System-Level Modeling with Open Source Tools

Jan Madsen, Professor, DTU

February 29, 2012, Nurenberg, Germany

Acknowledgement:
Mikkel K. Jakobsen (DTU), Seyed H.A. Niaki (KTH),
Ingo Sander (KTH), Jan Hansen (AuditData)

Design Challenge

- Moore's law leads to
  - Increasingly more powerful and complex architectures
  - Increasingly more advanced and demanding applications
- Verification costs are continuously increasing

We have already problems with complexity today, so how will we design tomorrow's systems?

Outline

- The SYSMODEL project
- Formal System Design with ForSyDe
- Case-study
- Conclusions

SYSMODEL

THE SYSMODEL PROJECT

The SYSMODEL Project

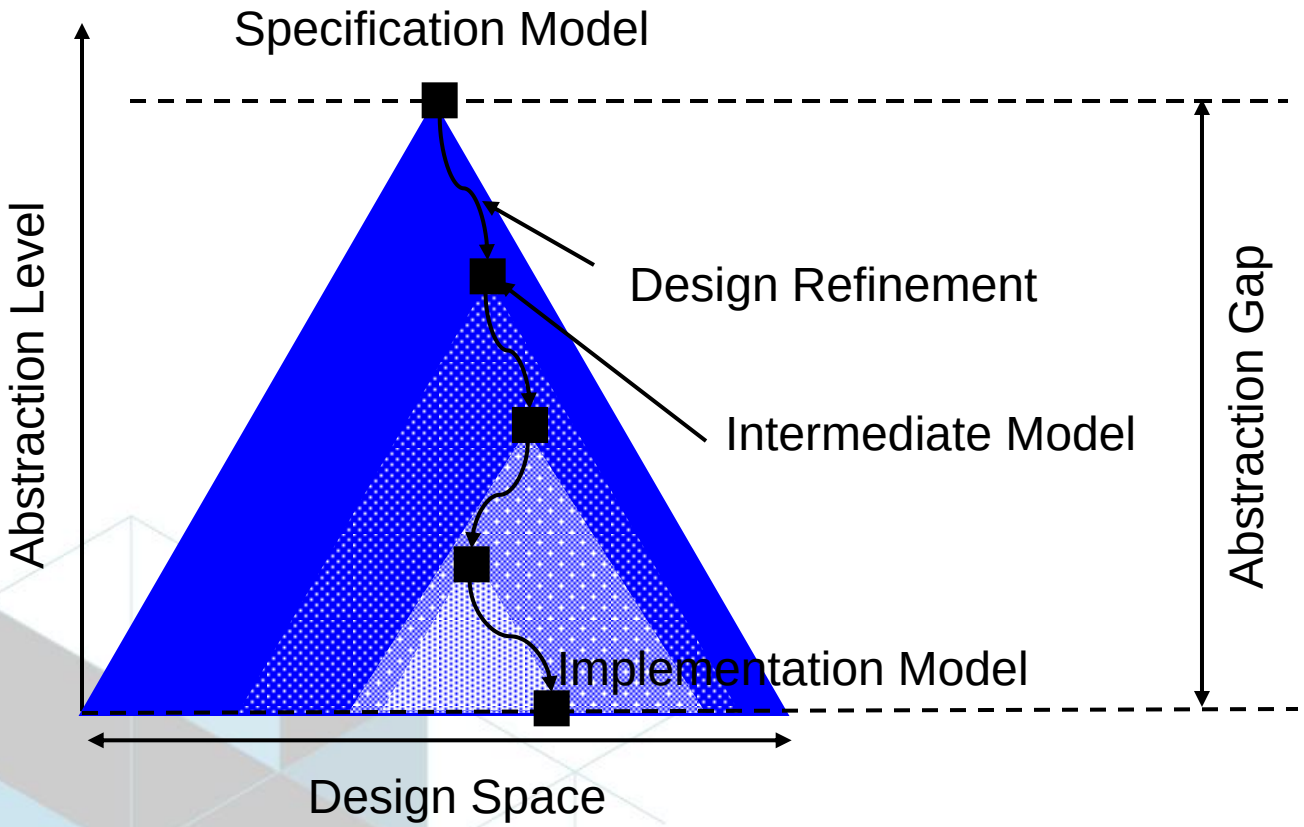
- Raise the level of abstraction for SMEs designing embedded systems by developing system-level modeling techniques:
  - Provide SMEs with system level modeling tools
  - Focus is on time and power critical, heterogeneous systems
  - Allow cost-efficient mapping of applications onto an embedded platform
  - While respecting constraints in terms of resources (time, energy, memory, etc.), safety, security and quality of service

Heterogeneity

- SYSMODEL focuses on the design of heterogeneous systems
  - Heterogeneous applications
    - Analog signals, digital signals
    - Variety of protocols
    - Control vs data-flow
  - Heterogeneous architectures
    - Digital HW, Analog HW, Software
- Most industrial tools focus on a single domain

How do we deal with heterogenity?

Abstraction Gap



How can a design methodology help us to arrive at a correct implementation?

Requirements on Design Methodologies

- Ad hoc methods will not bridge the design-productivity gap
- Formal approach is needed to support
  - Verification
  - Design exploration
  - Design refinement
  - Synthesis

But the world is not that simple...

