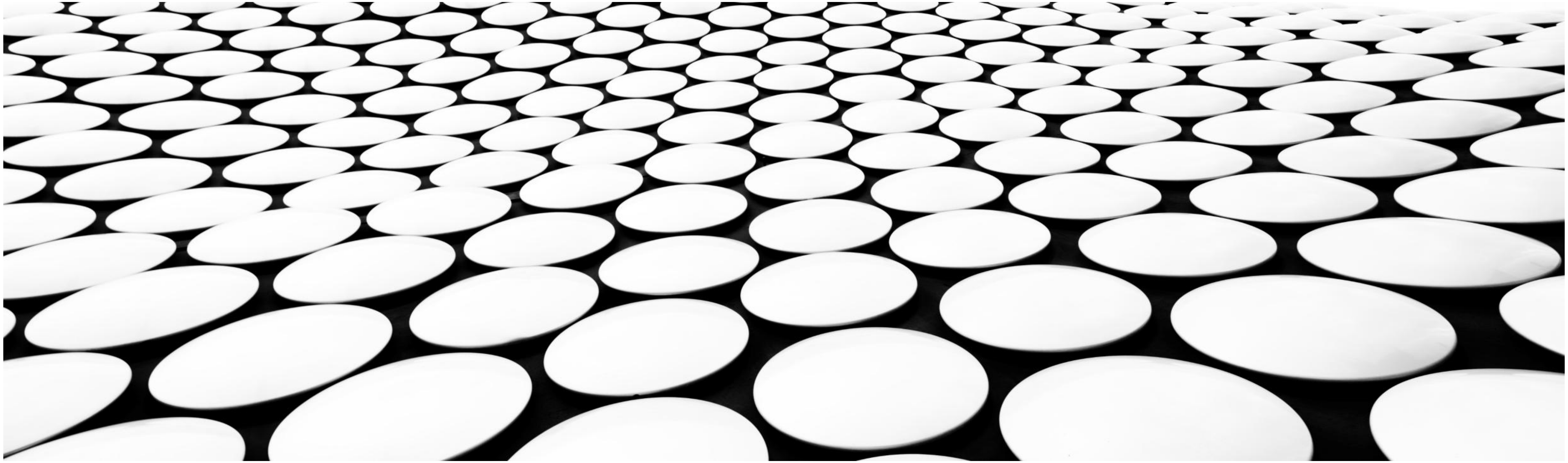
CS5102: FOUNDATIONS OF COMPUTER SYSTEMS

TOPIC 4: **COMPUTERS, HISTORY, AND INSTRUCTION SET**

DR. ARIJIT ROY

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INDIAN INSTITUTE OF TECHNOLOGY PATNA



These slides are based on the book: David Money Harris and Sarah L. Harris, Computer Organization and Design

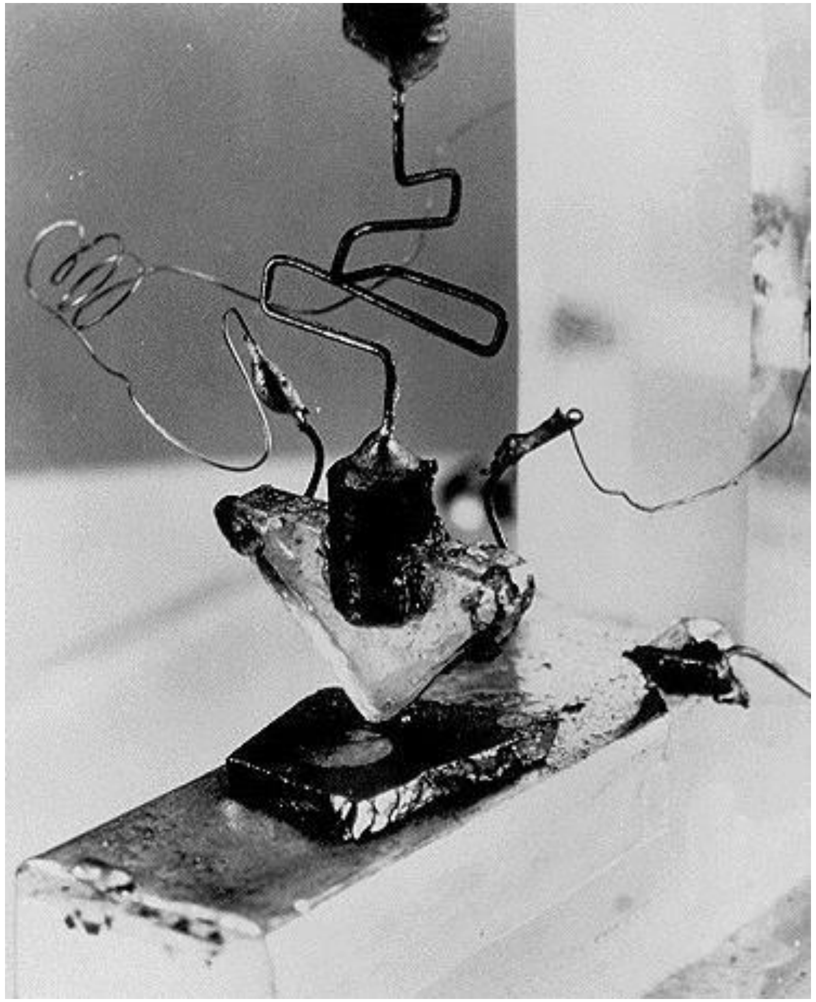
GENERATIONS OF COMPUTER

1. First Generation – 1940 -1956: Vacuum Tubes ENIAC
2. Second Generation – 1956 -1963: Transistors
3. Third Generation – 1964 -1971: Integrated Circuits
4. Fourth Generation – 1971-Present: Microprocessors
5. Fifth Generation - Present and Beyond: Artificial Intelligence

FIRST GENERATION 1940 - 1956

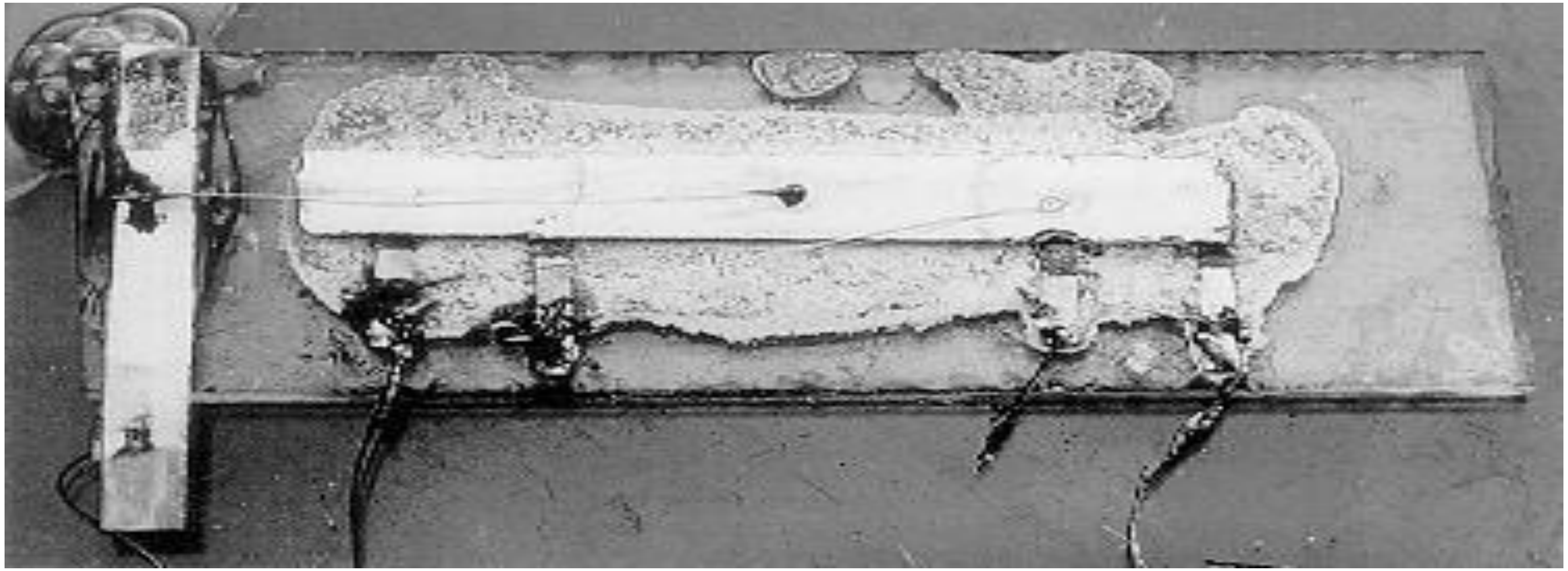
- First generation computers used Vacuum Tubes
- Vacuum tubes are glass tubes with circuits inside
- The word vacuum indicates that they have no air inside, which protects the circuitry
- Building a computer with these vacuum tubes would result in a very large machine occupying one full room

FIRST TRANSISTOR



- Uses Silicon
- developed in 1948
- won a ~~Nobel prize~~ John Bardeen
- on-off switch
- Second Generation Computers used Transistors, starting in 1956

