Sharif University of Technology
Department of Computer Engineering

Embedded System Design

Specification of embedded systems

A. Ejlali

System Specification

- **The first and the most important step in the design flow.**
  - **Requires human intelligence**
  - **Can we use natural language?**
  - **It is necessary to check specifications for**
    - **Completeness**
    - **Absence of contradictions**
  - **It should be possible to derive implementations from the specification in a systematic way.**

Required Features

Specification languages for ES should have the following features:

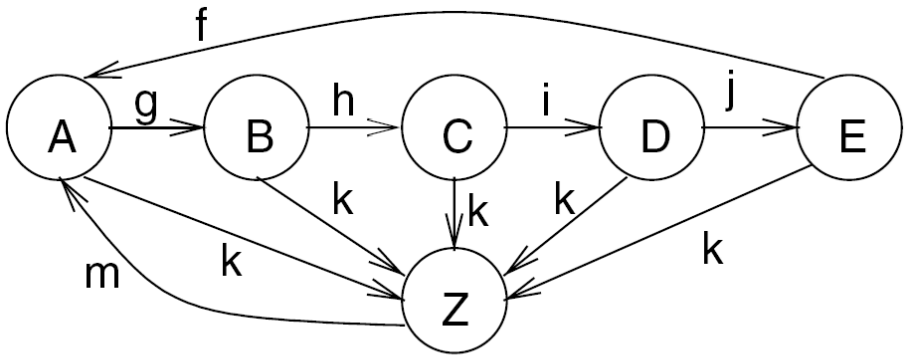
- **Hierarchy**
  - **Behavioral Hierarchies**
    - **e.g., Super-states**
  - **Structural Hierarchies**
    - **Like what is supported by VHDL and Verilog**
- **Timing Behavior**
  - **Delay**
  - **Cause and effect relationship**

Required Features (Cont.)

- **State-oriented behavior**
  - **Automata provide a good mechanism for modeling reactive systems.**
- **Event handling**
  - **The reactive nature of ES**
  - **Mechanisms for describing events must exist**
    - **External events (Caused by environment)**
    - **Internal events (Caused by components of the system)**
- **Support for efficient implementation**
  - **e.g. Hardware/Software Co-design**

Required Features (Cont.)

- **Support for dependable system design**
  - **Unambiguous semantics**
  - **Facilitate formal verification**
- **Exception-oriented behavior**
  - **It is not acceptable that exceptions have to be indicated for each and every state**



State diagram with exception k

Required Features (Cont.)

- **Concurrency**
  - **Real-life systems are concurrent systems.**
  - **It is necessary to be able to specify concurrency conveniently.**
- **Synchronization and communication**
  - **Concurrent actions have to be able to communicate and it must be possible to agree on the use of resources (e.g., mutual exclusion).**

Required Features (Cont.)

- **Presence of programming elements**
  - **Usual programming languages have proven to be a convenient means of expressing computations.**
  - **Classical hardware description techniques (e.g., state diagrams) do not meet this requirement.**
- **Executability**
  - **Simulation (Design Verification)**

Required Features (Cont.)

- **Readability and flexibility**
  - **Readable by human**
  - **Small changes of the system → Small changes of the specification**
- **Support for non-standard I/O-devices**
  - **to describe inputs and outputs for non-standard I/O-devices conveniently.**

Required Features (Cont.)

- **Non-functional properties**
  - **Reliability**
  - **Size**
  - **Power consumption**
- **Appropriate model of computation**
  - **Von Neumann paradigm is not suitable**

There is no hope to develop a formal language capable of meeting all these requirements.