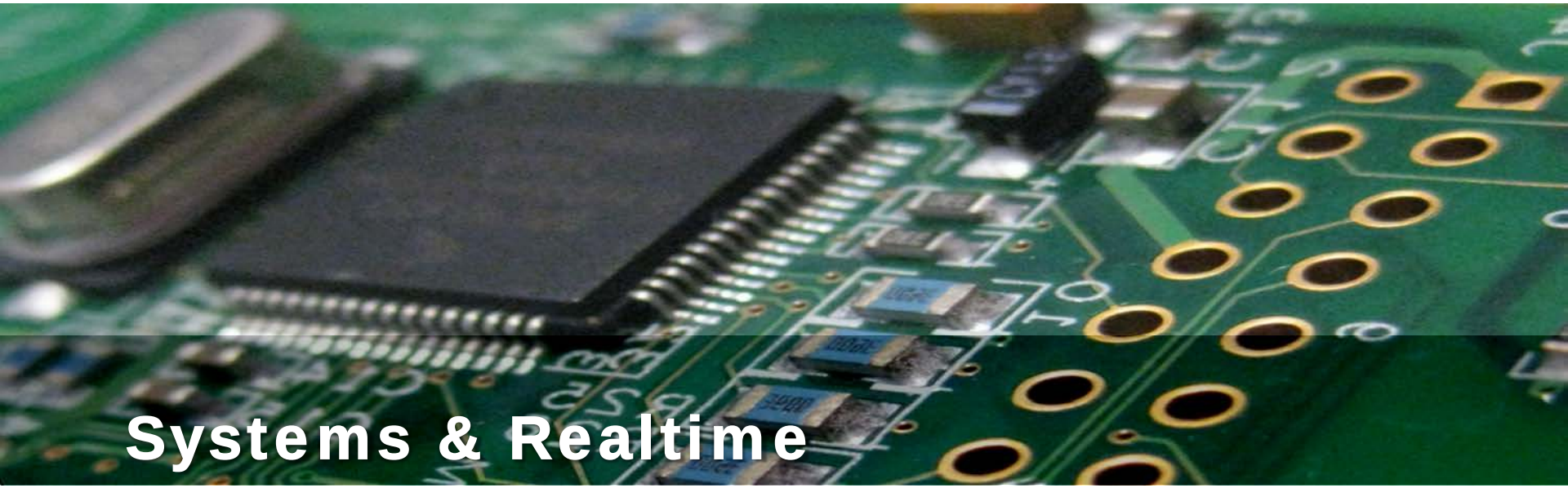
Lucerne University of
Applied Sciences and Arts
HOCHSCHULE
LUZERN
Technik & Architektur



Systems & Realtime

*"Yes, we are going to build a system for our INTRO Demonstrator.
And you bet it has to be real-time too!"*

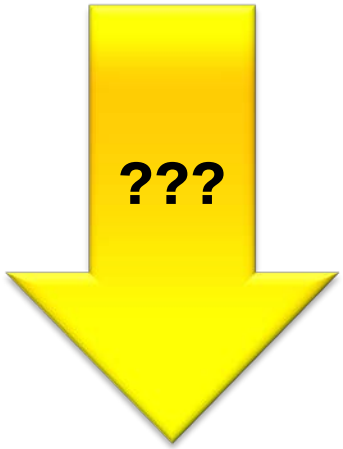
Prof. Erich Styger
erich.styger@hslu.ch
+41 41 349 33 01

