Embedded systems: An electronic device which is:

* Embedded in another system/product.
* a computer system dedicated to a certain task.
* having limited hardware & software functionality.

limits could be:

- Hw:
  - * Processing power
  - * Memory
  - * Power consumption
- Sw:
  - * limited or no OS
  - * Fewer Apps.

* ES contains microcontroller to process system's inputs & generate outputs. The link between system's inputs & outputs is provided by a coded algorithm stored within the processor's resident memory.

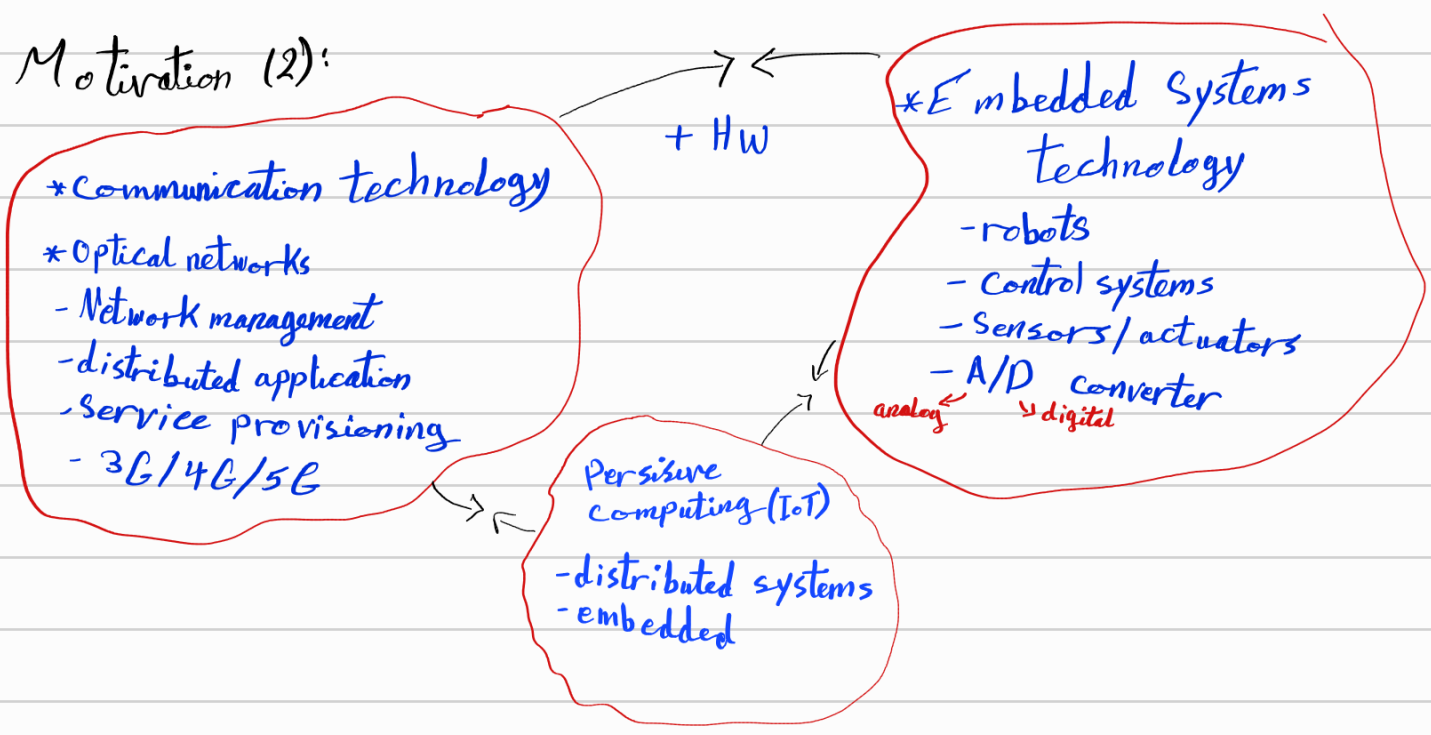
Cyper-physical system = Embedded system + physical system

Motivation: Trend in information Processing Systems towards:

- Ambient intelligence.
- IoT
- Ubiquitious computing

• requires a holistic approach involving embedded SW&HW and the physical enviroment.
• additional challenges:
  - power/energy
  - cost
  - dependability
  - real time processing
  - underrepresented teaching

Motivation (2):



Chapter 2:

Q_1: What are typical ES constraints?

Ans: REDs

Real-time *
Efficiency *
Environment
Extreme cost sensativity
Dependability *
Size
Safety

Q_2: What are the customer's expectations in ES Constraints?

Ans_2:
Increased overall dependability
Increased functionality
Increased performance
Reduced cost.

