Performance Analysis of Distributed Embedded Systems

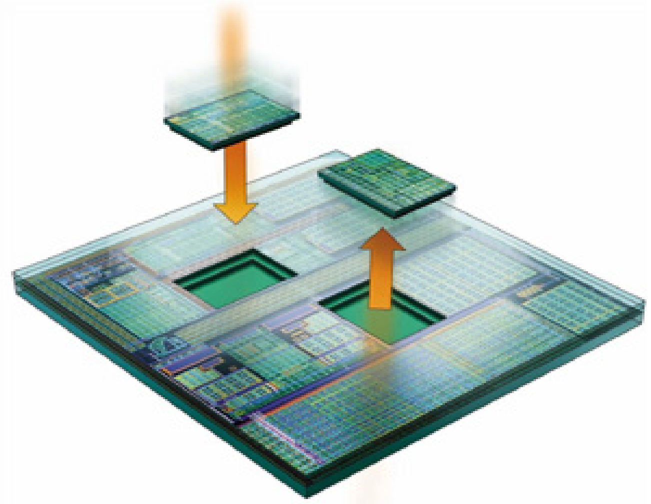
- Component-Based Design -

© Lothar Thiele

Analysis and Design

Embedded System =
Computation + Communication + Resource Interaction

Analysis:
Infer system properties from subsystem properties.



Design:
Build a system from subsystems while meeting requirements.

Challenges

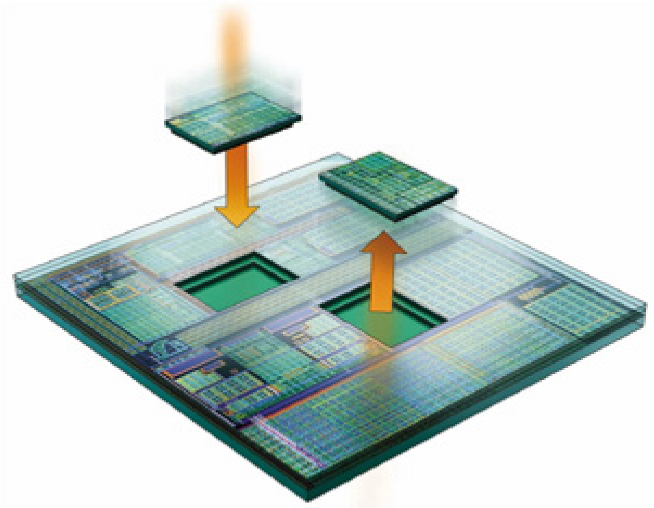
Make Analysis and Synthesis Compositional

Stepwise Refinement:

a. compose subsystems
b. refine subsystems

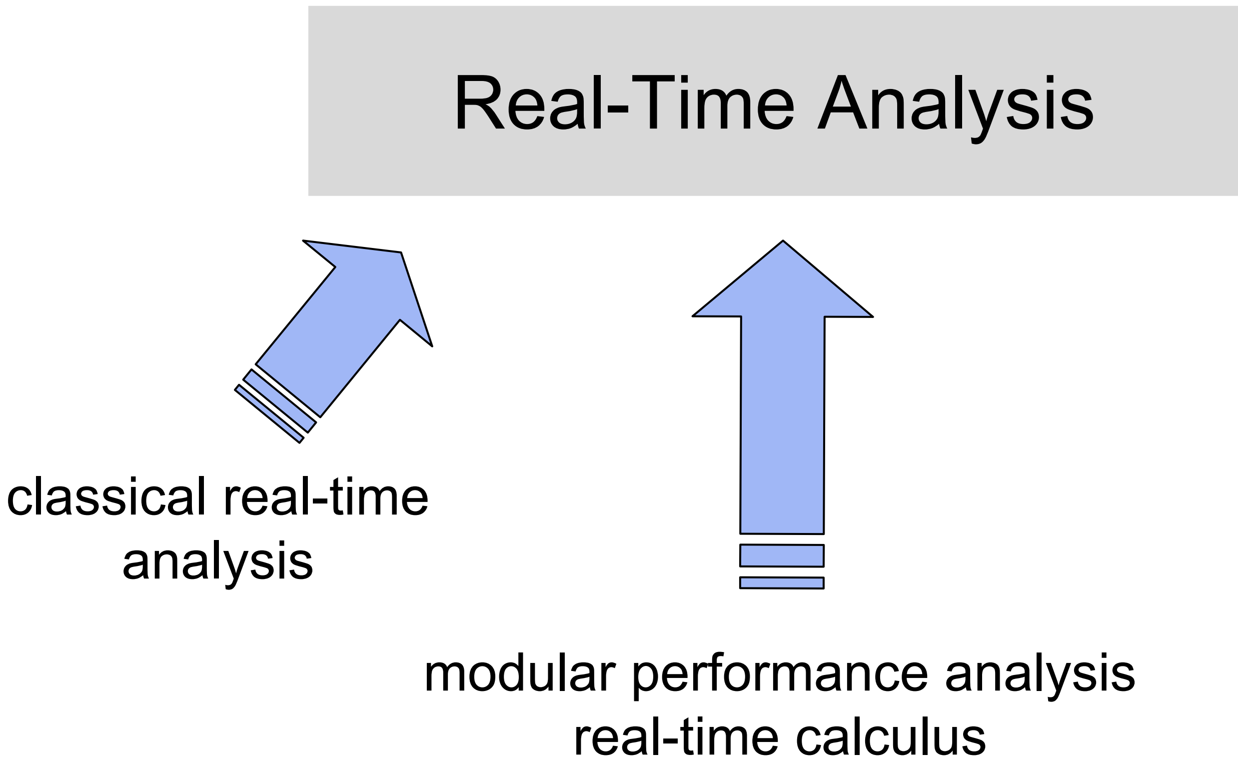
Adaptivity:

a. changes in environment
b. changes of requirements

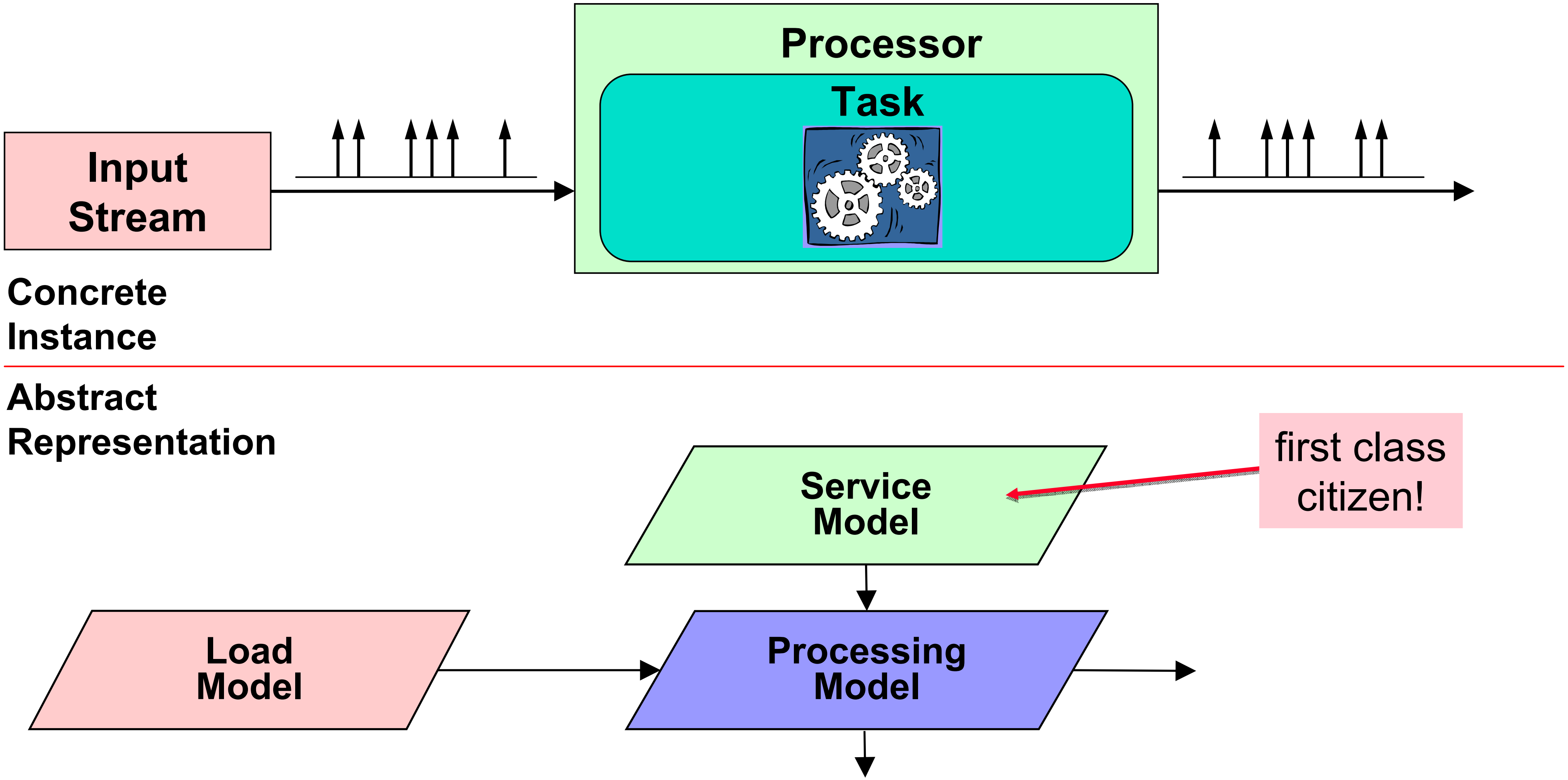


Concept

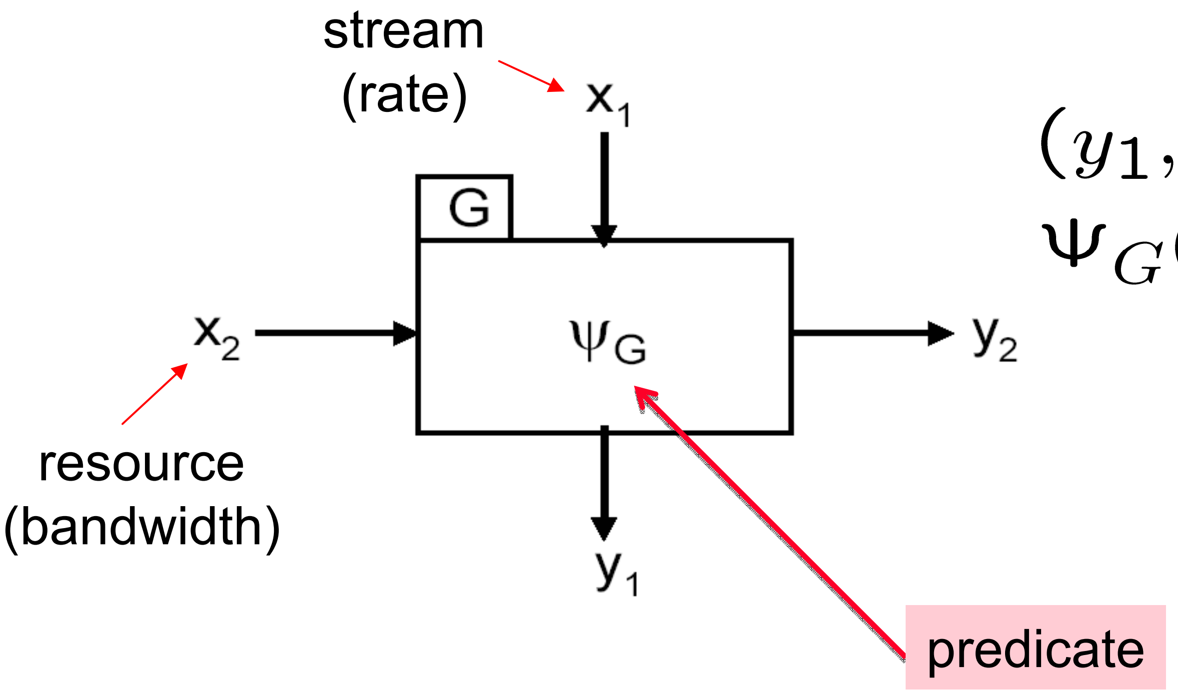
- Real-Time Interfaces provide a ***framework*** for dealing with resource interaction.
- First concept: Compositional real-time analysis