Scriptum:
Systems, Realtime

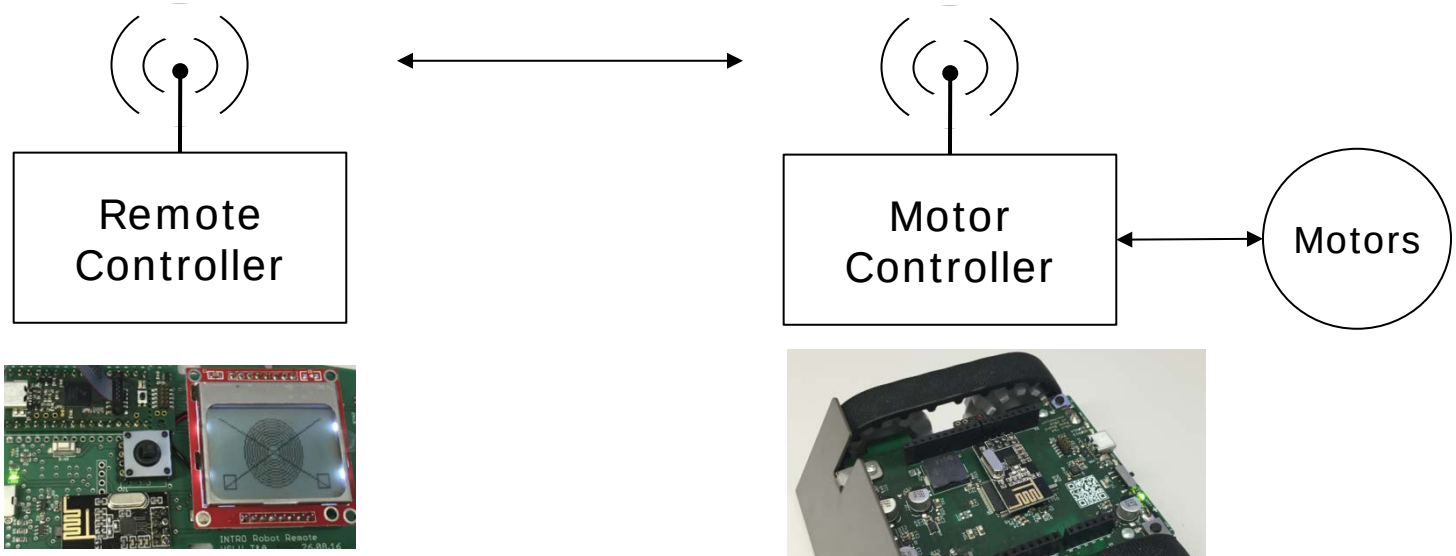
Learning Topics

- Problem: Understand and decompose the system

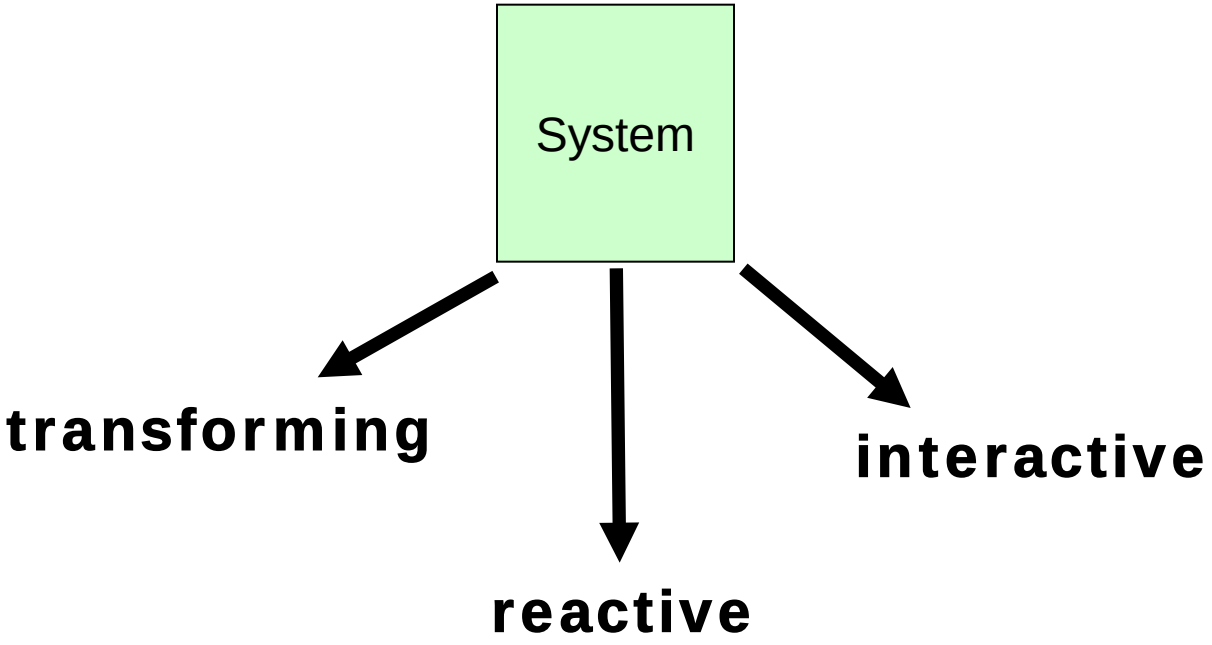
- Systems
- Classification
- Realtime
- Timeliness
- Reaction time



