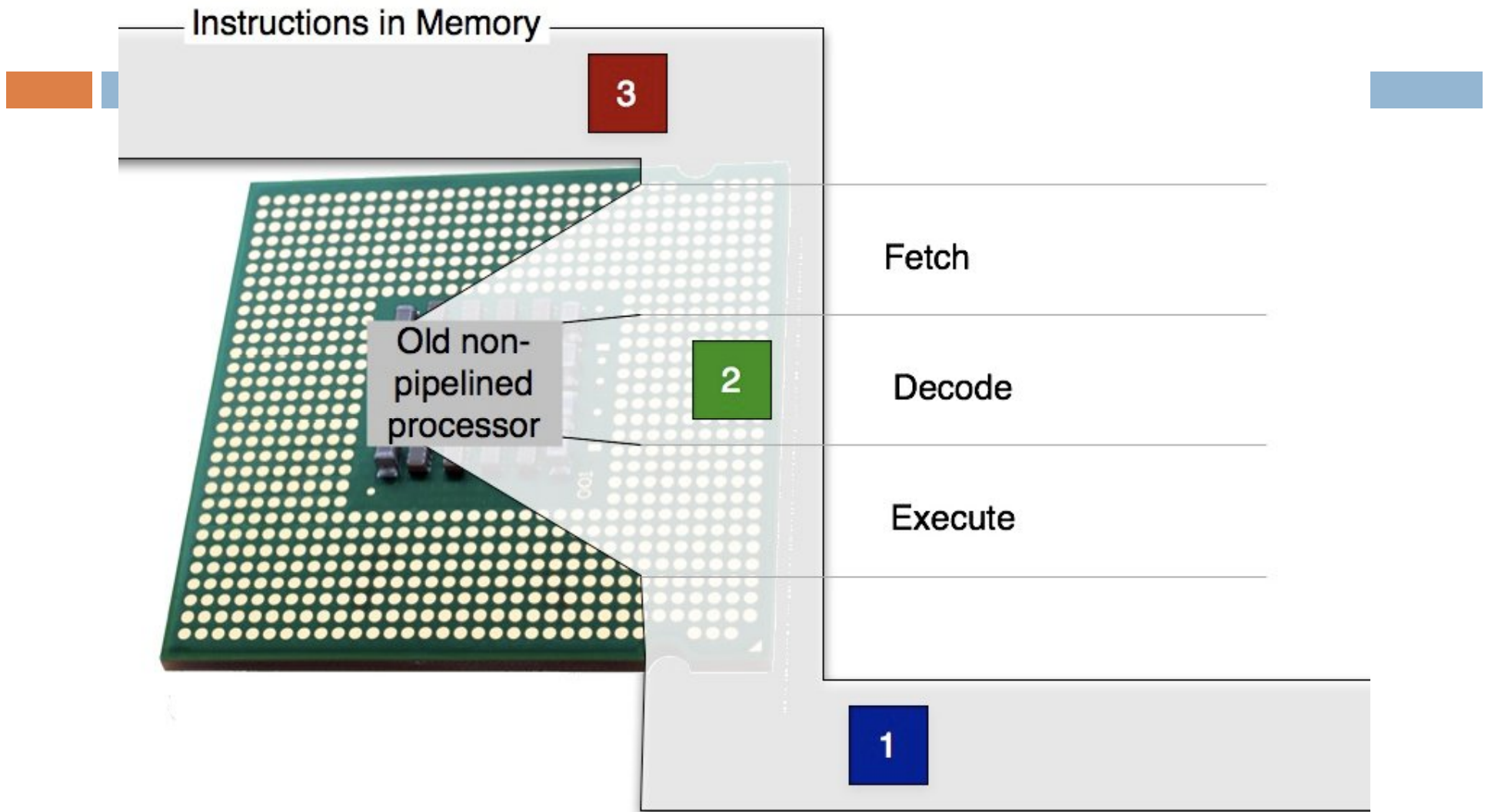
2

Fetch

Decode

Execute

1





Instructions in Memory

3

Old non-  
pipelined  
processor

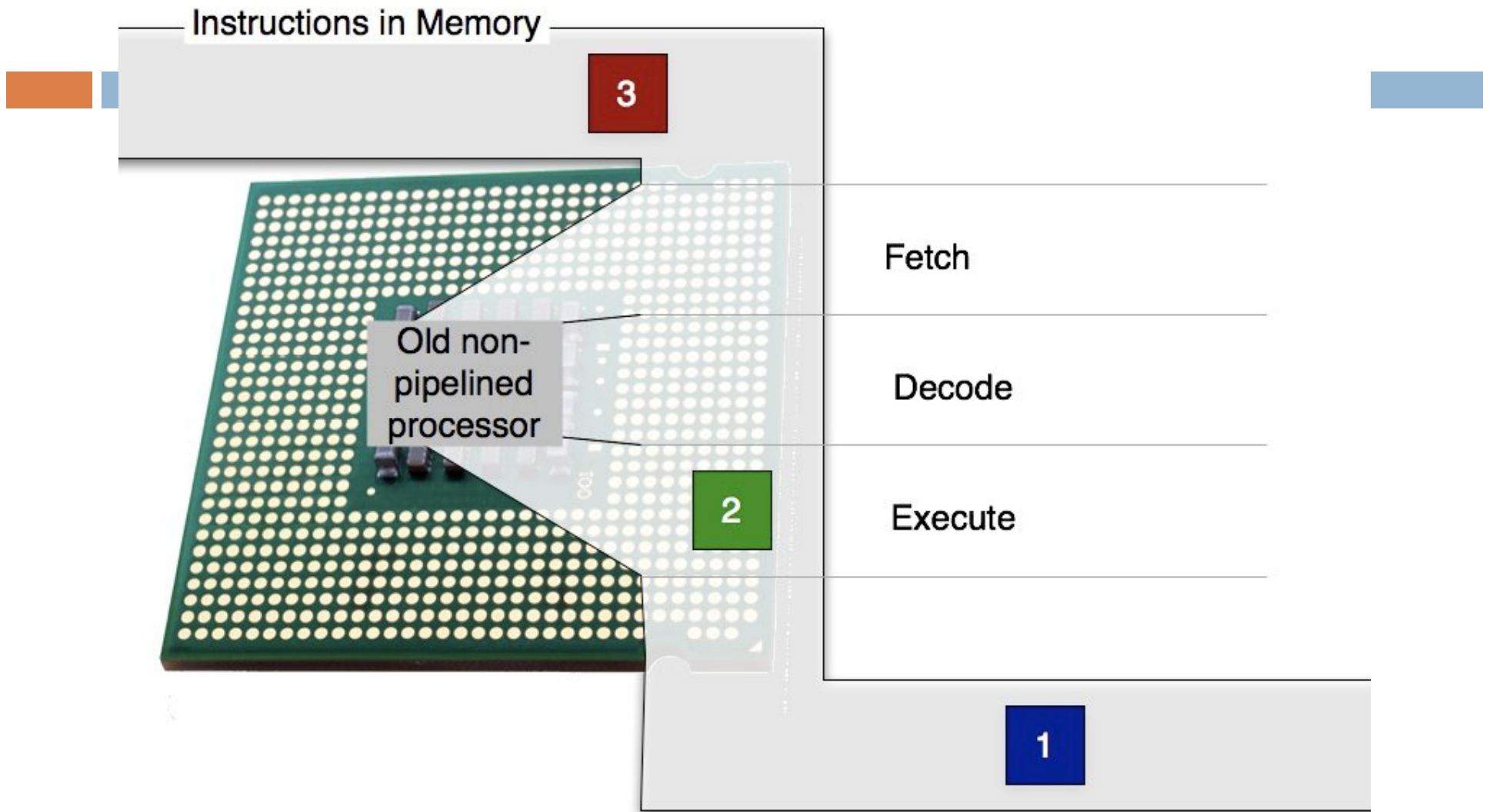
2

Fetch

Decode

Execute

1



Instructions in Memory

3

Fetch

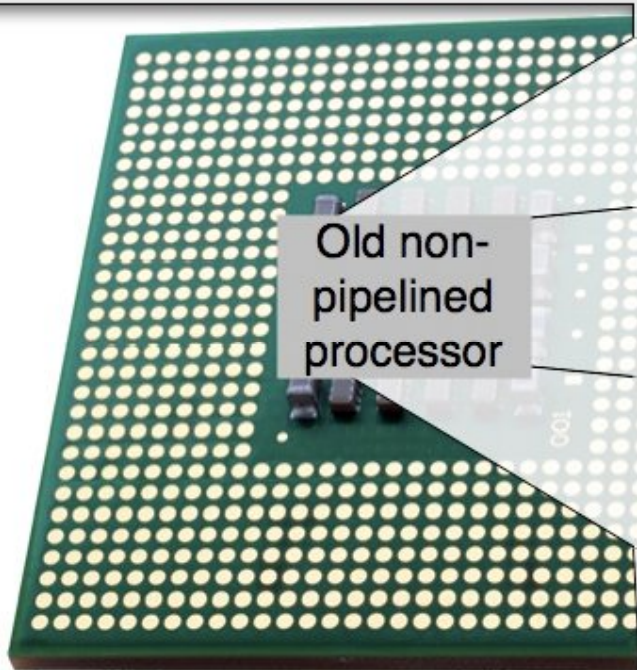
Old non-  
pipelined  
processor

Decode

