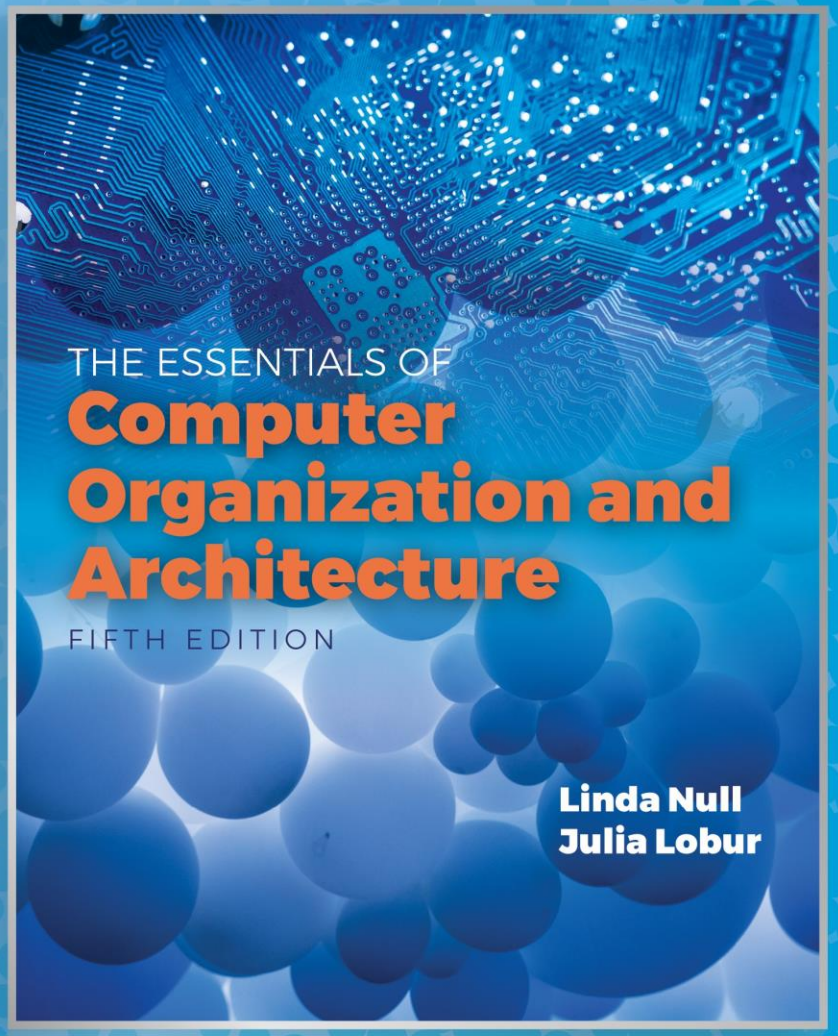
This is the first lecture of Chapter 10

Chapter 10

Topics in Embedded Systems (A)



Objectives

- Understand the ways in which embedded systems differ from general purpose systems.
- Be able to describe the processes and practices of embedded hardware design.
- Understand key concepts and tools for embedded software development.

10.1 Introduction (1 of 2)

- Embedded systems are real computer systems that support the operation of a device (or machine) that usually is not a computer.
- The user of the embedded system is rarely aware of its existence within the device.
- These systems are all around us. They are in watches, automobiles, coffeepots, TVs, telephones, aircraft, and just about any “intelligent” device that reacts to people or its environment.

10.1 Introduction (2 of 2)

- Embedded systems are different from general-purpose systems in several important ways. Some key differences are:
  - Embedded systems are resource constrained. Utilization of memory and power are critical. The economy of hardware and software is often paramount, and can affect design decisions.
  - Partitioning of hardware and software is fluid.
  - Embedded systems programmers must understand every detail about the hardware.
  - Signal timing and event handling are crucial.

10.2 An Overview Embedded Hardware (1 of 22)

- We will classify embedded hardware according to the extent to which it is adapted or adaptable by the people who program and install the system into the device that it supports.
- Accordingly, we say that embedded hardware falls into categories of:
  - Off-the-shelf
  - Configurable
  - Fully-customized

Note: There are many other taxonomies. This one is convenient for our purposes.