Abstract Models for Performance Analysis

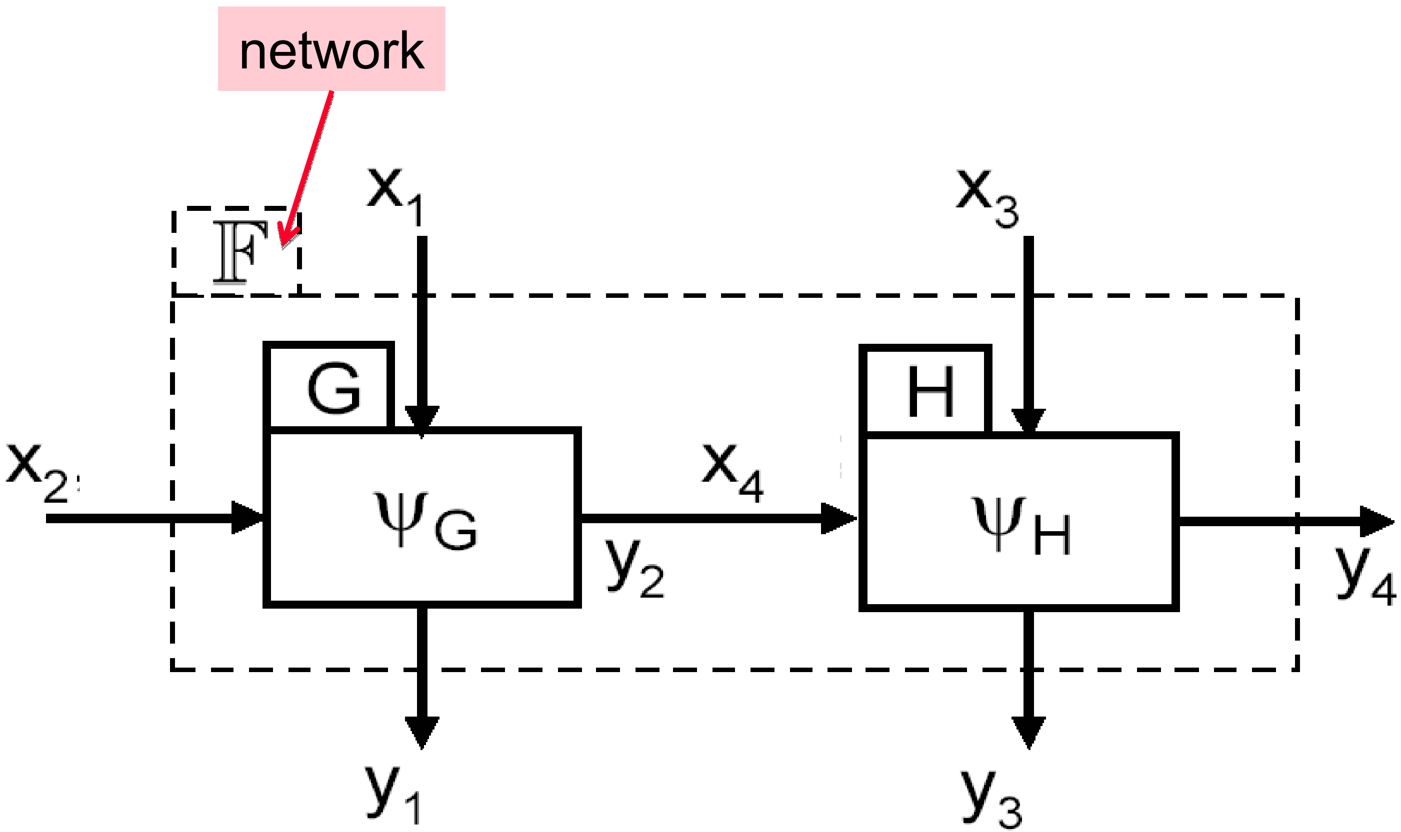


Abstract Component



