Lecture-3

Chapter-2

Computer Architecture and Organization-
Jhon P. Hayes

Design Methodology

Design Methodology

System Design:

A computer is an example of a system, which is defined informally as a collection – often a large and complex one – of objects called components, that are connected to form a consistent entity with a specific function or purpose. The function of the system is determined by the functions of its components and how the components are connected.

Hardware Description Language (HDL)

A system's structure and behavior can be described by means of a block diagram. We can convey the same detailed information by means of a *hardware description language* (HDL).

Hardware description language (HDL) is a specialized programming language used to design and describe digital circuits and systems in computer architecture. It allows engineers to specify how a computer's hardware components (like processors, memory units, and control logic) should behave and be structured.

Example Usage: Imagine you want to design a 4-bit adder, a fundamental part of an ALU. Instead of physically connecting logic gates.

Design Process

Design Process: Given a system's structure, the task of determining its function or behavior is termed analysis. The converse problem of determining a system structure that exhibits a given behavior is design or synthesis.

The complexity of computer systems is such that the design problem must be broken down into smaller, easier task involving various classes of components. Each smaller part can be solve independently by different designers and each major design step can be implemented by iterative process as shown in Figure 4

Design Methodology

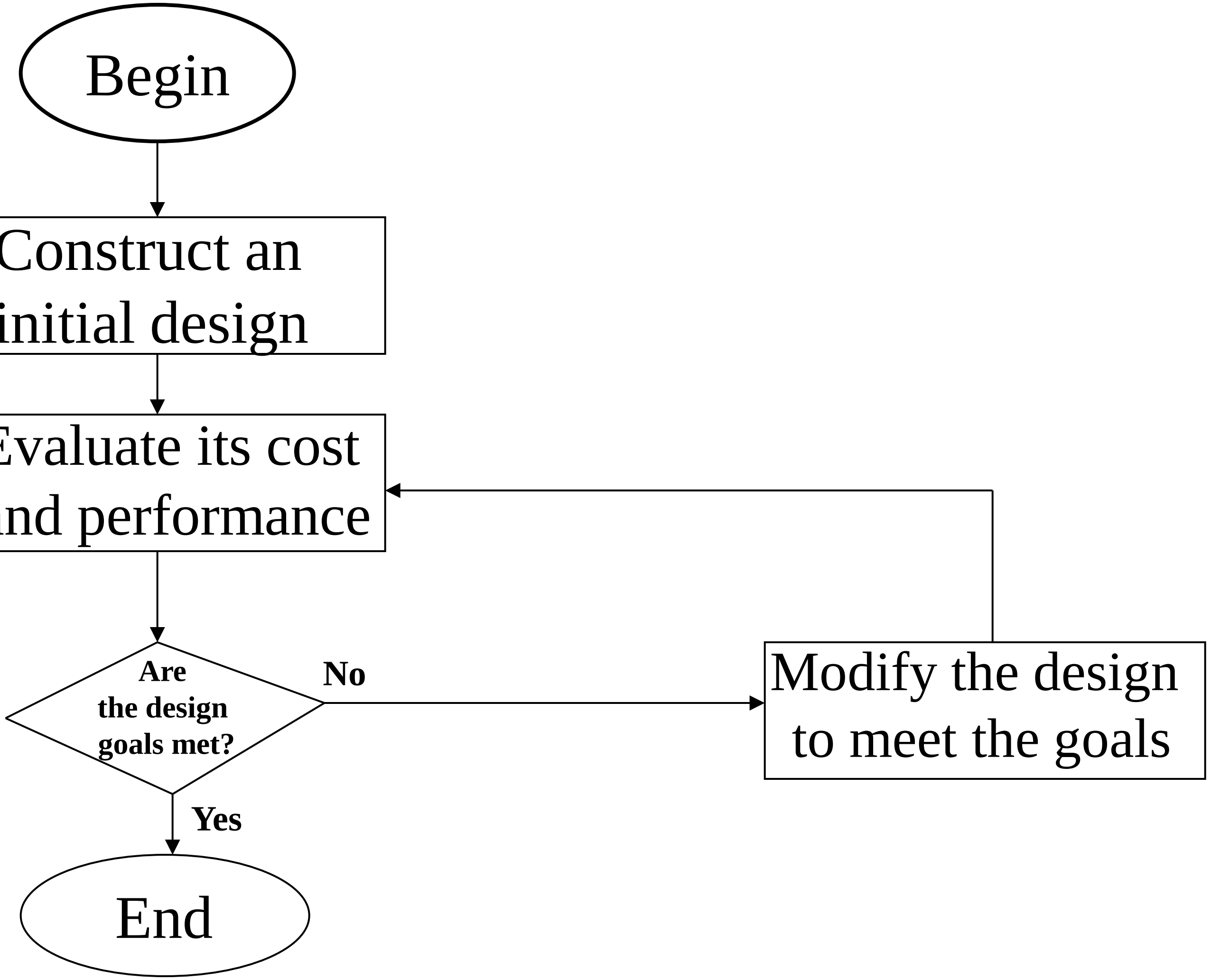


Figure 4 Flowchart of an iterative design process

Design Levels

Design levels:

1. The *processor level*, also called the architecture, behavior, or system level
2. The *register level*, also called the register-transfer level (RTL).
3. The *gate level*, also called the logic level

Level	Components	IC density	Information Units	Time units
Gate	Logic gate, Flip-Flop	SSI	Bits	10^{-12} to 10^{-9}s
Register	Registers, counters, Combinational circuits, small sequential circuit	MSI	Words	10^{-9} to 10^{-6}s
Processor	CPUs, memories, IO devices	VLSI	Blocks of Words	10^{-3} to 10^{3}s

The Gate Level

Gate level (logic):

- Design is concerned with processing binary variables whose possible values are restricted to be the bits (binary digits) 0 and 1. The design components are logic gates, which are si mple, memoryless processing elements and flip-flops which are bit-storage devices.

Combinational logic:

- Combinational logic is a type of digital logic circuit that performs a specific function based on the combination of its inputs. It does not store or remember past inputs or state, and its outputs depend only on the current inputs.
- A combinational function also referred to as a logic or a Boolean function, is to mapping from the set of 2^n input combinations of n binary variables onto the output values 0 and 1.

Common Logic Gates

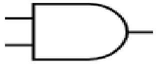
YES

INPUT	OUTPUT
A	
0	0
1	1

NOT

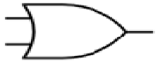
INPUT	OUTPUT
A	
0	1
1	0

AND



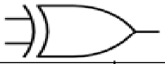
INPUT		OUTPUT
A	B	
0	0	0
1	0	0
0	1	0
1	1	1

OR



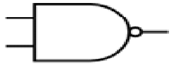
INPUT		OUTPUT
A	B	
0	0	0
1	0	1
0	1	1
1	1	1

