SE1101 : COMPUTER ORGANIZATION
DS1106 : COMPUTER SYSTEM ORGANIZATION

Ms.S.Sweshthika

Lecturer (Probationary)

Department of Software Engineering

Faculty of Computing

Sabaragamuwa University of SriLank

Evolution of Computer Generations

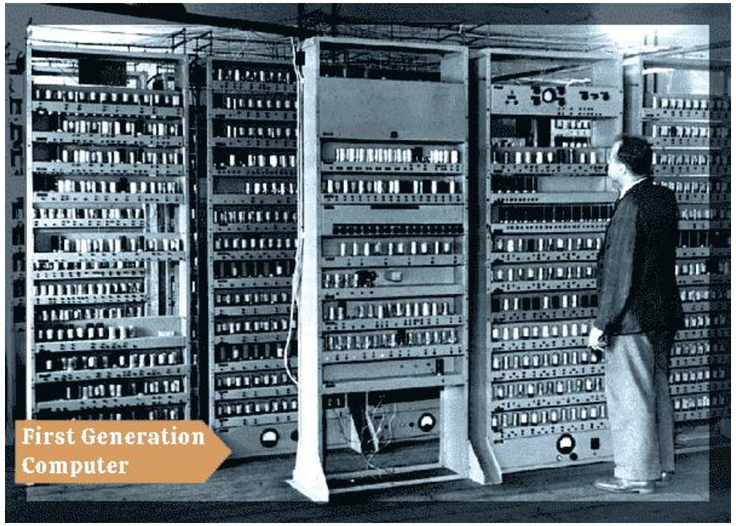
Objectives

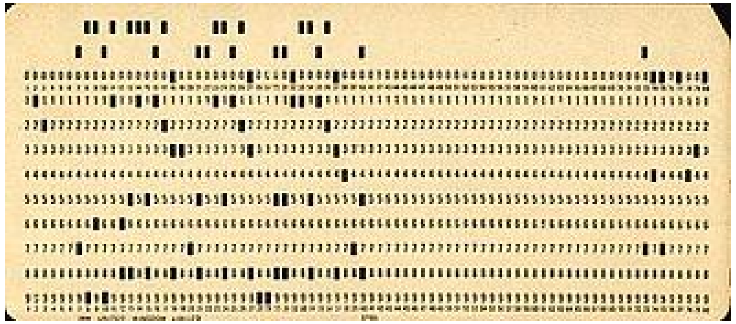
At the end of this lecture you will be able to explain,

- Understand the evolution of computer systems from the first to the sixth generation.
- Distinguish between key hardware and software trends across generations.
- Explain the importance and evolution of Intel x86 and ARM architectures.
- Recognize the characteristics and applications of embedded systems.
- Describe the fundamentals of cloud computing, its service models, and benefits/challenges.

First Generation (≈1940–1956) – Vacuum Tubes

- **Hardware:** Vacuum tubes
  : Magnetic drums
  : Punched cards
- **Software:** Only machine language
  : no OS
- **Systems:** ENIAC, UNIVAC, IBM 701
- **Characteristics:** Massive size
  : High power consumption
  : Reliable
  : Slow I/O





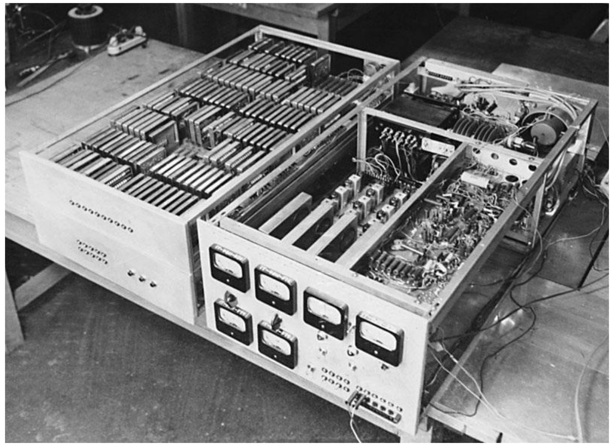
Punched Card

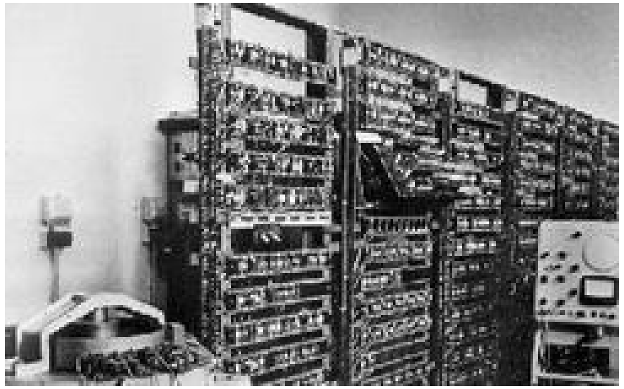
What is IAS Computer?