Execute

2

1



Instructions in Memory

3

Fetch

Old non-  
pipelined  
processor

Decode

Execute

2

1

Instructions in Memory

Old non-pipelined processor

3

Fetch

Decode

Execute

2





Instructions in Memory

Old non-pipelined processor

Fetch

Decode

Execute

3

2

Instructions in Memory

Old non-  
pipelined  
processor

Fetch

Decode

Execute

3

2

Instructions in Memory

Old non-pipelined processor

Fetch

Decode

Execute

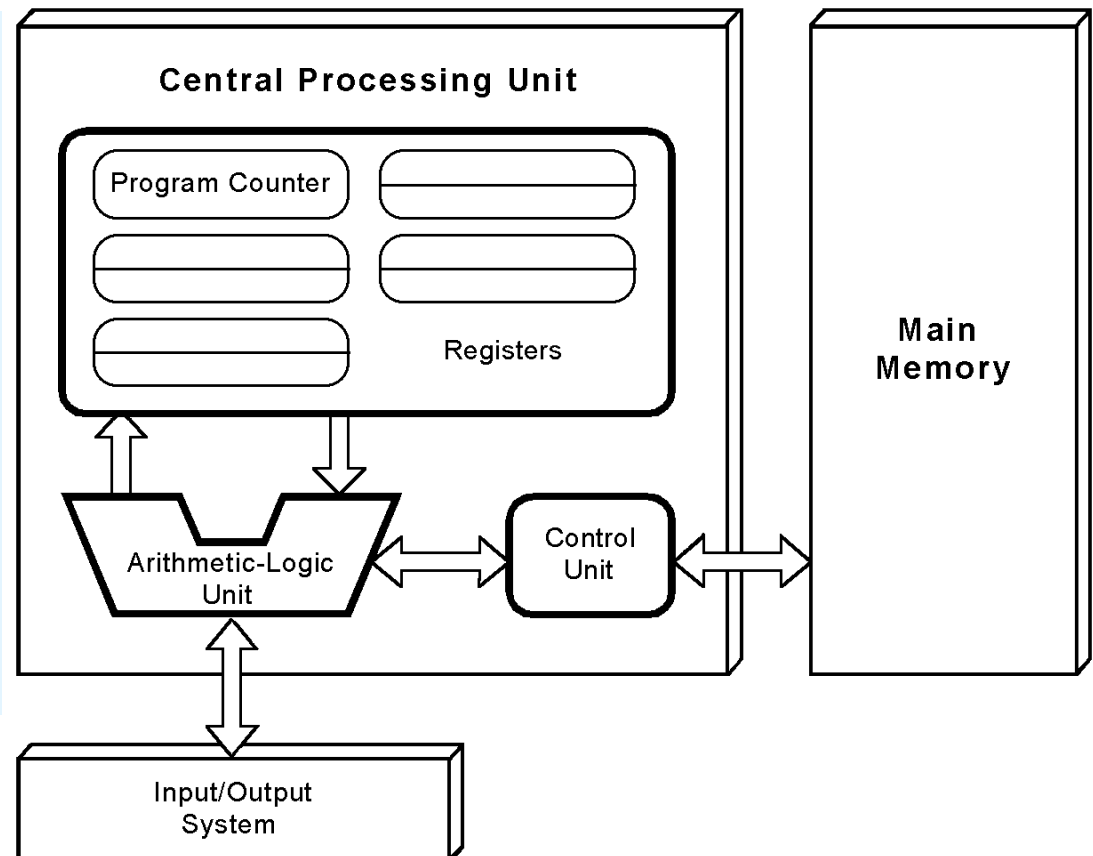
3

2

# The von Neumann Model

48

- This is a general depiction of a von Neumann system:
- These computers employ a fetch-decode-execute cycle to run programs as follows . . .