XOR



INPUT		OUTPUT
A	B	
0	0	0
1	0	1
0	1	1
1	1	0

NAND



INPUT		OUTPUT
A	B	
0	0	1
1	0	1
0	1	1
1	1	0

NOR

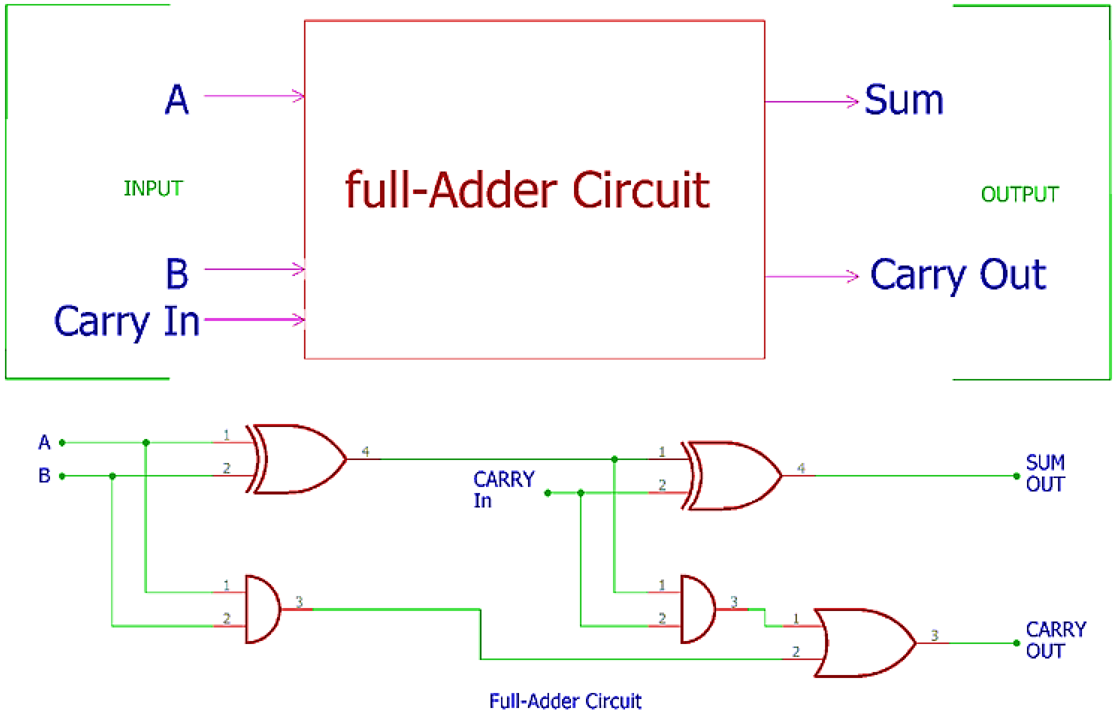
INPUT		OUTPUT
A	B	
0	0	1
1	0	0
0	1	0
1	1	0

XNOR

INPUT		OUTPUT
A	B	
0	0	1
1	0	0
0	1	0
1	1	1

Fig 5: Common Boolean logic gates with symbols and truth tables

Full Adder



Inputs			Outputs	
A	B	C_{in}	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Figure 6: Full Adder with circuit and truth table

Flip-flops

- A flip-flop is a digital logic circuit element that is used to store binary data.
- It is a type of sequential logic circuit, which means that its output depends not only on the present input, but also on its previous state.
- A flip-flop typically has two inputs and two outputs, with the inputs labeled as "clock" and "data". The clock input controls when the flip-flop will change its state, while the data input determines the new state that the flip-flop will switch to.
- The two outputs are the complementary states of the flip-flop.
- They are also designed to be unaffected by transient signal changes (noise) produced by t he combinational logic that feeds them.

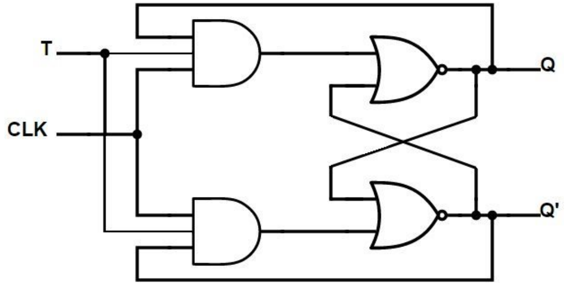
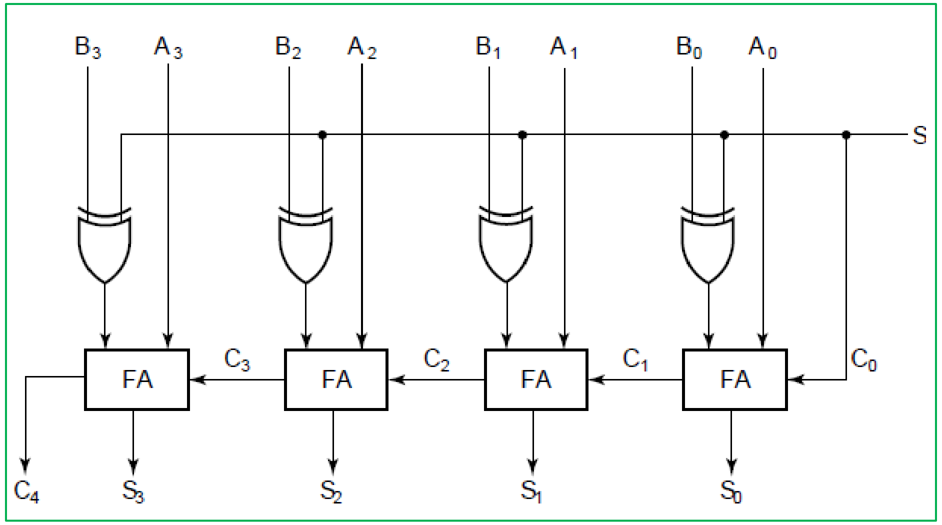


