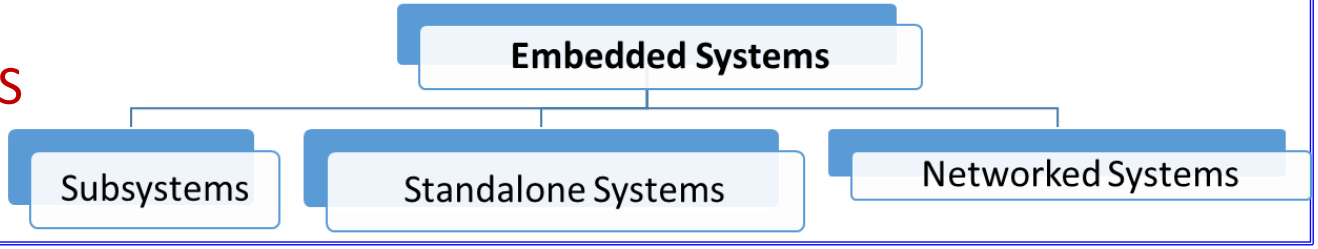
1. What is an Embedded System?

Embedded System

- What is a Embedded System (EESS)? Special purpose custom built to serve a specific purpose.
- Special purpose vs General purpose.

Special purpose	General purpose
Made to serve **one** particular function.	Made to serve **more than one** purposes.
Enough for some tasks (Cheaper)	Expensive for a specific purpose
Examples: • Calculator • Washing machine • Fridge • Microwave • …	Examples: • Personal computer • Smartphone • Tablet • Laptop • …

- Clasification of EESS

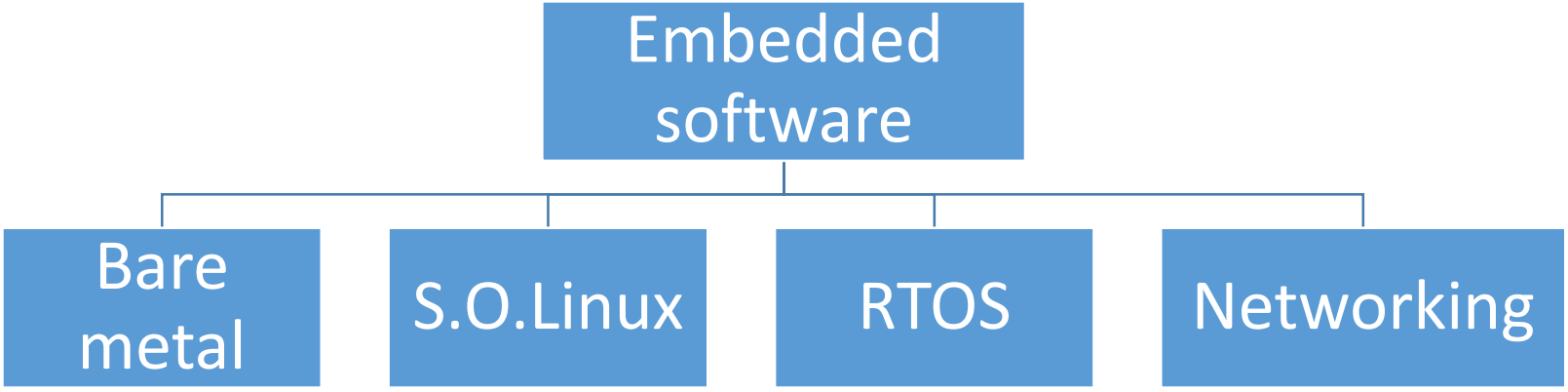


Clasification of the Embedded System

Subsystems	Standalone Systems	Networked Systems
Part of a larger system	Performs its functions independently	Separate electronic devices performing a collective function
Placed inside something bigger	No embeddedd into something bigger	When spanned througout the globe is IoT
The name comes from this idea	Same components and technique as the subsystem	
Examples: • Dashboard in a car • Displays is microvawe, oven, washing machine… • Computer parts: keyboard, mouse…	Examples: • USB pen • Camera • Digital watch	Examples: • Home automation

Embedded software: Clasification

Software that runs on devices like microwave ovens, Xbox controllers, blood pressure monitors, bluetooth headphones, smartwatches, and other such devices that has custom hardware.