- **Built at**: Princeton Institute for Advanced Studies (designed 1946, completed 1952).
- **Key Designer**: John von Neumann (with colleagues).
- **Concept**: Introduced the **stored-program concept**
- **Prototype**: Served as a model for most later general-purpose computers.
- **Architecture** (Von Neumann architecture):- Main memory
  :- Arithmetic and Logic Unit
  :- Control unit.
  :- Input/output equipment.
- **Legacy**: Almost all modern computers still follow this same architecture.

Second Generation (≈1956–1963) – Transistors

- **Hardware:** Transistors
  : Magnetic-core memory
- **Software:** Assembly language
  : Batch-processing OSs
- **Systems:** IBM 1401, IBM 7090
- **Characteristics:** Smaller
  : Faster
  : More Reliable
  : Lower Heat/Power





What is Batch Processing OS?

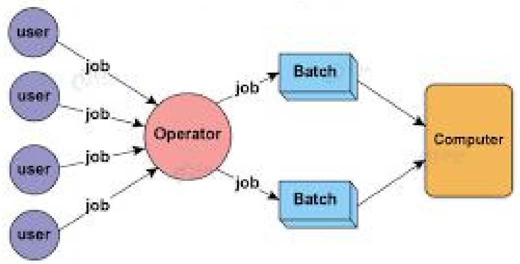
•**Definition**: A type of operating system where jobs (programs + data) are collected together (in batches) and executed one after another, without user interaction during execution.

•**How it worked**:- Users submitted their programs (often on punched cards).

:- The OS grouped these jobs into a batch.

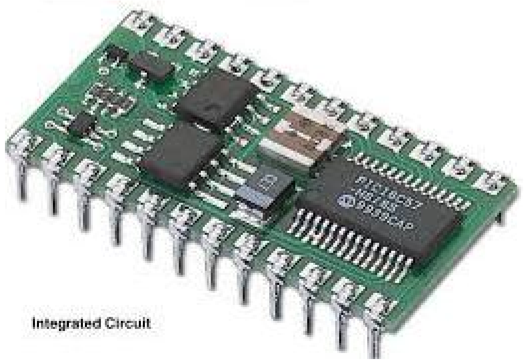
:- The computer executed them sequentially.

•**Key Point**: No real-time interaction. The computer works on one batch until it's done, then moves to the next.



Third Generation (≈1964–1975) – Integrated Circuits

- **Hardware:** ICs (SSI, MSI)
  : Keyboards/monitors
- **Software:** High-level languages
  : Early OS
  : Time-sharing
- **Systems:** IBM 360, PDP-8, CDC 6600
- **Characteristics:** Multiprogramming
  : Improved efficiency
  : Lower cost



(≈1975–1989) – Microprocessors

•**Hardware:** LSI/VLSI chips
: Microprocessors
: Optical/floppy disks

•**Software:** GUI-based OS (e.g. UNIX, early MS-DOS)
: More interactive

•**Systems:** IBM PC, Apple II

•**Characteristics:** Personal computing begins
: Compact
: Affordable
: Upgradeable

Fifth Generation (~1989–2010) – ULSI, Networking & AI Foundations

- **Hardware:** ULSI chips
  : Internet connectivity
  : Optical storage
  : Multimedia support
- **Software:** GUI OS with emerging AI features
  : Voice/handwriting recognition
- **Systems:** Laptops, notebooks, SUN workstations, early AI-oriented systems
- **Characteristics:** Rich multimedia
  : Portable
  : Highly reliable
  : Network-capable

Sixth Generation (≈2010–Now)

Hardware Trends: Ultra-scale integration
: Multicore
: GPGPUs (graphics processing)
: IoT devices
: Cloud servers

Software Trends: AI and machine learning–embedded software
: Parallel computing frameworks
: Virtualization
: Containerization

