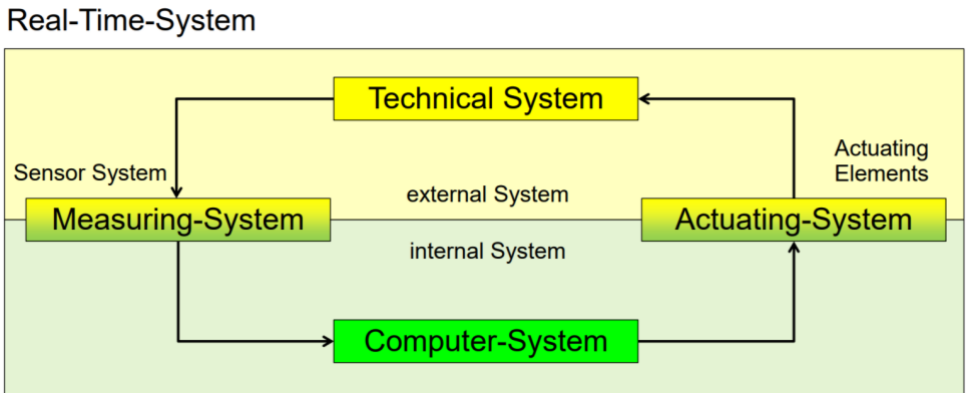
Realtime Systems Theoriefragen:



What are the two requirements for real time systems?

Real-time systems' requirements:

determinability (it comes to an well defined end!)

predictability (when it comes to the end!)

(Vorhersagbarkeit/Vorhersehbarkeit/Determiniertheit)

Defined response time

minimize the time for systems tasks

What is Real-Time?

Correctness and Response Time of the results are guaranteed

In non-Real Time Systems only the correctness of the result is guaranteed

An RTS will also guarantee that a certain deadline is met.

Definition of Realtime System:

Real-time systems are systems which, in addition to the functional requirements also meet temporal time requirements.

”. . . a Real-Time system is required to complete its work and deliver its services on a timely basis."

Give example of real time systems

The MP3 player is a real-time system, sat lite receiver and routers also.

What are three types of RTS? Give an example for each

There are three types of RTS:

1. Hard RTS: Missing a deadline is a total system failure (e.g. airbag in car, use case in aviation).

2. Soft RTS: The usefulness of a result degrades after its deadline, thereby degrading the system's quality of service. (e.g. warning systems, e.g. distance warner).

3. Firm RTS: Infrequent deadline misses are tolerable but may degrade the system's quality of service. The usefulness of a result is zero after its deadline. (e.g. ignition-point-optimizer for motor)

Classification of Real-Time Systems

1. Consequence of missing deadline
2. Reliability and fault tolerance
3. Distribution: centralized or distributed RTS
4. Interactive or autonomous system
5. Hierarchical or independent systems
6. Time-driven or event-driven RTS

    Time-driven :which job is next at specific time; there a chosen priori before execution

    Event-driven RTS: scheduling decisions are triggered by events not time instance events are release completion blocking, unblocking of jobs

7. periodic, aperiodic, sporadic RTS
8. Cyclic or asynchronous scheduling

What is an embedded system?

Embedded Systems

Definition: The computer is not a computer itself, but part of a system. In other words embedded systems are central and part of the overall system. (e.g. a missile control)

Charactristics of Embedded System

- Complex interaction with the environment and other Embedded Systems
- Often operated autonomously
- High requirements on reliability

What is a System (general definition)

Definition: A System is a set of Elements, Objects, Components or Modules that have a relations and may interact with each other.

A System has a well defined purpose.

What is a Process?

Definition: A process is set of relations or actions that converts a predefined input into a required outcome.

Ein Prozess ist ein Satz von Wechselbeziehungen oder Tätigkeiten, die Eingaben in Ergebnisse wandeln.

Time:

What is time?

Definition Time: Time is a mathematical construct to describe movement in a formal way.

