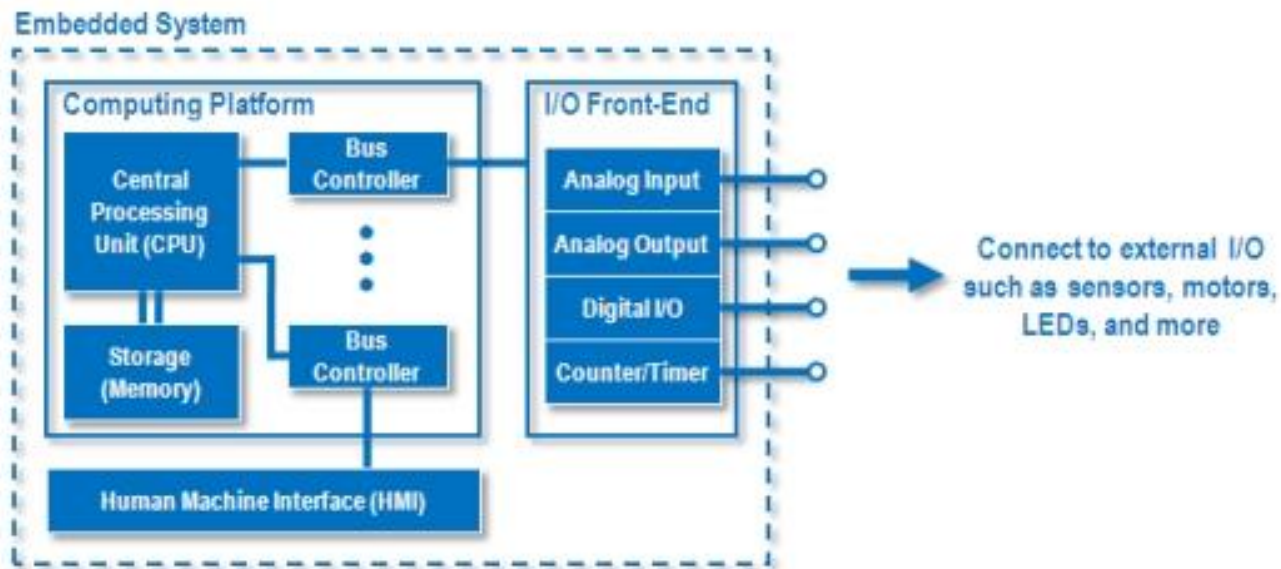


# CO3053 – Embedded Systems

## 1. What is Embedded System?

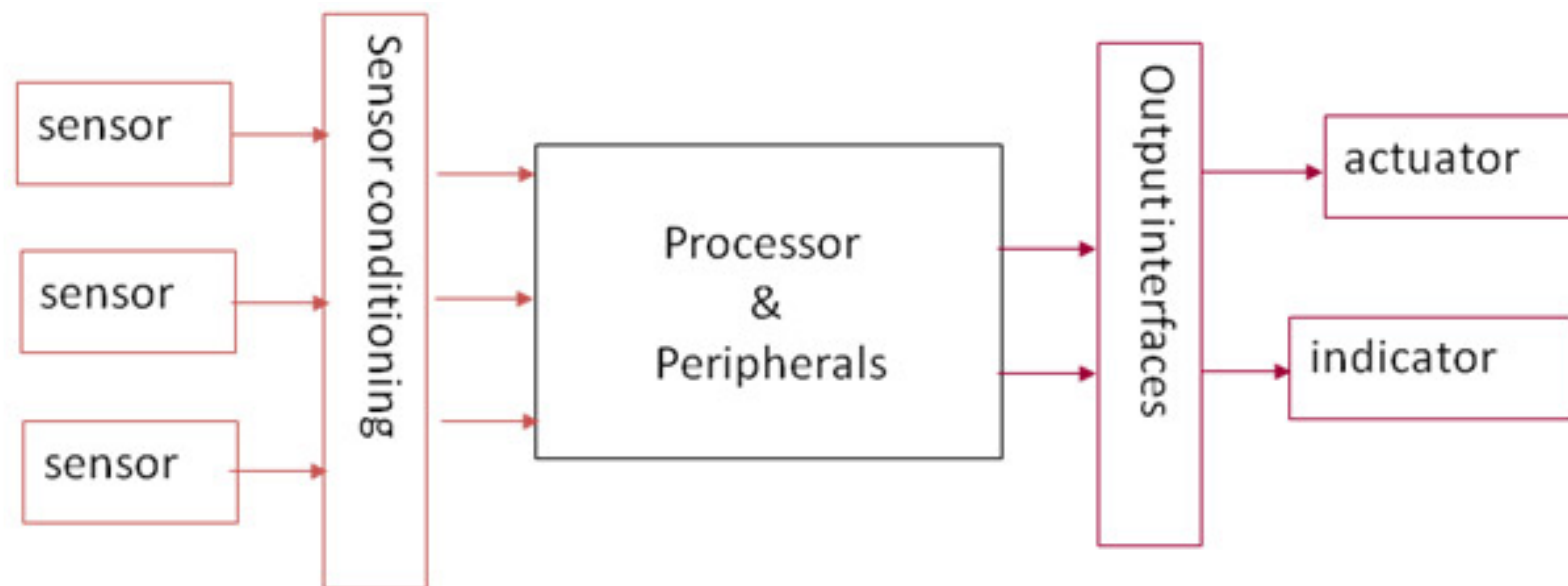


# Contents

- Definition and Concept
- Characteristic
- Design Challenge
- Design Metric

# What is Embedded System ?

- An embedded system is
  - **A computer system** that is **embedded** into **another larger system**.
  - Example: Cars, airplane, weapon, ...
- An embedded system is designed to perform a/some **dedicated task(s)**
  - Lighting control
  - Camera control