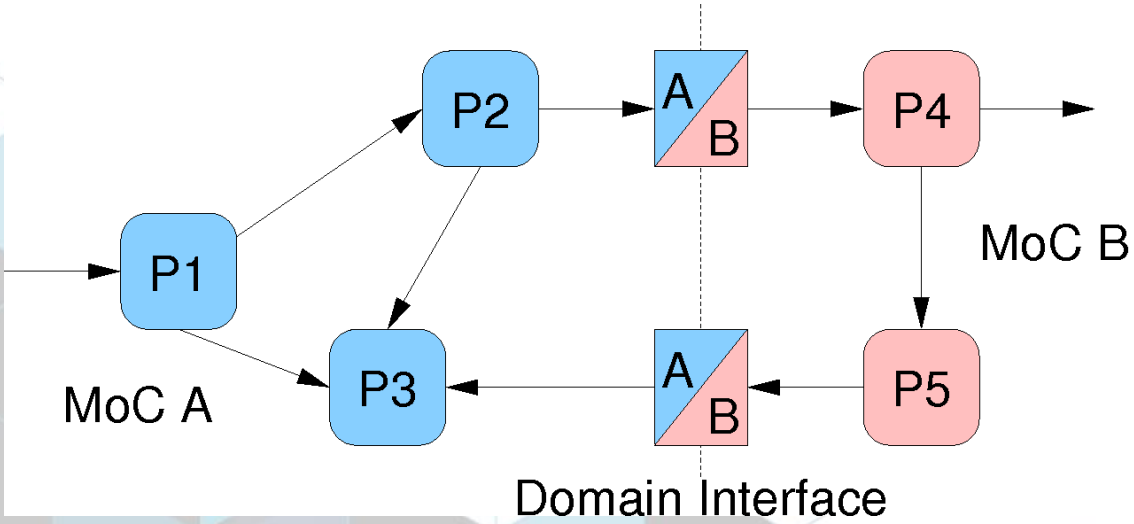
- Companies have invested a lot of effort into their existing design methodologies
- In particular companies want to resuse existing models, designs, and legacy code
- Also the design flow should support the integration of third-party IPs

SYSMODEL

FORMAL SYSTEM DESIGN WITH FORSYDE

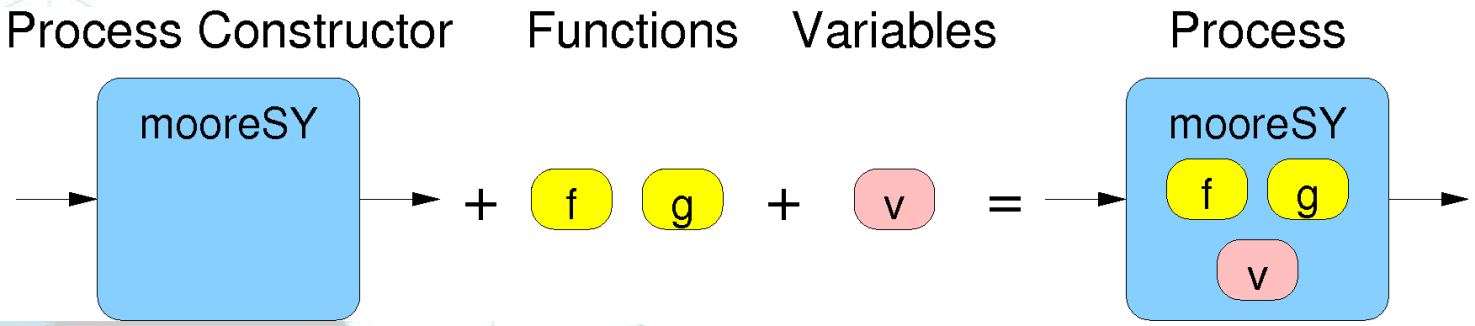
ForSyDe System Model

- A system is modeled as hierarchical concurrent process network
- Processes of different models of computation (MoC) communicate via domain interfaces
  - Supported MoCs: Synchronous, Untimed (SDF), Discrete Time, Continuous Time



Designing in ForSyDe Processes

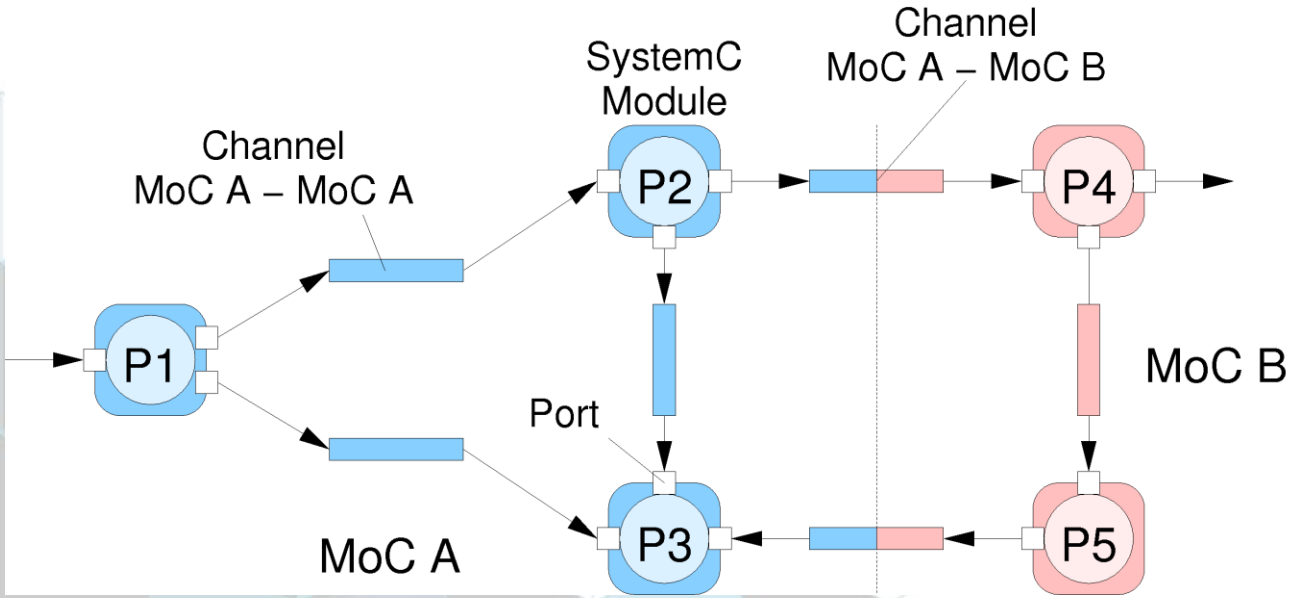
- A process is always designed by means of a process constructor
- The process constructor defines the communication interface of the process
- The process constructor takes side-effect free *functions* and *variables* as arguments and returns a process