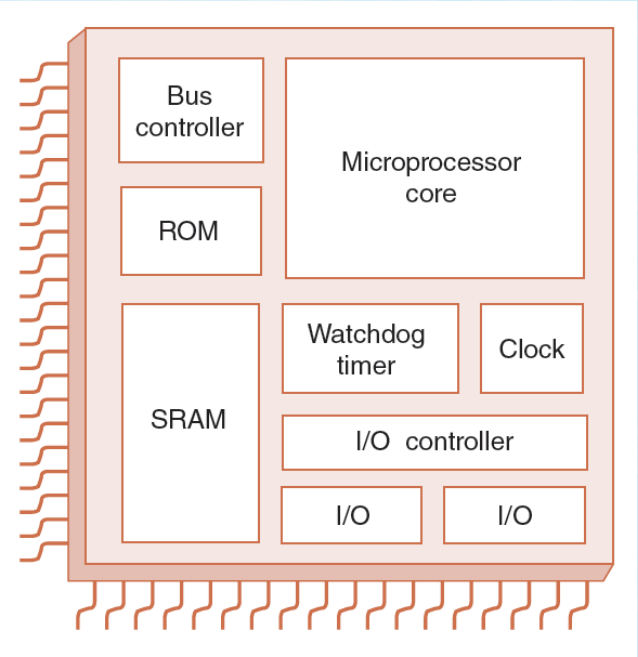
10.2 An Overview Embedded Hardware (2 of 22)

10.2.1 Off-the-shelf Hardware

- Using off-the-shelf hardware, minimal hardware customization possible.
  - Perhaps add memory or peripherals. The internal wiring stays the same.
- The most common off-the-shelf hardware is the microcontroller.
  - Microcontrollers are often derivatives of “old” PC technology. They are inexpensive because development costs were recouped long ago.
  - There are thousands of different microcontrollers.

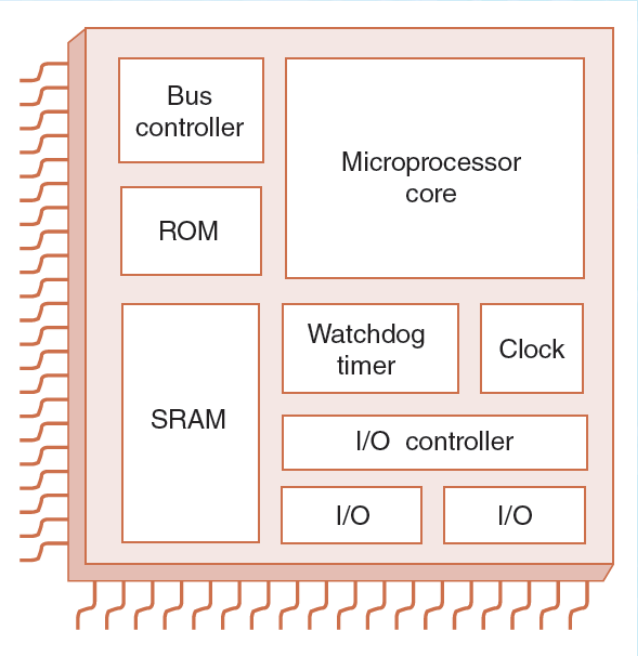
10.2 An Overview Embedded Hardware (3 of 22)

- Example: Microcontrollers are Motorola's 68HC12, Intel's 8051, Microchip's 16F84A, and the PIC family.
- A simplified block diagram of a microcontroller is shown at the right.



10.2 An Overview Embedded Hardware (4 of 22)

- We have seen all of these components before except for the watchdog timer.
- A watchdog timer helps guard against system hangs by continually checking for liveness.
- Watchdog timers are not used in all microcontrollers.



10.2 An Overview Embedded Hardware (5 of 22)

- For some applications, microcontrollers are too limited in their functionality.
- Systems-on-a-chip (SOCs) are whole computer systems—including all supporting circuits—that are etched on a single die.
  - Alternatively, separate chips are needed to provide the same services.
  - The additional chips are costly and consume power and space.
  - The advantages of SOCs is that they are faster, smaller, more reliable, and less power consumption.

10.2 An Overview Embedded Hardware (6 of 22)

- Semi-custom systems-on-a-chip can be fabricated whenever a suitable off-the-shelf SOC is unavailable.
- The chip mask is created using blocks of pre-designed, pretested intellectual property (IP) circuits.
- The semi-custom approach is costly. To save money, off-the-shelf SOCs are preferred, even when their functionality is not an exact fit for the application.

