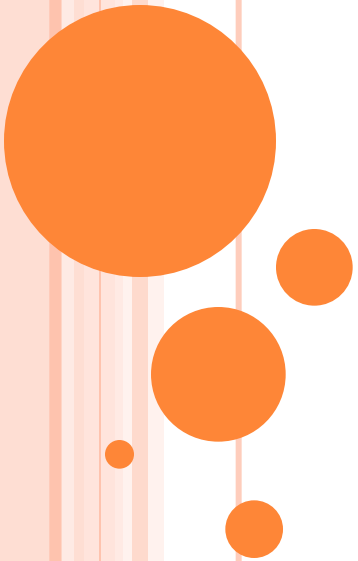
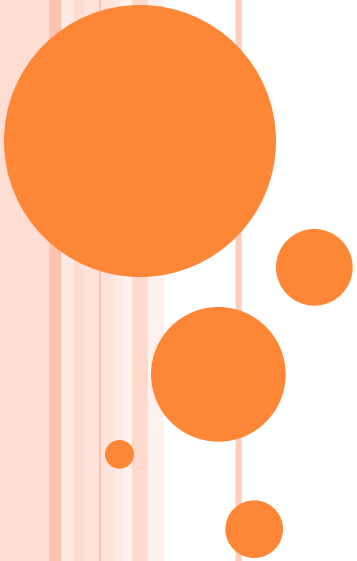
Software Engineering



LECTURE-30



Real-Time Systems Design and Analysis

Basic real-time concepts

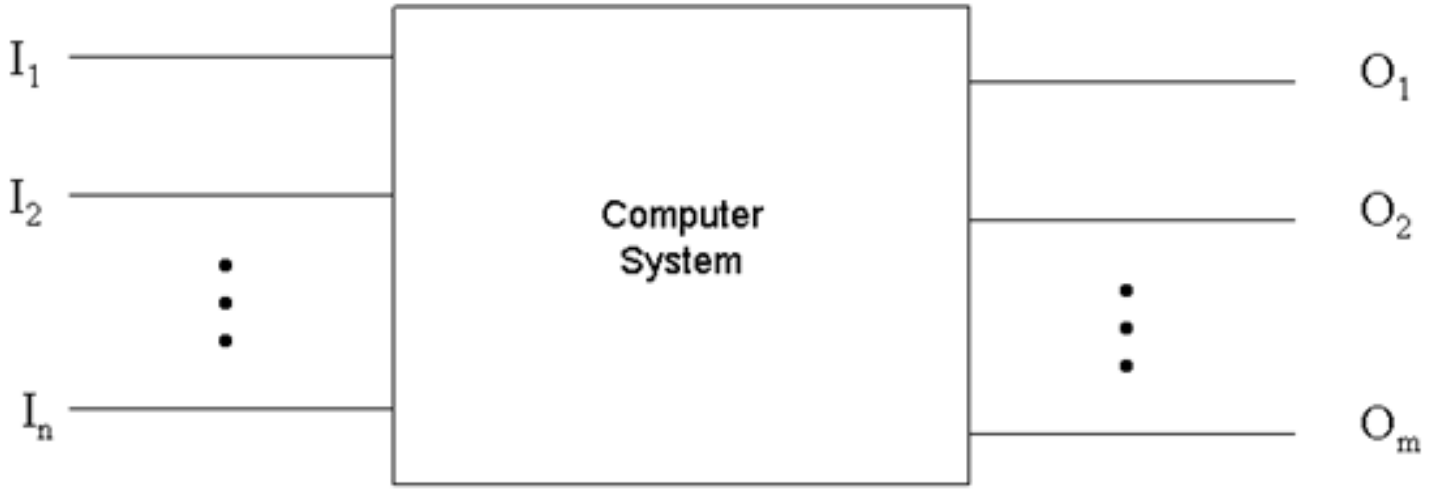
- Terminology
- Real-time design issues
- Example real-time systems
- Common misconceptions
- Brief history

Terminology

- Systems concepts
- Real-time definitions
- Events and determinism
- CPU utilization