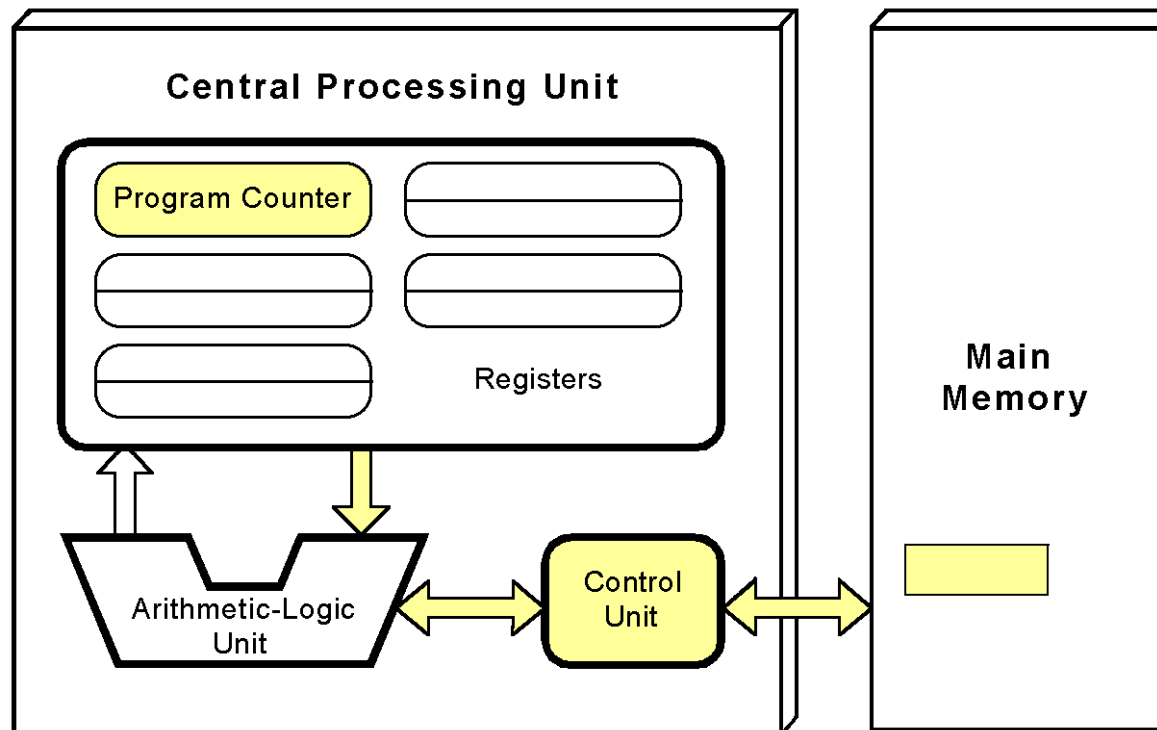




# The von Neumann Model

49

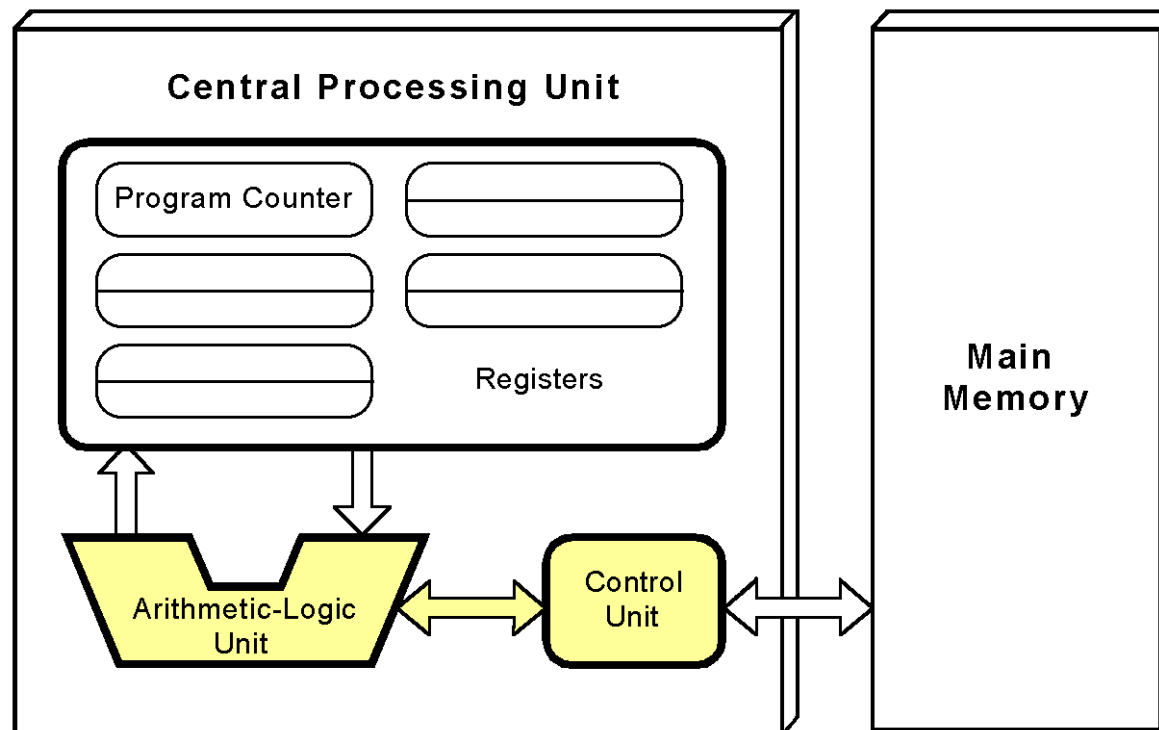
- The control unit fetches the next instruction from memory using the program counter