Systems: Smartphones, embedded systems, high-performance servers, edge devices, RISC-V and ARM platforms

Characteristics: Massive parallelism, energy-efficient performance, intelligent systems, ubiquitous connectivity

Summary

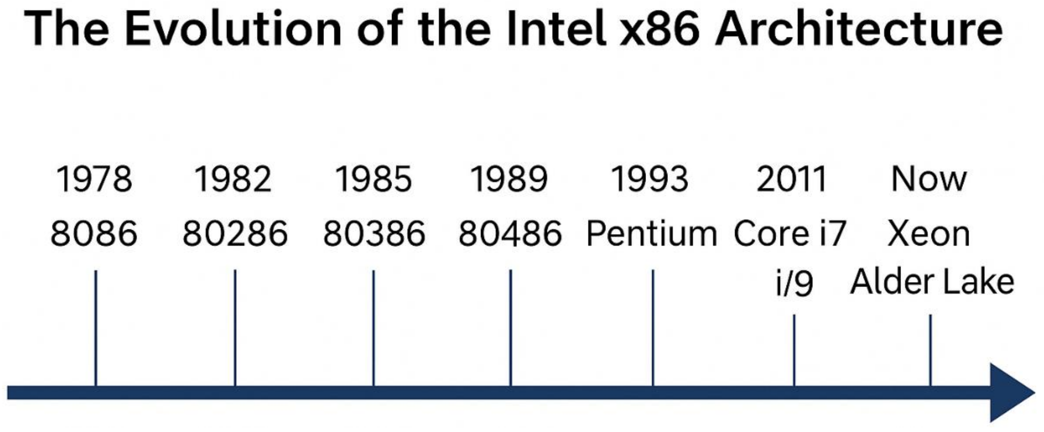
Generation	Time Frame	Hardware	Software/OS	Key Systems	Characteristics
1st	1940–1956	Vacuum tubes, drums	Machine language	ENIAC, UNIVAC	Huge, unreliable, high power
2nd	1956–1963	Transistors, core memory	Assembly, batch OS	IBM 1401, 7090	More reliable, lower cost/power
3rd	1964–1975	ICs, input/output devices	HLL, OS, time-sharing	IBM 360, PDP-8	Efficient, programmable
4th	1975–1989	Microprocessors, LSI/VLSI	GUI OS, interactive tools	IBM PC, Apple II	Personal computing era
5th	1989–2010s	ULSI, network connectivity	AI-capable GUIs	Laptops, workstations	Multimedia, portable, networked
6th+	2010s–Present	Multicore, GPGPU, cloud, IoT	Virtualization, AI frameworks	Smartphones, servers, IoT	Parallel, intelligent, connected

Intel x86 Architecture

- x86: a family of backward-compatible instruction set architectures.
- First introduced by Intel in 1978 (8086).
- Still dominant in desktops, laptops, and servers.
- Backward compatibility means code from decades ago can still run.

Why Study x86 Evolution?

•Shows how computer hardware evolved.
•Demonstrates the impact of Moore’s Law.
•Helps understand modern processor features.
•Links to computer organization concepts: registers, memory, pipelines, parallelism.



Intel x86 Evolution

Generation	Year	Word Size	Clock Speed	Key Features
8086	1978	16-bit	5–10 MHz	Segmented memory
80286	1982	16-bit	6–25 MHz	Protected mode
80386	1985	32-bit	12–40 MHz	Virtual memory
80486	1989	32-bit	20–100 MHz	On-chip FPU
Pentium	1993	32-bit	60–300 MHz	Superscalar
Core	2006+	64-bit	GHz range	Multi-core

Embedded System

An embedded system is a computer system designed for a specific function within a larger device, often with real-time computing constraints.

Example :- Washing Machine Controller
:- ATM Machine
:- Digital Camera
:- Automotive engine control

Key Characteristics of Embedded System

- **Specific Function:** Designed for a dedicated purpose.
- **Real-Time Operation:** Must respond within strict timing limits.
- **Resource Constraints:** Limited memory, CPU speed, and power.
- **Reliability:** Must operate continuously without failure.
- **Low Power Consumption:** Especially in battery-powered devices.