ForSyDe processes and system models are deterministic!

SYSMODEL System Model ForSyDe compliant SystemC

- Project develops SystemC libraries that
  - are based on the formal foundations of ForSyDe
    - Concept of process constructor
    - Well-defined execution semantics
- Project develops modeling guidelines



SYSMODEL

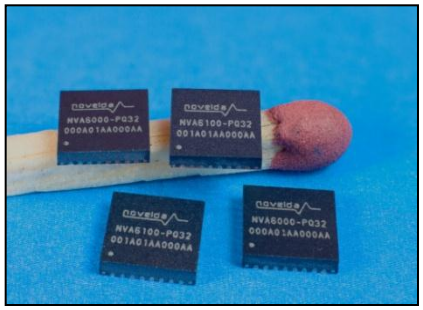
CASE-STUDY

SYSMODEL Case-Studies

AuditData Case-Study

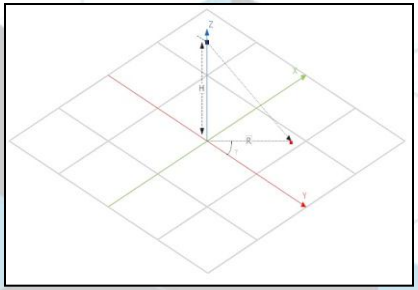
AuditData A/S
Audio calibration device

Novelda AS
Nanoscale impulse radio

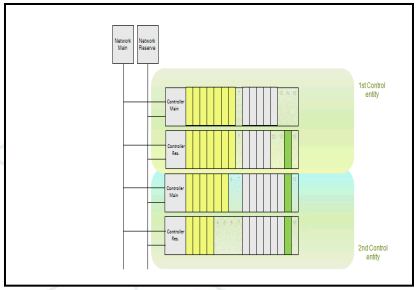


Nito Telecom AB
Architecture for VoIP

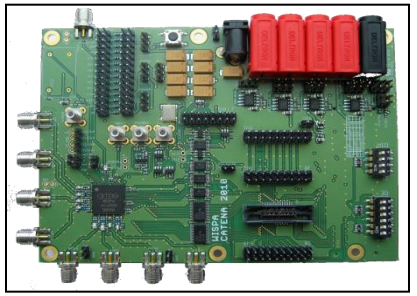