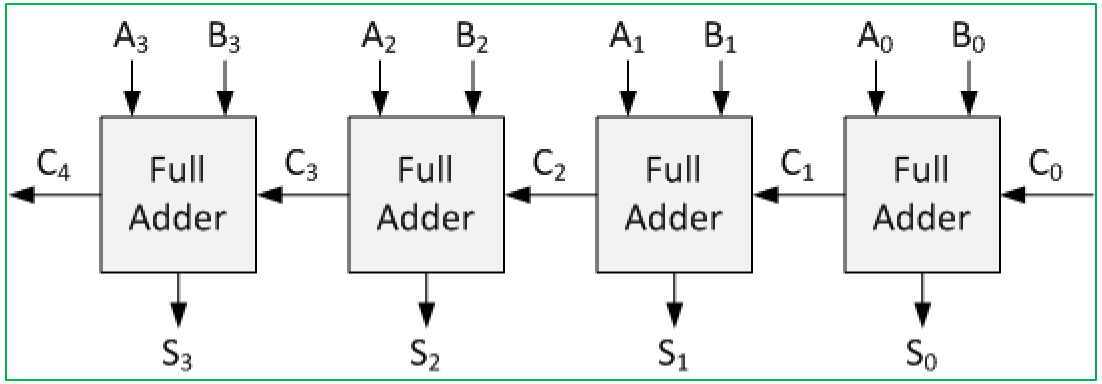
Fig 7: Flip-flop

Ripple Carry Adder

Four bit ripple carry: a) Circuit Diagram; b) Schematic Diagram.

	A_3	A_2	A_1	A_0	
	1	0	1	0	= 10
	B_3	B_2	B_1	B_0	
	1	0	1	1	= 11
1	0	1	0	1	= 21





Use to add two n-bits binary numbers

Inputs			Outputs	
A	B	C_{in}	Sum	Carry
0	0	0	0	0
0	0	1	1	0
0	1	0	1	0
0	1	1	0	1
1	0	0	1	0
1	0	1	0	1
1	1	0	0	1
1	1	1	1	1

Sequential Circuits Flip-Flops

- By adding memory to a combinational circuit in the form of 1-bit storag e elements called flip-flops, we obtain a sequential circuit.
- A sequential circuit consists of a combinational circuit and a set of flip-flops.
- The combinational logic forms the computational or data processing part of the circuit.
- The flip-flops store information of the circuit's past behavior; this store d information defines the circuits internal state. (output depends on pre sent input and past input)

The Register level

At the register or register transfer level, related information bits are grouped into ordered sets called words or vectors. The primitive (Modern) components are small combinational or sequential circuits intended to process or store words.

Type	Component	Functions
Combinational	Word Gates	Logical (Boolean) Operation
	Multiplexers	Data routing; general, combinational function
	Decoder, encoder	Code checking and conversion
	Adders	Addition and subtraction
	Arithmetic Logic Unit	Numerical and logical operations
	Programmable logic devices	General sequential functions.
Sequential	(Parallel) registers, Shift registers	Information storage; serial-parallel conversion
	Counter, Programmable logic devices	Control/timing signal generation, General sequential functions.

Fig 8: Register level Components

The Processor level

There are four main groups of processor level components:

- Processors
- Memories
- IO devices
- Interconnection networks

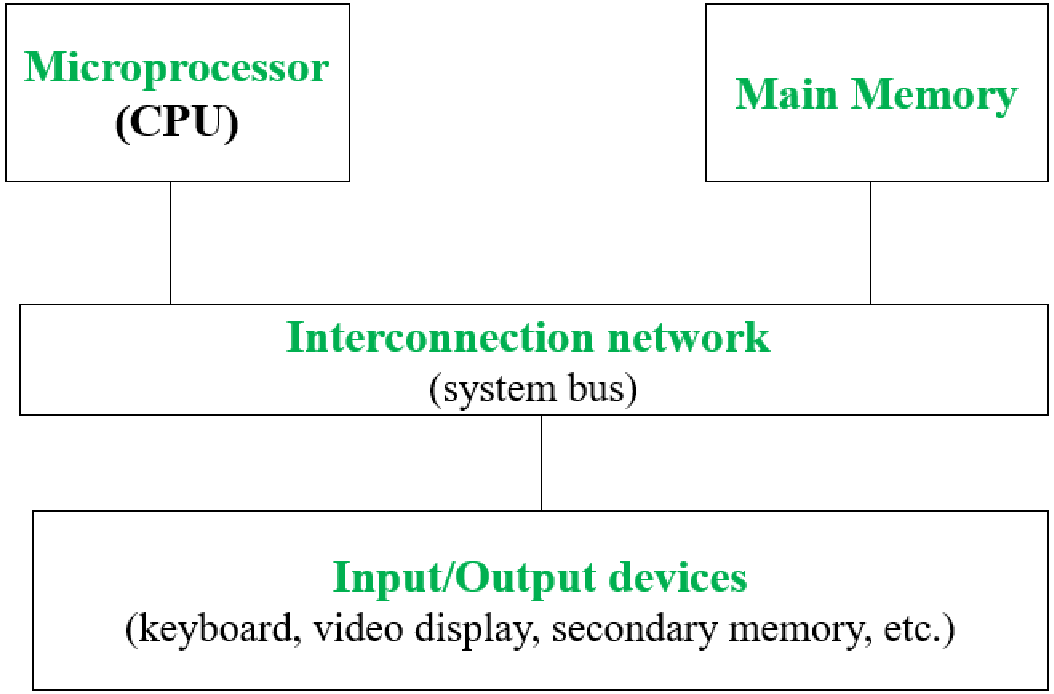


Figure: Major components of a computer system

The Processor level

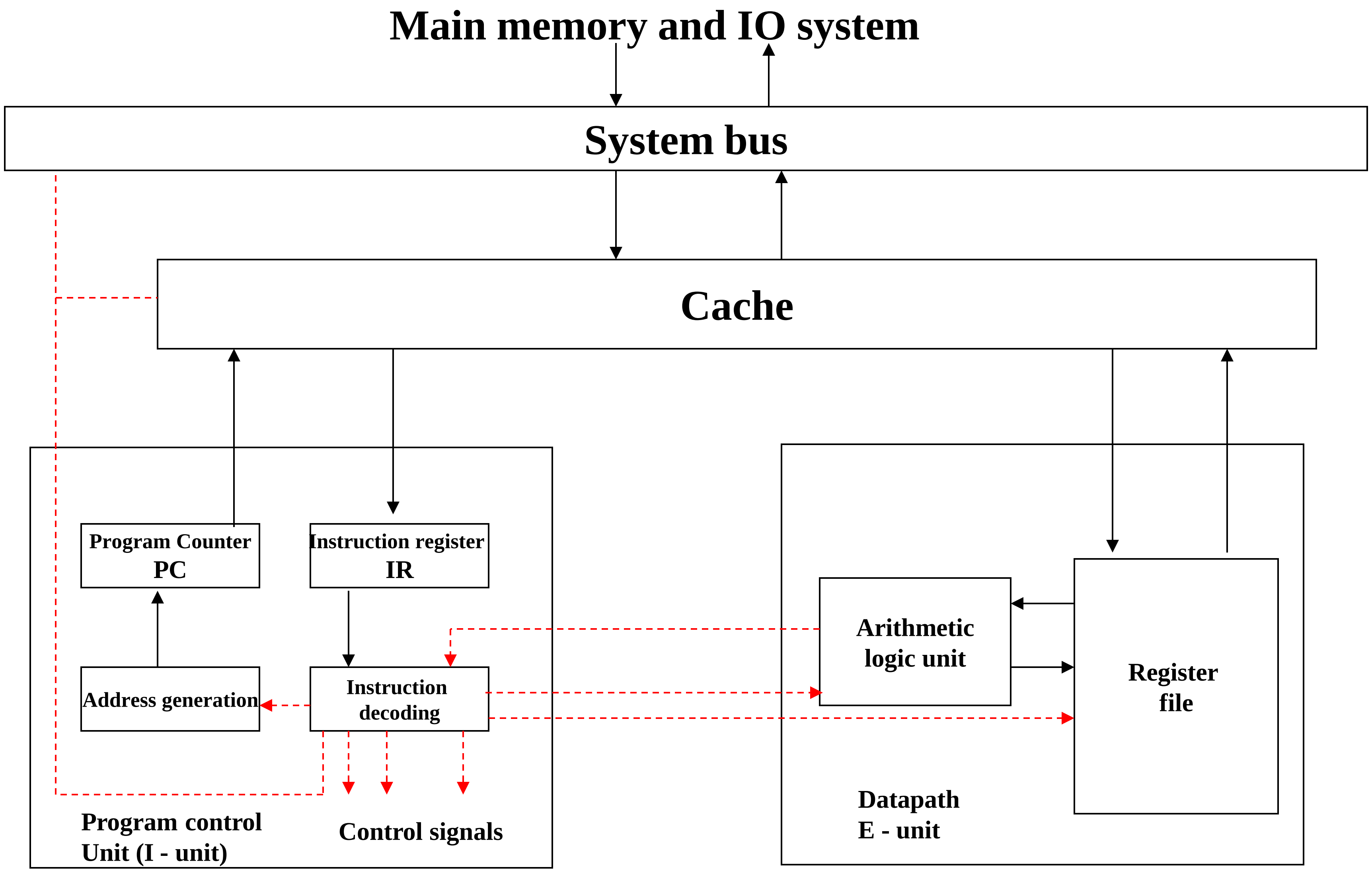
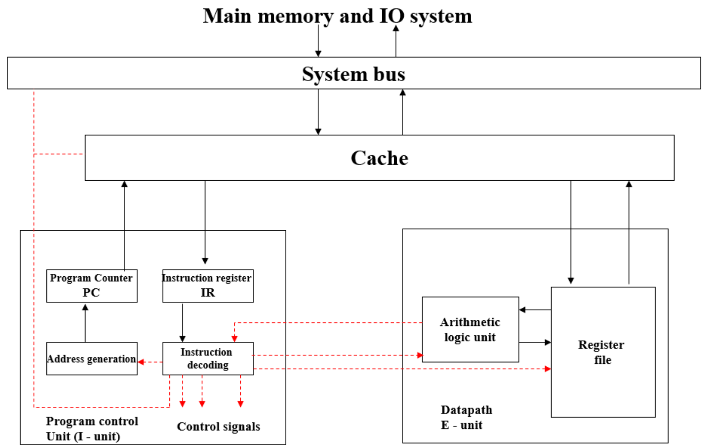


Figure 10: Internal organization of a CPU and cache memory

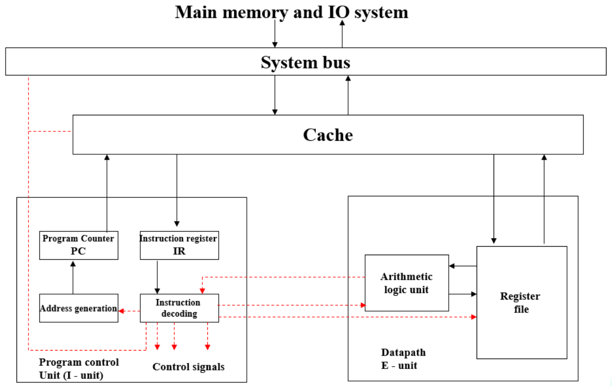
The Processor level

- **Figure 10** shows the essential internal organization of a CPU at the register level. The CPU contains the logic needed to execute its particular instruction set and is divided into datapath and control unit.
- The control part (***I - unit***) generates the address of instructions and data stored in external memory. In this particular system a cache memory is interposed between the main memory M and the CPU.
- Each memory request generated by the CPU is first directed to the cache.



The Processor level

- If the required information is not currently assigned to the cache, the request is redirected to M and cache is automatically updated from M.
- The **I – unit** fetches instructions from the cache or M and decodes them to derive the control signals needed for their execution.
- The CPU’s datapath (***E - unit***) has the arithmetic – logic circuits that execute most instructions; it has also a set of registers for temporary data storage.
- The CPU manages a ***system bus***, which is the main communication link among the CPU cache subsystem, main memory and the I/O devices.