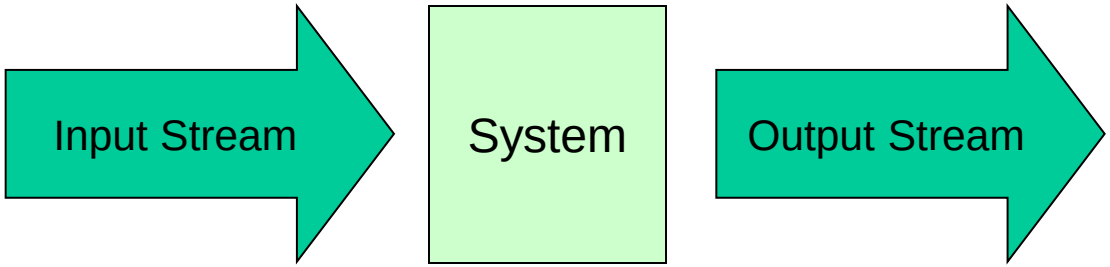
Intro Systems



Embedded System – System?

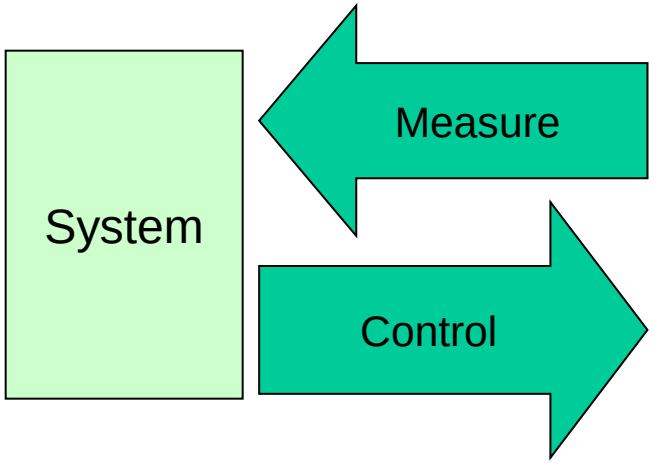


Transforming Systems



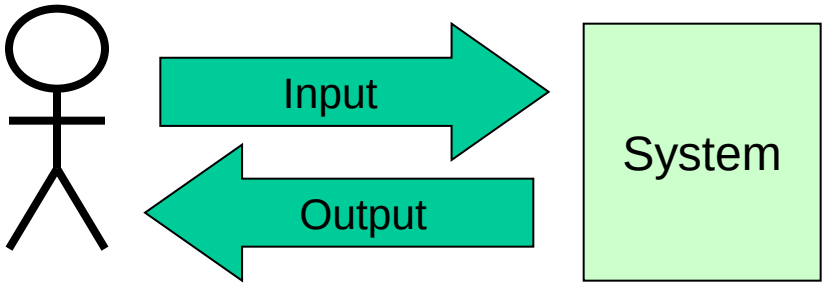
- Typical
  - Data processing quality
  - Throughput
  - Optimized system load
  - Optimized Memory Usage

Reactive Systems



- Typical
  - External events are driving system
  - Guaranteed response time
  - Control loop
  - Realtime

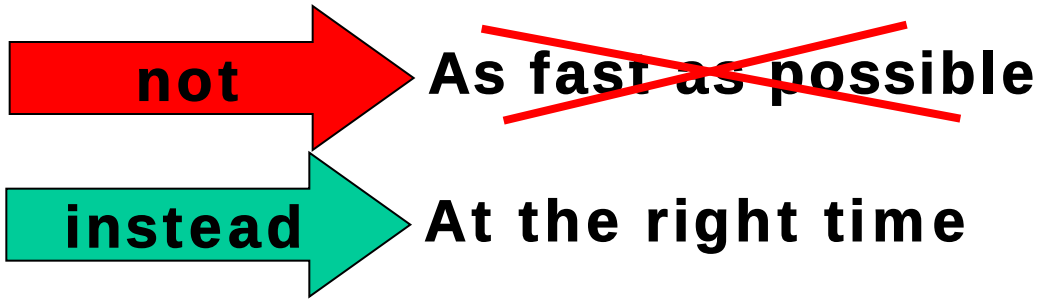
Interactive Systems



- Typical
  - Short response time
  - High system load
  - Human-Machine Interaction (HMI)