DA-Design Oy
Device stabilization



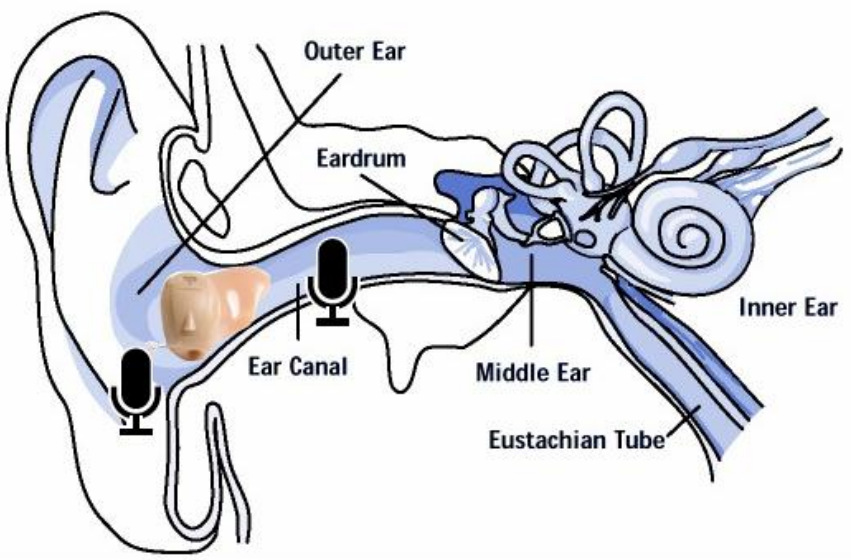
Finnelpro Oy
UART-based protocol



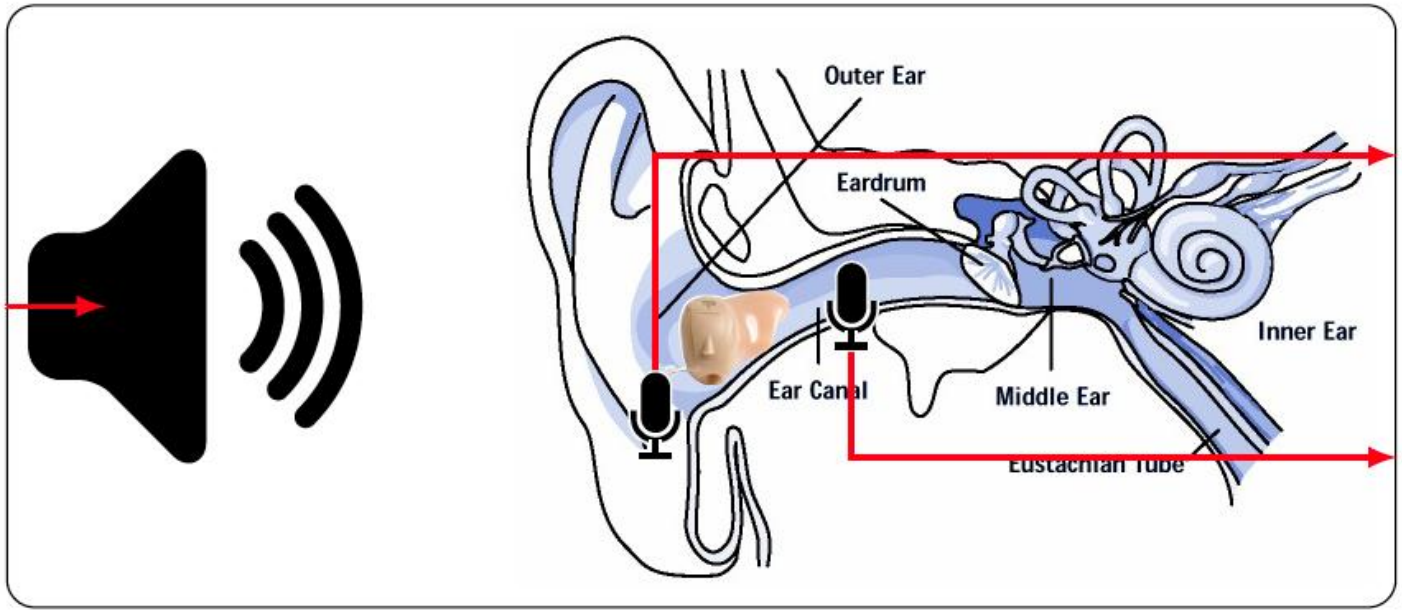
Catena AB
WISPA platform



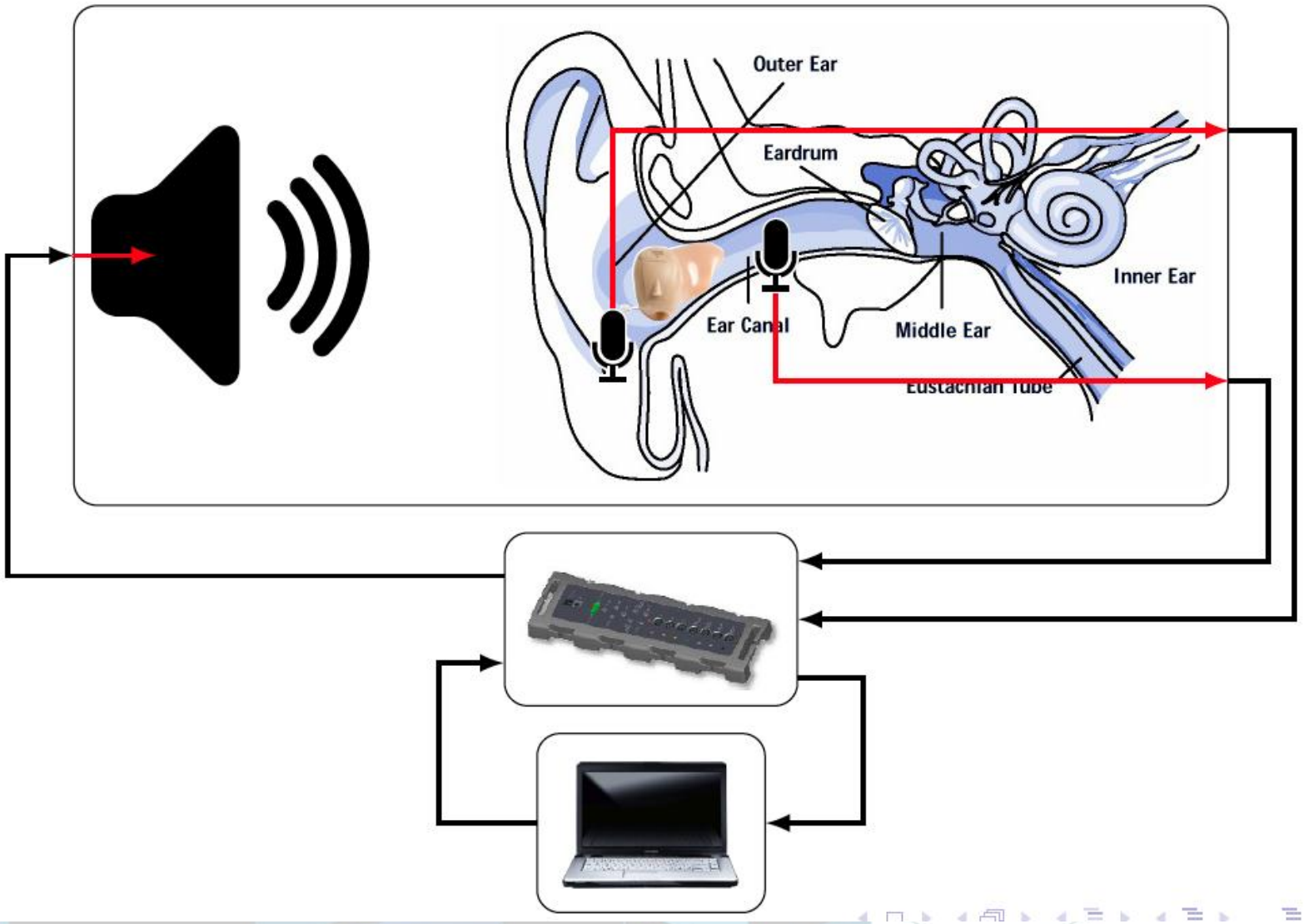
Hearing aid adjustment device



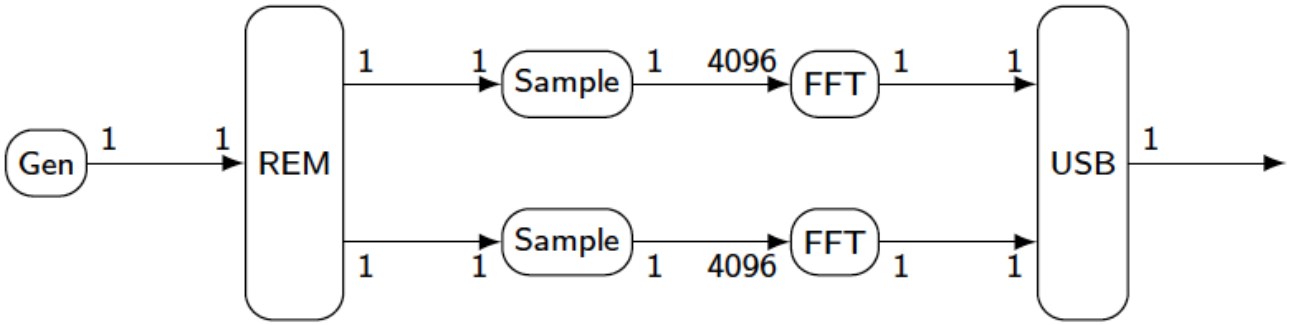
Hearing aid adjustment device