Basic Architecture of Embedded System

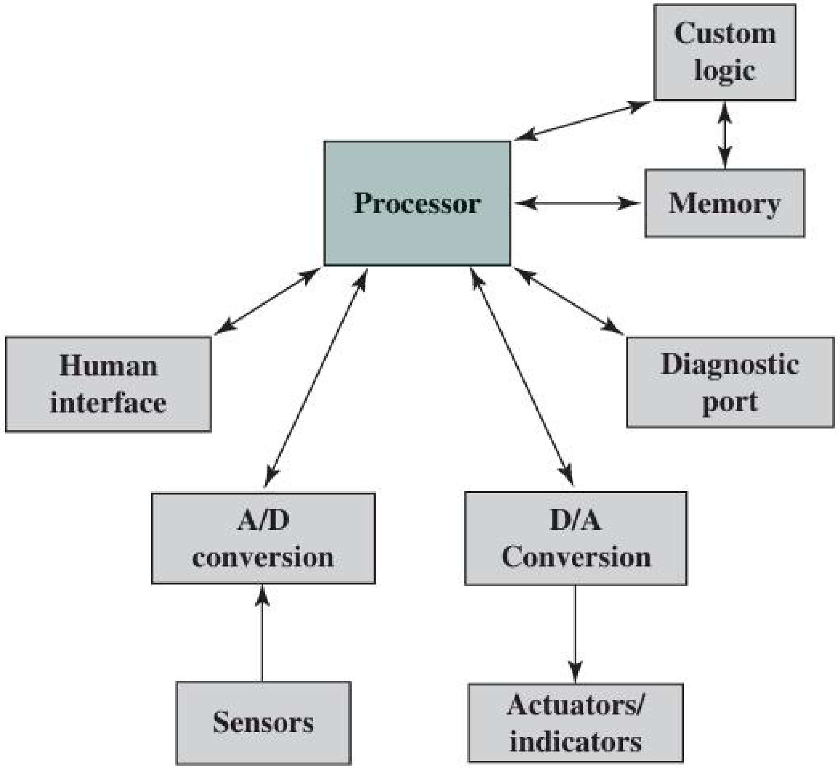


Figure 1.14 Possible Organization of an Embedded System

A/D conversation
Analog to Digital

Types of Embedded System

Type	Processor / Capability	Real-World Examples
Small-Scale	8- or 16-bit microcontroller, limited memory	Simple home appliances (microwave ovens, washing machines)
Medium-Scale	16- to 32-bit controllers, RTOS-capable	ATMs, industrial controllers
Sophisticated / Complex	32- to 64-bit processors, multi-function, networked	Industrial robots, advanced instrumentation

ARM Architecture

- ARM = Advanced RISC Machines (originally Acorn RISC Machine)
- Based on RISC principles (Reduced Instruction Set Computer)
- Designed by ARM Holdings (Cambridge, UK)
- ARM Holdings licenses designs - does not manufacture chips
- Used in embedded systems, mobile devices, and IoT products

Arm architecture is type of computer processor architector designed by ARM Holdings.It based on RISC Principles,instruction can be execute quickly and efficiently.

What is RISC Machine?

•RISC = **Reduced Instruction Set Computer**
•Processor with a **small, simple set of instructions**
•Goal: **faster execution** (usually 1 instruction per clock cycle)

Key Features:- Simple instructions
:- Load/Store architecture (memory only with LOAD / STORE)
:- Many registers → less memory use
:- Pipeline friendly → multiple instructions in parallel

Why ARM Architecture is Popular?

- High performance with low power consumption
- Small die size → fits in compact devices
- Cost-effective for mass production
- Found in:- Smartphones (Android, iPhone)