Embedded software: Types

<table>
<tr><th>Bare Metal</th><th>Linux</th><th>RTOS</th><th>Networking</th></tr>
<tr><td>The simplest type which does its job without an operating system</td><td rowspan="3">Operating system to suit their particular needs by keeping just the necessary parts, keeping the size small. Then on top of the kernel, the application code can be developed to make an embedded device.</td><td>React to events within the specified time constraints. Execute code within the specified deadline.</td><td>2 modules talk to each other.</td></tr>
<tr><td>From drivers to schedulers must be designed</td><td rowspan="2">Hard real-time systems are those that a late answer is a wrong answer.</td><td></td></tr>
<tr><td>Much more responsive and efficient</td><td rowspan="3">This reduces the development time and costs compared to building your own embedded operating system.</td><td rowspan="2">Soft real-time systems a late answer is an inaccurate answer but still useful.</td></tr>
<tr><td>Runs by manipulating and reading the registers.</td><td></td></tr>
<tr><td></td><td></td><td>Multitasking</td><td></td></tr>
</table>

Embedded software: Needed skills

Bare Metal	Linux	RTOS	Networking
Good mastery of C and C++ programming Debugging skills using oscilloscope and logic analyzers Version management software like Git			
Microcontroller and its peripherals	Linux kernel	Task Management	Develop and manage custom network stacks
Ability to read data sheets and working with registers and manipulating bit-level data	Operating System concepts: Scheduling, queues, power and memory management	Heap memory management	Basic Networking: Ethernet, Wifi, Bluetooth, Zigbee and Zwave protocols for home automation.
Assembly programming	Linux driver development	Queue Management Software timer management	ANT+ protocols for the health industry
Software engineering principles	Linux application development	Interrupt Management and Resource management	Networking devices like switches, routers, IDS, IPS, and firewalls

Key Features/Constraints of an Embedded System

➢ User Interface

- A GUI (graphical user interface).
- GUI could be very simple like just push buttons

and in some systems it could be really complex like graphical LCDs.

➢ Processing power

- Millions of instructions per second (MIPS)
- Register width: range from 8 to 64

➢ Memory

- To hold the executable software and the data it manipulates
- The amount of memory required can also affect the processor selection.

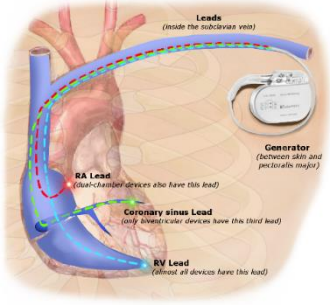
Key Features/Constraints of an Embedded System

➢ Real Time Operation

- It is a subclass of embedded systems
- They can react to events happening in real time

and work according to them.

- Frequently connected to physical environment through sensors and actuators.
- It has the ability to make calculations or decisions in a timely manner. They have a deadline for completion those actions.
- These systems are divided into:
  - Hard real-time: Severe consequences
  - Soft real-time: Not severe consequences (microwave)
- Key software → Polling or interrupts



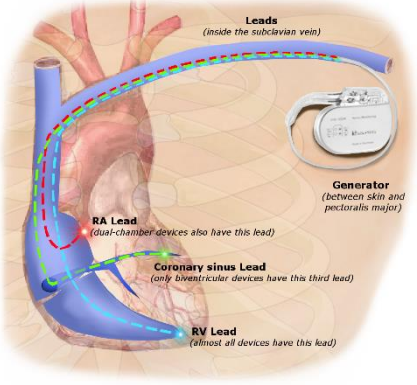
Key Features/Constraints of an Embedded System

Cost

- The design process is costly but once a system is designed, customized and produced in bulk, overall cost becomes minimum.
- Examples: toys, watches, smartphones, etc.

Size

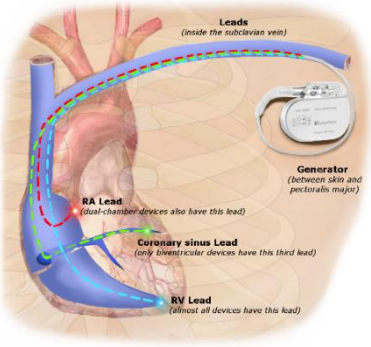
- The size should be small and it is done by adding more functionality in a single chip so that the need for external parts is reduced.