Year: Time distance between the recurrence of a distinct position of the sun

Month: Time distance between 2 times new moon

Day: Time distance between two maximum altitudes of the sun

Hour: A day has 24 hours

Minute: An hour has 60 minute

Second: A minute has 60 seconds

There are three major types of movement:

1. force-less, un-accelerated, linear movement
2. periodic movement
3. by a force accelerated movement

Definition of Job: a Job J is a single CPU-time requirement to perform a computational sequence.

Definition execution time / Ausführungszeit: execution time is the duration of a specific Job between Job request and Job completion.

Definition net execution time / Netto-Ausführungszeit: net execution time is the execution time for a Job when the CPU is used exclusively for this specific Job.

What are the important execution times ?

• Minimal net execution time (ideal case)

• Maximum net execution time (worst case)

• Average net execution time

Soft Realtime Systems= Average execution time

Hard Realtime Systems= Maximum execution time

Definition of Worst Case Execution Time:

- Relevant for deadlines
- maximum limit of the variable execution time

World Clock / Weltzeit Uhr

This time is a point in time! This point in time is an absolute time! Nothing to set. No trigger. No reaction.

Alarm Clock / Wecker

you can set an absolute alarm time

the alarm time is a point in time

The alarm clock compares the actual time with the alarm time. When the alarm time lower or equal the actual time the alarm clock trigger a sound.

Worst Case Execution Time – WCET

• Relevant for deadlines is the maximum execution time of a program on a certain hardware.

• The WCET is the maximum limit of the variable execution time.

Calculating WCET

Two different levels:

1. macroscopic examination: What does a computer program do?

2. microscopic examination: What happens inside the microprocessor?

Two methods:

1. Dynamical WCET Analysis: (deductive, let's measure) Measurement of an adequate number of execution runs on a certain hardware

2. Static WCET Analysis: (inductive, let's think) Calculating execution time based on the computer program.

Theoretical planning of a system: Using the maximum net execution time and name it as execution time Δtexec

Pratical planning of a system: Measuring the time for the job and estimate the WCET, which is the maximum gross execution time / maximale Brutto-Ausführungszeit and name it as execution time Δtexec

Concurrency and Scheduling

Concurrency

Block design:

• Which HW-Block (e.g. Co-processor) or SW-algorithm (executed on a standard processor) should be used for implementation.

Communication design:

• Which and how many communication channels are needed between the design blocks?

Concurrency Issues

Deadlocks: One wait for another

Livelocks: One triggers the other (distributed enless loops)

Race conditions: The output is dependent on the sequence or timing of other uncontrollable events

Definition of scheduling: Decision how the processes are assigned to run on available CPU(s)/Core(s).

Goals of Scheduling:

• Minimizing the average response time: e.g. interactive systems

• Maximizing throughput: e.g. server

• Maximizing processor load: e.g. super computer

• Fairness: fair utilization of resources

• Compliance with deadlines

Constraint of RTS:

-Deadline has to be met

-Tasks have different importance

Important parameters: Period and Inter task dependencies

Generalisation of Scheduling:

- Offline-Scheduling:

  -Before execution of the system

  -Inflexible

  -Always maximum utilization

  -Low costs at execution

- Online-Scheduling :

  -During execution of the system

  -Flexible

  -Utilization depends on scheduling algorithm

  -Higher cost of execution

  -Normally priority based

Classification of Real Time scheduling algorithms

1. **Point of time**
   - during execution (online scheduling)
   - before execution (offline scheduling)
2. **Interruptibility**
   - Non-interruptible
   - Cooperative scheduling
   - Preemptive scheduling
3. **Time- or event-driven**
   - Time-driven scheduling
   - Event-driven scheduling

Real Time scheduling:

Without Priority :

- FIFO (First in First Out)
- Round Robin

Static Priority :

- EDD (Earliest Due Date First)
- RMS (Rate Monotonic Scheduling)

  -the task with the shortest period has the highest priority

-if another task with an even shorter period is scheduled, the current task is interrupted and the shorter task runs

- DMS (Deadline Monotonic Scheduling)

Dynamic Priority :

- EDF (Earliest Deadline First)

  -the task with the nearest deadline always comes first

  -if another task with an even closer deadline is scheduled, the current task is interrupted and the one with the closer deadline runs

Schedulabilitytest

There are two major types of test

- necessary tests (notwendig)
- sufficient tests (hinreichend)

Necessary means:

- if one of the appropriate necessity test fails then there is no feasible schedule!