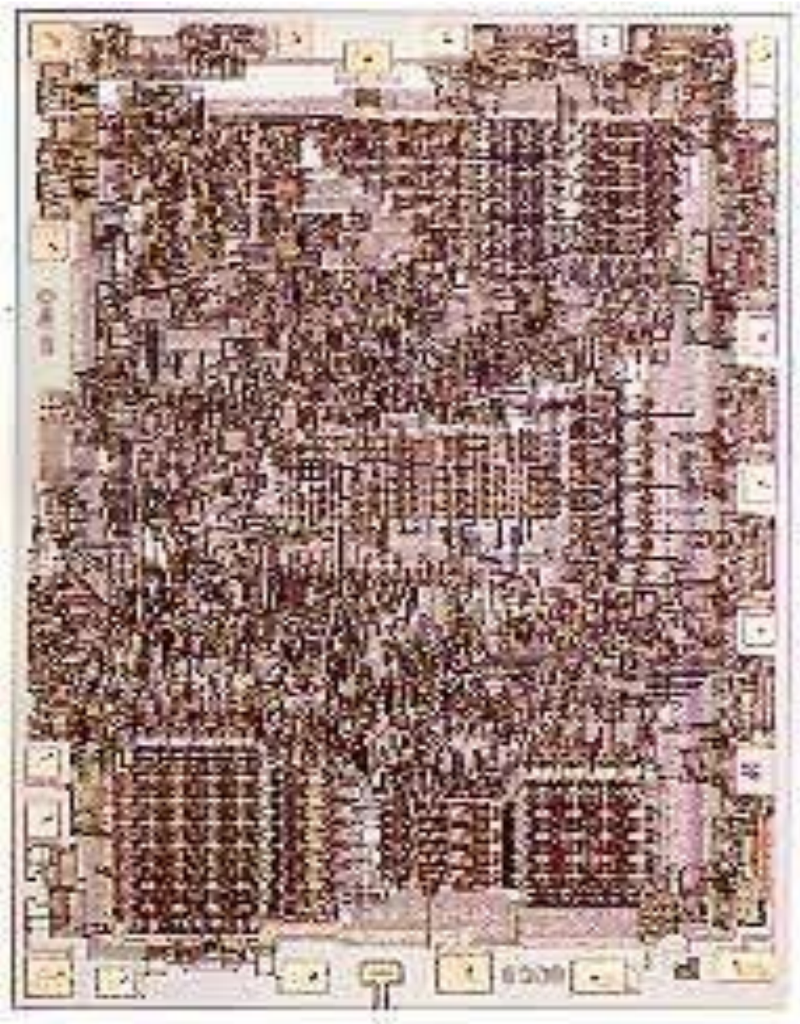
INTEGRATED CIRCUITS



- Third Generation Computers used Integrated Circuits (chips)
- Integrated Circuits are transistors, resistors, and capacitors integrated together into a single “chip”
- Instead of punched cards and printouts, users started interacting with keyboards and mouse

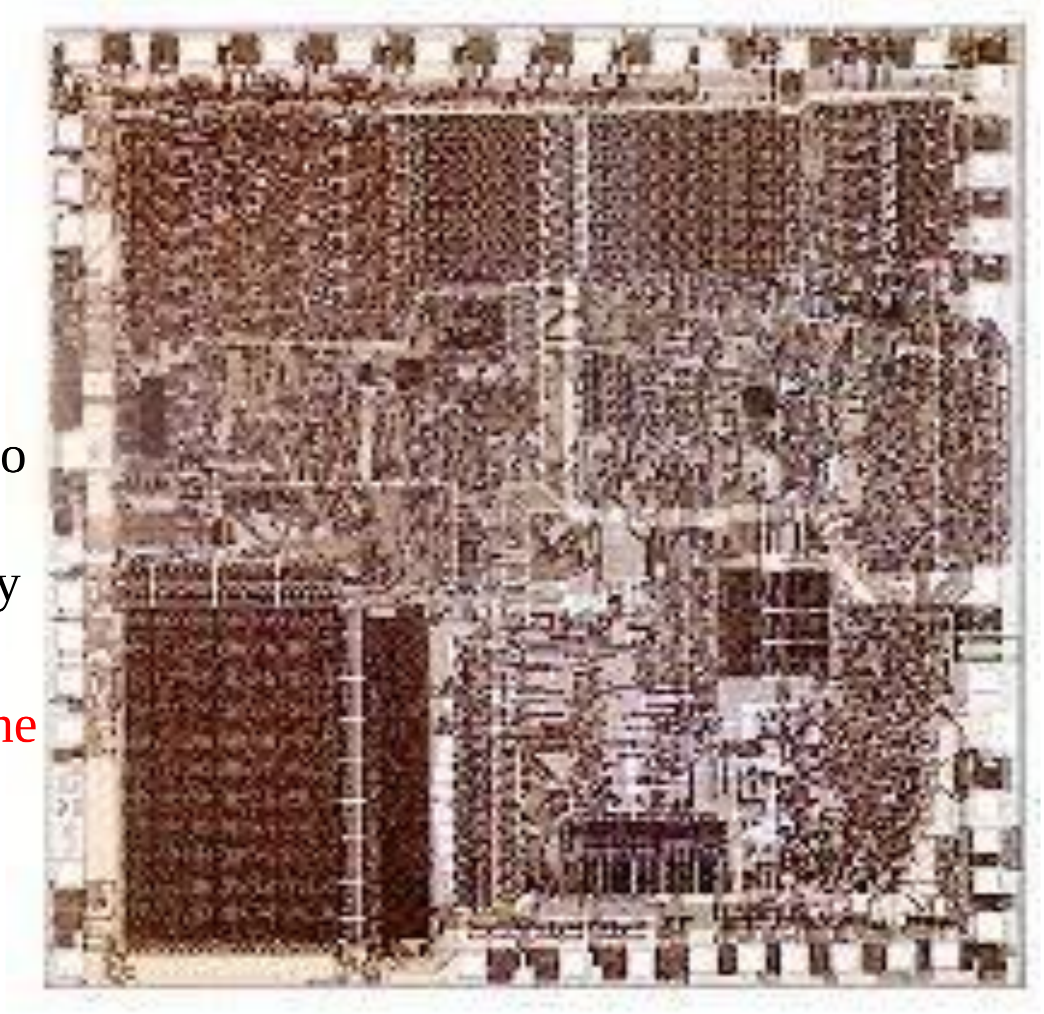
1972: 8008 MICROPROCESSOR

- The 8008 was twice as powerful as the 4004.
- According to the magazine *Radio Electronics*, Don Lancaster, a dedicated computer hobbyist, used the 8008 to create a predecessor to the first personal computer, a device *Radio Electronics* dubbed a "TV typewriter." It was used as a dumb terminal.



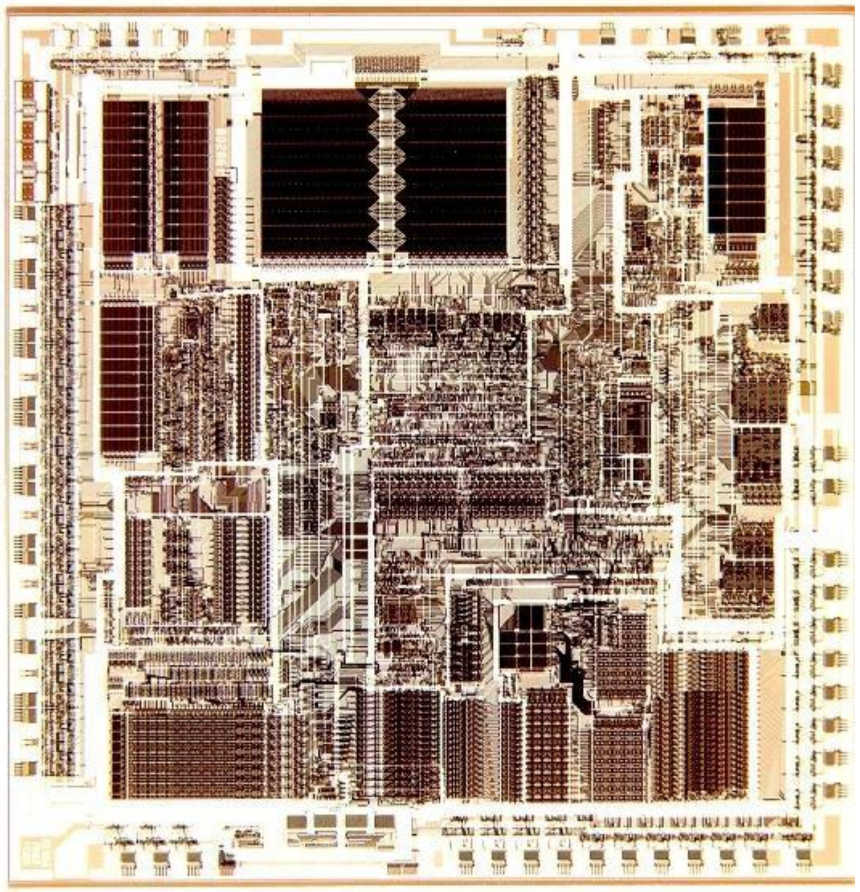
1978: 8086-8088 MICROPROCESSOR

- In 1978 intel
  - 16-bit
  - 2.5MIPS **millions of instructions per second**
  - **1M-byte memory**
  - 4- or 6-byte instruction (cache) queue that prefetch instructions
  - over 20,000 variations instructions.
  - 64K bytes of memory found in 8-bit microprocessors to execute efficiently
  - The 16-bit 8086 and 8088 provided 1M byte of memory for these applications
  - Popularity of Intel ensured in 1981 when IBM chose the 8088 in its personal computer