# Embedded System Applications

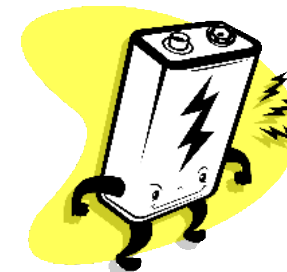
- Require very high performance

- Communications
- Multimedia
- Graphics



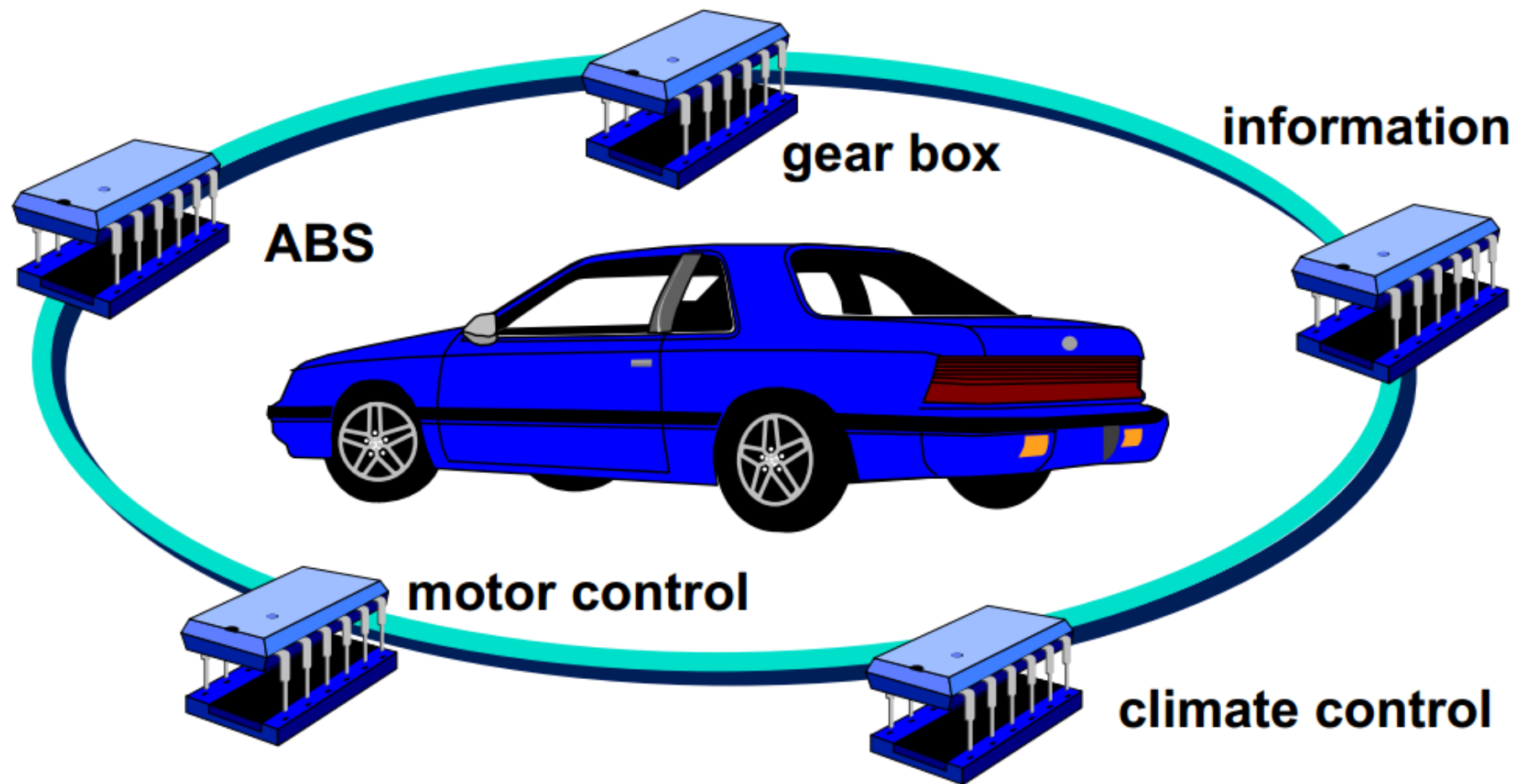
- Must also meet strict design goals

- Real-time performance
- Power/energy consumption
- Cost



# Example – Automotive

- Car as an integrated control, communication and information system.