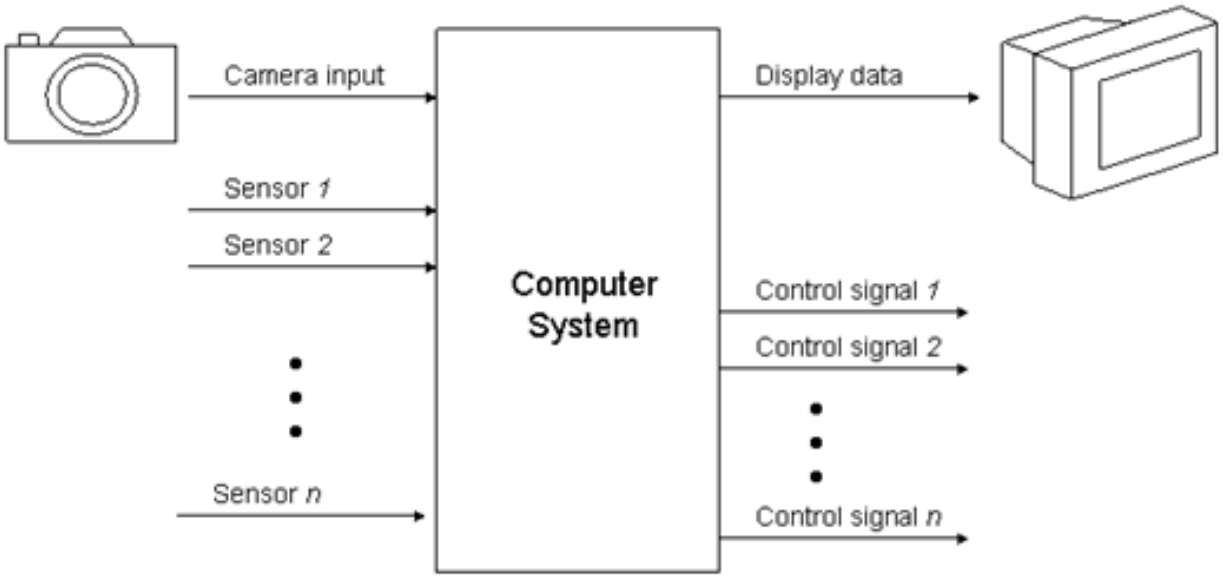
Systems concepts

Definition: A system is a mapping of a set of inputs into a set of outputs.



A system with *n* inputs and *m* outputs.

Systems concepts



Typical real-time control system including inputs from sensors and imaging devices and producing control signals and display information.

Systems concepts

Definition: The time between the presentation of a set of inputs to a system (stimulus) and the realization of the required behavior, including the availability of all associated outputs, (response) is called the response time of the system.

Real-time definitions

Definition: A real-time system is a system that must satisfy explicit (bounded) response-time constraints or risk severe consequences, including failure.

Definition: A failed system is a system that cannot satisfy one or more of the requirements stipulated in the formal system specification.

Definition: A real-time system is one whose logical correctness is based on both the correctness of the outputs and their timeliness.

Real-time definitions

Definition: A real-time system is a system that must satisfy explicit (bounded) response-time constraints or risk severe consequences, including failure.

Definition: A failed system is a system that cannot satisfy one or more of the requirements stipulated in the formal system specification.

Definition: A real-time system is one whose logical correctness is based on both the correctness of the outputs and their timeliness.

Real-time definitions

Definition: A soft real-time system is one in which performance is degraded but not destroyed by failure to meet response-time constraints.

Definition: A hard real-time system is one in which failure to meet a single deadline may lead to complete and catastrophic system failure.

Definition: A firm real-time system is one in which a few missed deadlines will not lead to total failure, but missing more than a few may lead to complete and catastrophic system failure.