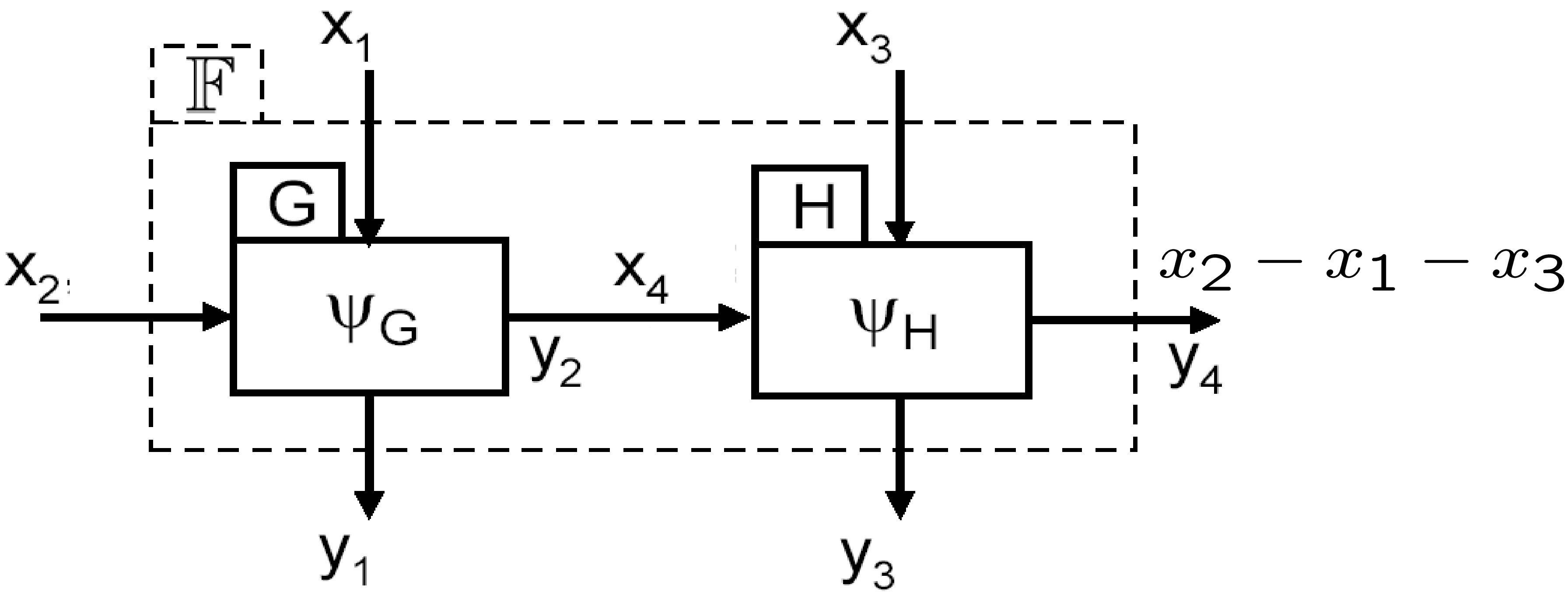
$$(y_1, y_2) = (x_1, x_2 - x_1)$$
$$\Psi_G(x_1, x_2) = (x_2 \geq x_1)$$