Realtime

- System interaction with the environment
- Different speed domains of events
- System has to deal with the time constraints of the real world (real time)
- Realtime ➔ real time



Realtime Processing

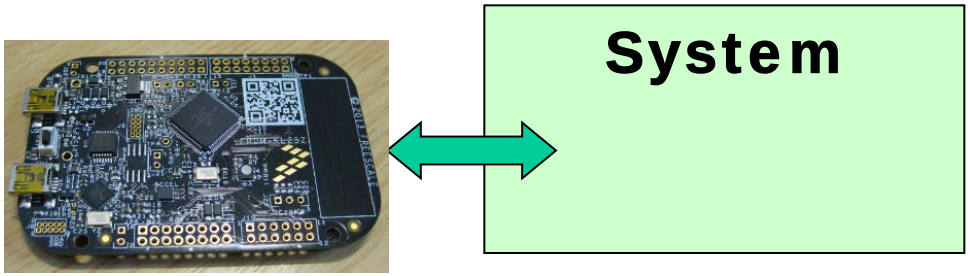
- Systems
  - Transforming
  - Reactive
  - Interactive
- Realtime System Requirements
  - Correctness
  - External time conditions compliance
- Examples
  - Train system schedule computation
  - Railroad switch

Source: Wikipedia

Source: Wikipedia

The correct result at the correct time

Realtime for Computer Systems



- Computer is connected with a system
- Computer has to comply with the real time
  - No time short cut
  - No time expansion
  - Regardless current system load

Realtime

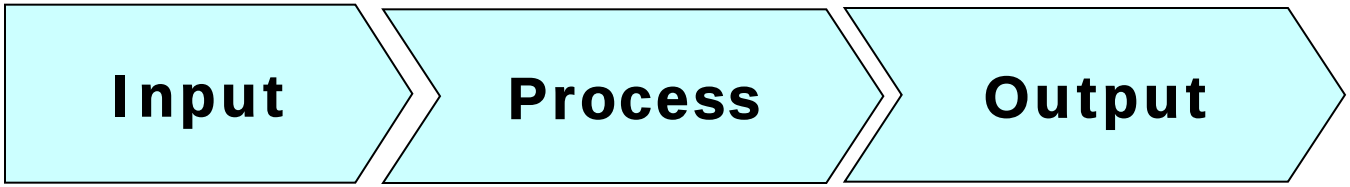
A computer is classified as Realtime if it can react on external events in the real world:

- **With the correct result**
- **At the correct time**
- **Independent of current system load**
- **In a deterministic and foreseeable way**

