Course: Embedded Systems Design [Professional Core – 1]

Handout (Introduction)

Faculty:

Mr. N Abid Ali Khan

Assistant Professor – Dept Of ECE

Vasavi College Of Engineering (Autonomous)

abid.Khan@staff.Vce.Ac.In|8897727547

Dept of ECE, ME ECE (ESVLSID) Sem – 1st Acad. Year 2020-2021

Date: 04-01-2021

Definition of Embedded System

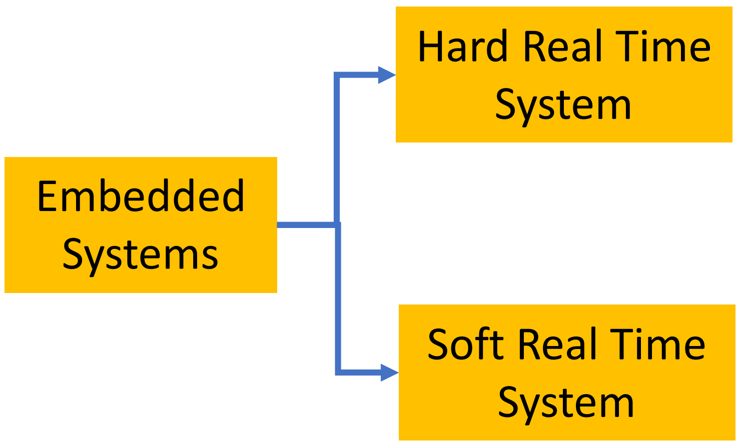
From Customer Point-of-view:

"Any electronic system with a CPU in it that does a specific (dedicated) task (job) under the environmental conditions".

From Designer Perspective:

"A Computer which is built inside the system that does a specific task; but, the final end user does not know that the system is being computing"

Embedded System Classifications (Based on Real-Time Responsiveness)



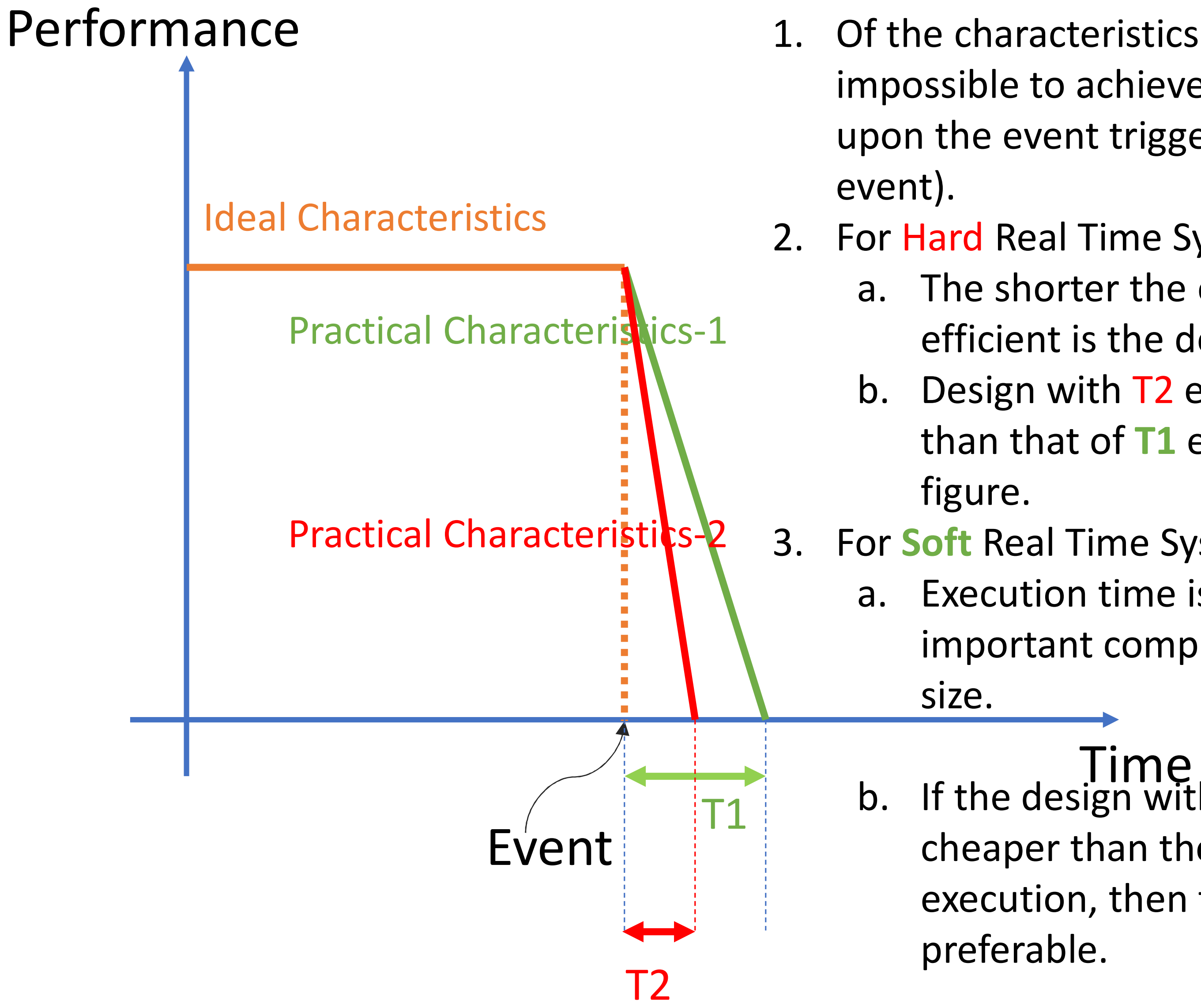
Hard Real Time System:

1. Extremely Tight Timing Deadlines
2. Examples include Life Control Applications
   a. Guided Missile electronic Control Sys.
   b. Airbag System
3. Execution time is important than other parameters

Soft Real Time System:

1. No Tight Timing Deadlines (Compromisable Delays)
2. Examples include Non-Life Control Applications
   a. Bus Ticketing machine
   b. Random blood glucose measurement unit
3. Product cost and Size are important than the execution time

Ideal & Practical Characteristics of Hard & Soft Real-Time Systems:

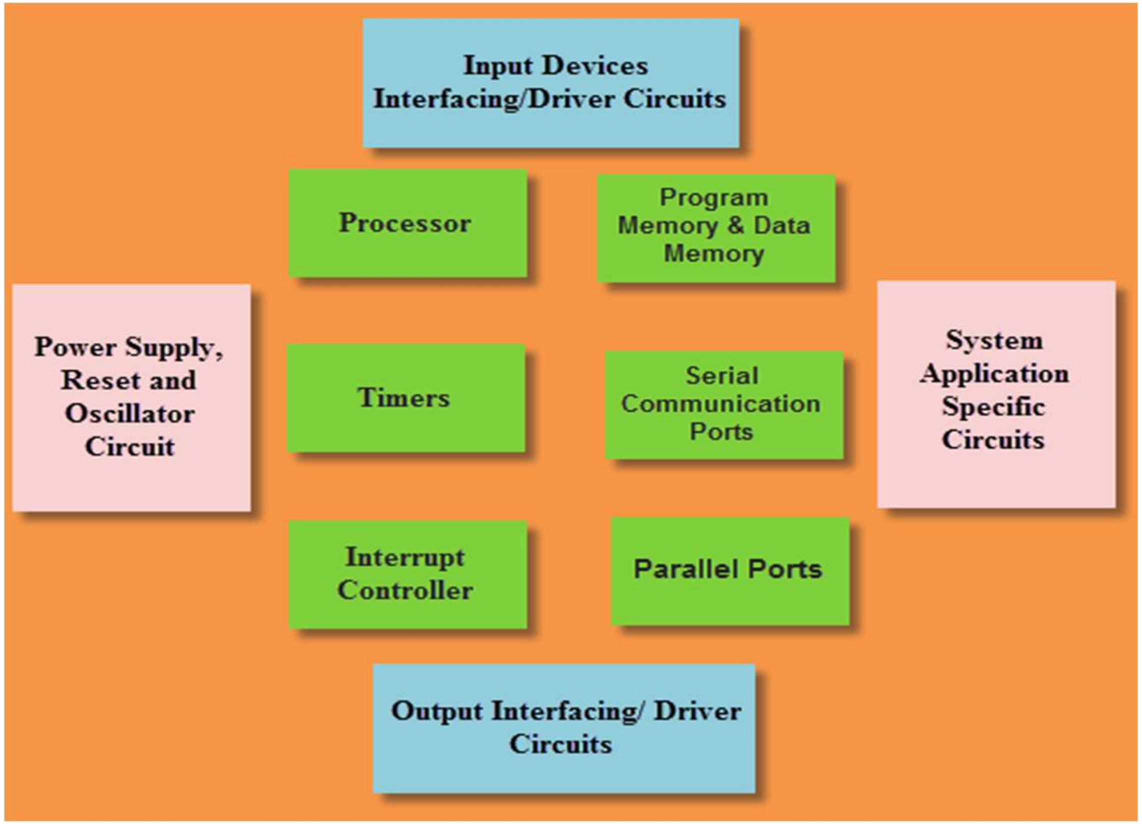


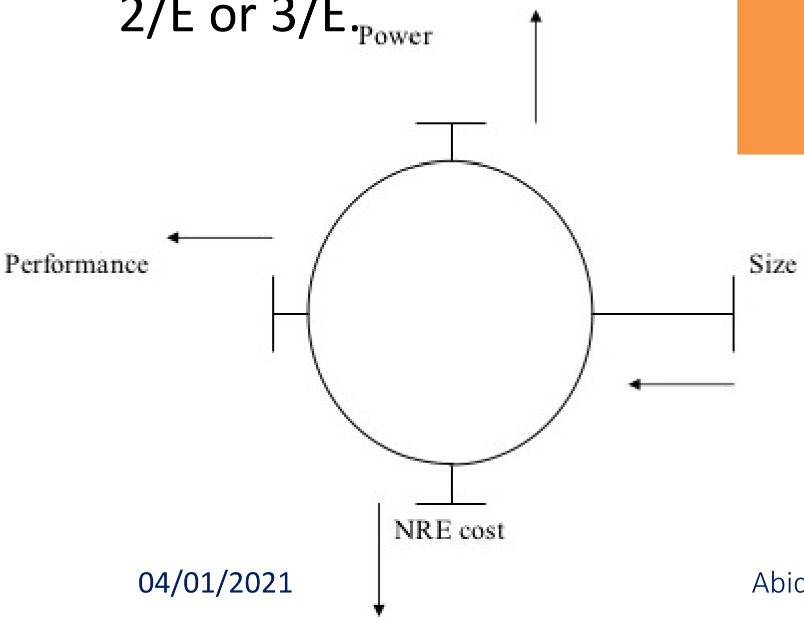
1. Of the characteristics shown, it is almost impossible to achieve the ideal performance upon the event triggering (occurrence of the event).
2. For Hard Real Time Systems:
   a. The shorter the execution time, the efficient is the design.
   b. Design with T2 execution time is better than that of **T1** execution time in the figure.
3. For **Soft** Real Time Systems:
   a. Execution time is not that much important compared with the cost and size.
   b. If the design with **T1** execution time is cheaper than the design with T2 execution, then the design with **T1** is preferable.

Embedded System Block Diagram

Mandate Building Blocks

1. CPU
2. I/O Circuitry
3. Memories
   a. Program Memory (ROM)
   b. Data Memory (RAM)
4. Power Supply
5. Reference: Embedded Systems Design, Raj Kamal, 2/E or 3/E.