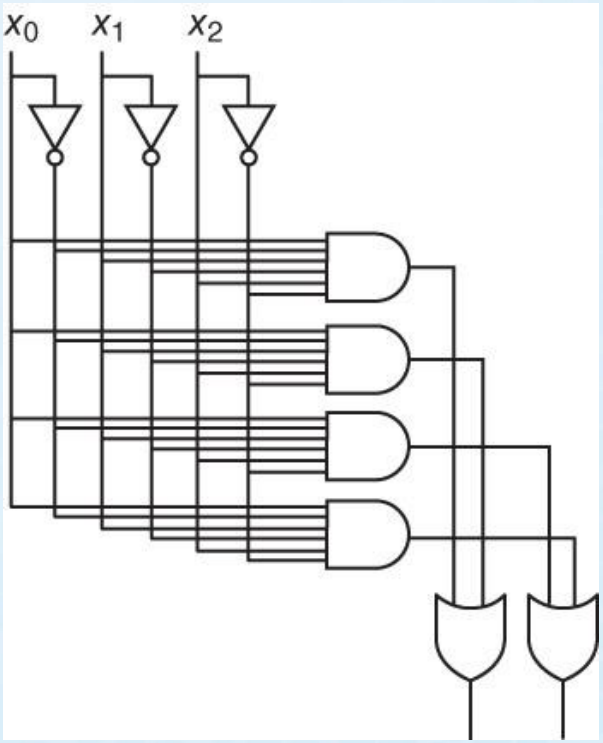
10.2 An Overview Embedded Hardware (7 of 22)

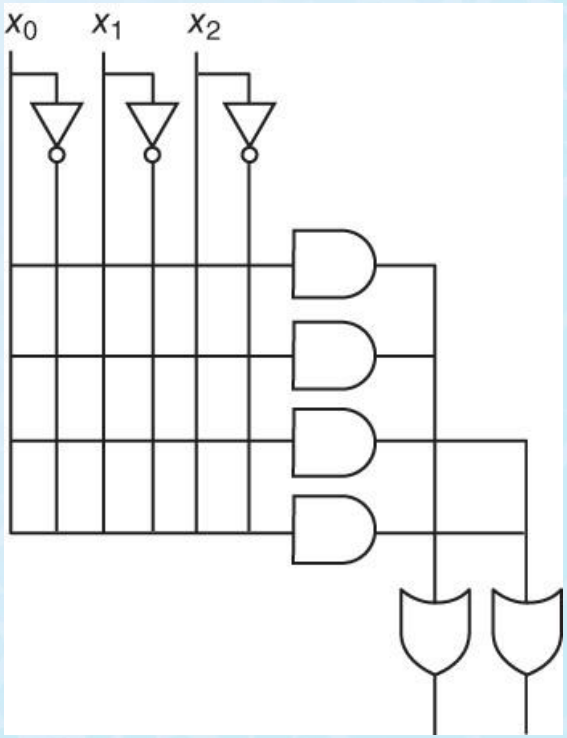
10.2.2 Configurable Hardware

- Programmable logic devices (PLDs) are configurable devices in which the behavior of the circuits can be changed to suit the needs of an application.
  - Programmable array logic (PAL) chips consist of programmable AND gates connected to a set of fixed OR gates.
  - Programmable logic array (PLA) chips consist of programmable AND gates connected through programmable OR gates.

10.2 An Overview Embedded Hardware (8 of 22)

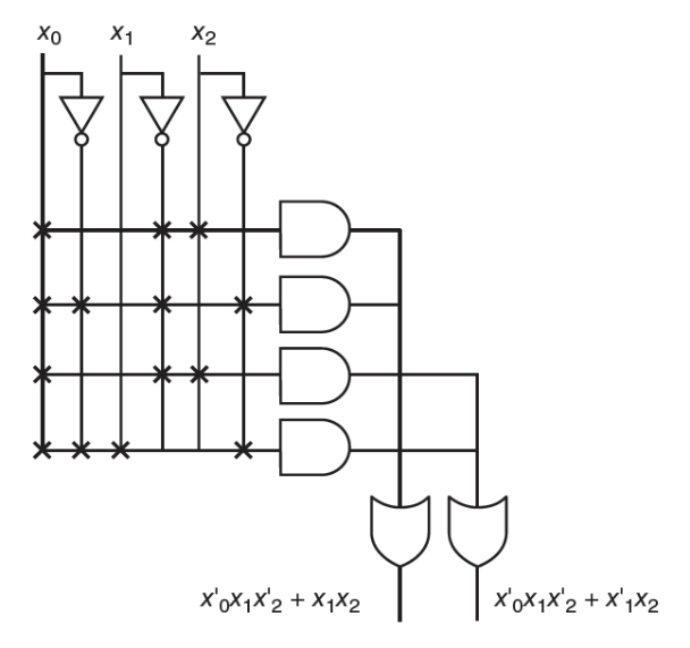
- Logic diagram for a PAL: Detailed and Simplified

