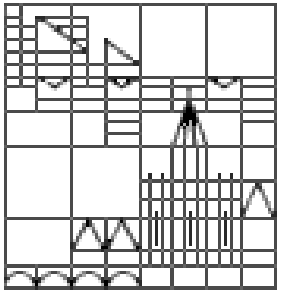
Software Engineering in Embedded Systems

Stephan Heidinger

Seminar: Software Engineering
Fachbereich für Informatik und Informationssysteme
Universtität Konstanz

19. January 2012

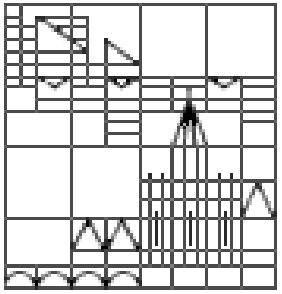


Embedded Systems - What's that? - I

Definition

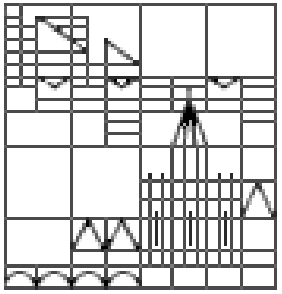
"An **embedded software system** *is part of a hardware/software system that reacts to events in its environment. The software is 'embedded' in the hardware. Embedded systems are nominaly real-time systems."*

Software Engineering, p.561, Edited by Ian Sommerville, Ninth Edition



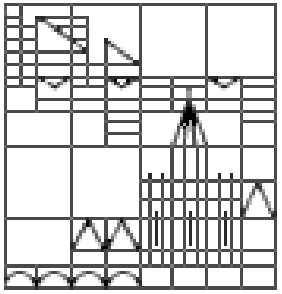
Embedded Systems - What's that? - II

- Embedded Systems: ...
  - ...respond to physical world
  - ...respond in real time ("have a *deadline*")
  - ...often have little resources
  - ...run on special purpose hardware
  - ...run in real-time operating systems



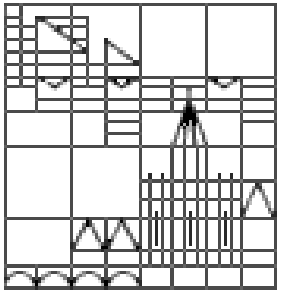
Embedded Systems - What's that? - III

- Examples for Embedded Systems:
  - airbag
  - cell phone / 'modern' phone
  - burglar alarm
  - (fully automatic) coffee machine
  - danger detection
  - ...



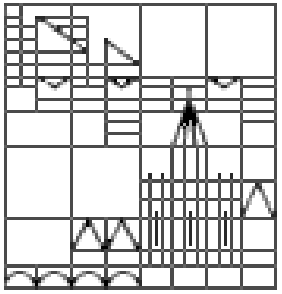
Motivation

- Why embedded systems:
  - Embedded Systems are everywhere!
  - There are probably more Embedded Systems than computers out there!
  - Man, they must be important.
  - There sure is some money in this.
- I did an internship producing an embedded system.



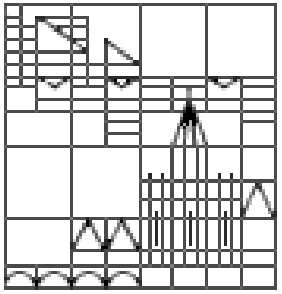
Problems

- Problems in Embedded Systems:
  - deadlines
  - environment
  - continuity
  - direct hardware interaction
  - safety & reliability