Typical Design Constraints

1. Low Cost
2. Low Energy Consumption
3. Limited Memory
4. Real-Time Response

Key Design Issues or Challenges for Designing Embedded System

1. Cost
2. Size
3. CPU selection
   a. Microcontroller Vs
   b. Microprocessor Vs
   c. DSP
   d. FPGA + CPU
4. Speed (fosc) (Reliability)
   a. The power dissipated by the CPU is proportional to the core operating voltage
   b. The power dissipated by the CPU is proportional to square of operating voltage
5. Memories
   a. Program memory (ROM): Boot Code + Look-up table + Application executable
   b. Data memory (RAM): Registers + variables for temporary data storage
   c. Cache or DMA: To improve the CPU performance
   d. SD Card/USB: For secondary storage devices as a part of information retrieval
6. Time-to-market
7. Proven Case-Studies
8. Support of IDEs (Availability of cross-compilers)
   a. The software packages which are installed in the PC by using which the developers may start designing the embedded software even though the hardware is not ready is called as cross compiler
   b. Host: Computer
   c. Target: Actual hardware or SBC (Single Board Computer)
9. Interoperability
10. Validation by OEM standards
11. Design & Development Cost
12. Operational and Maintenance Issues

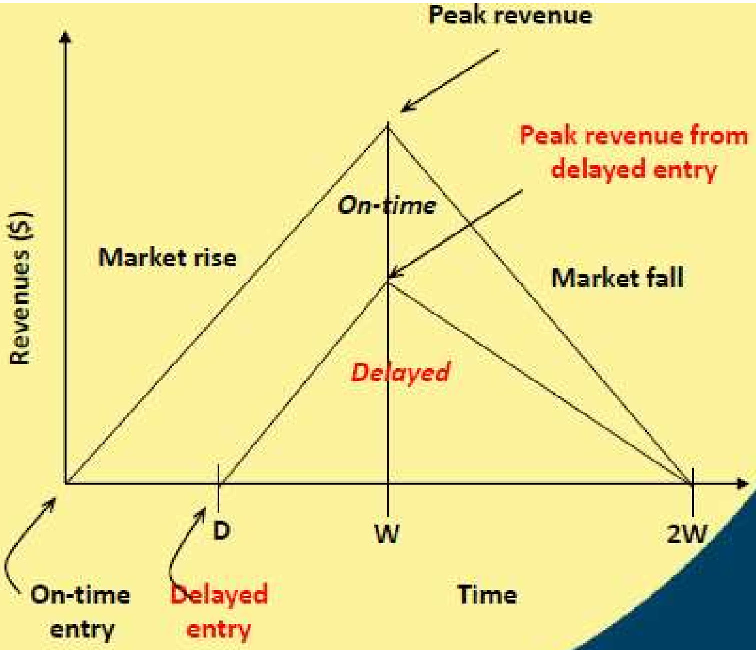
Time to Market Design Metric

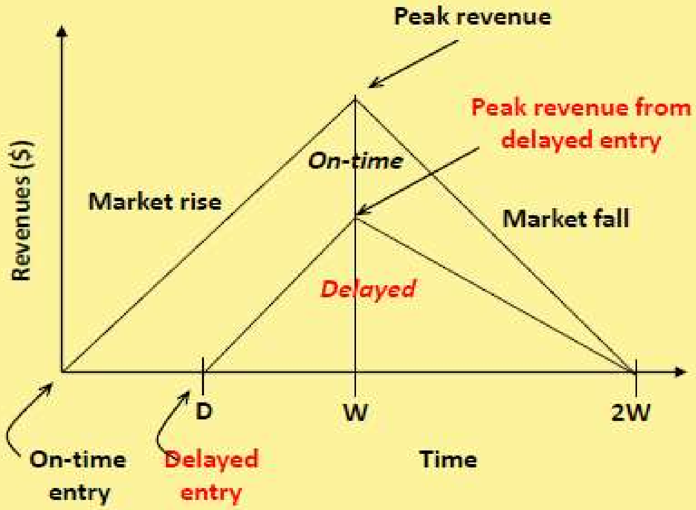
- This is a very crucial design metric.
  - Must be strictly followed to make a product commercially viable.
  - Requires exhaustive market study and analysis.
- Starting from the point a product design starts, we can define a *Market Window* within which it is expected to have the highest sales.
  - Any delay can result in drastic reductions in sales.