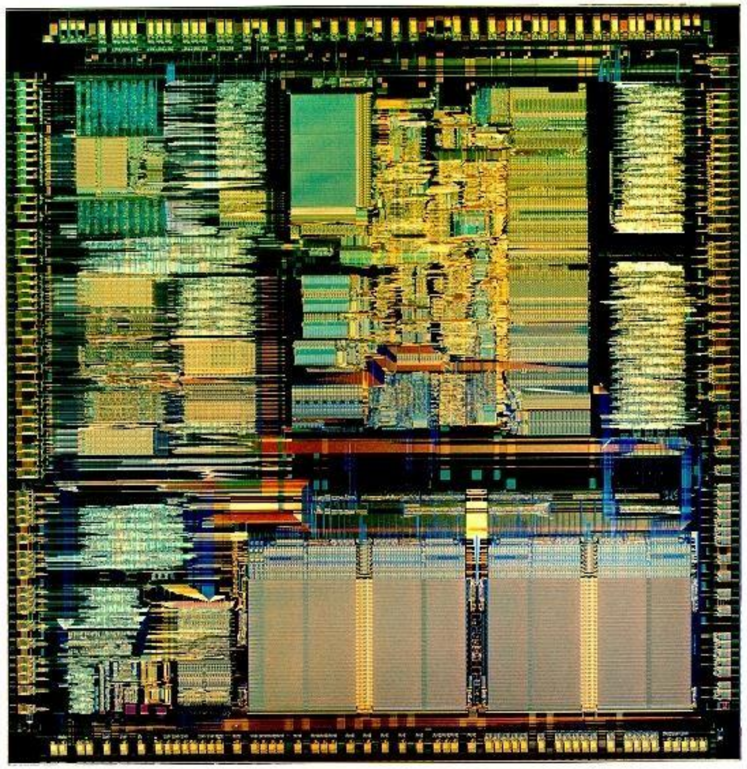
1982: 286 MICROPROCESSOR

- The 286, also known as the 80286, was the first Intel processor that could run all the software written for its predecessor.
- This software compatibility remains a hallmark of Intel's family of microprocessors.
- Within 6 years of it release, there were an estimated 15 million 286-based personal computers installed around the world.
    - 80286: updated 8086
        - 16M byte memory addressing
        - Instructions identical to 8086 few more added
        - 4MIPS
        - 8Mhz clock speed



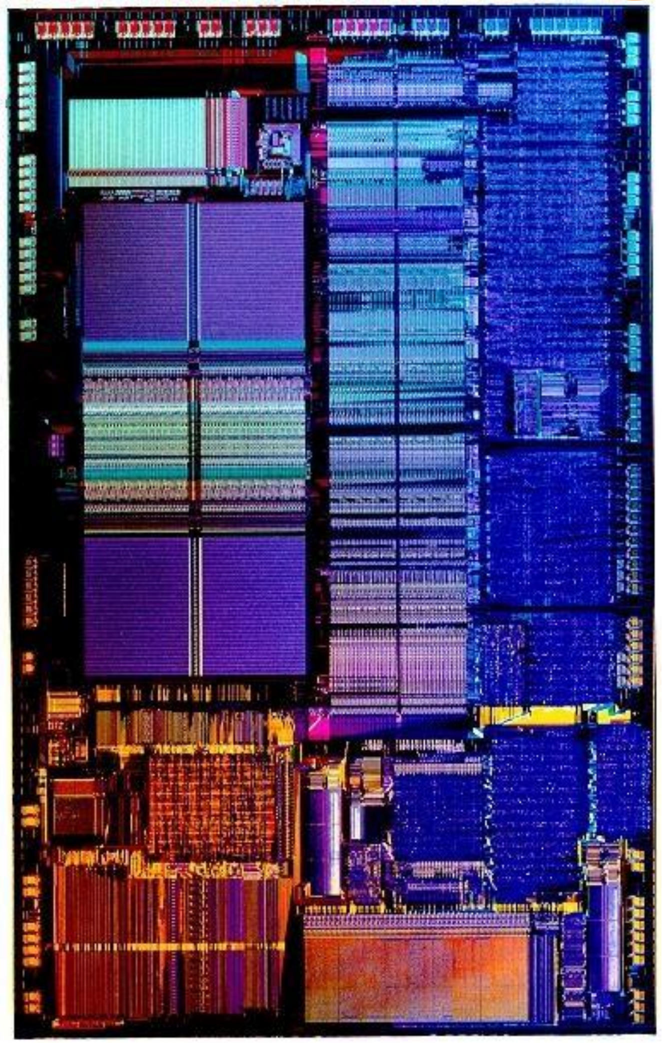
1985: INTEL 386(TM) MICROPROCESSOR

- The Intel 386TM microprocessor featured 275,000 transistors -- more than 100 times as many as the original 4004.
- It was a 32-bit chip and was "multi tasking," meaning it could run multiple programs at the same time.
- 80386 (1985)
  - 32-bit address bus and 32-bit data bus
  - 4GB memory
  - Hardware circuitry for memory management
  - Additional instructions referenced 32-bit registers and managed the memory system



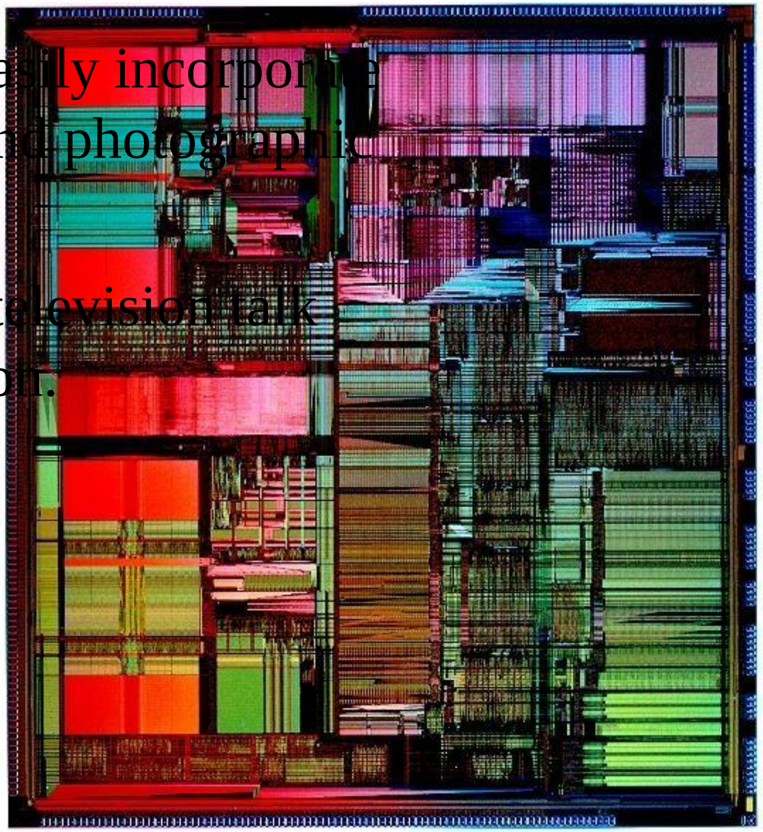
1989: INTEL 486(TM) DX CPU

- The 486™ generation really allowed the shift from command-level computing into point-and-click computing.
- The Intel 486™ processor was the first to offer a built-in math coprocessor, which speeds up computing because it offers complex math functions from the central processor.
- 80486
  - Highly integrated package.
  - 80386-like microprocessor.80387-like numeric coprocessor.
  - 50 MIPS
  - 8K-byte cache memory system
  - Half of its instructions executed in 1 clock cycle rather 2



1993: PENTIUM® PROCESSOR

- The Pentium® processor allowed computers to more easily incorporate "real world" data such as speech, sound, handwriting and photographic images.
- The name Pentium®, mentioned in the comics and on television talk shows, became a household word soon after introduction.

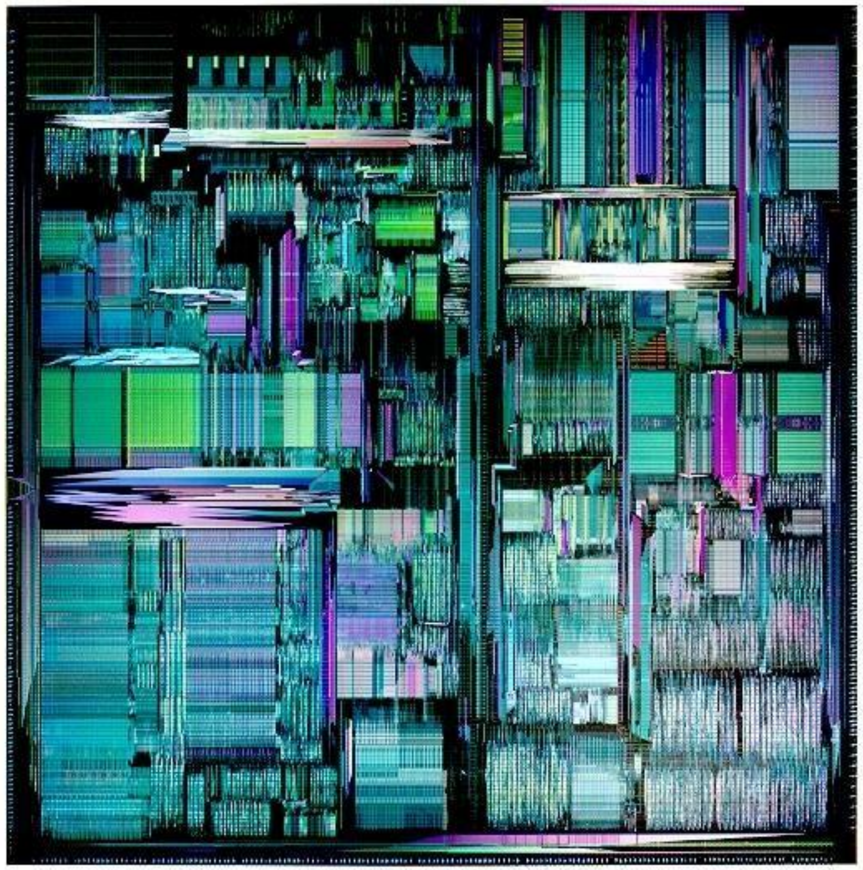


- Originally named P5 or 80586
- Clock speed of 60MHZ
- Executes 110MIPS
- Cache size: 16K bytes (8K cache in 80486)
- 8K-byte instruction cache and data cache.
- Memory system up to 4G bytes.
- Data bus width increased to a full 64 bits.
- Data bus transfer speed 60 MHz or 66 MHz depending on the version of the Pentium