Q_3: Explain dependability in your words:-

Ans_3: ES/CPS must be:

MARS:-

Maintainable:

Available: a(t) = probability that system works at time t.

Reliable: r(t) = probability of system working correctly considering it was working correctly at time t=0.

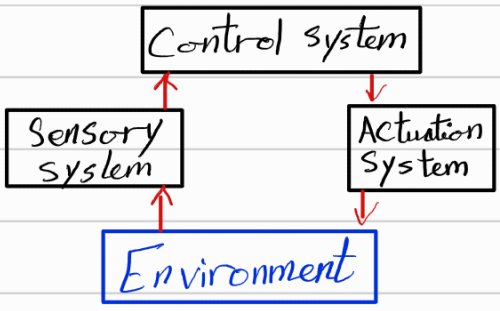
Safety: No harm to be caused, not dangerous.

Security: Encrypted, and authentic communication.

*What kind of efficiencies do we have?

- Energy efficiency
- Cost //
- Size //
- Weight //
- Code size //
- real-time // (should use the avaliable HW as good as possible)

*Real-time ≠ Fast: real-time means the system time (internal time) must be measured with the same scale used for measuring the controlled environment time (external time)



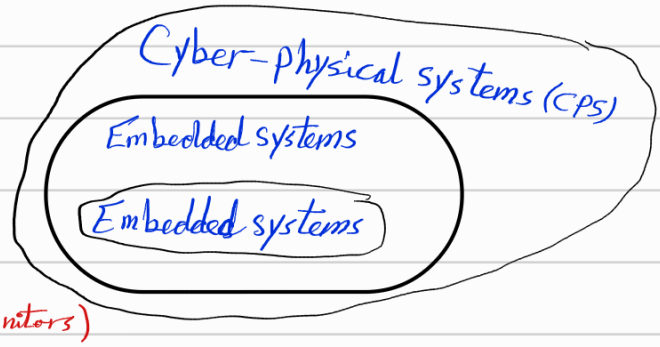
* Name different Categories of real-time constraint?

1- Hard: If the result is produced after the deadline, it may cause catastrophic consequences.

2- Soft: If the result is produced after the deadline, it still has some utility of the system.

3- Firm: nowaday it is also categorized into Soft.

* Real-time system ≡ CPS



* What does dedicated system means?

- For embedded system, it means two things:-

1- dedicated to a certain task (not like computers)

2- dedicated user interface (no mouse, keyboard, or monitors)

* Why dedicated, or why no games or apps avaliable on ES?

1- Would make the system less dependable.

2- No unused resourses should be avaliable in a ES.

* What challenges for ES/CPS in terms of software?

- Less efficient resource usage
- Sw engineers don't have contact with environment
- // // // // access to ES/CPS
- There is an exponential growth in Sw complexity
- More than 70% of development costs for complex systems are due to software development.

* What is a Model ? (define a model?)

- simplification of another entity, which can be a physical thing or another model.

* What are the Requirments to a model? (صِيَغَة)

- Synchronization & Communication
- Component-based design
- Concurrency
- Timing-behaviour
- Hierarchy

* What does it mean "Concurrency" of a model/system?

- Real-life systems are usually concurrent systems.
- Humans aren't very good in understanding concurrent systems & its complex behaviour.

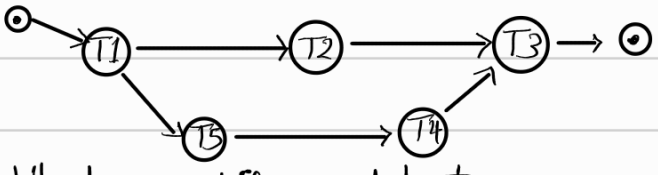
* What are requirements for specs & modeling techniques? FEETS

- Flexibility & portability
- Event-handeling
- Excuctability & Readability
- Termination
- State-oriented behaviour

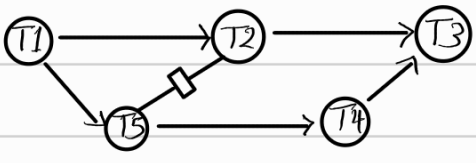
* Dependence graph could contain: (need to know all)

1- Timing information:

2- I/O information:



3- Shared resorces



* What are diff. models to communication? and explain

1- Shared memory: Communication is been carried out by access to the same memory from all component. We have to make sure that two components are not writing in at the same time.

Pros: very fast communication processing.

Conss Multiple concurrent systems are very complex with shared memory.

2- Message passing:-

a) Asynchronous messaging passing (non-blocking Commucation):

* Communicationis carried out through channels.
* Each channel can buffer messages.
* The sender doesn't have to wait for the recipient to send next message
* If a buffer is full, it buffer overflows.