# Example – Consumer Electronics

- MP3 audio, digital camera, home electronics

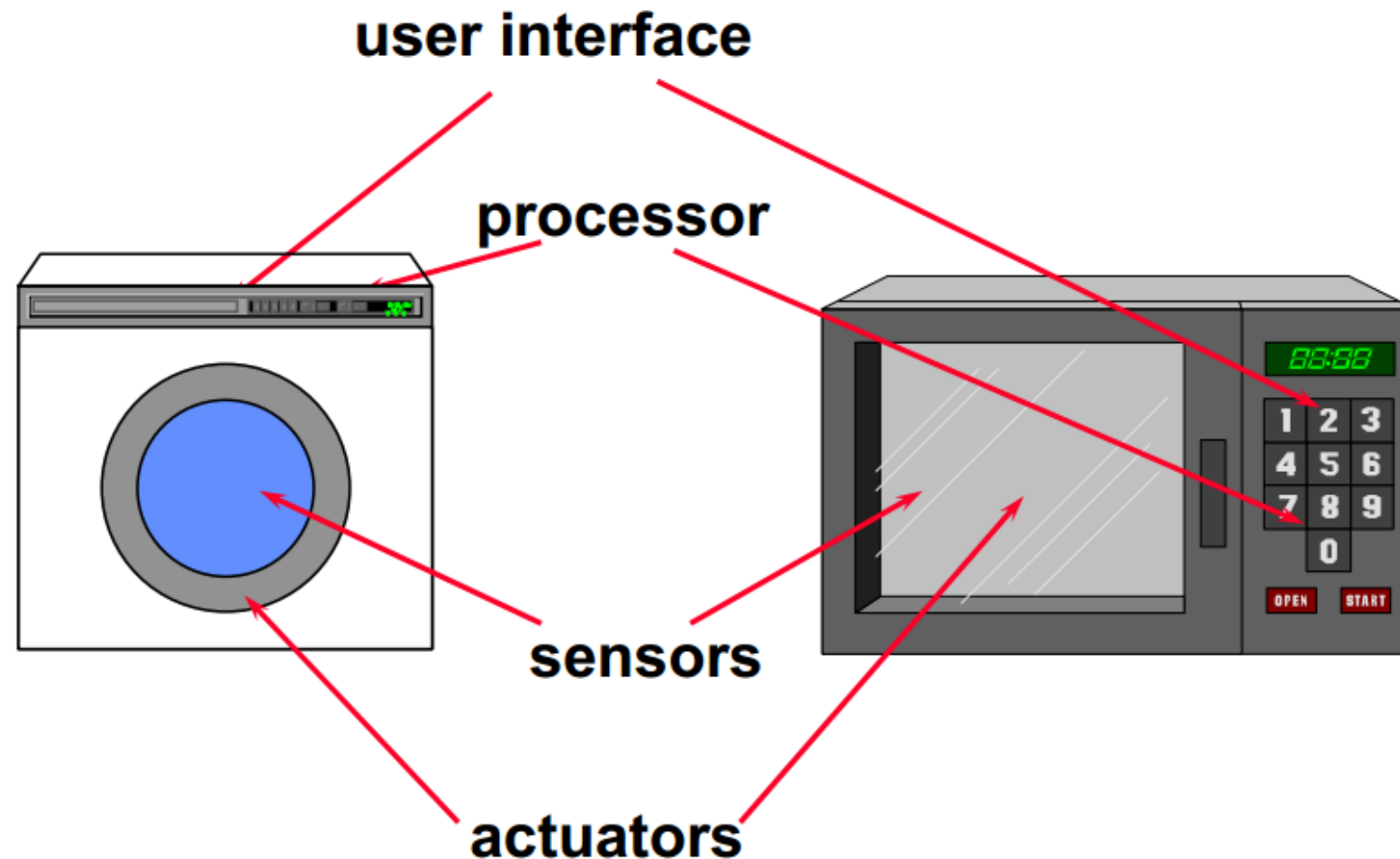
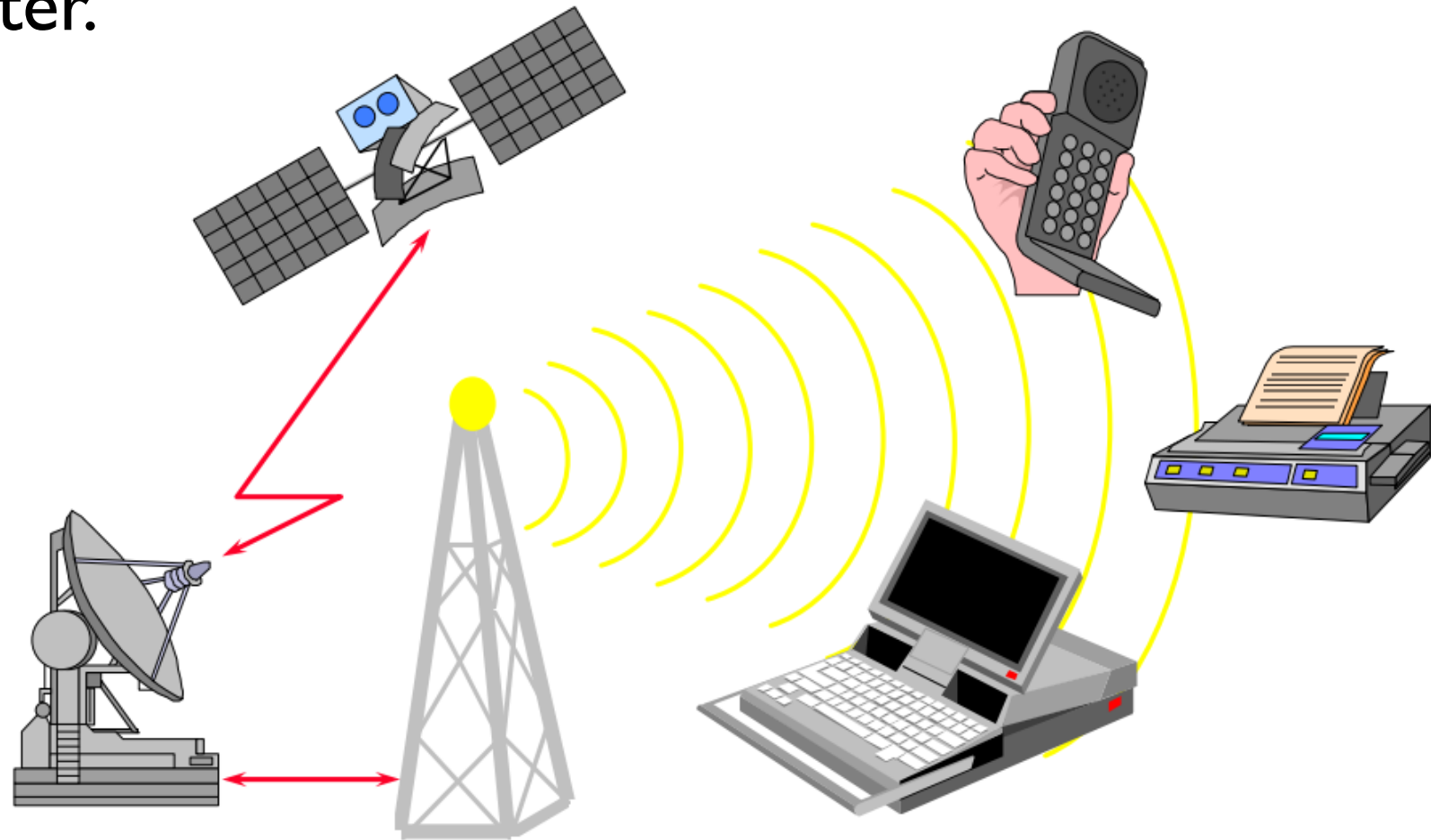


Image Source: Internet

# Example – Information Systems

- Wireless communication (mobile phone, wireless LAN, etc), end-user equipment, router.