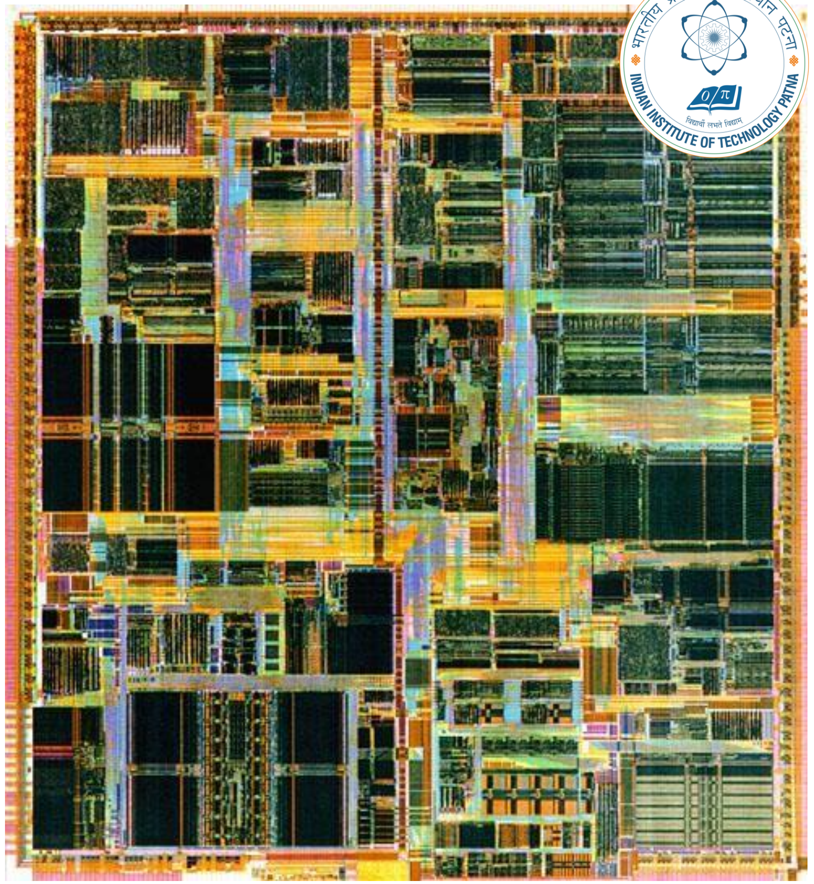
Vinod Dham *was known as the "Father of the Pentium chip"*

1995: PENTIUM® PRO PROCESSOR

- Released in the fall of 1995 the Pentium® Pro processor is designed to fuel 32-bit server and workstation-level applications, enabling fast computer-aided design, mechanical engineering and scientific computation.
- Each Pentium® Pro processor is packaged together with a second speed-enhancing cache memory chip.
- The powerful Pentium® Pro processor boasts 5.5 million transistors.



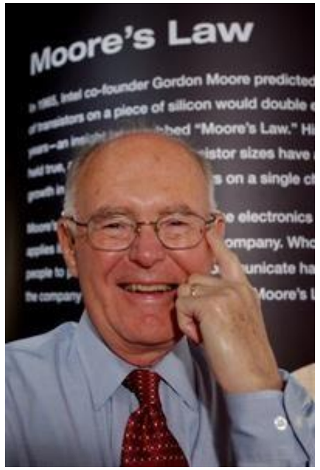
1997: PENTIUM® II PROCESSOR

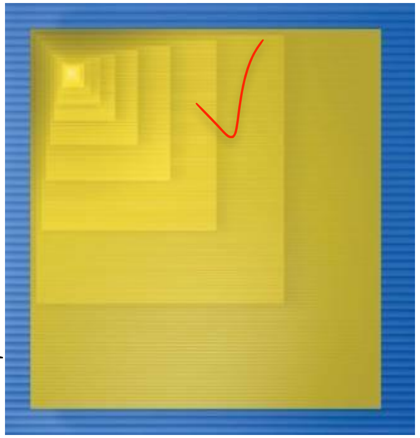


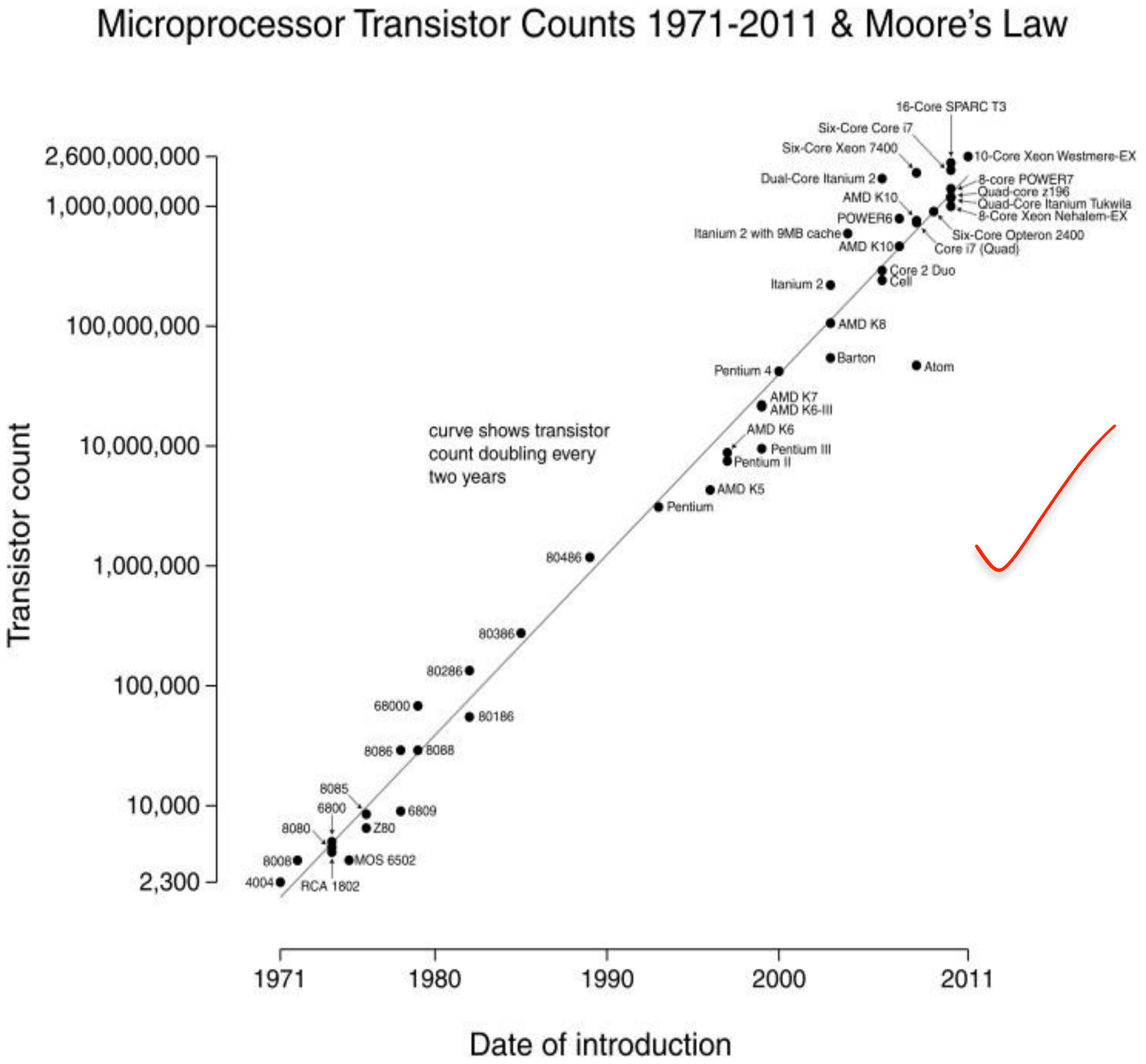
- The 7.5 million-transistor Pentium® II processor incorporates Intel MMX™ technology, which is designed specifically to process video, audio and graphics data efficiently.
- It is packaged along with a high-speed cache memory chip in an innovative Single Edge Contact (S.E.C.) cartridge that connects to a motherboard via a single edge connector, as opposed to multiple pins.
- With this chip, PC users can capture, edit and share digital photos with friends and family via the Internet; edit and add text, music or between-scene transitions to home movies; and, with a video phone, send video over standard phone lines and the Internet.

MOORE'S LAW

- IC capacity doubling about every 18 months for several decades
  - Known as "Moore's Law" after Gordon Moore, co-founder of Intel
    - Predicted in 1965 predicted that components per IC would double roughly every year or so
  - Picture depicts related phenomena
    - For a particular number of transistors, the IC shrinks by half every 18 months
      - Notice how much shrinking occurs in just about 10 years
      - Enables incredibly powerful computation in incredibly tiny devices
  - Today's ICs hold *billions* of transistors
    - The first Pentium processor (early 1990s) needed only 3 million







Number of transistors on an integrated circuit doubles ~ every two years

Image source: Wikipedia

REQUIREMENTS FOR FUTURE TECHNOLOGY

- Must be suitable for portable, low-power operation
  - Consumer products
  - Internet of Things components
  - Not cryogenic, not quantum
- Must be inexpensive to manufacture
  - Comparable to current semiconductor technology
    - O(1) cost to make chip with O(*N*) devices
- Need not be based on transistors
  - Memristors, carbon nanotubes, DNA transcription, ...
  - Possibly new models of computation
  - But, still want lots of devices in an integrated system

*A **memristor** is a non-linear two-terminal electrical component relating electric charge and magnetic flux linkage. -- Wiki*

WHY STUDY COMPUTER ARCHITECTURE TODAY?