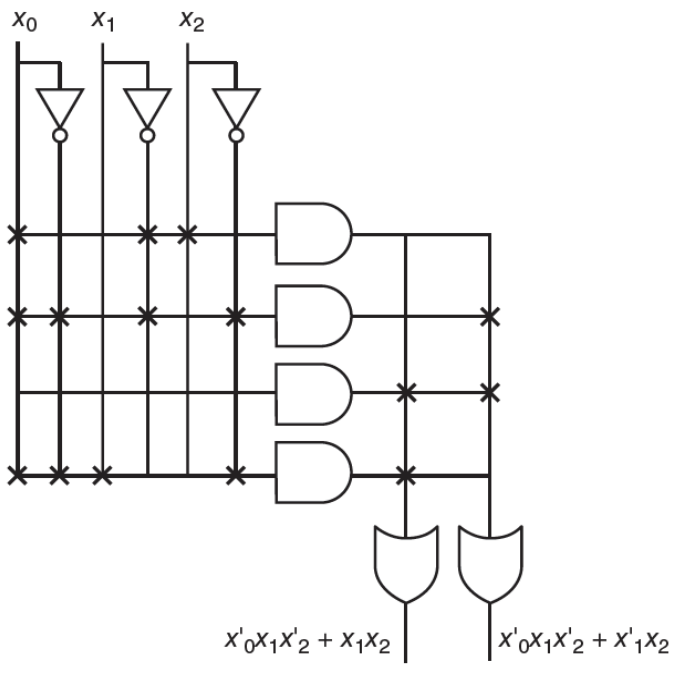




10.2 An Overview Embedded Hardware (8 of 22)

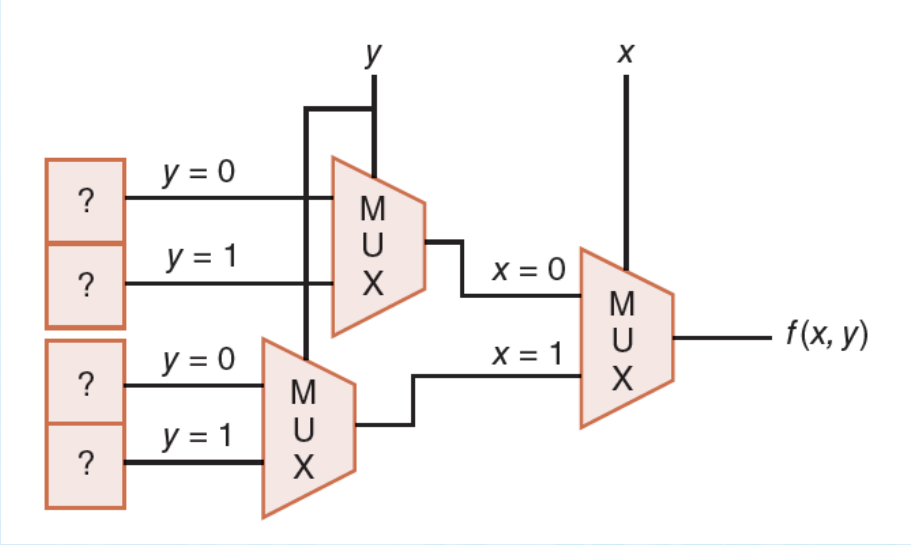
- A programmed PAL and a programmed PLA:





10.2 An Overview Embedded Hardware (9 of 22)

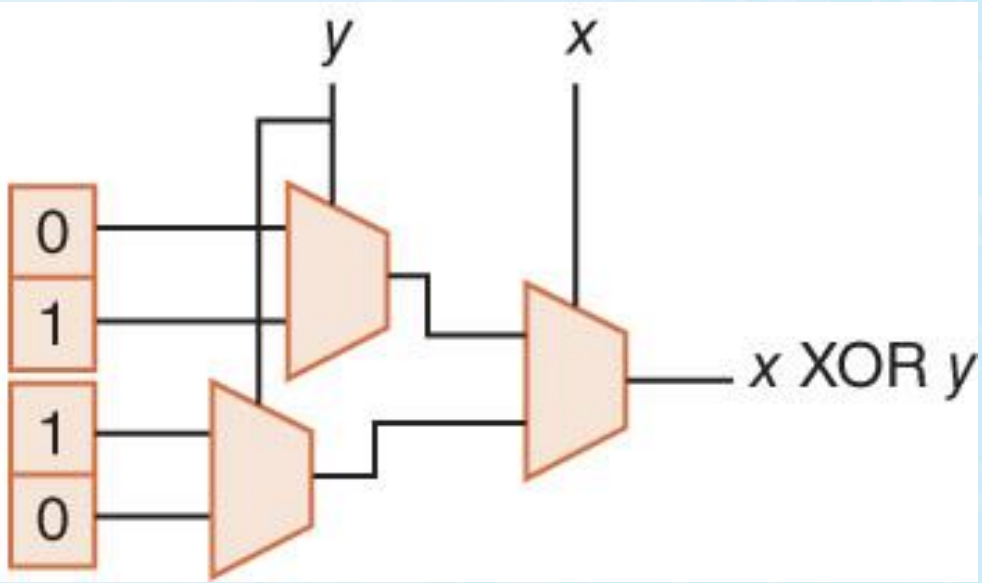
- The behavior of field programmable gate arrays (FPGAs) is controlled through values stored in memory lookup tables rather than by changing connections between logic elements.



10.2 An Overview Embedded Hardware (10 of 22)

- Truth tables are entered directly into FPGA memory.