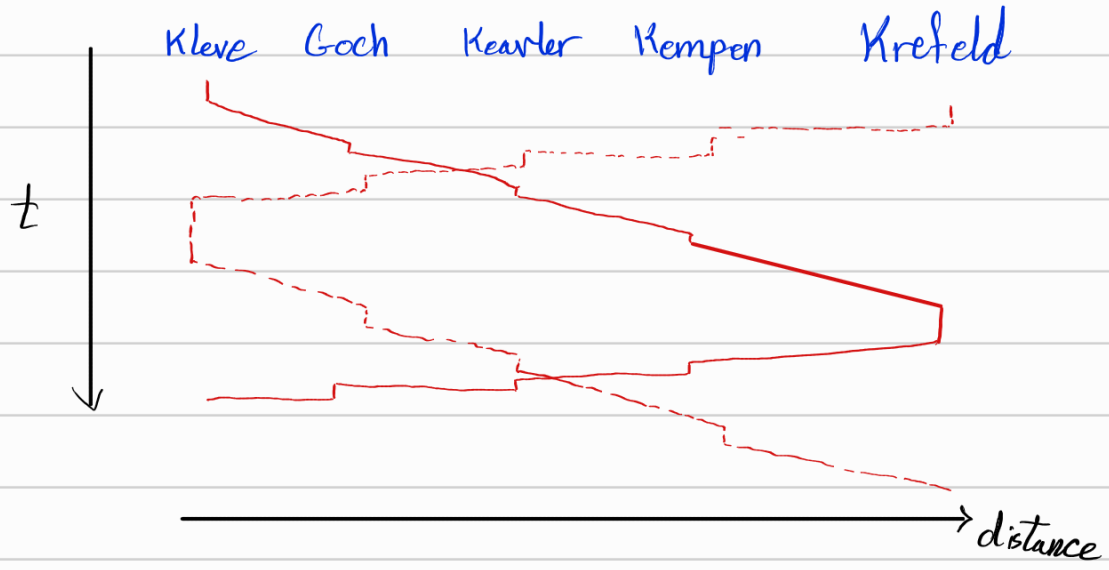
b) Synchronous message passing (blocking communication, rendzvous):

* Communication is an instantaneous action called rendzvous.
* Sender will wait until message is recieved.
* No buffer overflow
* Reduced performance

C) Extended rendzvous:

* ACKnowlegement must be send from the receiver for the sender to be able to communicate again.
* ACKnowlegement is not sent, until the message is checked.

* Sketch a Time distance diagram (TDD) for Kleve-Krefeld bus/train?

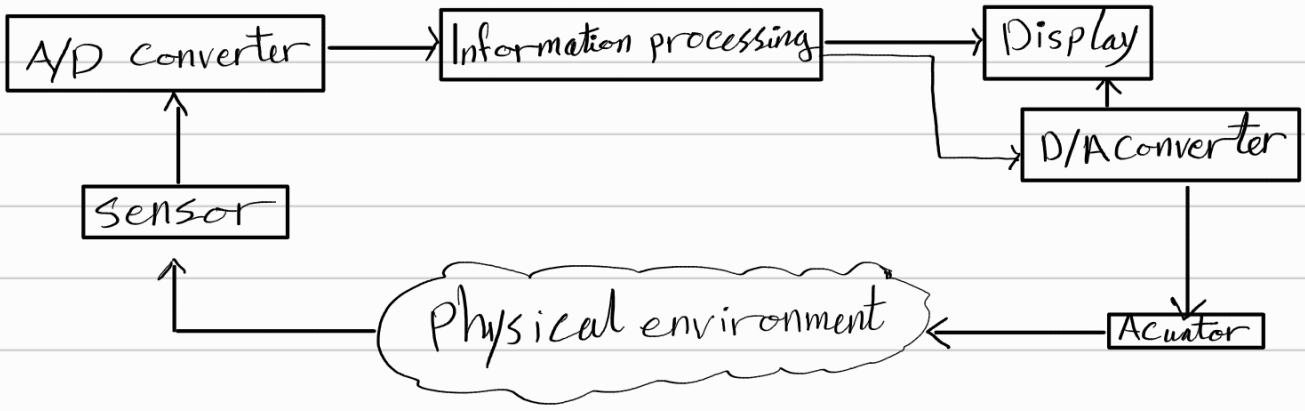


* What are the two classical Automata Finite State Machines? and explain?

1-Moore: Output is only based on current state

2-Mealy: Output is based on current state & input.

* What are the Key elements of petri nets?
* Conditions: are satisfied or not satisfied
* Events: May happen if certain Conditions are met.
* Flow relation: Describes the conditions that must be met before events can happen, describes Conditions which become true if events happen.