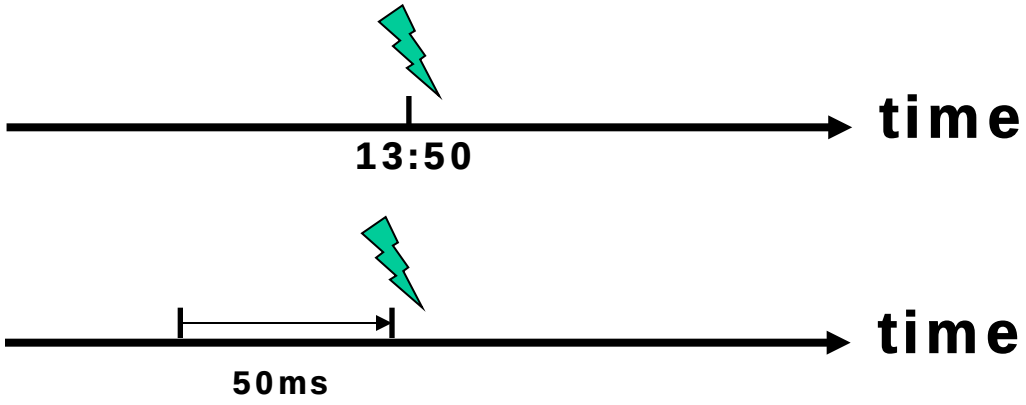
Timeliness
Concurrency

Timeliness

- For all processing stages



- Categories
  - absolute
  - relative



Concurrency

- Real World: is concurrent
- Problem: Computers are sequential

For slow and few tasks

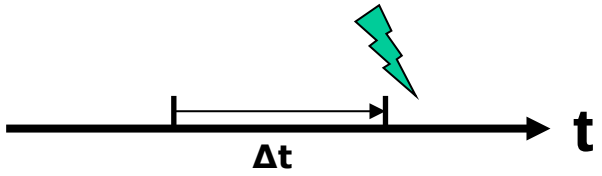
Multitasking, Nesting

‘Simultaneous’ Processing

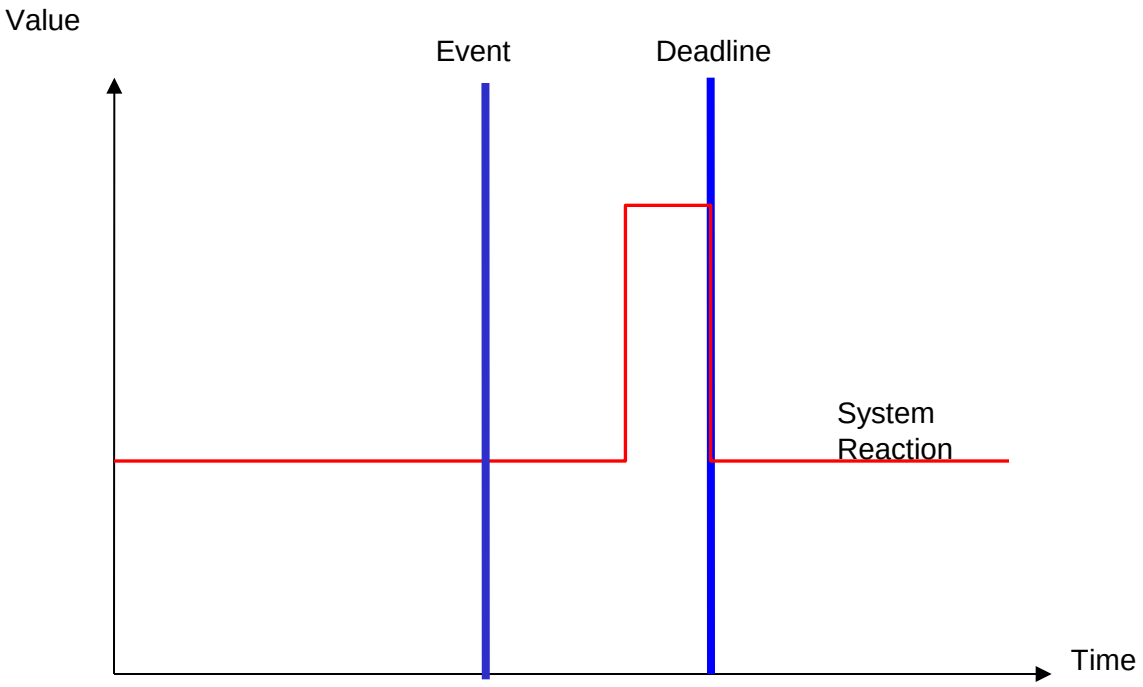
Does NOT work for many and fast tasks

Reaction Time

- Realtime systems require a defined reaction time
  - Absolute
  - Relative
- Interactive Systems
  - seconds
- Reactive & Transitive Systems
  - Milliseconds
  - Microseconds
- System load defined with
  - Number of concurrent events/tasks
  - Interval of events
  - Reaction time for events
  - Processing time for events