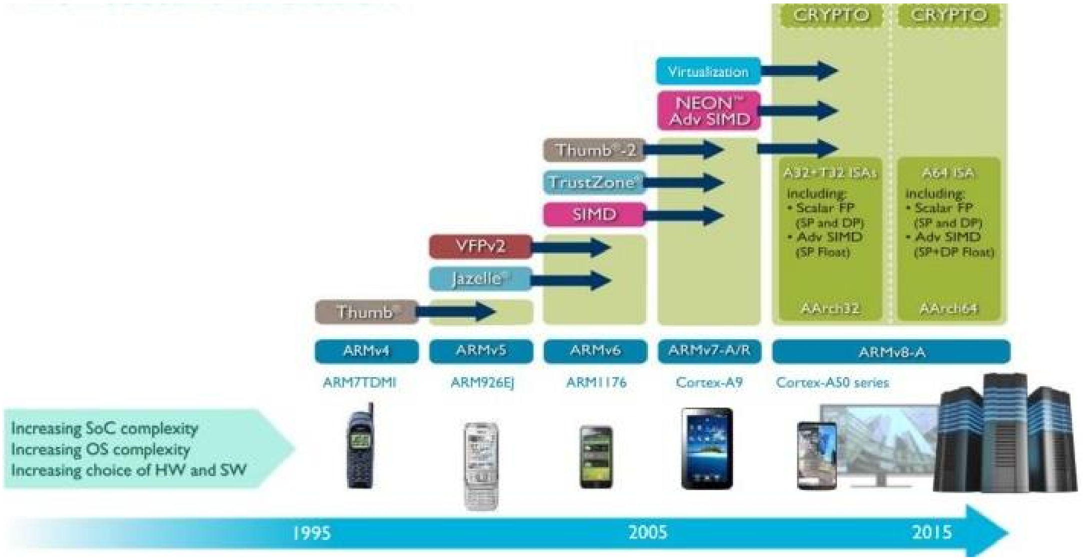
  :- Gaming systems

  :- Smart home devices

  :- Industrial controllers

Evolution of ARM Architecture

- Early 1980s: Developed by Acorn Computers for BBC Micro project
- 1985: ARM1 — first RISC processor
- 1990: ARM Ltd formed (Acorn + VLSI + Apple)
- Evolved into the world's most widely used processor architecture

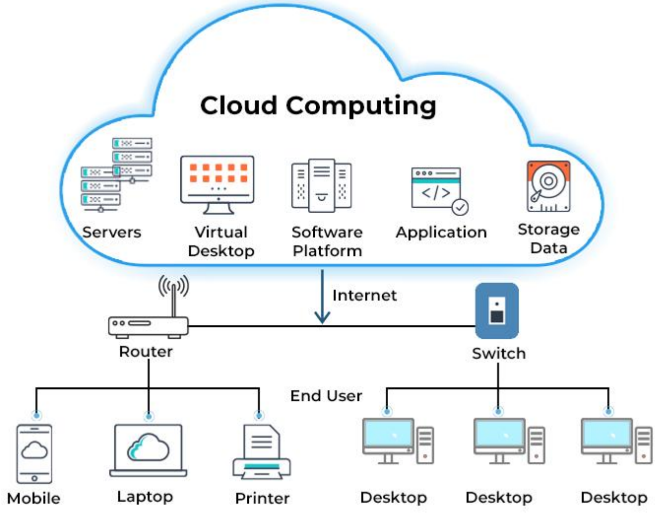


ARM Cortex Product Families

Family	Target Use	Key Features	Memory Management	Examples
Cortex-A	Application processors (smartphones, tablets, smart TVs, home gateways)	High performance, supports OS (Linux, Android, Windows), runs at >1 GHz	MMU (supports virtual memory & paging)	Cortex-A53 (64-bit), Cortex-A72
Cortex-R	Real-time systems (automotive control, storage controllers, industrial automation)	Low latency, predictable timing, high reliability	MPU only (no MMU)	Cortex-R4, Cortex-R7
Cortex-M	Microcontrollers (IoT devices, sensors, embedded systems)	Ultra-low power, deterministic interrupts, compact design	MPU only, Thumb-2 ISA	Cortex-M0, M0+, M3, M4