Curtesy: Slide No – 8, Lecture – 2, NPTEL Online Course on "Embedded System Design with ARM", By Prof. Indranil Sen Gupta, IIT Kharagpur

Loss Due to Delayed Time-to-Market

- Consider a simplified revenue model:
  - Product life is *2W*
  - Maximum sale occurs at time *W*
  - Market rise and market fall defines a *triangle* in the revenue graph.
  - *Area of the triangle determines the total revenue.*
- For delayed entry, can estimate the loss.
  - Difference in areas of the expected and actual triangles.





- Area of a triangle = ½ * base * height
  - Area (on-time) = ½ * 2W * W
  - Area (delayed) = ½ * (2W – D) * (W – D)
  - Percentage revenue loss = $D(3W - D)/2W^2 * 100$
- Examples:
  - 2W = 52 weeks, D = 4 weeks → LOSS = 22%
  - 2W = 52 weeks, D = 10 weeks → LOSS = 50%

Curtesy: Slide No – 9, 10, Lect. 2, NPTEL Course on "Embedded System Design with ARM", By Prof. Indranil Sen Gupta, IIT Kharagpur

NRE & Unit Cost Metric

- If C_{NRE} denotes the NRE cost and C_{unit} the unit cost of a product, then the total cost for manufacturing N units is given by:

  $$Total\ Cost = C_{NRE} + N * C_{unit}$$

- Therefore, per-unit cost is given by: $C_{NRE} / N + C_{unit}$
- Example:
  - C_{NRE}= Rs. 5,00,000 and C_{unit} = Rs. 5,000
  - Total cost for manufacturing 100 units = 5,00,000 + 5000 * 100 = 10,00,000
  - Per unit cost = 5,00,000 / 100 + 5000 = 10,000

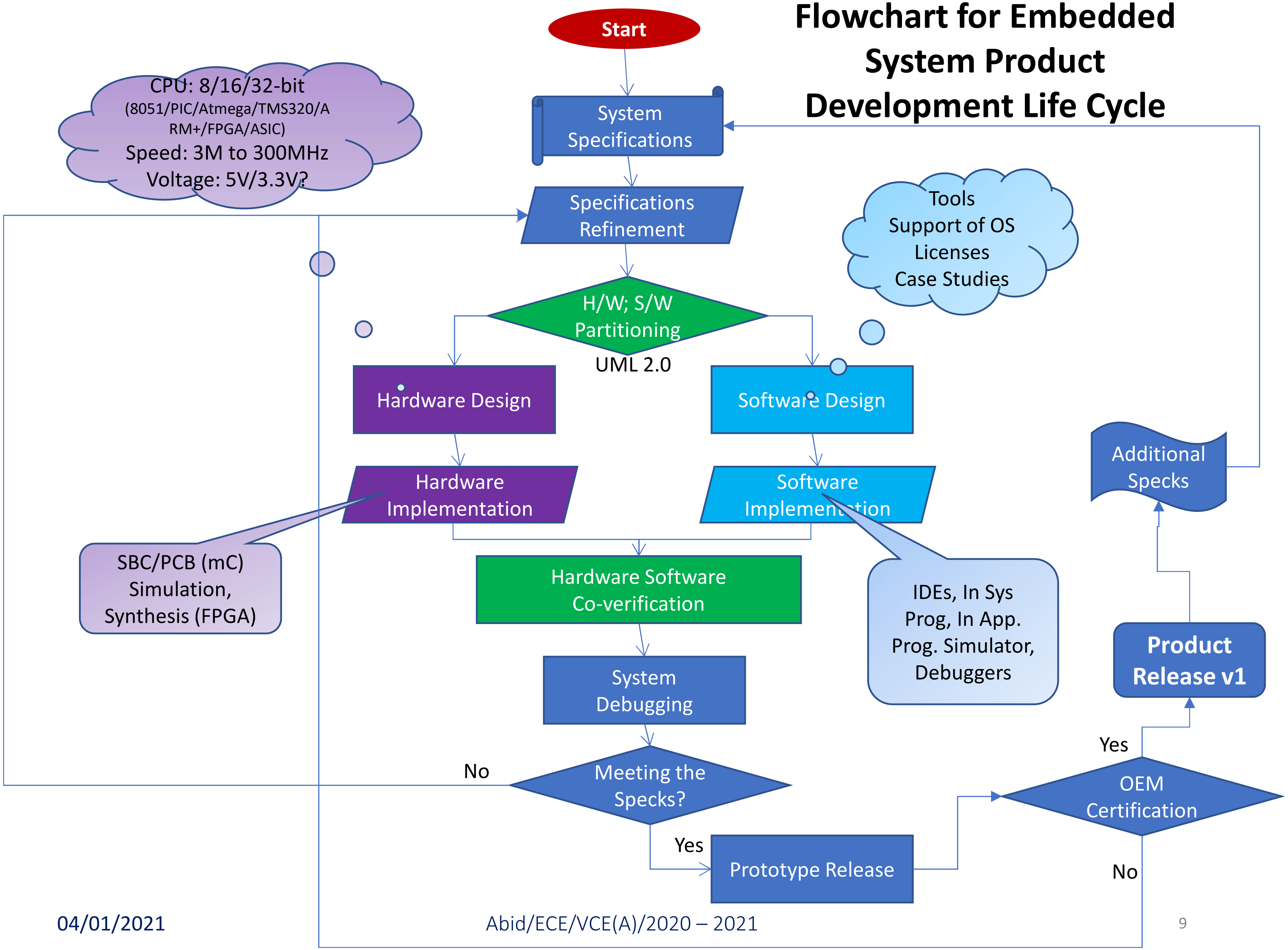
$$\boxed{Per\text{-}unit\ cost = C_{NRE} / N + C_{unit}}$$