- Enable better systems: make computers faster, cheaper, smaller, more reliable, …
  - By exploiting advances and changes in underlying technology/circuits

- Enable new applications
  - Life-like 3D visualization 20 years ago?
  - Virtual reality?

- Enable better solutions to problems
  - Software innovation is built into trends and changes in computer architecture
    - > 50% performance improvement per year has enabled

- Understand why computers work the way they do

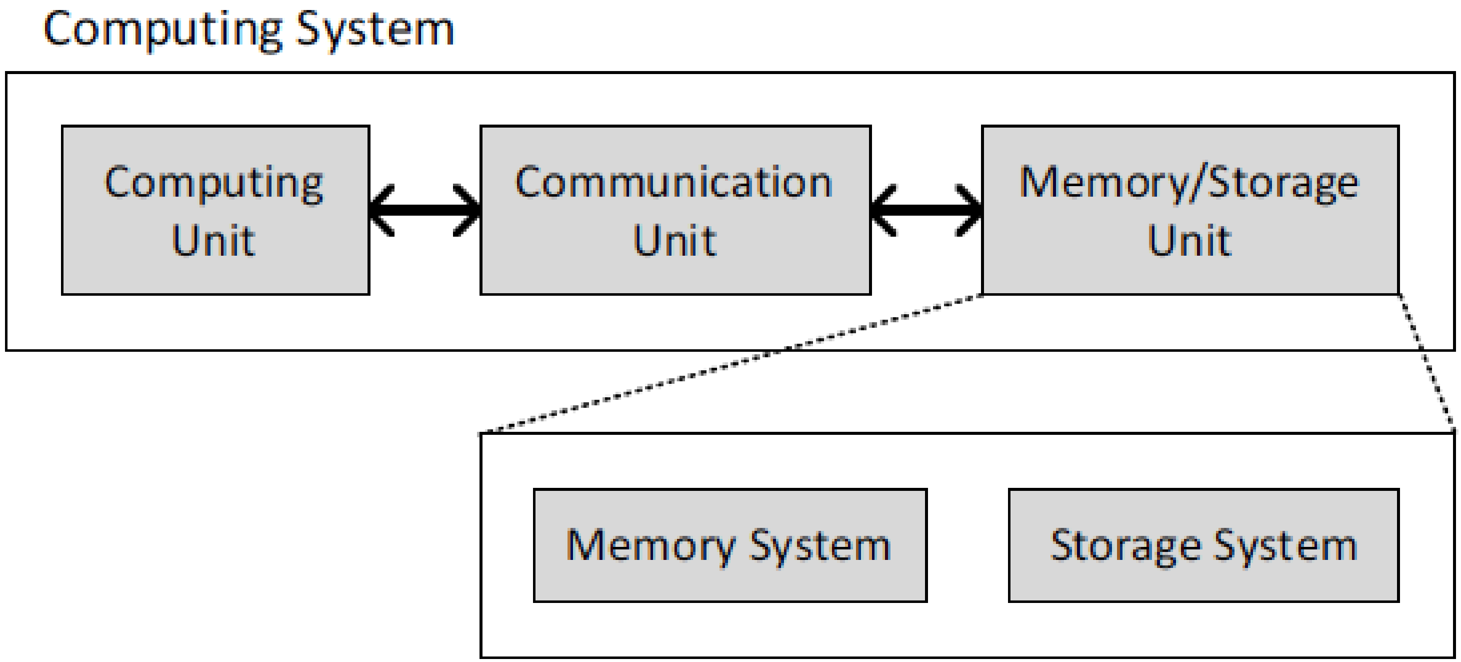
COMPUTER ARCHITECTURE TODAY

- Today is a very exciting time to study computer architecture
- Industry is in a large paradigm shift (to multi-core)
- Many problems motivating and caused by the shift
  - Power/energy constraints
  - Complexity of design -- multi-core
  - Technology scaling -- new technologies
  - Memory wall/gap
  - Reliability wall/issues
  - Programmability wall/problem

60

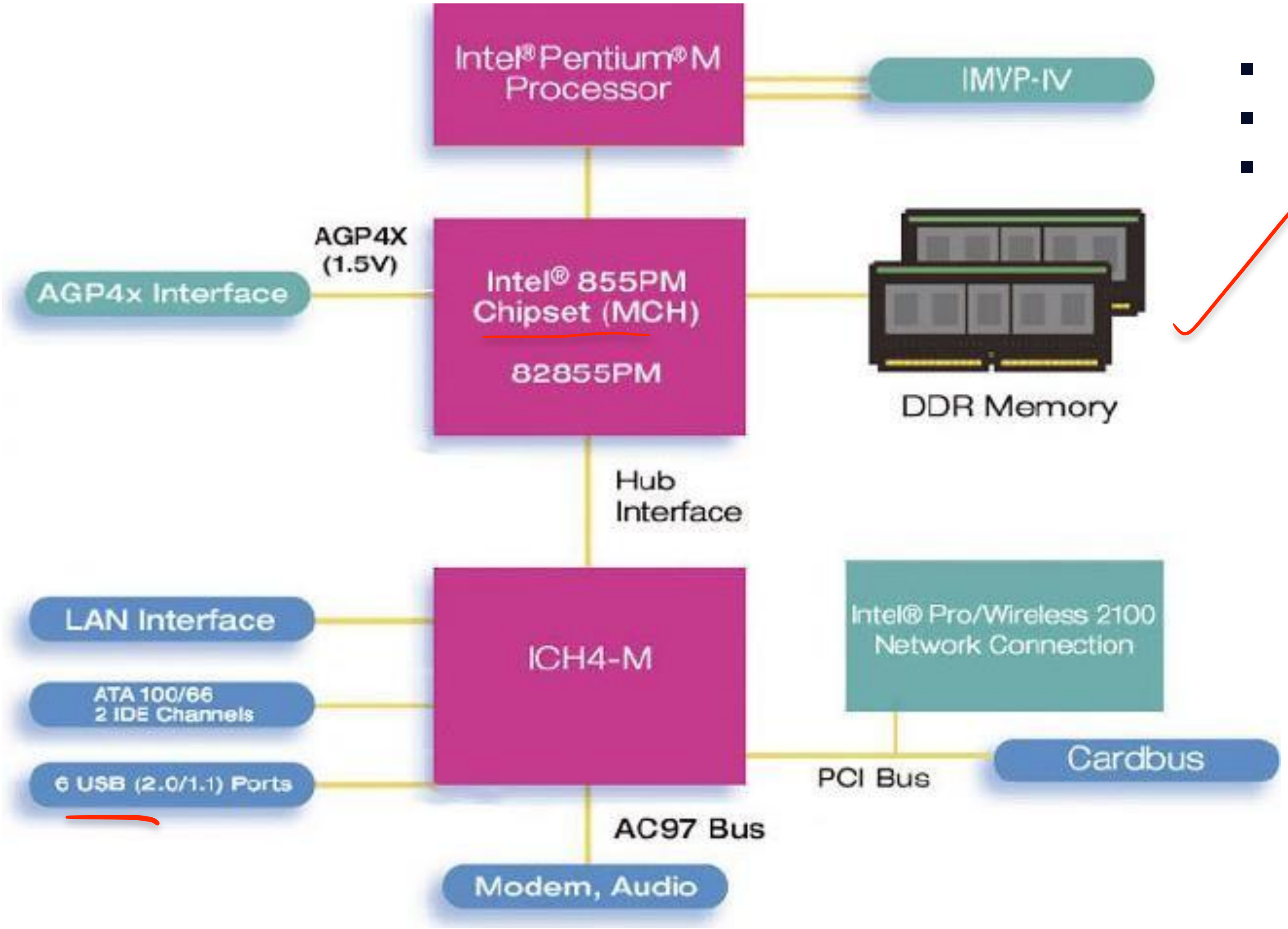
WHAT IS A COMPUTER?

- Three key components
- Computation
- Communication
- Storage (memory)



61

WHAT IS A COMPUTER?