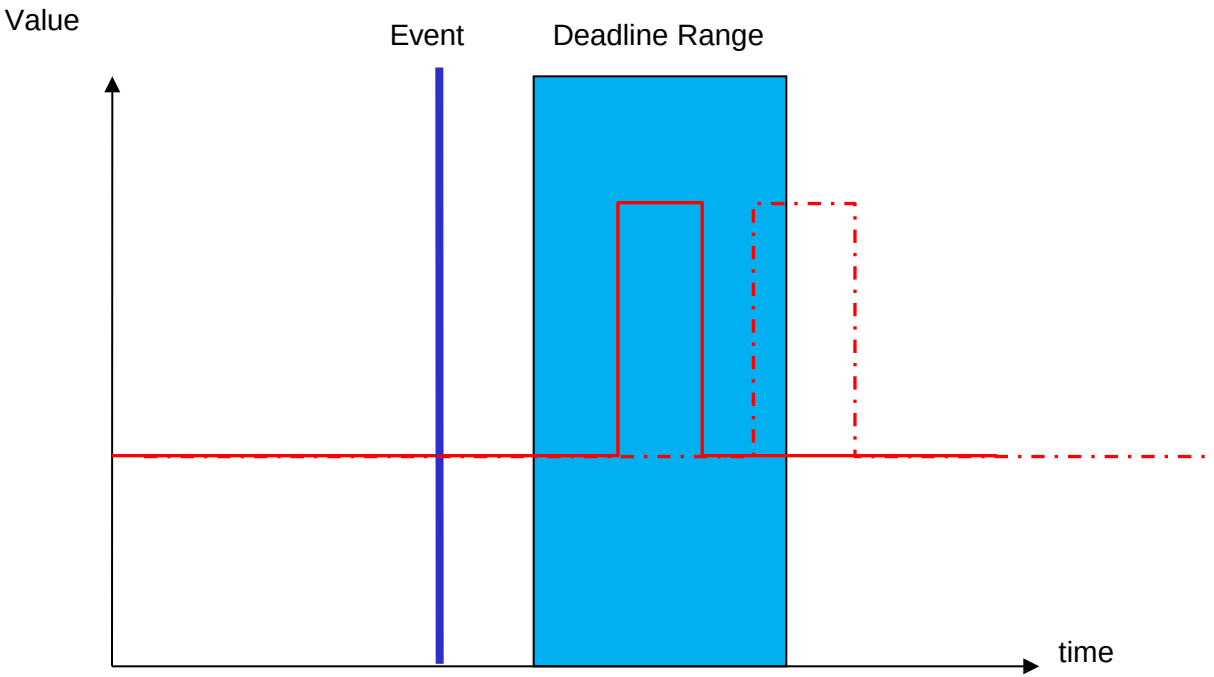


Hard Realtime



- Incorrect if correct result does not meet time conditions

Soft Realtime



- Degradation, if correct result does not meet the time conditions

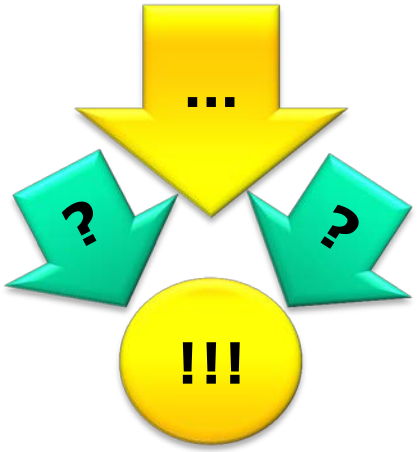
Summary

- *Problem: Understand and decompose the system*