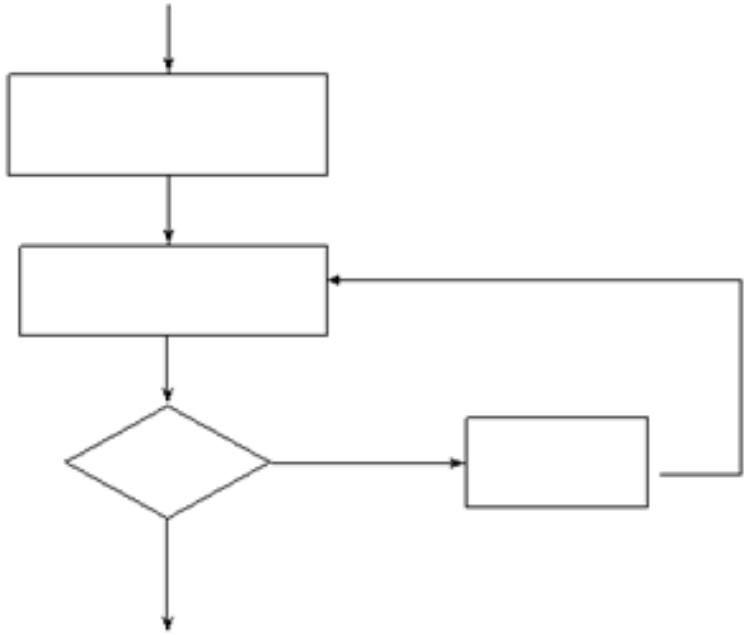
Real-time definitions

	Real-time Classification	Explanation
Automated teller machine	soft	Missing even many deadlines will not lead to catastrophic failure, only degraded performance.
Embedded navigation controller for autonomous robot weed killer	firm	Missing critical navigation deadlines causes the robot to veer hopelessly out of control and damage crops.
Avionics weapons delivery system in which pressing a button launches an air-to-air missile	hard	Missing the deadline to launch the missile within a specified time after pressing the button can cause the target to be missed – which will result in catastrophe.

A sampling of hard, soft, and firm real-time systems.

Real-time definitions

Definition: Any occurrence that causes the program counter to change non-sequentially is considered a change of flow-of-control, and thus an event.



A simple program flowchart showing a branch as a change in flow of control, represented by the diamond icon.

Real-time definitions

Definition: The release time is the time at which an instance of a scheduled task is ready to run, and is generally associated with an interrupt.

Real-time definitions

	Periodic	**Aperiodic**	**Sporadic**
Synchronous	Cyclic Code Processes scheduled by internal clock	Typical branch instruction Garbage collection	Branch instruction e.g. error recovery Traps
Asynchronous	Clock generated interrupt	Regular, but not fixed period interrupt	Externally generated exception “Random events”

Taxonomy of events and some examples.

Real-time definitions

Definition: A system is deterministic if, for each possible state and each set of inputs, a unique set of outputs and next state of the system can be determined.

Note, deliberately non-deterministic machines are hard to build, while it is easy to fall into accidentally non-deterministic machines!

Real-time definitions

Definition: The (CPU) utilization or time-loading factor, , is a measure of the percentage of non-idle processing.

Utilization (%)	Zone Type	Typical Application
0-25	significant excess processing power – CPU may be more powerful than necessary	various
26-50	very safe	various
51-68	safe	various
69	theoretical limit	embedded systems
70-82	questionable	embedded systems
83-99	dangerous	embedded systems
100+	overload	stressed systems

CPU utilization zones and typical applications and recommendations.

Real-time definitions

- Determining CPU Utilization

Suppose a system has $n \geq 1$ periodic tasks, each with an execution period of p_i and hence execution frequency $f_i = 1/p_i$. If task i is known to have (or has been estimated to have) a maximum (worst case) execution time of e_i then the utilization factor u_i for task i is