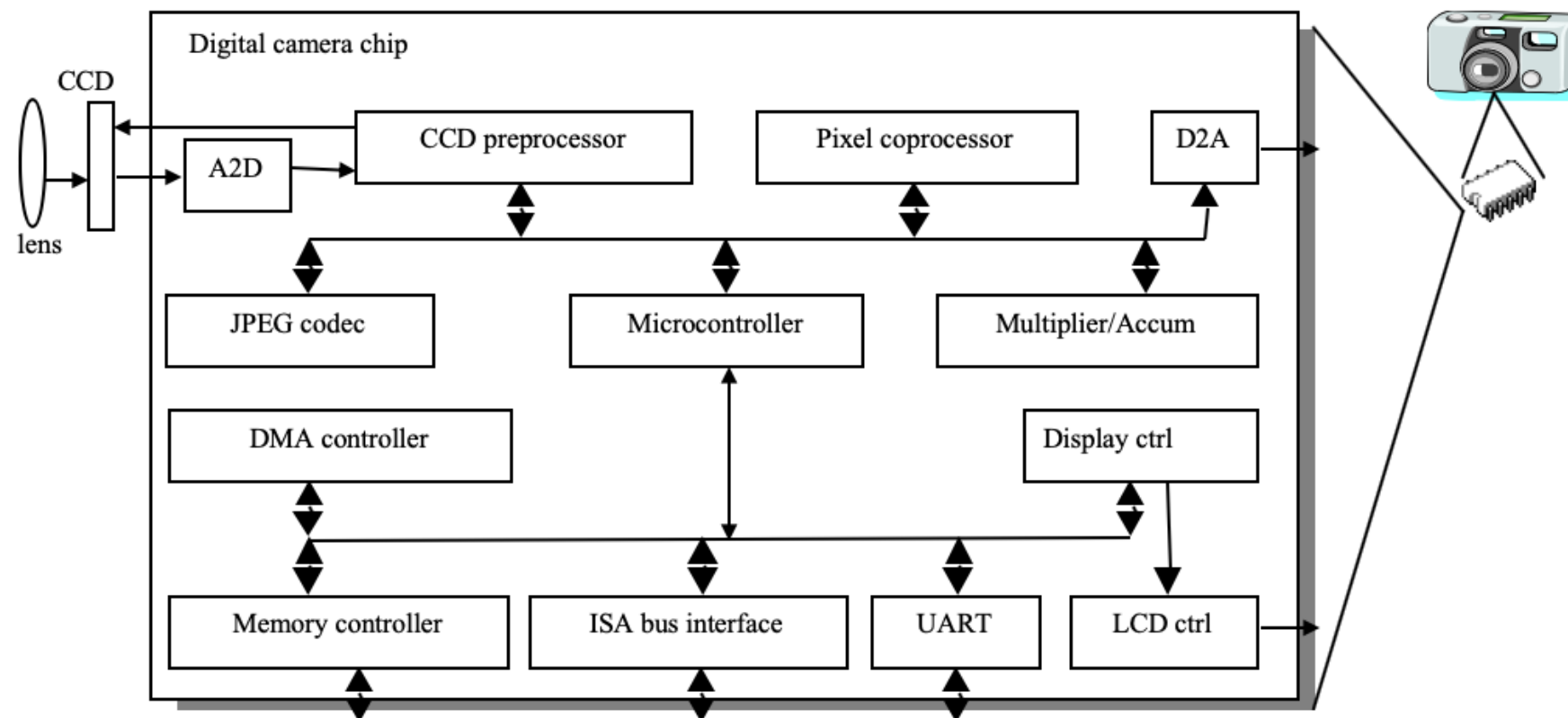
# Common Characteristics of Embedded Systems