to determine where the instruction is located.



# The von Neumann Model

50

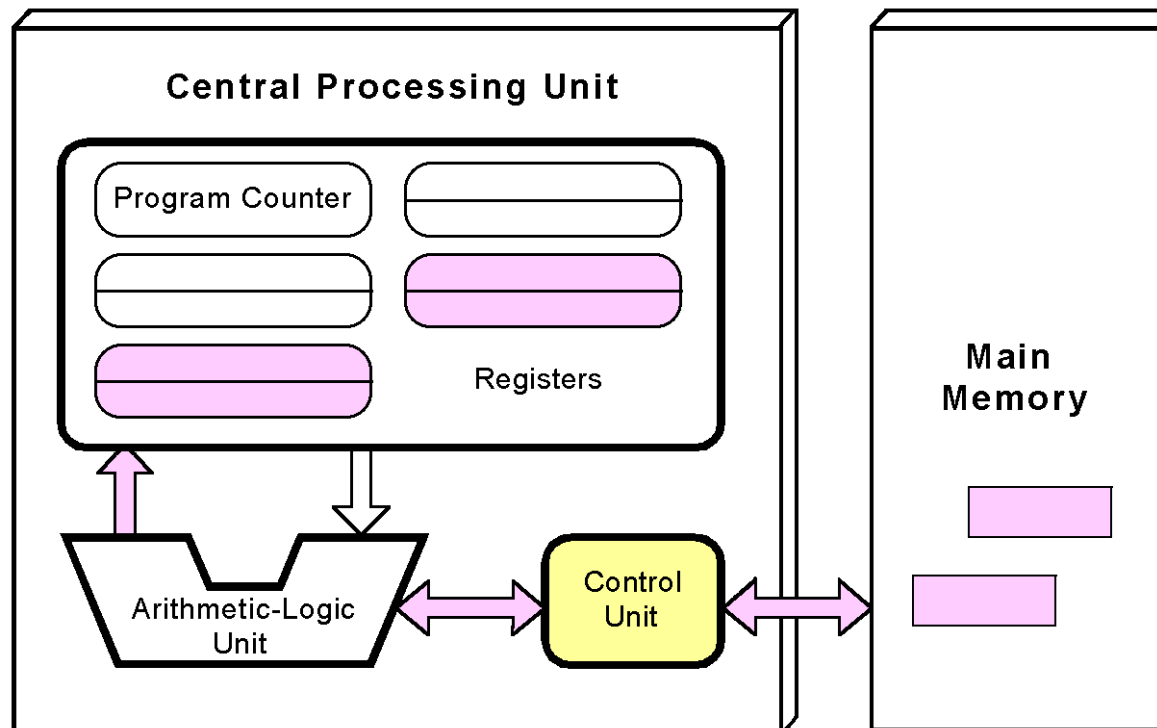
- The instruction is decoded into a language that the ALU can understand.