Key Features/Constraints of an Embedded System

➢ Power consumption

- Isolated systems running for a very long time → critical!!!
- Low power consumption leads to other characteristics: less heat, smaller batteries, less weight, smaller size,…
- Cut down consumption techniques (processors)
  - Active/Power-down Modes
  - Code-size efficient:
    - Limited resources
    - Algorithm (polling)
    - A sensor activated although it is not operating



Key Features/Constraints of an Embedded System

➢Reliability

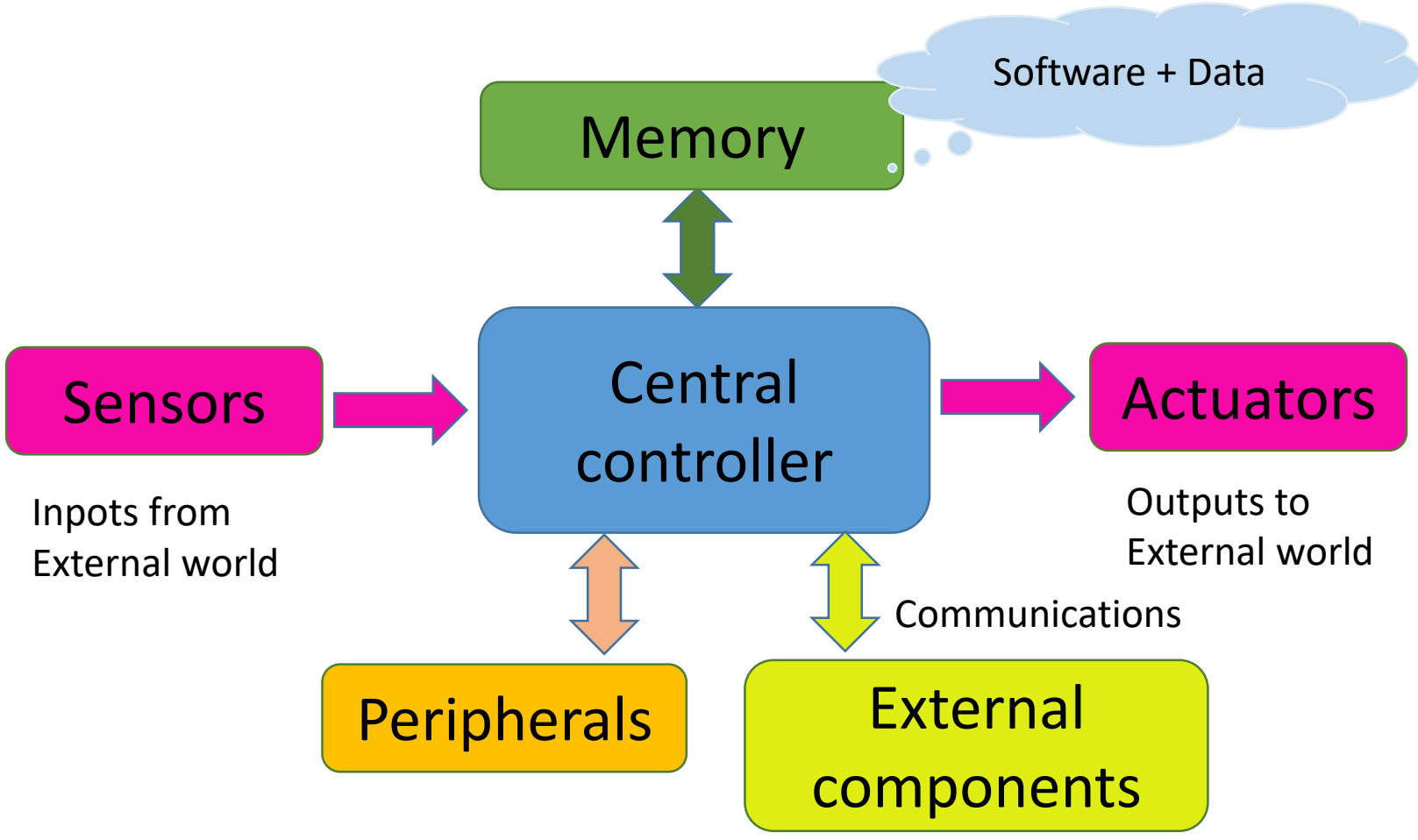
- Maintainability: it must work correctly after an error occurs.
- Availability: probability of system working at anytime.
- Safety: no harm to be caused.
- Security: confidential and authentic communication.
- Temporary external hazards: EMI, Cosmic ray, High Temperature
- Fault-tolerance
  - Redundancy (HW / SW).
    - RAID, Power-supply, Memory, Processor (Lockstep)
    - Drawbacks → High cost

https://en.wikipedia.org/wiki/Lockstep_(computing)