$$u_i = e_i / p_i \tag{1.1}$$

Then the overall system utilization is

$$U = \sum_{i=1}^{n} u_i = \sum_{i=1}^{n} e_i / p_i \tag{1.2}$$

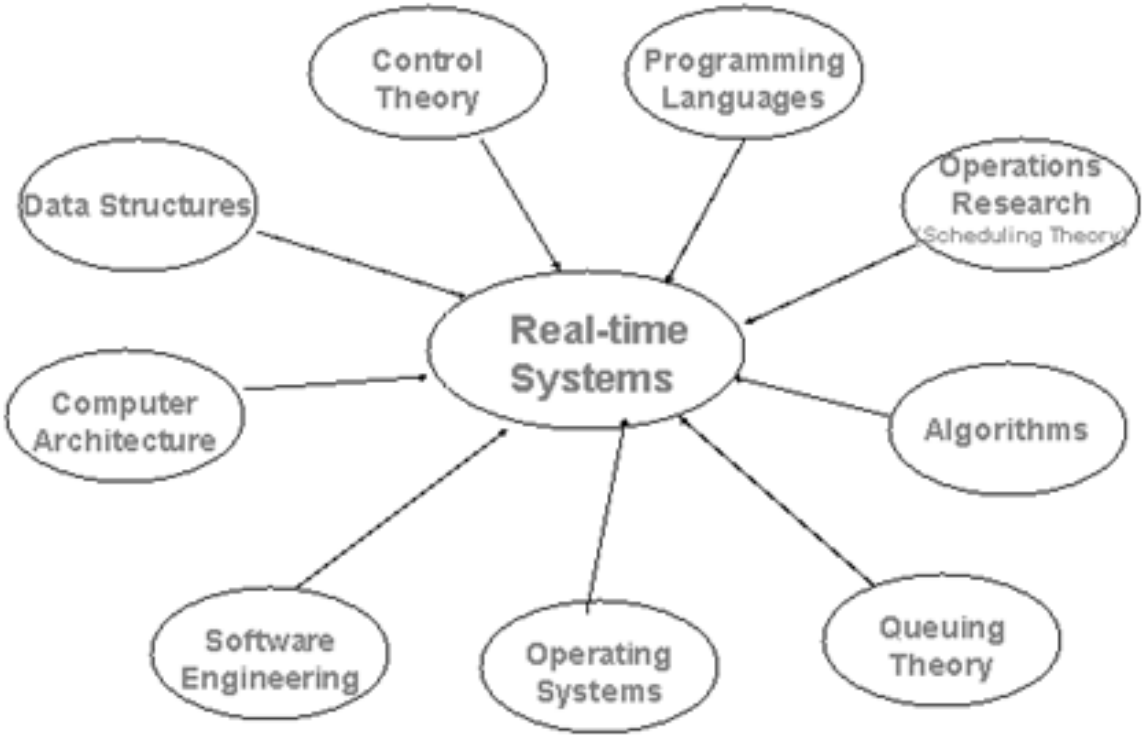
Real-time definitions

- The nature of time
  - Where do deadlines come from?
  - Challenge “conventional” wisdom as it may place undue constraints on the system
  - Clocks can be used for time stamping and synchronization, but clocks are imperfect

Basic real-time concepts

- Terminology
- Real-time design issues
- Example real-time systems
- Common misconceptions
- Brief history

Real-time design issues



Disciplines that impact real-time systems.

Real-time design issues

- The selection of hardware and software, and evaluation of the tradeoff needed for a cost-effective solution, including dealing with distributed computing systems and the issues of parallelism and synchronization.
- Specification and design of real-time systems and correct representation of temporal behavior.
- Understanding the nuances of the programming language(s) and the real-time implications resulting from their translation into machine code.

Real-time design issues

- Maximizing of system fault tolerance and reliability through careful design.
- The design and administration of tests, and the selection of test and development equipment.
- Taking advantage of open systems technology and interoperability.
- Measuring and predicting response time and reducing it.
- Performing a schedulability analysis, that is, determining and guaranteeing deadline satisfaction, *a priori*, is largely “just” scheduling theory.

Basic real-time concepts

- Terminology
- Real-time design issues
- Example real-time systems
- Common misconceptions
- Brief history

Example real-time systems

Domain	Applications
Avionics	• Navigation • Displays
Multimedia	• Games • Simulators
Medicine	• Robot surgery • Remote surgery • Medical imaging
Industrial Systems	• Robotic assembly lines • Automated inspection
Civilian	• Elevator control • Automotive systems

Real-time application domains.

Example real-time systems

- Aircraft inertial measurement system
- Nuclear plant control
- Airline reservation system
- Pasta sauce bottling plant
- Traffic light control system for 4-way intersection

Basic real-time concepts

- Terminology
- Real-time design issues
- Example real-time systems
- Common misconceptions
- Brief history

Common misconceptions

- Real-time systems are synonymous with “fast” systems.
- Rate-monotonic analysis has solved “the real-time problem.”
- There are universal, widely accepted methodologies for real-time systems specification and design.
- There is never a need to build a real-time operating system, because many commercial products exist.
- The study of real-time systems is mostly about scheduling theory.

Basic real-time concepts

- Terminology
- Real-time design issues
- Example real-time systems
- Common misconceptions
- Brief history

Brief history

Year	Landmark	Developer	Development	Innovations
1947	Whirlwind	MIT/US Navy	Flight simulator	Ferrite core memory, “real response times”
1957	SAGE	IBM	Air defense	Specifically designed for real-time
1958	Scientific 1103A	Univac	General purpose	Hardware interrupt
1959	SABRE	IBM	Airline reservation	Hub-go-ahead policy
1962	Basic Executive	IBM	General purpose	First real-time executive
1963	Basic Executive II	IBM	General purpose	Diverse real-time scheduling, Disk resident user/systems programs
1970s	RSX, RTE	DEC, HP	Real-time operating systems	Hosted by mini-computers
1973	Rate-monotonic system	Liu and Layland	Theory	Stated upper bound on utilization for schedulable systems
1980s	RMX-80, MROS 68K, VRTX, etc.	Various	Real-time operating system	Hosted by microprocessors
1983	Ada 83	US Department of Defense	Programming language	Intended for mission critical, embedded, real-time systems
1995	Ada 95	Community	Programming Language	Refinement to Ada 83