- Single-functioned
  - Executes a single program, repeatedly
- Tightly-constrained
  - Low cost, low power, small, fast, etc.
- Reactive and real-time
  - Continually reacts to changes in the system's environment
  - Must compute certain results in real-time without delay





# An Embedded System Example



- Single-functioned -- **always a digital camera**
- Tightly-constrained -- **low cost, low power, small, fast**
- Reactive and real-time -- **only to a small extent**

# Embedded Systems – Design Challenge

- Obvious design goal
  - Construct an implementation with desired functionality
- Key design challenge:
  - Simultaneously optimize numerous design metrics
- Design metric
  - A measurable feature of a system's implementation
  - Optimizing design metrics is a key challenge



# Embedded Systems – Common Design Metrics

- **Unit cost**

- The monetary cost of manufacturing each copy of the system, excluding NRE cost

- **NRE cost** (Non-Recurring Engineering cost)

- The one-time monetary cost of designing the system

- **Size**

- The physical space required by the system

- **Performance**

- The execution time or throughput of the system

- **Power**

- The amount of power consumed by the system

- **Flexibility**

- The ability to change the functionality of the system without incurring heavy NRE cost



# Embedded Systems – Common Design Metrics

- **Time-to-prototype**
  - The time needed to build a working version of the system
- **Time-to-market**
  - The time required to develop a system to the point that it can be released and sold to customers
- **Maintainability**
  - The ability to modify the system after its initial release
- Correctness, Safety, CPU Technologies, Integration Level, Form Factor, Application Specific Hardware, User Interface, Connectivity, Security



# Embedded Systems – Other Design Metrics

- CPU Technologies
- Integration Level
- Form Factor
- Application Specific Hardware
- User Interface
- Connectivity
- Security



# The CPU

- **General purpose**

- 32-bit, 64-bit long time ago.
- Clock-rate reach several GHz.
- Included Floating Point Unit (FPU) and/or Graphic Processing Unit (GPU) with parallel processing capability.

- **Embedded System**

- Large range of compute capability.
- Still many 8-bit and 16-bit in small ES (PIC, AVR,...).
- Now, many systems requires 32-bit, 64-bit CPU (ARM, ATOM,...)