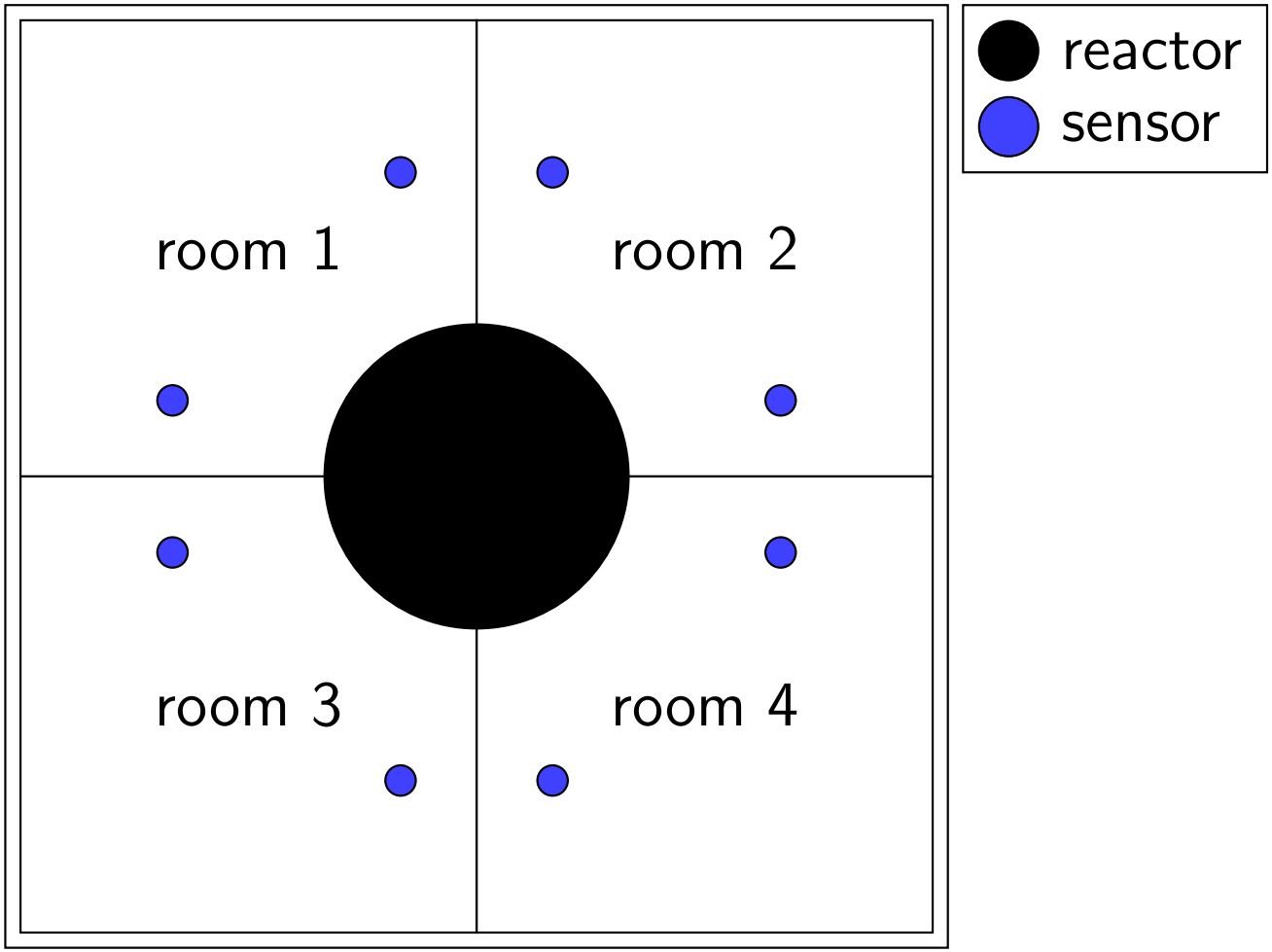


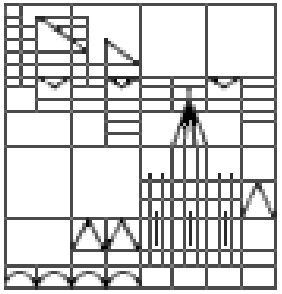
Embedded Systems Design

- design steps
  - platform selection
  - special purpose hardware
  - stimuli:
    1. periodic stimuli
    2. aperiodic stimuli
  - timing analysis
  - process design
  - algorithm design
  - data design
  - process scheduling