Key Features/Constraints of an Embedded System

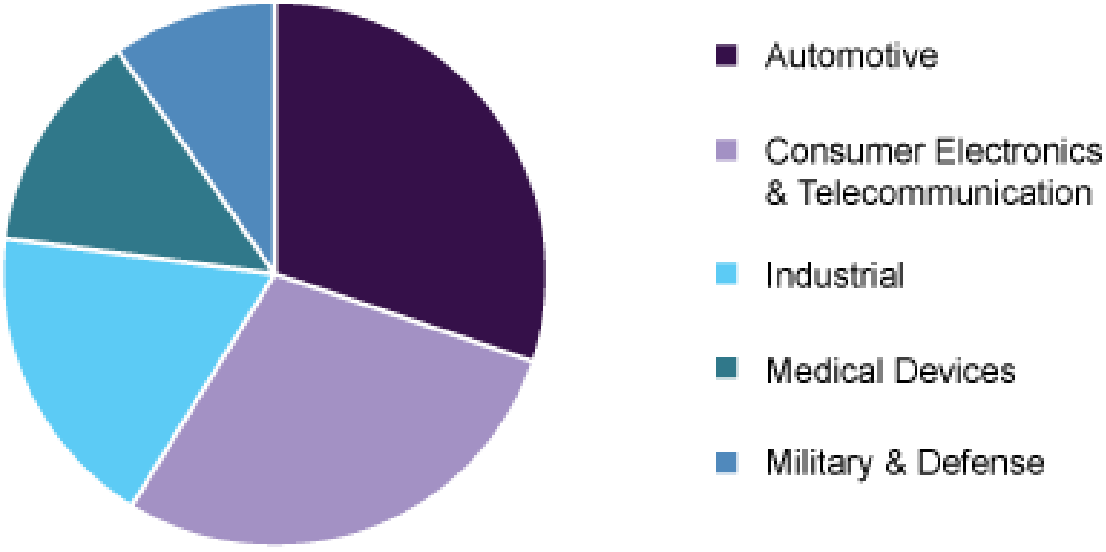
- Reliability
  - Radiation and electromagnetic noise immunity
    - Electromagnetic interference (EMI) : Electromagnetic energy which affects the functioning of an electronic device.
      - Sources of EMI:
        - Electrical storms and solar radiation
        - Another electronic device or electrical system: cellphones, welders, motors and LED screens
    - Electromagnetic compatibility (EMC):
      - Evaluating how a device will react when exposed to electromagnetic energy (immunity).
      - Measuring the amount of EMI generated by the device's internal electrical systems (emissions).

Embedded System's Components



Fields of application

Global microcontroller market share, by application, 2020 (%)



Source: www.grandviewresearch.com

Examples of embedded systems

- **CONSUMER ELECTRONICS:** Cameras, etc
- **HOUSE HOLD APPLIANCES:** Television, DVD Players, washing machine, fridge, microwave oven, etc.
- **HOME AUTOMATION AND SECURITY SYSTEMS:** Air conditioners, closed circuit television, cameras, fire alarms etc
- **AUTOMOTIVE INDUSTRY:** Anti lock breaking system(ABS) engine control, automatic navigation systems etc
- **TELECOM:** Cellular telephones, telephone switches, etc.
- **COMPUTER PERIPHERALS:** Printers, Scanners, fax machines
- **COMPUTER NETWORKING SYSTEMS:** Network routers, switches, hubs, firewalls etc
- **HEALTH CARE:** Different kinds of scanners, EEG , ECG machines, etc.
- **MEASUREMENT AND INSTRUMENTATION:** Digital multimeters, logic analyzers,…
- **BANKING AND RETAIL:** Automatic teller machines(ATM) and currency counters, point of sales(POS),…
- **CARD READERS:** Barcode, smart card readers, hand held devices,…
- **WEARABLES:** smartwatch**, ….**
- **AEROSPACE:** Mars Rovers,…