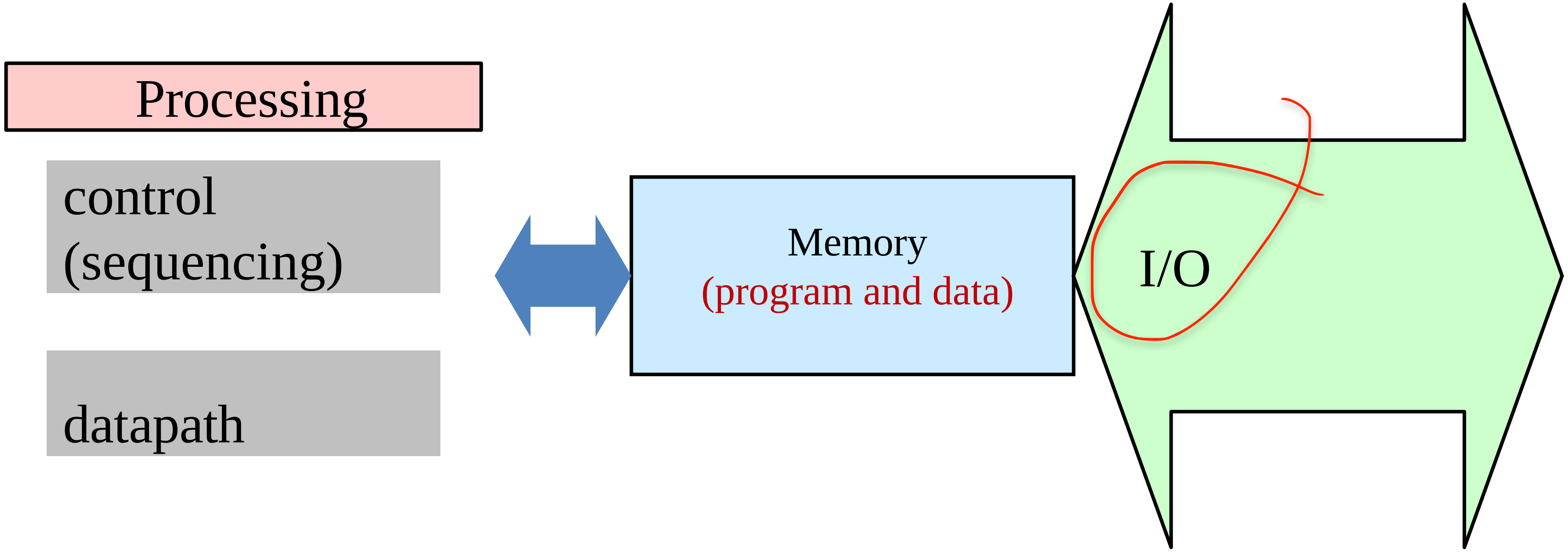


- PCI: Peripheral Component Interconnect
- DDR: Double Data Rate
- USB: Universal Serial Bus

Courtesy, Intel

WHAT IS A COMPUTER?

- We will cover all three components



CLASSES OF COMPUTERS

- Desktop computers: Designed to deliver good performance to a single user at low cost usually executing 3rd party software, usually incorporating a graphics display, a keyboard, and a mouse
- Servers: Used to run larger programs for multiple, simultaneous users typically accessed only via a network and that places a greater emphasis on dependability and (often) security
- Supercomputers: A high performance, high cost class of servers with hundreds to thousands of processors, terabytes of memory and petabytes of storage that are used for high-end scientific and engineering applications
- Embedded computers (processors): A computer inside another device used for running one predetermined application. Very often cost, power, and failure rate are more important than performance.

CISC VERSUS RISC

CISC

- Complex Instruction Set Computer
- Emphasis on hardware
- Includes multi-clock complex instructions
- Memory-to-memory: "LOAD" and "STORE" incorporated in instructions
- Small code sizes, high cycles per second
- Transistors used for storing complex instructions

RISC

- Reduced Instruction Set Computer
- Emphasis on software
- Single-clock,reduced instruction only
- Register to register: "LOAD" and "STORE“ are independent instructions
- Low cycles per second, large code sizes
- Spends more transistors on memory registers

INSTRUCTION SET ARCHITECTURE

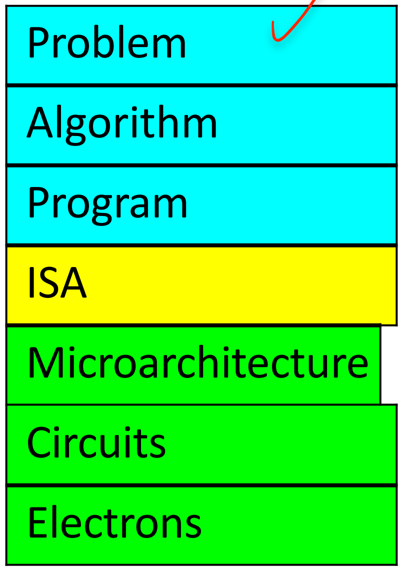
- A very important abstraction
  - interface between hardware and low-level software
  - standardizes instructions, machine language bit patterns, etc.
  - advantage: *different implementations (cost, performance, power) of the same architecture*
  - disadvantage: *sometimes prevents using new innovations*
- Common instruction set architectures:
  - IA-32, PowerPC, MIPS, SPARC, ARM, and others

INSTRUCTION SET ARCHITECTURE

- ISA, or simply architecture – the abstract interface between the hardware and the lowest level software that encompasses all the information necessary to write a machine language program, including instructions, registers, memory access, I/O, …
- ISA Includes
  - Organization of storage
  - Data types
  - Encoding and representing instructions
  - Instruction Set (or opcodes)
  - Modes of addressing data items/instructions
  - Program visible exception handling
- Specifies requirements for binary compatibility across
- implementations.

ISA VS. MICROARCHITECTURE

- ISA
  - Agreed upon interface between software and hardware
    - SW/compiler assumes, HW promises
  - What the software writer needs to know to write and debug system/user programs
- Microarchitecture
  - Specific implementation of an ISA
  - Not visible to the software
- Microprocessor
  - **ISA, uarch**, circuits
  - "Architecture" = ISA + microarchitecture



MIPS ISA

- Instruction Categories
  - Load/Store
  - Computational
  - Jump and Branch
  - Floating Point
  - Memory Management
  - Special

R0 - R31

PC
HI
LO

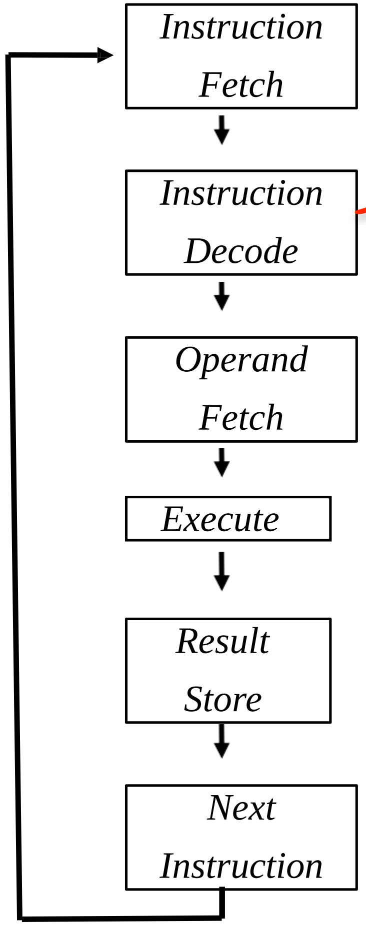
3 Instruction Formats, 32 bits wide

OP	rs	rt	rd	sa	funct

OP	rs	rt	immediate

OP	jump target

EXECUTION CYCLE



Obtain instruction from program storage

Determine required actions and instruction size

Locate and obtain operand data

Compute result value or status

Deposit results in storage for later use

Determine successor instruction

LEVELS OF REPRESENTATION

High Level Language Program

temp = v[k];
v[k] = v[k+1];
v[k+1] = temp;

Compiler

Assembly Language Program

```
lw   $15,   0($2)
lw   $16,   4($2)
sw   $16,   0($2)
sw   $15,   4($2)
```

Assembler

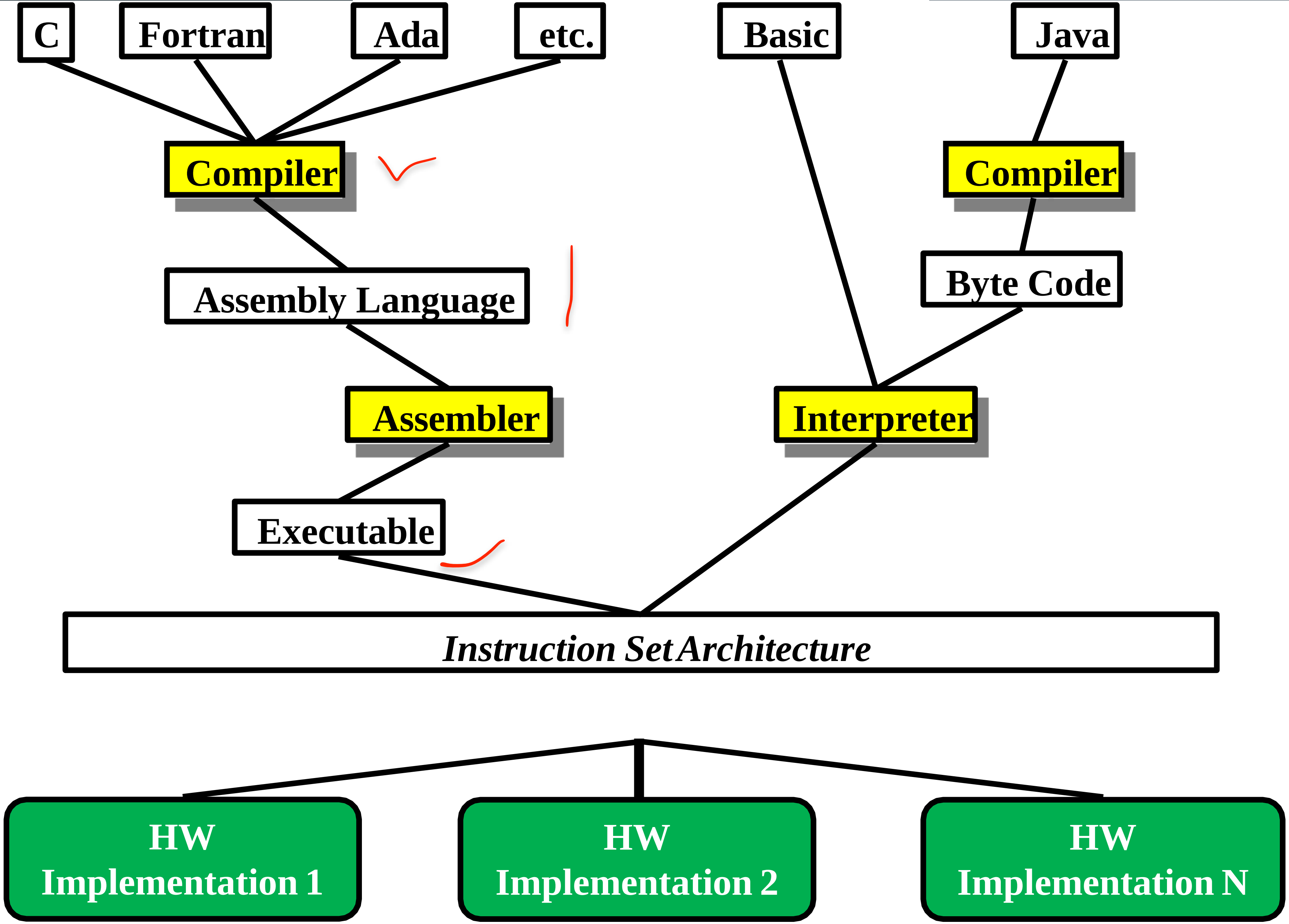
Machine Language Program

```
0000  1001 1100  0110  1010  1111  0101  1000
1010  1111 0101  1000  0000  1001  1100  0110
1100  0110 1010  1111  0101  1000  0000  1001
0101  1000 0000  1001  1100  0110  1010  1111
```