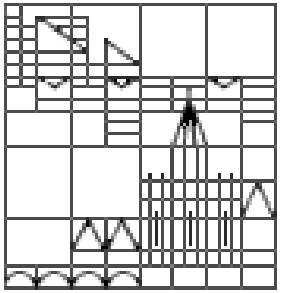
Example: radiation warning system





Example: Stimuli-List of a radiation warning system

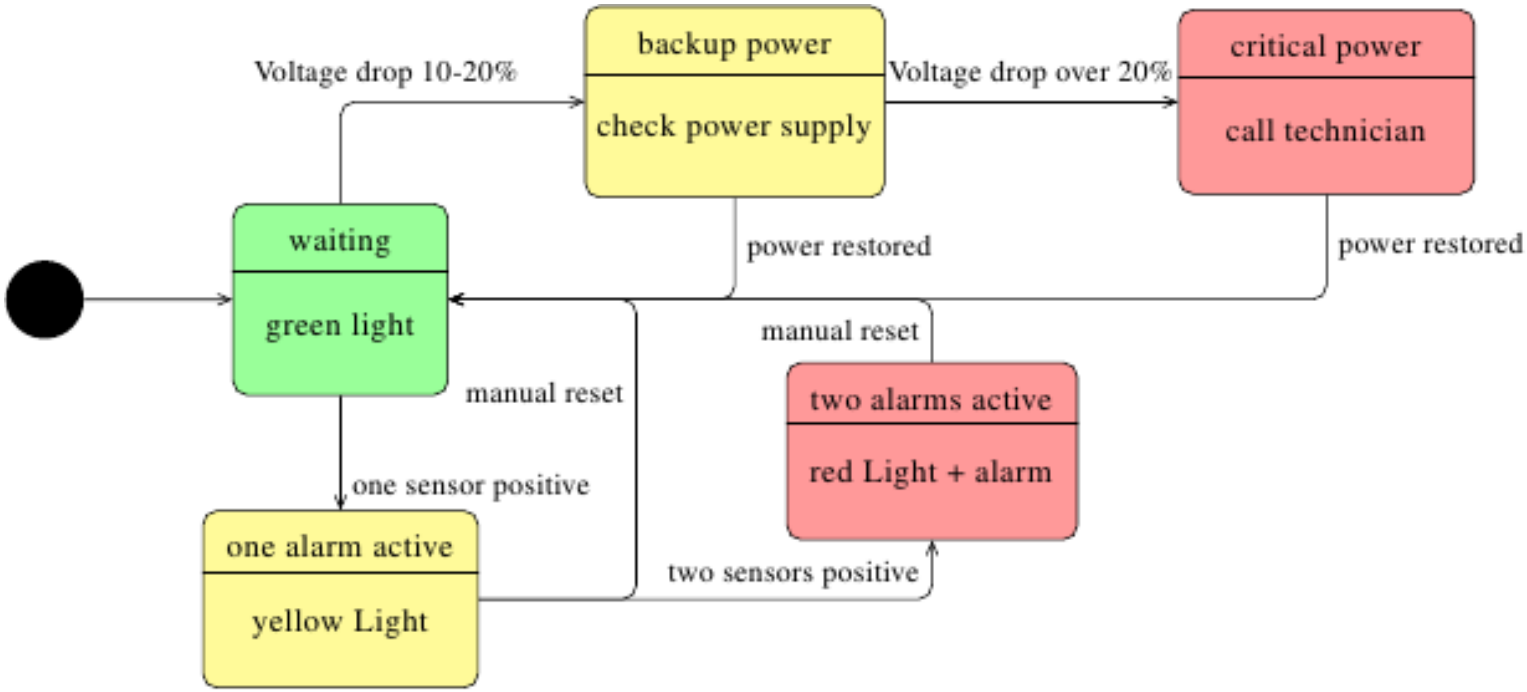
Stimulus	Response
single sensor positive	flash yellow light around sensor
both sensors in one area positive	flash red light in area, sound acoustic alarm in area
Voltage drop of 10-20%	switch to backup power; run power supply test
Voltage drop of more than 20%	switch to backup; run power supply test; call technician

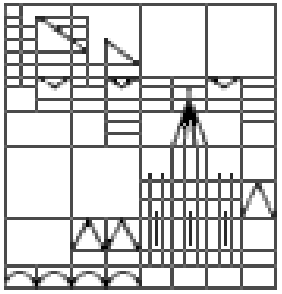


Embedded system modeling

- Embedded Systems are often built as state machines.

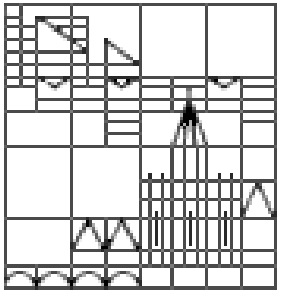
$\Rightarrow$ UML state diagrams





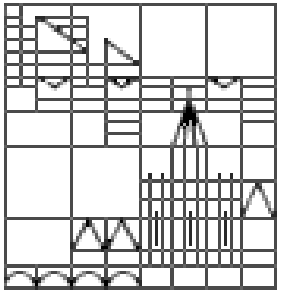
Programming language

- program has to be...
  - ...fast (i.e. C, Assembler)
  - ...concurrent (i.e. C++, real time Java, ...)
- speed looses importance
- it's up to you in the end ...