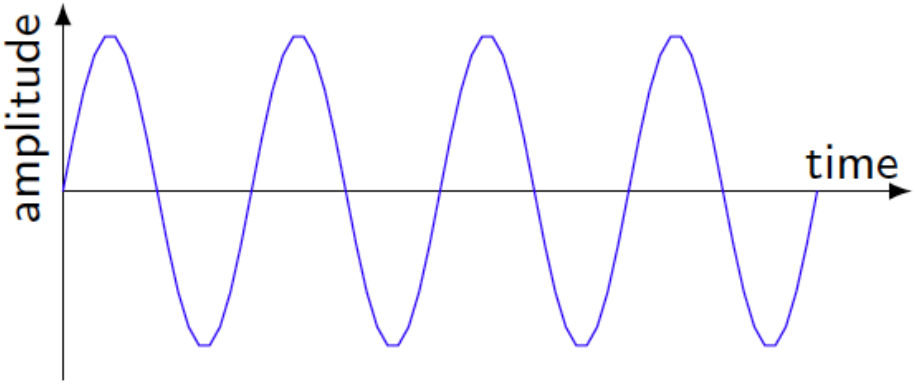


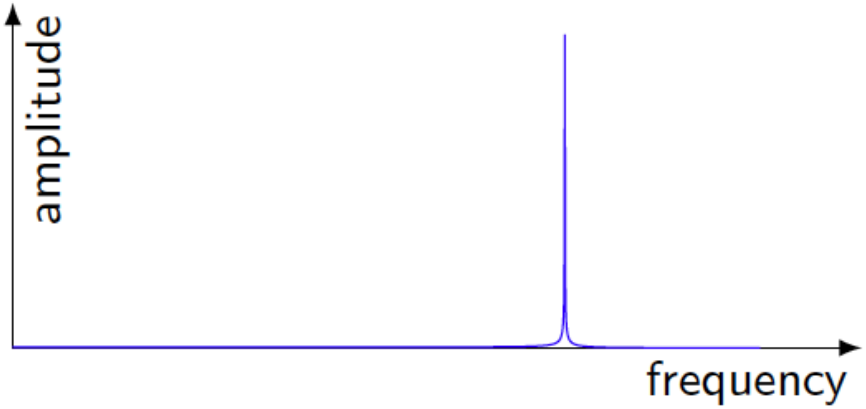
Hearing aid adjustment device



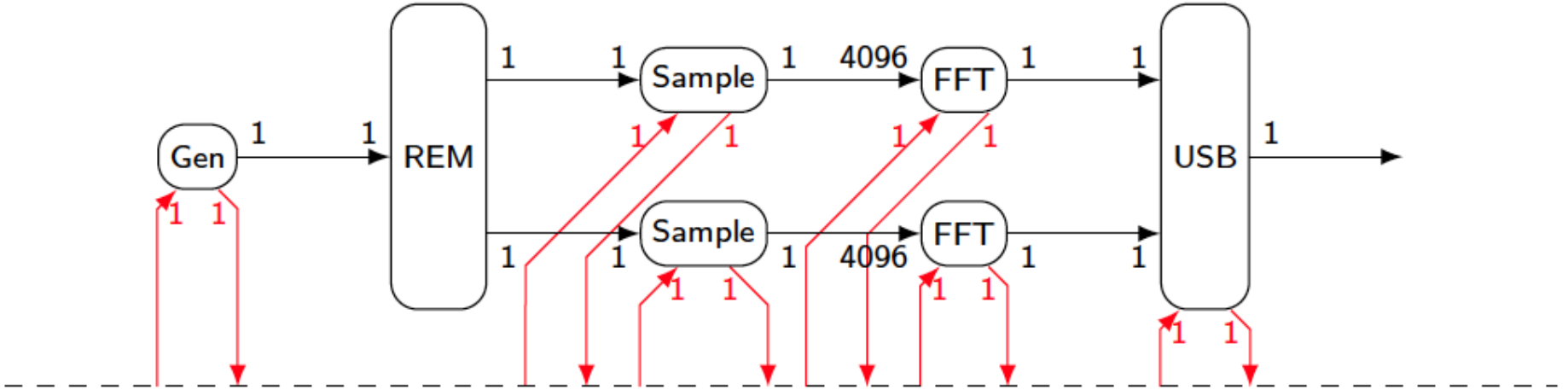
Behavioral model



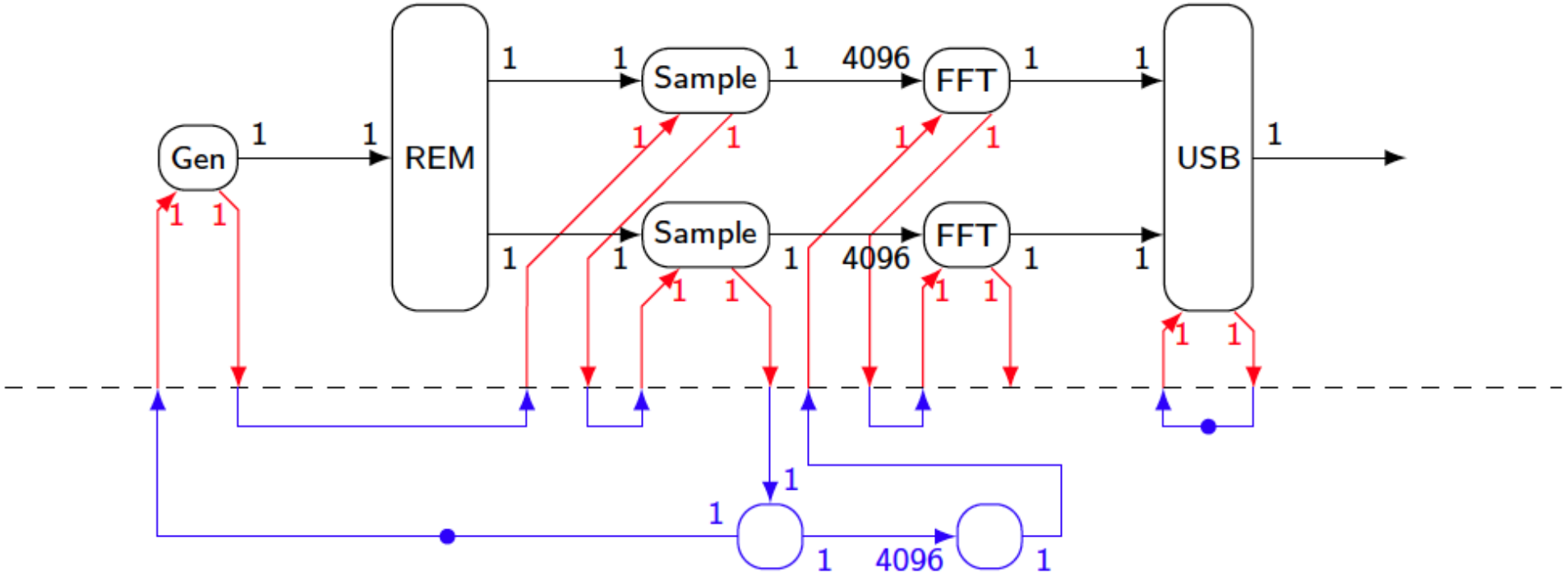




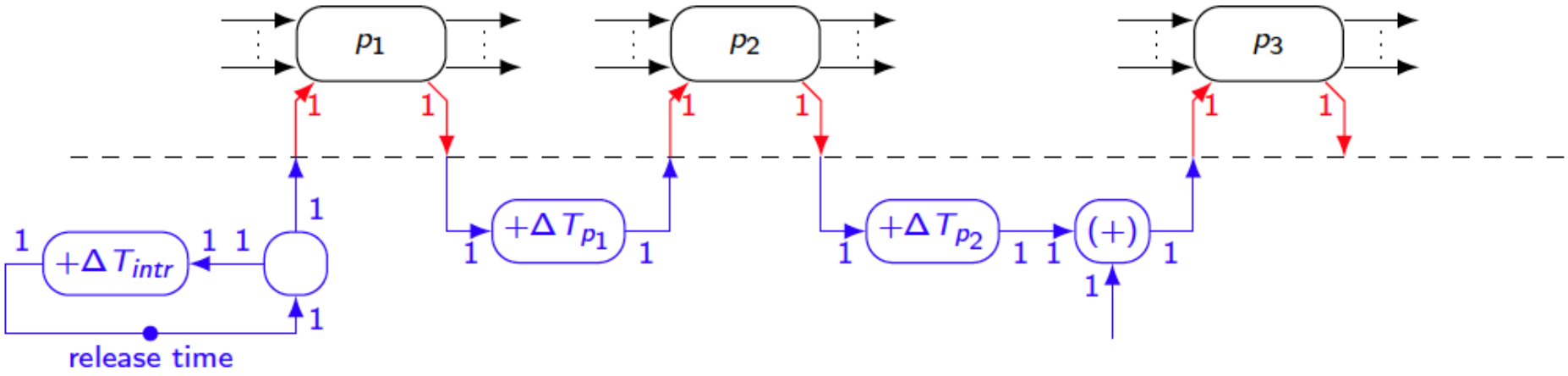
Concepts for execution dependencies related to OS/HW