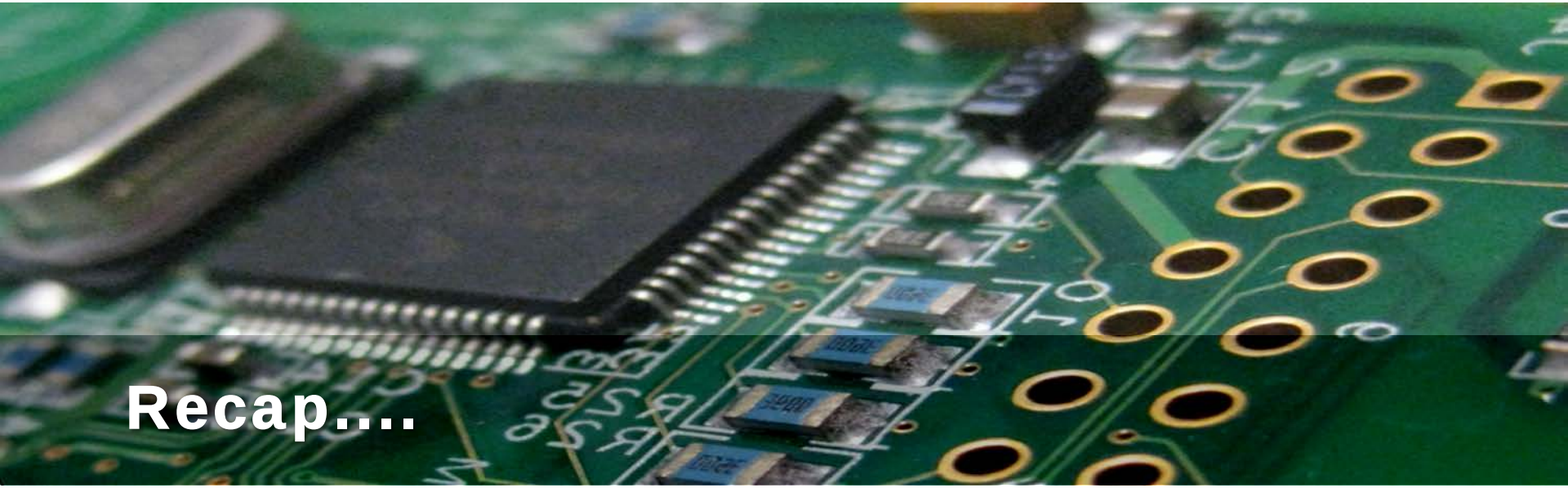
- Systems
  - Reactive
  - Interactive
  - Transformative

- Realtime: hard & soft

- Characterization of different systems



Lucerne University of
Applied Sciences and Arts
HOCHSCHULE
LUZERN
Technik & Architektur

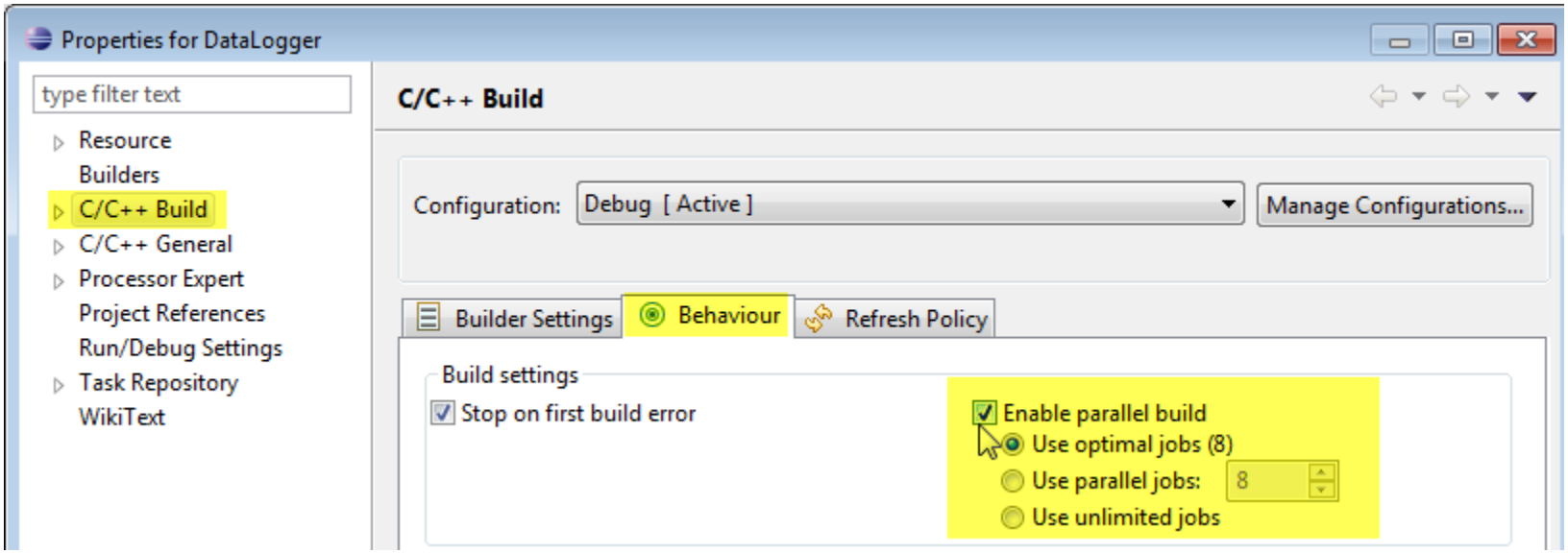


Recap....