Machine Interpretation

Control Signal Specification

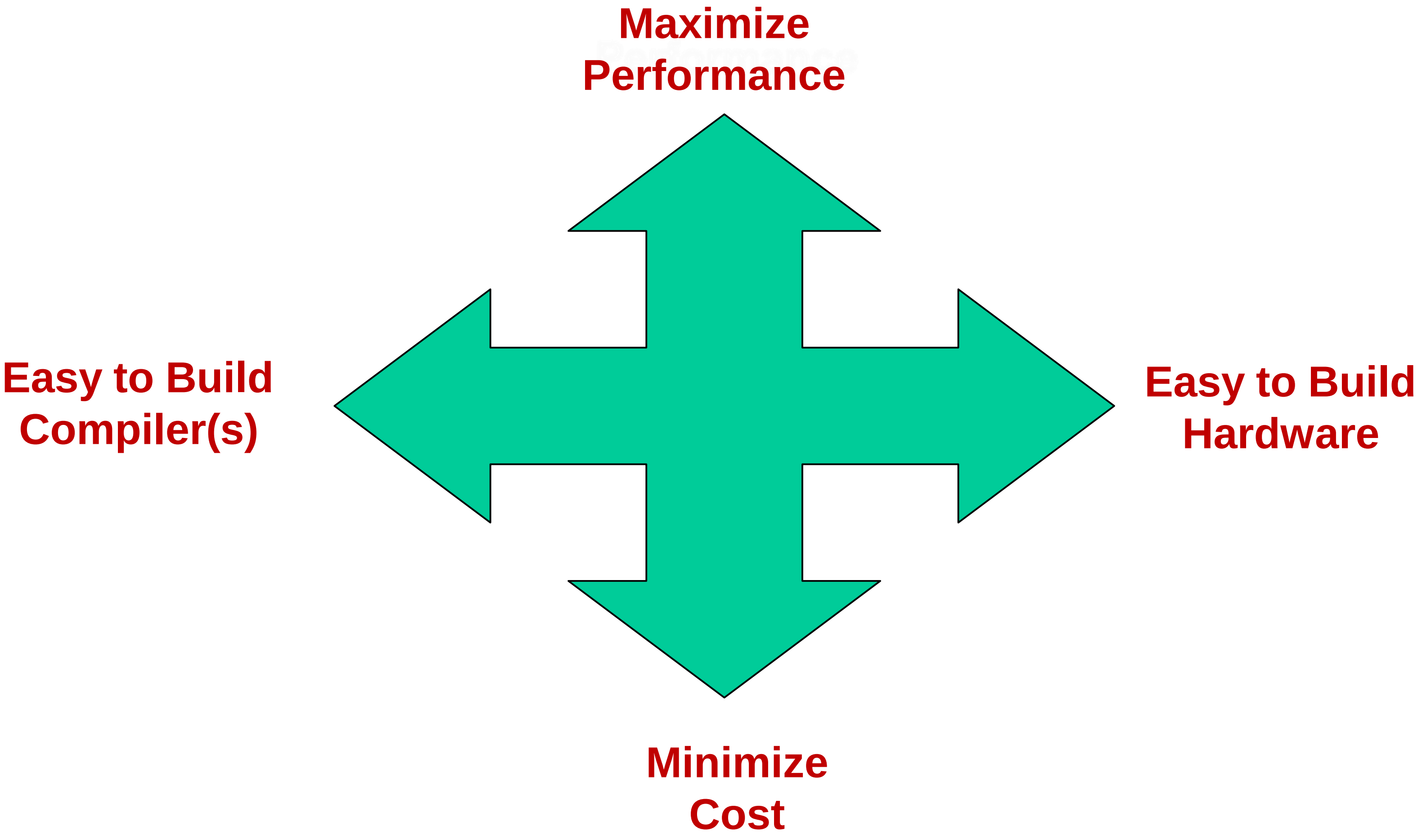
ALUOP[0:3] <= InstReg[9:11] & MASK
[i.e.high/low on control lines]


C
Fortran
Ada
etc.
Basic
Java
Compiler
Compiler
Assembly Language
Byte Code
Assembler
Interpreter
Executable
Instruction Set Architecture
HW Implementation 1
HW Implementation 2
HW Implementation N

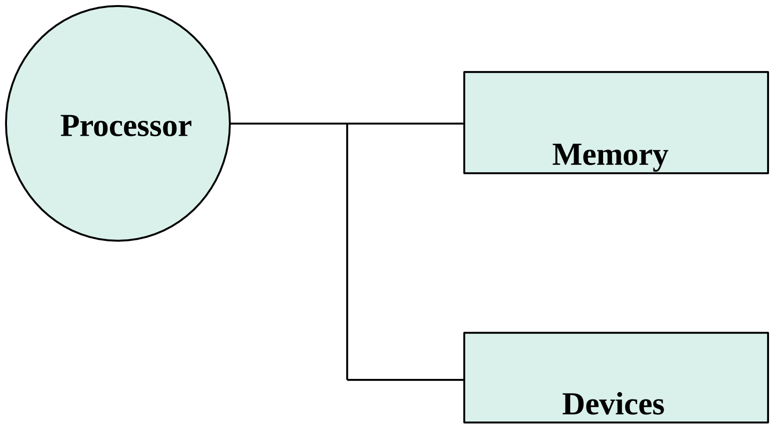
ADVANTAGES OF HLLS

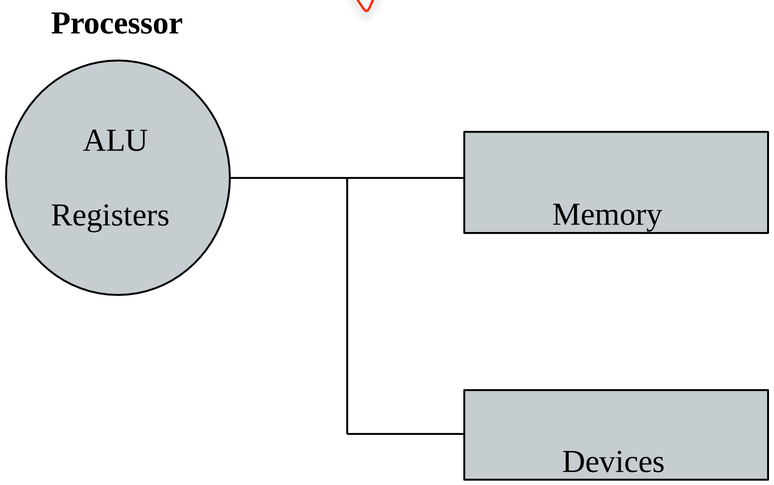
- Higher-level languages (HLLs)
  - Allow the programmer to think in a more natural language and for their intended use (Fortran for scientific computation, Cobol for business programming, Lisp for symbol manipulation, Java for web programming, …)
  - Improve programmer productivity – more understandable code that is easier to debug and validate
  - Improve program maintainability
  - Allow programs to be independent of the computer on which they are developed (compilers and assemblers can translate high-level language programs to the binary instructions of any machine)
  - Emergence of optimizing compilers that produce very efficient
- assembly code optimized for the target machine
- Compilers convert source code to object code
- Libraries simplify common tasks

INSTRUCTION SET DESIGN GOALS



WHERE TO KEEP THE OPERANDS





PROGRAM: IN MEMORY

Address	Memory	
	Reserved	
0x7FFFFFFC	Stack ↓	← $sp = 0x7FFFFFFC
0x10010000	↑ Heap	
	⋮	← $gp = 0x10008000
	y	
	g	
0x10000000	f	
	⋮	
	0x03E00008	
	0x00851020	
	0x03E00008	
	0x23BD0004	
	0x8FBF0000	
	0xAF828008	
	0x0C10000B	
	0xAF858004	
	0x20050003	
	0xAF848000	
	0x20040002	
	0xAFBF0000	
0x00400000	0x23BDFFFC	← PC = 0x00400000
	Reserved	

THE MIPS MEMORY MAP

Address	Segment
0xFFFFFFFC 0x80000000	Reserved
0x7FFFFFFC 0x10010000	Stack ↓ Dynamic Data ↑ Heap
0x1000FFFC 0x10000000	Static Data
0x0FFFFFFC 0x00400000	Text
0x003FFFFC 0x00000000	Reserved