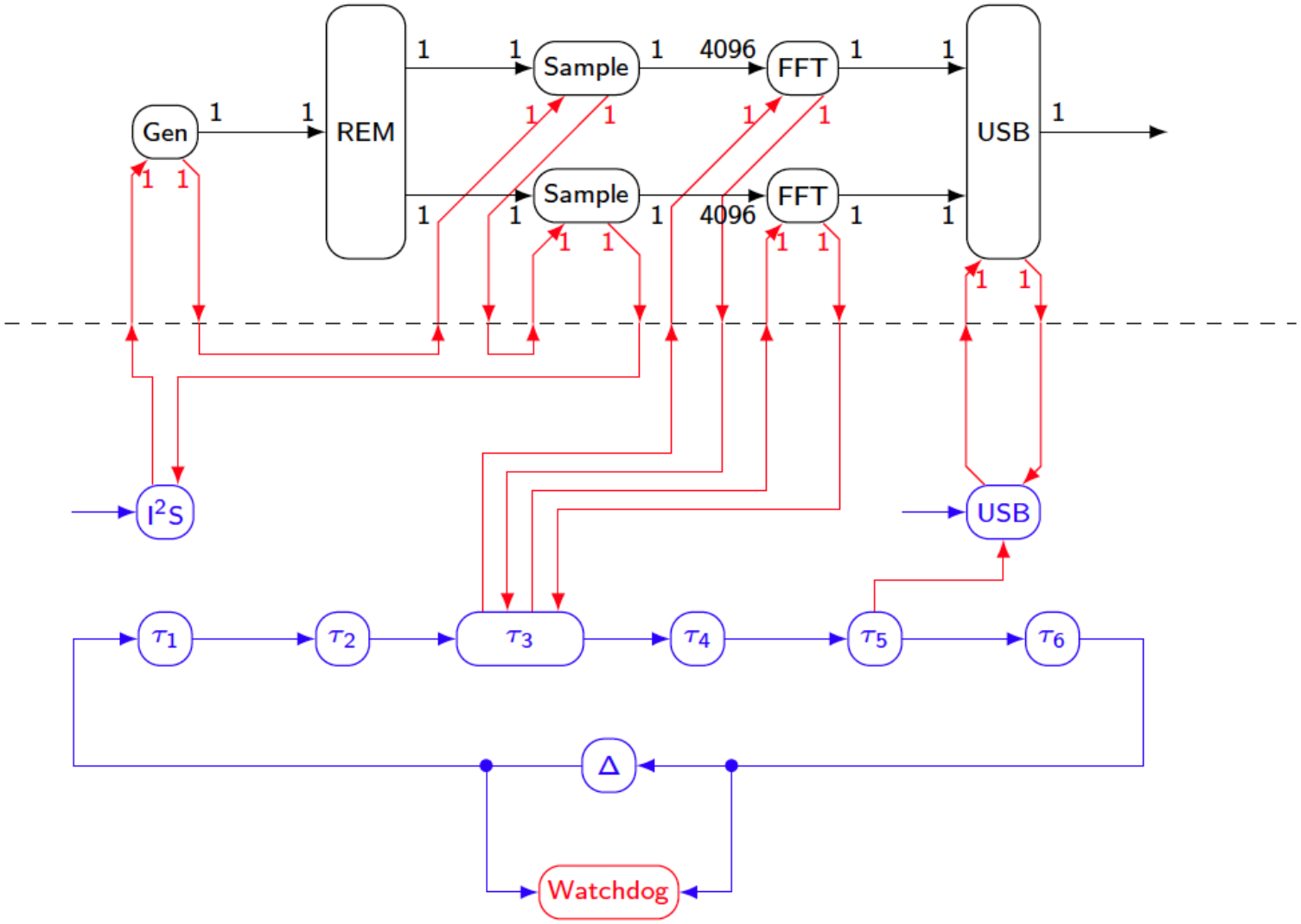
Implementing a certain execution schedule