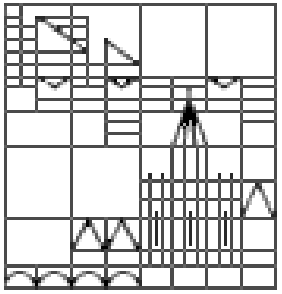
Architectural patterns

- Architectural patterns are used to describe a system in an abstract way and help to understand the architecture.
  - Observe and React
  - Environmental Control
  - Process Pipeline



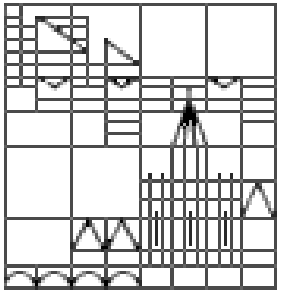
Observe and React

- Observe and React
  - monitor the system with a set of sensors
  - display something
  - primarly used in: Monitoring Systems



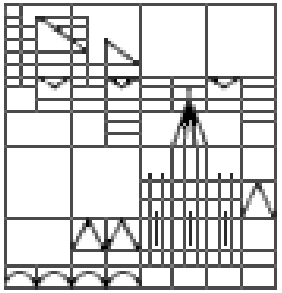
Environmental Control

- Environmental Control
  - monitor the system and react to any changes
  - Used when there is no requirement for user interaction. . .
  - . . . or no time for the user to interact . . .
  - . . . no way a user can interact . . .
  - . . . or there is too much information for users to process.



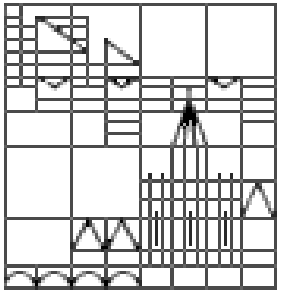
Process Pipeline

- Process Pipeline
  - transform data
  - often huge amounts of data to be converted in real time
  - data aquisition system: storing of data may need to be fast



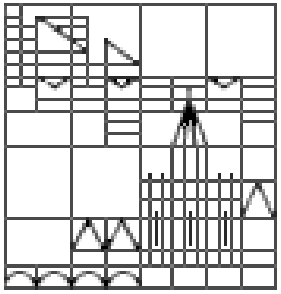
Timing Analysis - I

- timing analysis
  - Correctness of systems depends not only on result, but also on the time at which the result is produced.
  - How often does each process need to be executed?
  - aperiodic stimuly $\Rightarrow$ make assumptions

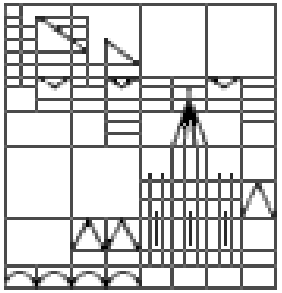


Timing Analysis - II

- Consider:
  - deadlines
  - frequency
  - execution time

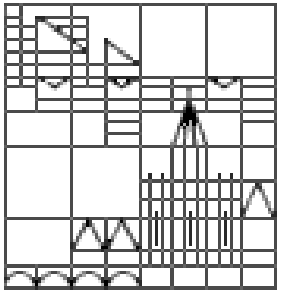


Stimulus/Response	Timing requirements
voltage drop	switch to backup: 50ms
sensor reaction	poll twice a second
turn on light	500ms
call technician	5000ms



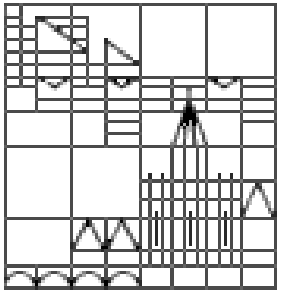
Real-time operating systems

- normal operating systems not feasible
- special “real-time operating systems” exist
- RTOS must include:
  - real-time clock
  - interrupt handler
  - process manager: scheduler & resource manager
  - dispatcher



30 minutes in short

- What you should (at least) remember:
  - Embedded Systems react to events in real time.
  - Embedded Systems are a set of processes reacting to stimuli
  - State models help understanding the System.
  - Architectural patterns can be used to help in designing the system.
  - Always do timing analysis in (hard) Embedded Systems.



Questions?

Questions?