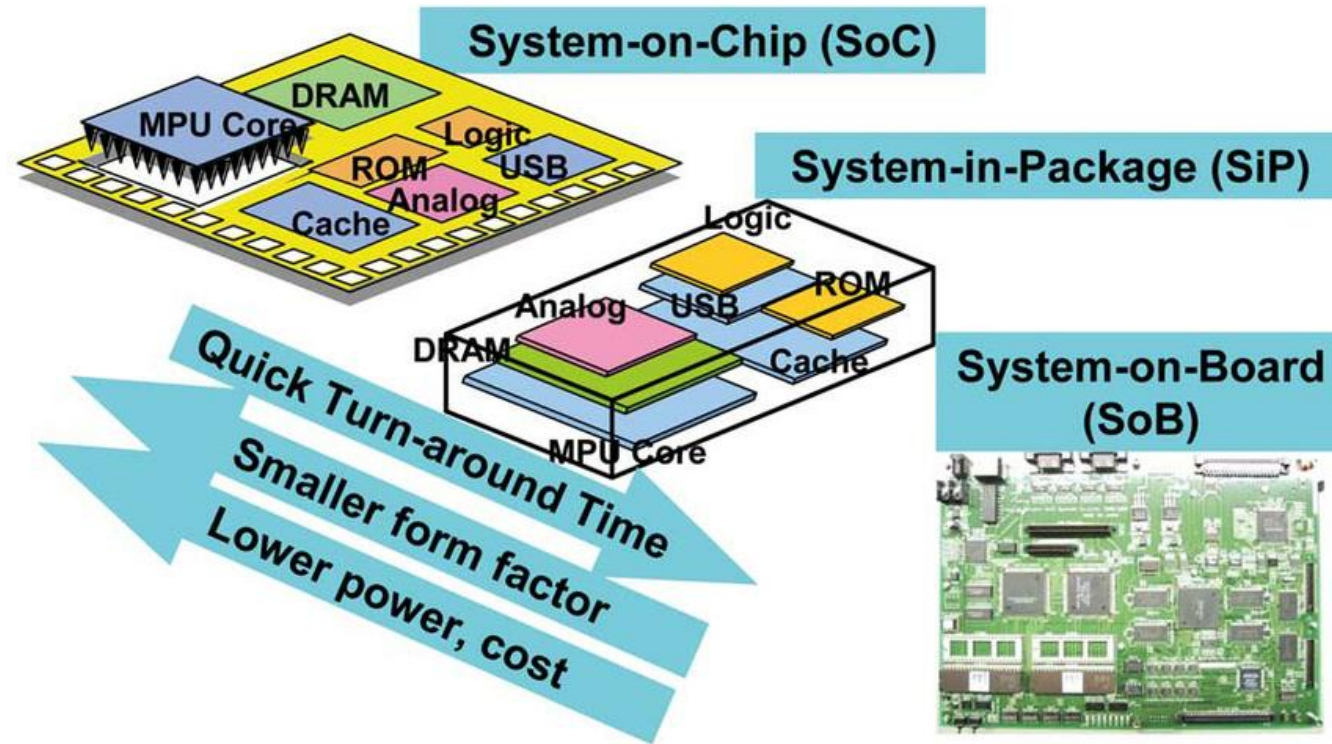
# Integration Level

- Early Embedded Systems

- Separate CPU + Peripherals + ...
- Cost highly

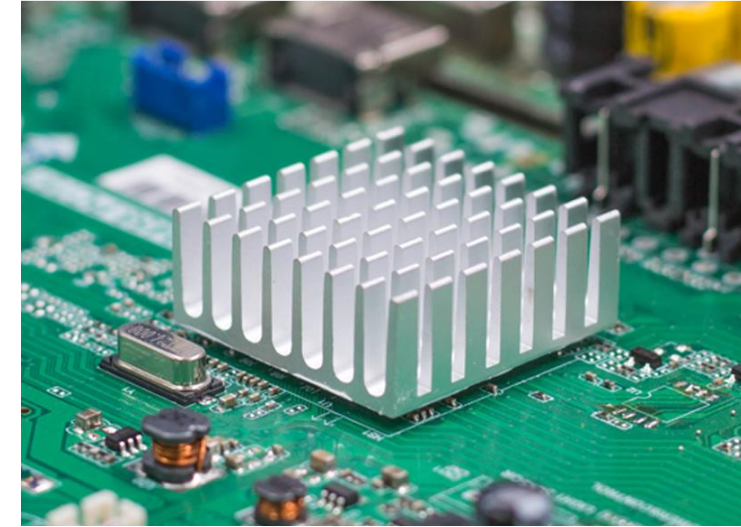
- SoC (System on a Chip)

- Many ICs are integrated into a single Chip
- High-density → Low-cost
- CPU + Memory + IO or CPU + Memory + ADC
- Many levels of integration



# Power Consumption

- Many ES rely on battery
  - Power consumption is very important
- Apply power optimization technique
  - Sleep mode
- Heat is also very important
  - Heat sink must be efficient
  - E.g. Cell phone





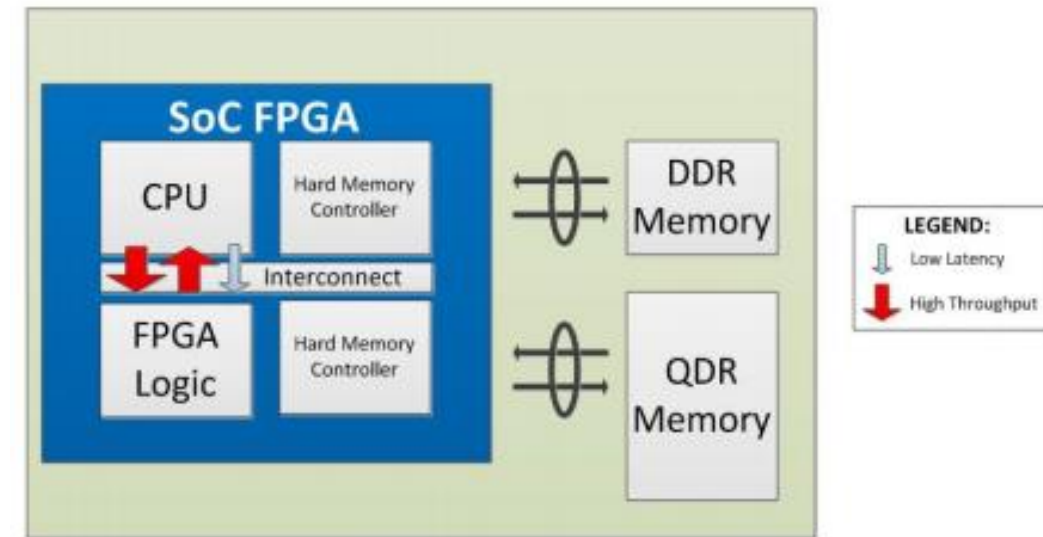
# Form Factor & Expansion

- Size, pin-out, connector.
- Define some standard (Ex COM) help software development, hardware expansion
- Sometime for expansion, resources are reserved (DRAM, Flash). But these have trade-off and need to be consider carefully.