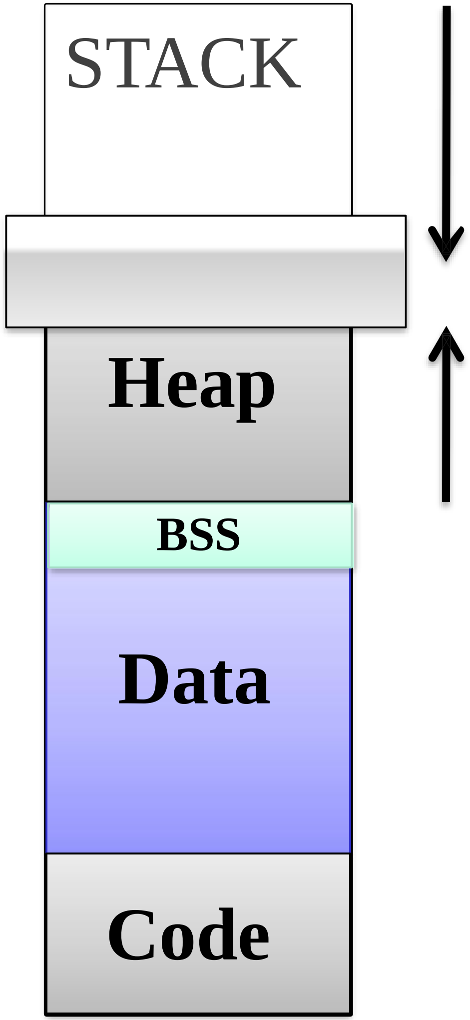
Function calls,
Local variable,
dynamic memory

Global variables

Instructions

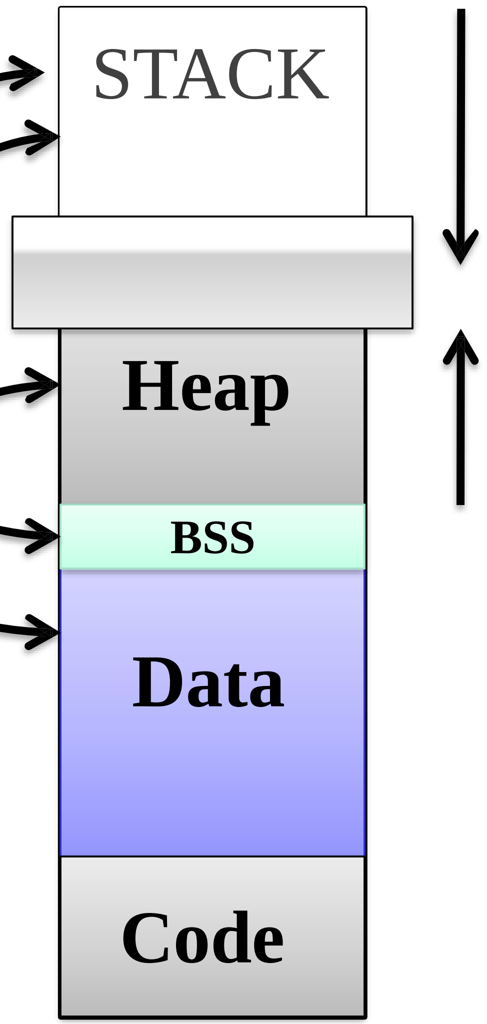
MEMORY LAYOUT OF C PROGRAM

- Stack
  - automatic (default), local
  - Initialized/uninitialized
- Data
  - Global, static, extern
  - BSS: Block Started by Symbol
- Code
  - program instructions
- Heap
  - malloc, calloc



MEMORY LAYOUT OF C PROGRAM

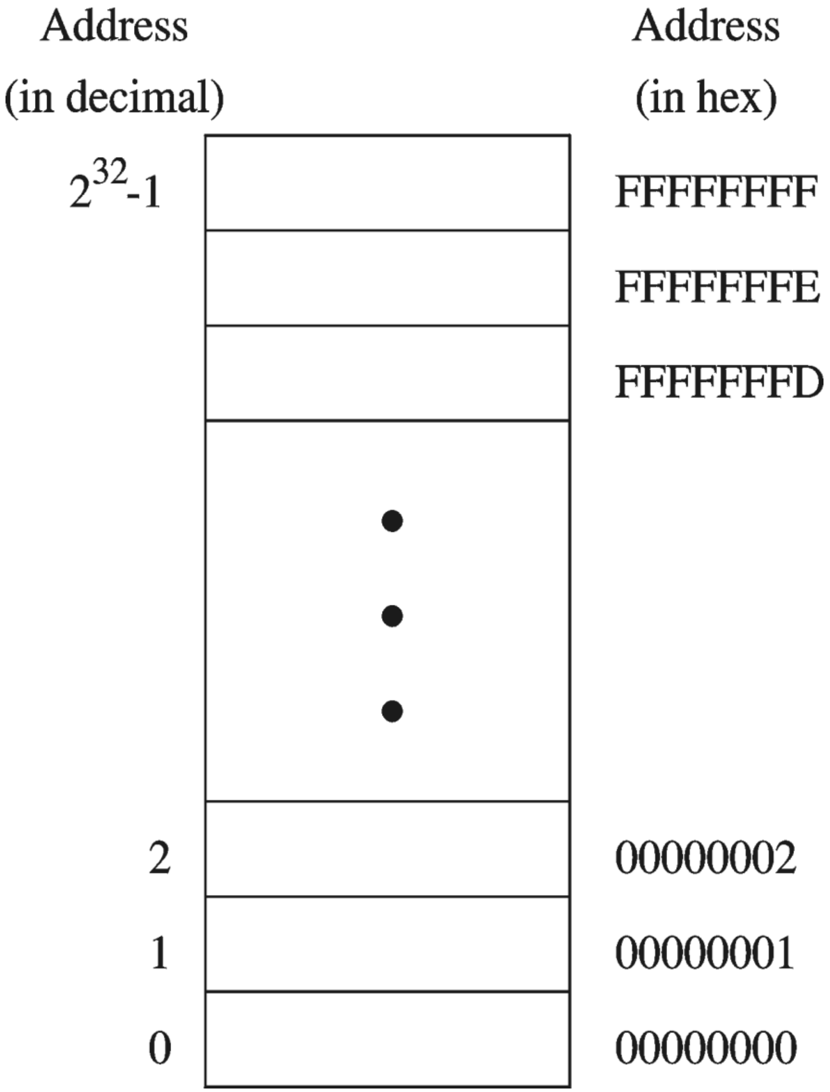
```
int A;
int B=10;


main(){
int Alocal;
int *p;
p=(int*)malloc(10);
}
```



- 4048
4044
4040

0

ADDRESS SPACE

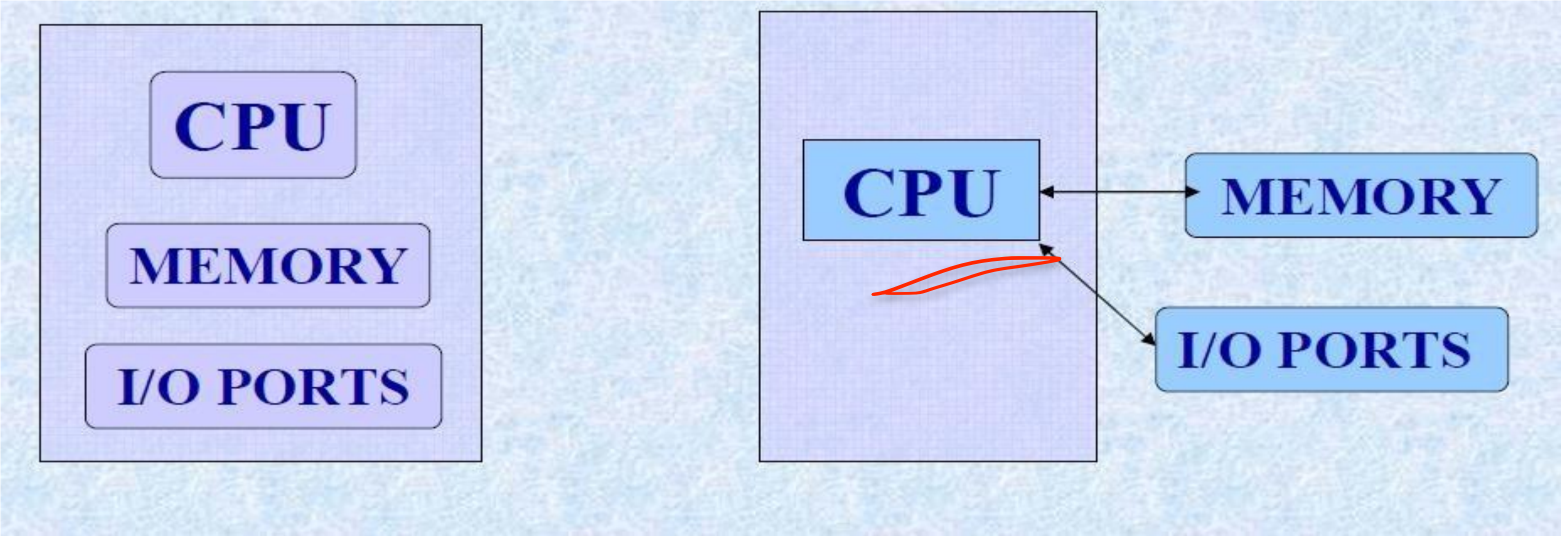


Address Space is the set of memory locations (bytes) that can be addressed

MICROPROCESSOR AND MICROCONTROLLER

- Micro Controller
- It is a single chip Consists Memory, I/O ports

- Micro Processor
  - It is a CPU
  - Memory, I/O Ports to be connected externally



THANK YOU!