*What is von Neumann Model?

- Reflectes the principles of operation of standard Computers by Sequential excution of instructions.
- It allow almost unrestricted access to the global memory (shared variable).

* What Languages are used in Von Neumann Model? ♡

- Machine (binary) languages. - Assembly languages.
- C, C++, Java.

* What is a Race condition, and how to avoid it?

Allowing several threads to access the same memory might lead to a race condition.

Race condition: When more than one thread is accessing the same resource at the same time, it could happen that you have an unexpected outcome because of rais condition.

We can avoid it by using critical sections. Any resource which can not be accessed by more than 1 thread, will be considered as Critical section, and threads will be accessing any critical section only by turn.

What is Deadlock & how to avoid it? (deadlock free system)

Deadlock happens if the following 4 conditions are satisfied:

1- Mutual exclusion: A resource cannot be used by > 1 thread.

2- Hold & wait: Thread already holding resources may request new resources.

3- No pre-emption: Resources cannot be preempted

4- Circular wait: There exist a set of threads $(T_1, T_2, \dots, T_N)$, such that T_1 is waiting for T_2, T_2 for T_3, and T_N for T_1 (where $N \geq 2$)

There are many techniques to turn one of these conditions false to have Deadlock free.

*Name five different model Levels:- Gaits

* Gate-level
* Algorithmic Level
* Instruction-set Level
* Transction Level
* System Level

*Name 5 diff. sensors?

Light, rain, temprature, velocity, acceleration, and wieght.

* How rain sensor is working? please sketch:-

Based on internal reflection

What is signal, and how do we measure it?

A signal s is a mapping from time domain D_T to value Domain D_V.

Signals are generated by sensors.

The problem that the signal is continuous in time & value (Analog signal), however, in a digital system we can only work with discrete signals in time & value.

Analog signal: Continuous one line.

Discrete: Points

* What do we use to go from analog to digital signals?

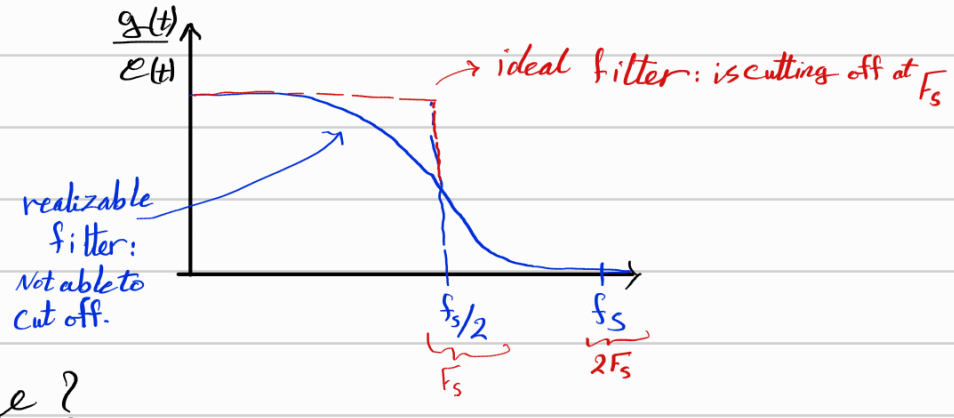
Capacitors ★

* When does Aliasing happen?

= When you don't sample enough.

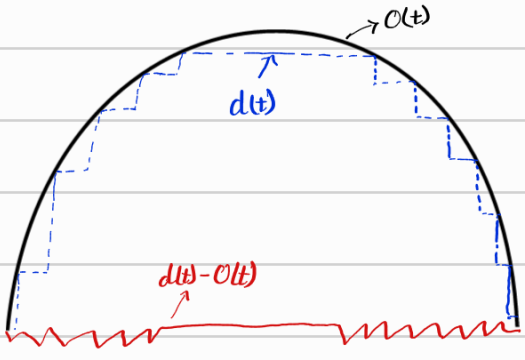
* What can we do to prevent Aliasing?

To remove aliasing we place a Low-pass filter in front of the signal.



* What is Quantization voltage?

The difference between the original & digital voltage is called Quantization voltage.



* Which net is better for dining philosophers prolem?
Predicate/transition net because:
1- Model can be easily extended by just adding an individual token.
2- No change of structure needed.
3- Much smaller in size / Large and complex in C/E net.

* What is C/E net?
A simple net consists of two nodes with the same pre-set and post-set.

* What is place/transition nets?
Can allow more than one token per condition
- Place correspond to condition in (C/E nets)
- Transition correspond to E event in (C/E nets)
Number of tokens per place is called marking

* What is predicate/transition nets?
predicate/transition nets can be used to reduce the size of representation of complex systems.