Prof. Erich Styger
erich.styger@hslu.ch
+41 41 349 33 01

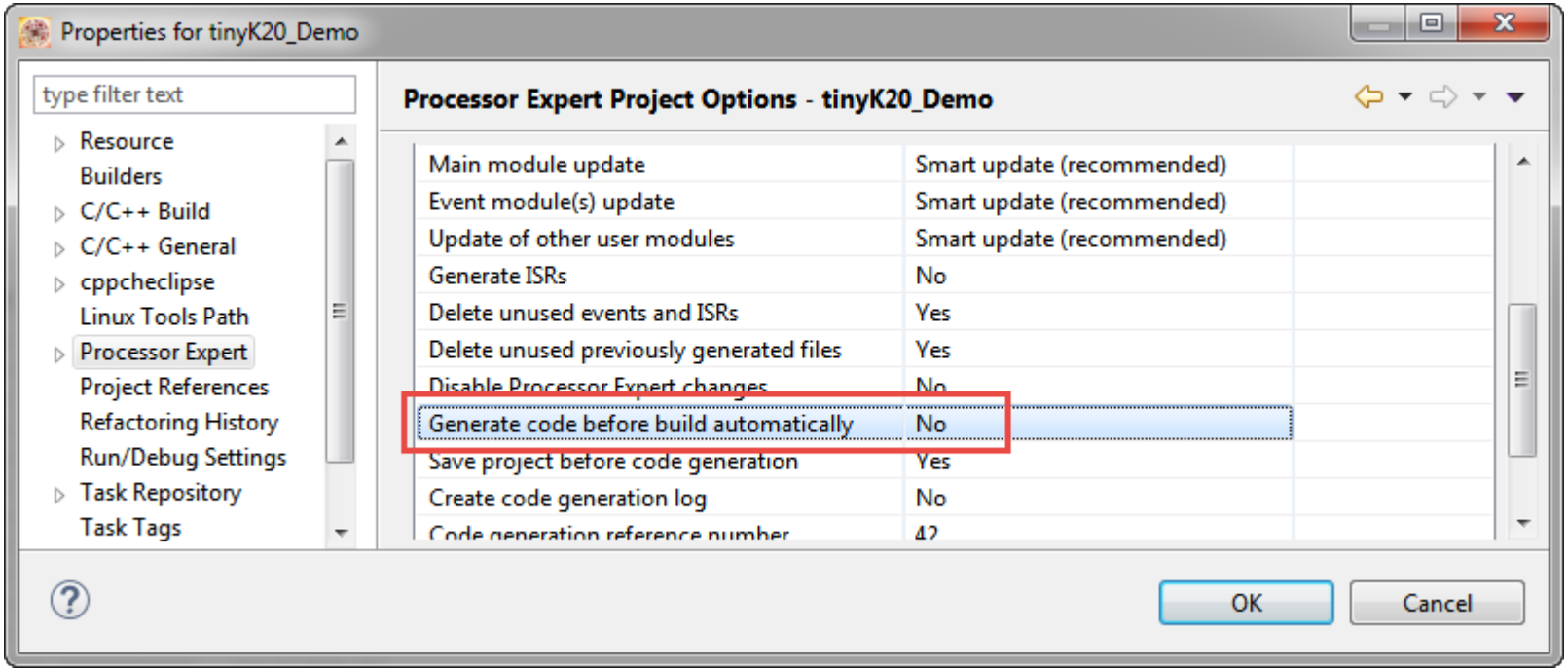
Tips & Tricks: Parallel Build

- Project Setting
- http://mcuoneclipse.com/2014/05/25/reducing-build-time-in-eclipse-with-parallel-build/
- http://mcuoneclipse.com/2012/04/26/using-parallel-builds-what-is-optimal/
- http://mcuoneclipse.com/2013/01/09/reducing-the-build-time-with-gcc-for-arm-and-codewarrior/



Tips & Tricks: Code Generation

- Project setting
- http://mcuoneclipse.com/2013/10/19/how-to-avoid-slow-processor-expert-projects/



Tips & Tricks: Debug without Build

- Workspace setting
- http://mcuoneclipse.com/2012/10/30/speeding-up-the-debug-launch-in-codewarrior/