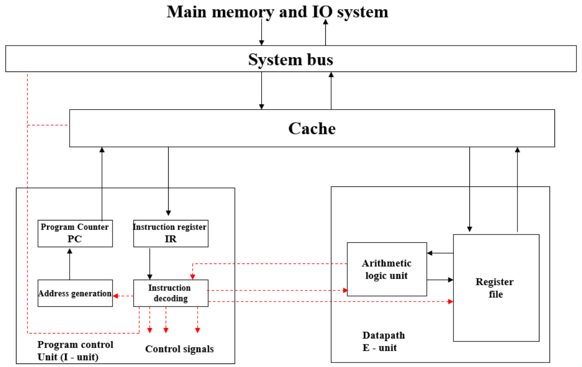


The Processor level

- In one clock cycle the CPU can perform a register-transfer operation, such as fetching an instruction word from M via the system bus and loading it into the instruction register IR. This operation can be expressed formally by:

  IR:= M(PC);

  Where PC is the program counter used by the CPU to hold the expected address of the next instruction word. Once in the I-unit, an instruction is decoded to determine the actions needed for its execution.
- The entire process of fetching, decoding and executing an instruction constitutes the CPU's ***instruction cycle***.



The Processor level

Memories: The memory part of a computer can be divided into several major subsystem.

- *Main memory M,* consisting of relatively fast storage ICs connected directly to, and controlled by, the CPU.

- *Secondary memory,* Consist of less expensive devices that have very high storage capacity. These devices often involve mechanical motion and so are much slower than M. They are generally connected indirectly (via M) to the CPU and form part of the computer's I/O system.

- *Other memory,* many computer have a third type of memory called a cache, which is positioned between the CPU and main memory. Some or all of cache may be integrated on the same IC chip as the CPU itself.

The Processor Level (Cache memory)

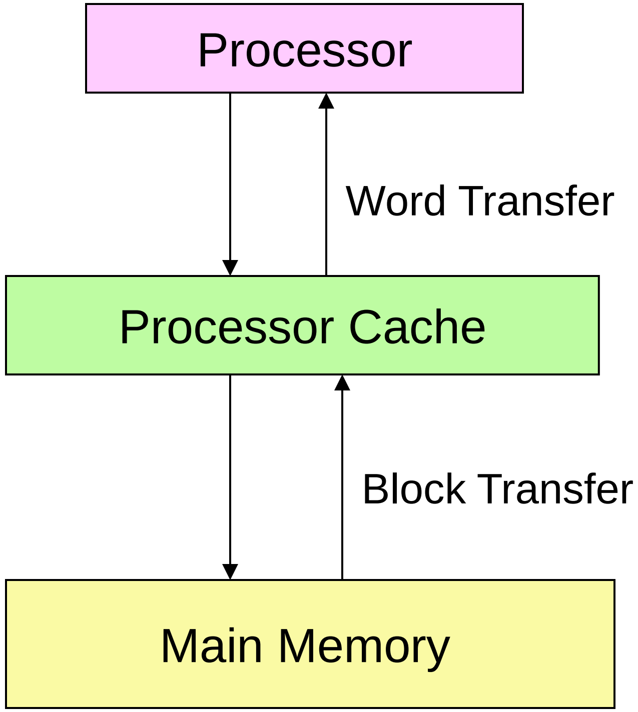


Fig 11: Processor Cache Memory

Cache memory is a small amount of fast but expensive memory placed between the processor and main memory (512 KByte on the P-III processor). In some PCs there may be extra cache attached directly to the motherboard. When the processor attempts to read a word of main memory, a check is made first to determine if the word is in the cache. If it is, a copy of the word is transferred to the processor. If not, a block of main memory, consisting of a fixed number of words, is transferred into the cache and then a copy of the required word is transferred to the processor.

There are two types of cache memory:

- External cache memory,
- Internal cache memory

The Processor level

Interconnection networks: Processor level components communicate by word-oriented buses. In systems with many components, communication may be controlled by a subsystem called an *interconnection network*; terms such as *switching network*, *communications controller*, and *bus controller* are also used in this context.

Internal Communication:

The System BUS: A bus is a set of parallel wires connecting two or more component s of the computer.