- We can compare technologies by cost:
  - Choice A: C_{NRE} = Rs. 20,000, C_{unit} = Rs. 8,000
  - Choice B: C_{NRE} = Rs. 4,00,000, C_{unit} = Rs. 3,000
  - Choice C: C_{NRE} = Rs. 10,00,000, C_{unit} = Rs. 8,000
- Of course, time-to-marker cost must also be considered.

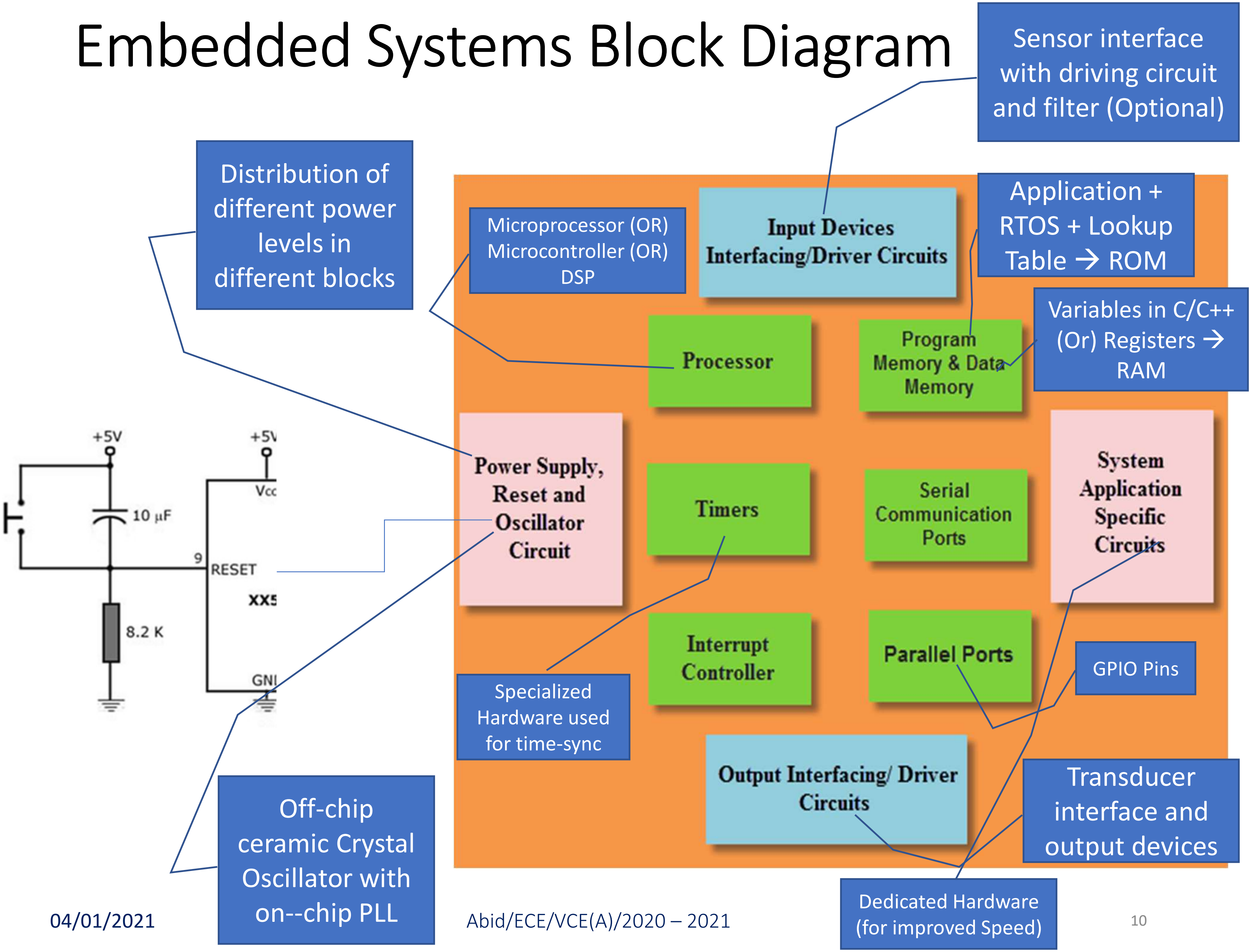
Which one to Choose?

Curtesy: Slide No – 11, 12, Lect. 2, NPTEL Course on "Embedded System Design with ARM", By Prof. Indranil Sen Gupta, IIT Kharagpur

Flowchart for Embedded System Product Development Life Cycle



Embedded Systems Block Diagram



Classifications of Embedded Systems

1. Based on Real-Time responsiveness:
   a. Hard Real-Time System
   b. Soft Real-Time System

2. Based on foot-print or Size or Cost:
   a. Small-scale embedded system
   b. Medium-scale embedded system
   c. Large (Sophisticated) embedded System

3. Based on Connectivity:
   a. Stand-alone embedded system
   b. Embedded System connected to a new

Embedded System Classifications (Based on Size & Cost)

1. Small Scale Embedded Systems
   a. Normally designed using a Microcontroller or DSP
   b. Costs less than Rs. 1, 000/-
   c. Very small memory and I/O footprints of the order of KBs
   d. Mostly Soft Real Time Embedded Systems in nature.
   e. Typical examples includes:
      i. Bus ticketing machines
      ii. Digital blood pressure measurement
2. Medium Scale Embedded Systems
   a. Normally designed using 32-bit Microcontroller Unit (MCU) or MPU
   b. Costs less than Rs. 10, 000/-
   c. Often comes with the Real Time Operating System (Open-source)
   d. Medium memory (MB) & occasionally with the secondary memory devices such as USB/SD Cards.
   e. Typical Examples include:
      i. Mobile Phones
      ii. Connected Printer or POS
3. Large Scale Embedded Systems
   a. Normally designed on FPGA with CPU cores.
   b. Largely these systems are Hard Real Time in nature
   c. Operates at much higher clocks with higher volumes of memories.
   d. Typical Examples include:
      I. Guided missile electronic control unit.
      II. Airbag controller etc.