# The von Neumann Model

51

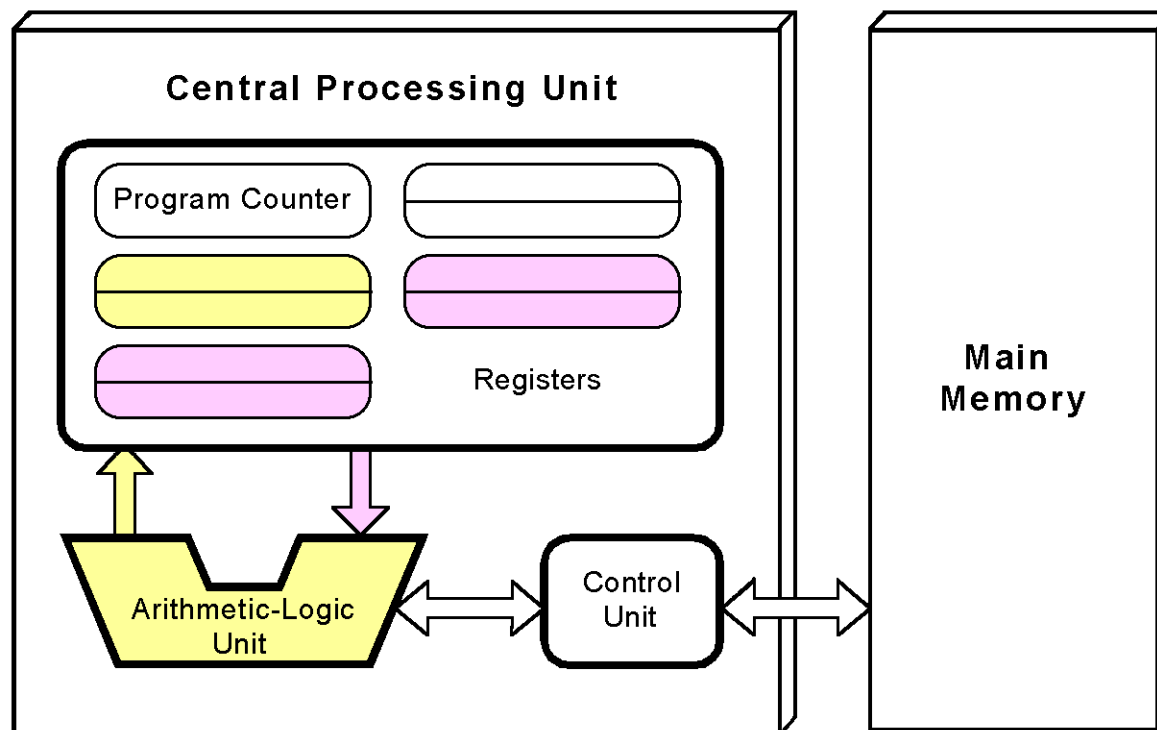
- Any data operands required to execute the instruction are fetched from memory and placed into registers within the CPU.



# The von Neumann Model

52

- The ALU executes the instruction and places results in registers or memory.



# Machine Language Instructions

53

- Can be decoded and executed by control unit
- Parts of instructions
  - ▣ Operation code (op code)
    - Unique unsigned-integer code assigned to each machine language operation
  - ▣ Address field(s)
    - Memory addresses of the values on which operation will work



Figure 5.14  
Typical Machine Language Instruction Format