The number of lines is referred to as the width of the bus. Bus lines are classified into three functional groups:

–Control Bus

–Data Bus

–Address Bus

The Processor Level (System Bus)

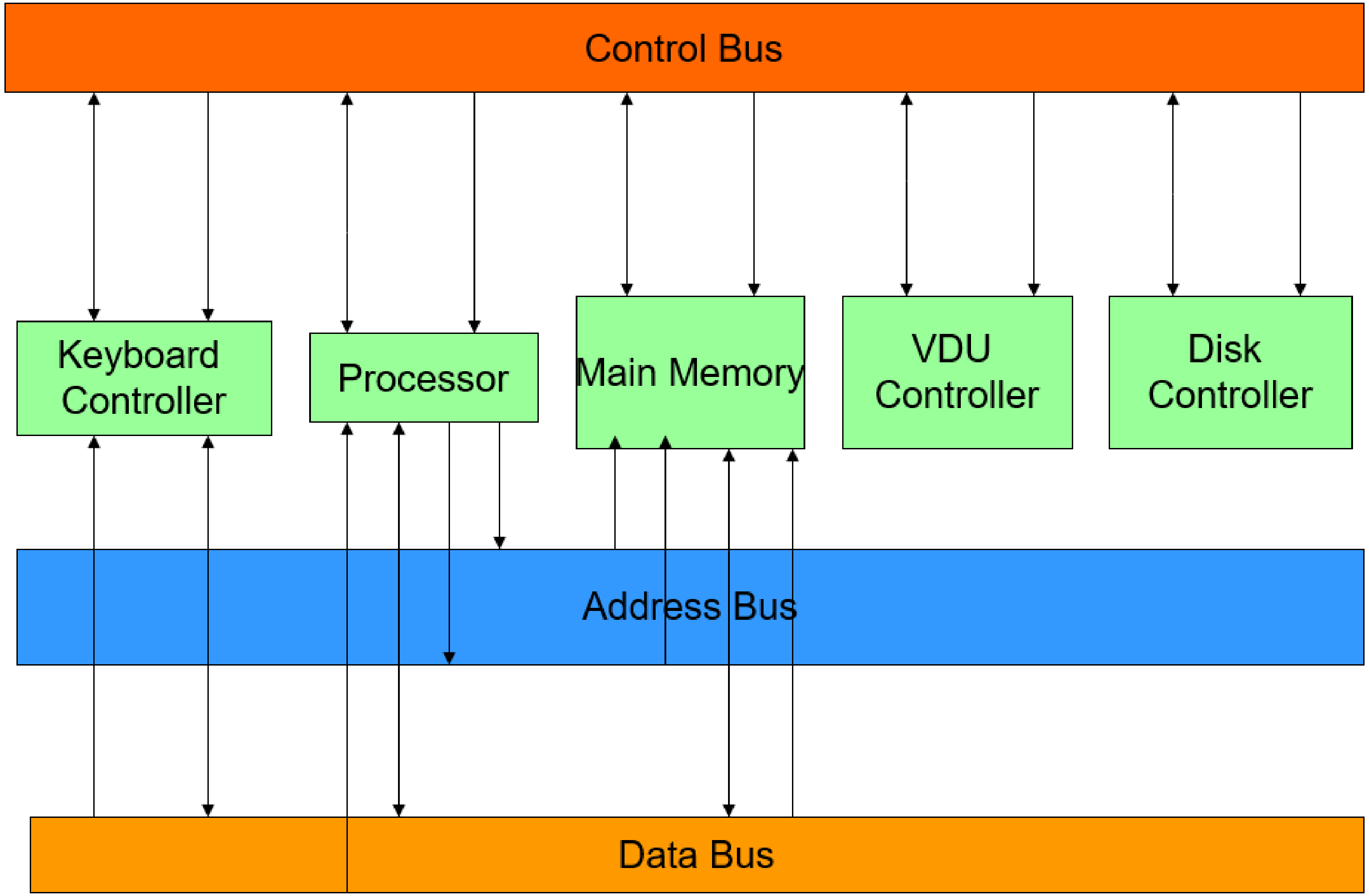


Fig 12: System Bus

That’s All

Thank You