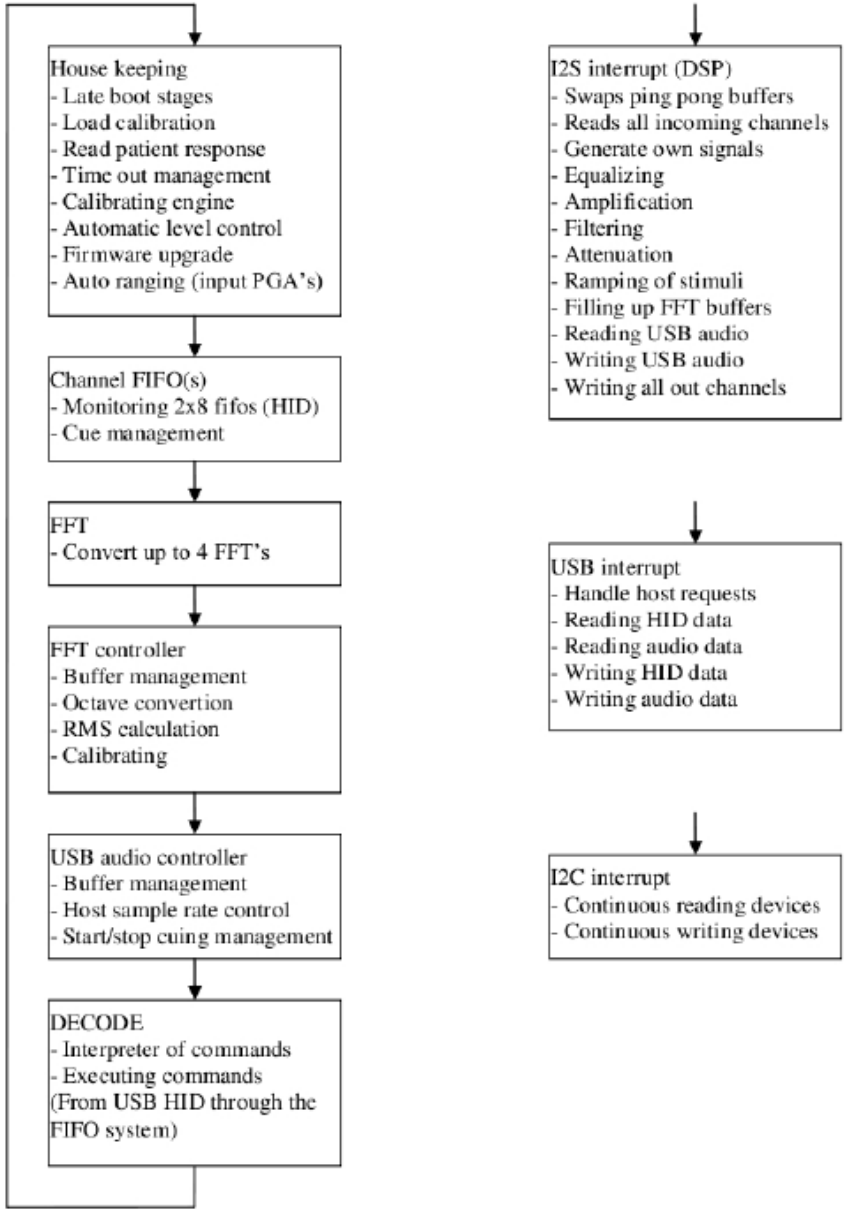
Concepts for timed execution models



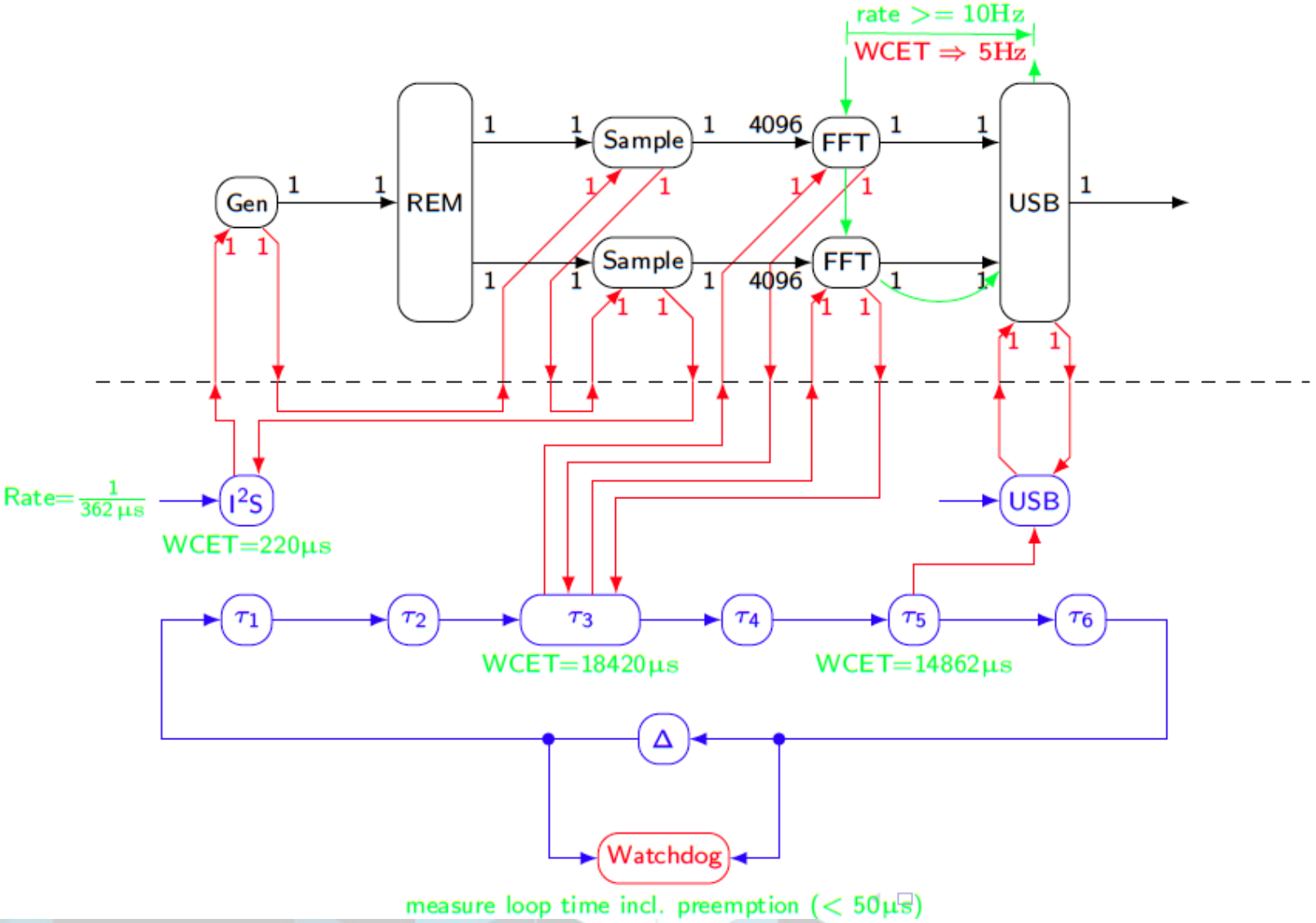
System model



Firmware implementation



System analysis



Conclusions

- Early performance analysis (hints of potential problems)
- Abstract models close to system specification
- Reusing abstract model for mapping

- More information on ForSyDe: https://forsyde.ict.kth.se/

Thank you for your attention!

Jan Madsen [jan@imm.dtu.dk]

Advanced Research & Technology for Embedded Intelligence and Systems