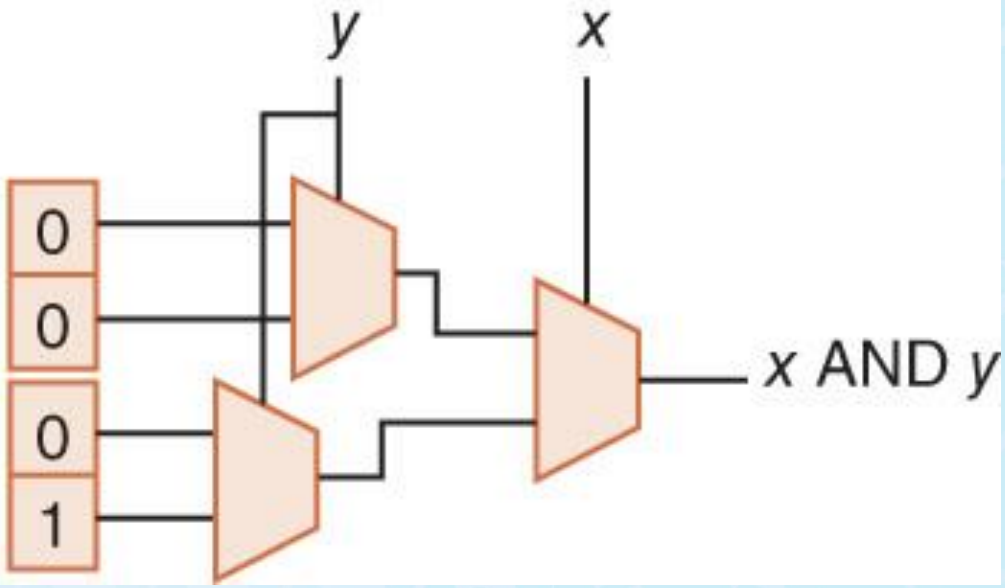
x	y	x XOR y
0	0	0
0	1	1
1	0	1
1	1	0



10.2 An Overview Embedded Hardware (10 of 22)

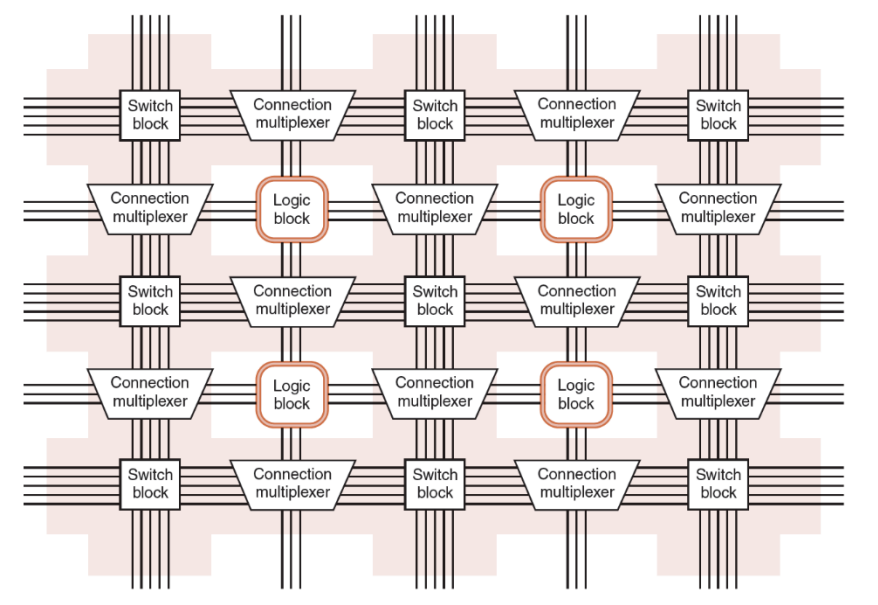
- Truth tables are entered directly into FPGA memory.

x	y	x AND y
0	0	0
0	1	0
1	0	0
1	1	1



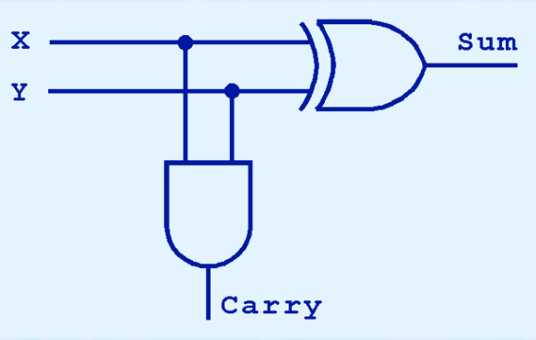
10.2 An Overview Embedded Hardware (11 of 22)

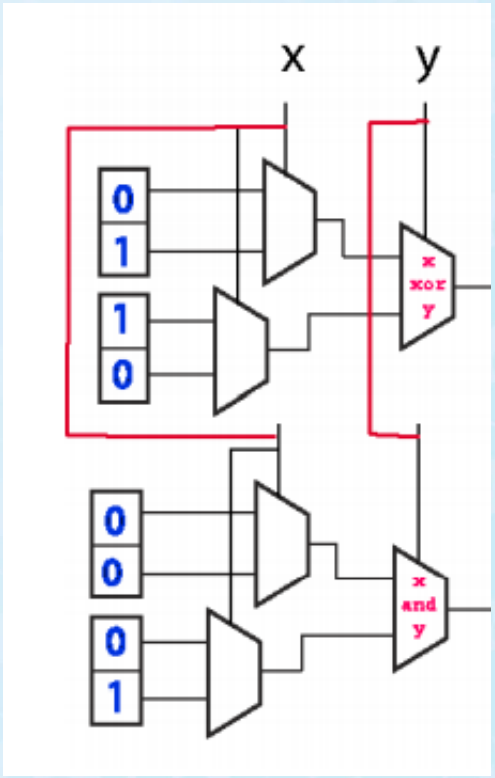
- FPGAs typically consist of blocks of logic elements interconnected by switches and multiplexers in an “island” configuration.



Implement a Half Adder Using FPGA

Inputs		Outputs	
X	Y	Sum	Carry
0	0	0	0
0	1	1	0
1	0	1	0
1	1	0	1





10.2 An Overview Embedded Hardware (12 of 22)

10.2.3 Custom-Designed Hardware

- When:
  - Off-the-shelf microcontrollers and SOCs do not have sufficient functionality for the task at hand...
  - Or off-the-shelf microcontrollers and SOCs have too much functionality, with the excess consuming resources needlessly…
  - And a semi-custom chip cannot be economically fabricated from commercially available IP designs...
  - And PLDs are too expensive or too slow…
- The only option left is to design an application-specific integrated circuit (ASIC) from scratch.