Sufficient means:

- if you find at least one sufficient necessity test, than the task package is feasible schedulable.

Load test

Load U = 1 means the processor never idles

Load U > 1 there is no feasible schedule

Load U < 1 means: this test does not exclude that a feasible schedule may exist

<u>Process/ Thread</u>

- Process is that executing unit of compulation, which is controlled by some processes of the OS
- Process control block

- Threat two process can run at the same time but they do not share memory. Suppose we promised software

Difference:

Process:

-Process can be divided into multiple threads

- Each process has its own memory space

Threat:

-Threads cannot be sub divided

- It is easy to create a thread

What is a minor page fault?

If the page is loaded in memory at the time the fault is generated, but is not marked in the memory management unit as being loaded in memory, then it is called a minor or soft page fault.

What is a race condition? How can you avoid them?

- When two or more tasks access common data, this may lead to a "Race Condition".

The result of a Race Condition depends on the relative progress of the different tasks.

Avoidance of Race Conditions

1. Race conditions can be avoided by allowing access to a critical section for one task only at a time. This is called mutual exclusion (gegenseitiger Ausschluss).
2. The mathematical instrument to solve this issue is called a **semaphore**.

Mutex: A Semaphore with N=1 is called Binary Semaphore or a Mutex.
It tells you only free or not free and
The size of the queue respectively.

Absolute time vs. relative time

Absolute time: time expressed with numbers
Relative time: time expressed as a comparison

Sensors : A sensor converts a physical or chemical measure into electrical signals.

Interrupt

An interrupt is an event that stops a currently running program A, starts another program B and returns to the program A that is continuing exactly where it was interrupted.

There are two types of interrupts:

1. hardware interrupts and

2. software interrupts.

Hardware interrupts are used by devices to communicate that they require attention from the operating system.

A software interrupt is caused either by an exceptional condition in the processor itself, or a special instruction in the instruction set which causes an interrupt when it is executed.

Types of Interrupts

1. Level-triggered interrupts: is an interrupt signaled by maintaining the interrupt line at high or low level

2. Edge-triggered interrupt: is an interrupt signaled by a level transition on the interrupt line, either a falling edge (high to low) or a rising edge (low to high).

3. Hybrid: the hardware not only looks for an edge, but it also verifies that the interrupt signal stays active for a certain period of time

4. Message signaled: the device signals its request by sending a short message over some communication medium

Latency time

Latency time is the time that is needed for computer internal processing (operation system and processor internal) and leads to couple of time delay between an initiation and the start of the requested task.

There are

• Interrupt-Latency • Task-Latency • Kernel-Latency • Preemption Delay (Verdrängungszeit)

Petri net: place / transition net is one of several mathematical modeling to describe dist.system. Have places and transitions having directed are in between Places gave marks /tokens teach place represent a state.

Safeness: is related to bounded memory capacity

Liveness: is related to the absence of deadlocks. The transition is deadlocked if it is not possible to fining the token.

Microcontroller: you can find micro controller in difference d

What are typical components of a micro controller?

3 components: - processes core, access

-input/output, to processing external data

How do you use a micro controller?

Working:

1. Semiconductor Physics: silicon is doped to create semiconductor properties
2. Transistors: semiconductor material is arranged for form transistors

3. CMOS: transistors come together to form circuits that perform boolean algebra
4. Logic: both combinational and sequential logic used to form the major building blocks of a CPU
5. Microcontroller Architecture: digital logic circuitry comes together to execute instructions

What are the three ways to do priority inversion for RMS + Ressource

There are three possibilities to „heal“ Priority Inversion:
1. NPCS (Non Preemptive Critical Section) „Unterbrechungssperre“

2. Priority Inheritance Protocol (PIP)

3. Priority Ceiling Protocol (PCP)

Calculate the load of the following tasks(period, execution time):
T1(10,2)
T2(25,4)

And calculate to total load:

Load= execution time/period

T1 Load = 2/10= 0,2
T2 Load=4/25=0,16