# Application Specific Hardware

- Can any SoC suit the user requirements?
- Adding other ASIC or FPGA: trade-off
  - ASIC can show better performance
  - But FPGA can show the flexibility
- Future ES: **SoC + FPGA**
- Many parts of ES need **certification**
  - E.g. **radio frequency** need to be licensed (US, Japan,...)
- Consider to use pre-certified module



# User Interface

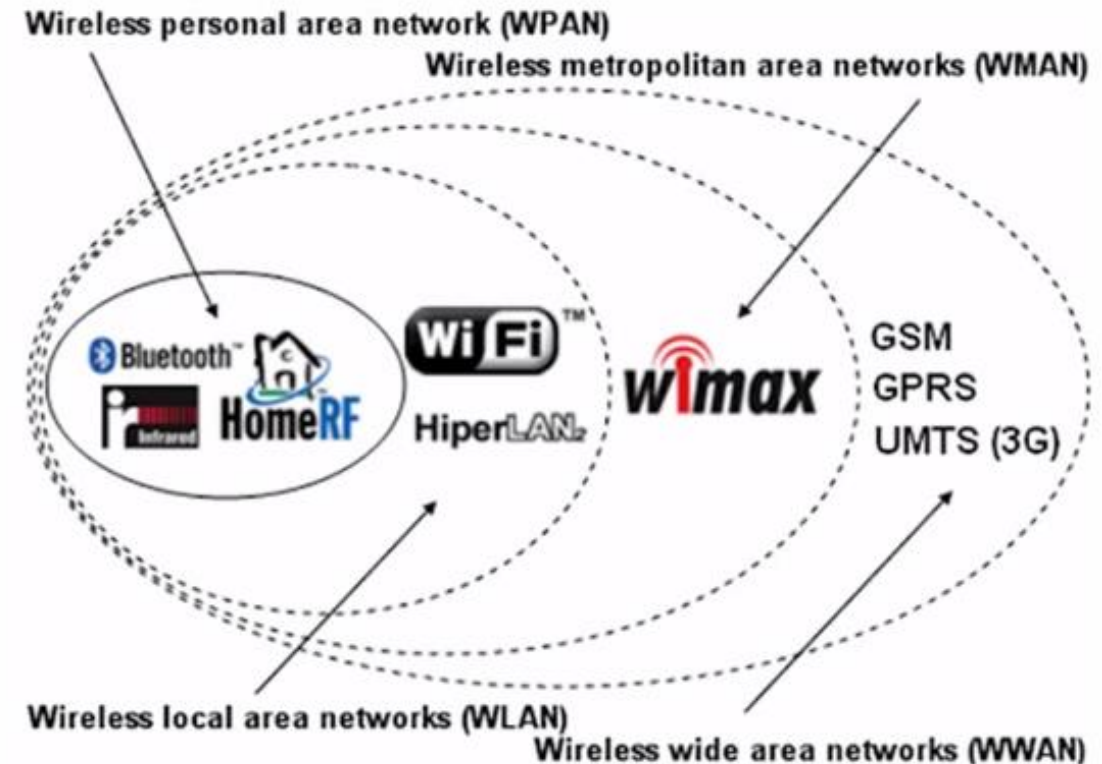
- Many ES is interactive
  - Headed provide the display
    - E.g. touch screen. The appearance of application
  - Headless: no display.
    - Use console, key/button, or via web page



Image Source: Internet

# Connectivity and Security

- For current and future ES, connectivity is very important
  - Wifi, Xbee, Bluetooth, LoRa
  - 3G/4G/5G
  - ...
- **Ubiquitous:** every device can connect each other
- When the **connectivity is easy**, the **security become critical**



# Further Reading

- Audio Lecture

- [https://lectures.tik.ee.ethz.ch/es/recordings/ES\\_I/ES\\_I.html](https://lectures.tik.ee.ethz.ch/es/recordings/ES_I/ES_I.html)