Cloud Computing

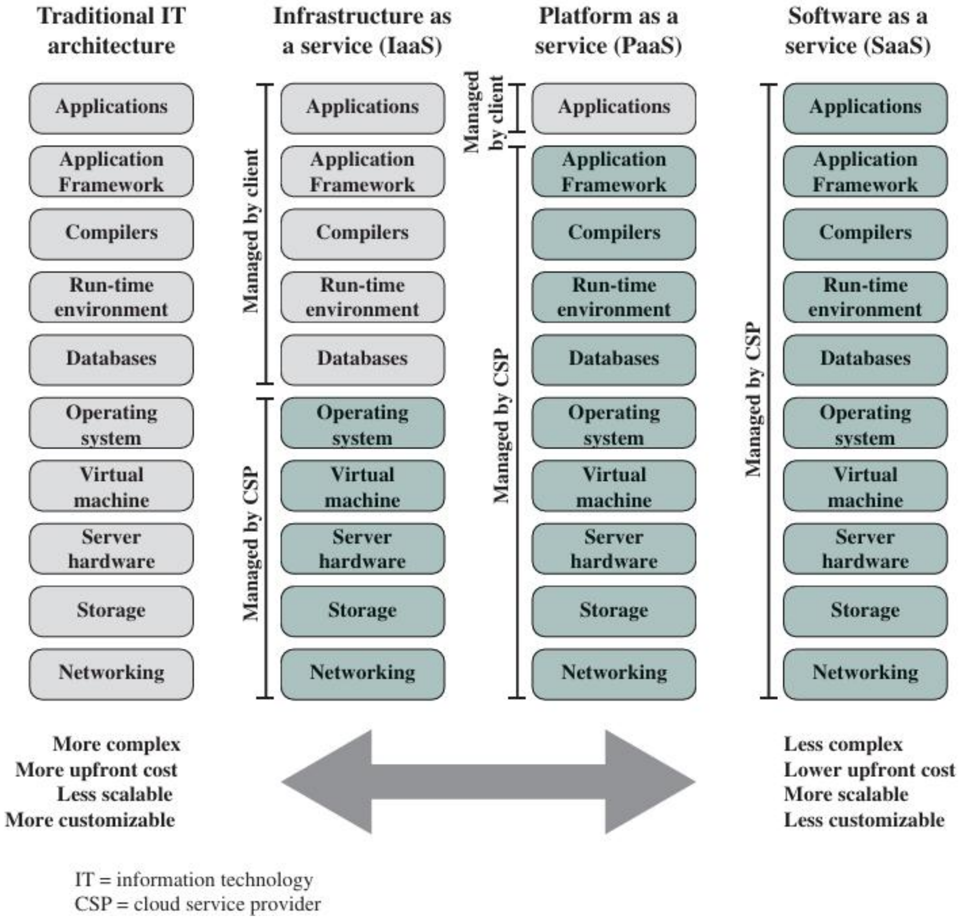
- A model for enabling ubiquitous, convenient, on- demand network access to a shared pool of configurable computing resources (e.g., networks, servers, storage, applications, and services) that can be rapidly provisioned and released with minimal management effort or service provider interaction *(NIST SP- 800-145)*



Key Characteristics of Cloud Computing

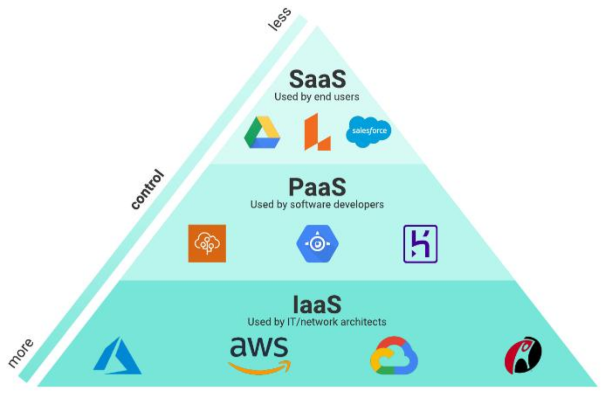
- On-Demand Self-Service – resources available instantly
- Broad Network Access – accessible from anywhere with internet
- Resource Pooling – multiple customers share the same infrastructure
- Rapid Elasticity – scale resources up/down quickly
- Measured Service – pay for what you use

Cloud Service Models



Examples

- **IaaS :-** AWS EC2, Google Compute Engine
- **PaaS :-** Google App Engine, Microsoft Azure App Service
- **SaaS :-** Gmail, Microsoft 365, Dropbox



Advantages And Disadvantages of Cloud Computing

Advantages:- Reduced hardware & maintenance costs
:- Scalability & flexibility
:- Accessible anywhere
:- Disaster recovery & backup options

Challenges:- Security and Privacy Concerns
:- Internet Dependency
:- Downtime/Service Outages
:- Limited control over resources
:- Hidden Costs
:- Compliance and Legal Issues

Key Takeaways

- Computer generations reflect advancements in hardware, software, and system capabilities.
- Intel x86 remains a cornerstone in desktop, laptop, and server computing due to backward compatibility.
- Embedded systems are specialized, efficient, and found in numerous everyday devices.
- ARM architecture dominates mobile and IoT markets due to its performance–power balance.
- Cloud computing offers scalable, on-demand access to shared resources with diverse service models.


Q
A


Thank You