Network of Abstract Components



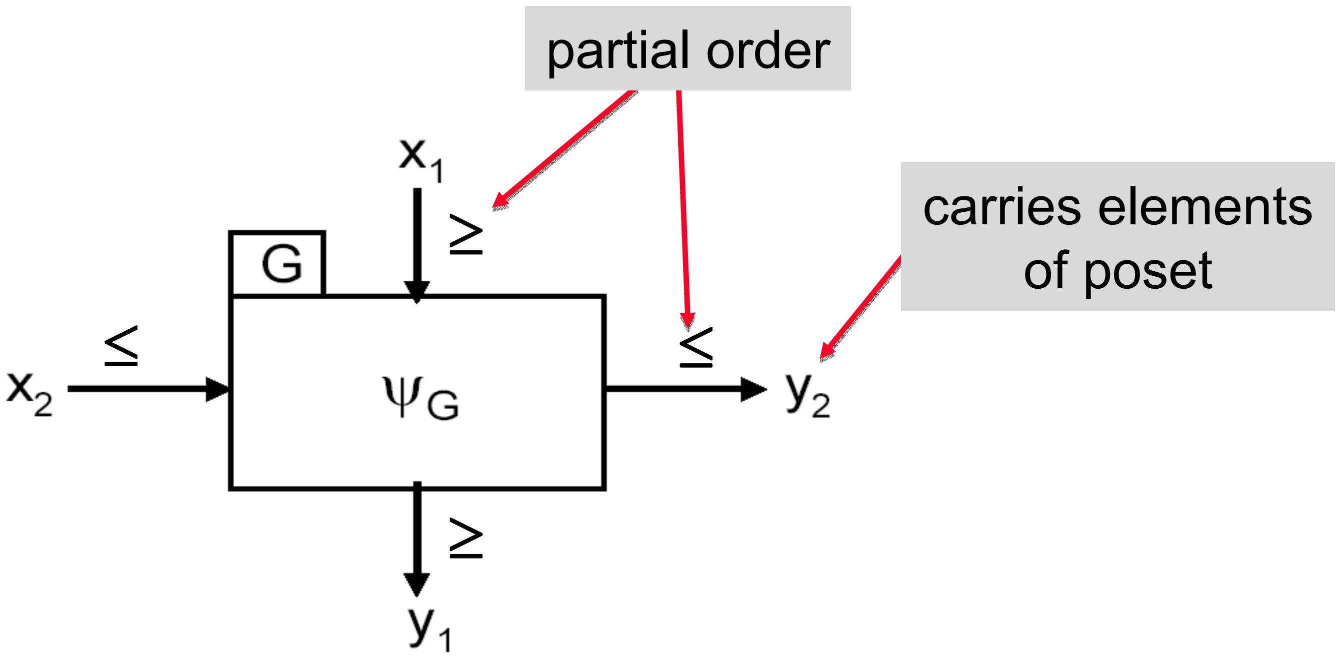
predicate: $\Psi_{\mathbb{F}} = \bigwedge_{F \in \mathbf{F}} \Psi_F$

Network of Abstract Components



$$\Psi_{\mathbb{F}}(x_1, x_2, x_3) = (x_2 \geq x_1) \wedge (x_2 - x_1 \geq x_3)$$

Monotone Components



monotone component:

$$X_F \geq \tilde{X}_F \Rightarrow T_F(X_F) \geq T_F(\tilde{X}_F)$$

$$X_F \geq \tilde{X}_F \wedge \Psi_F(X_F) \Rightarrow \Psi_F(\tilde{X}_F)$$

Challenges

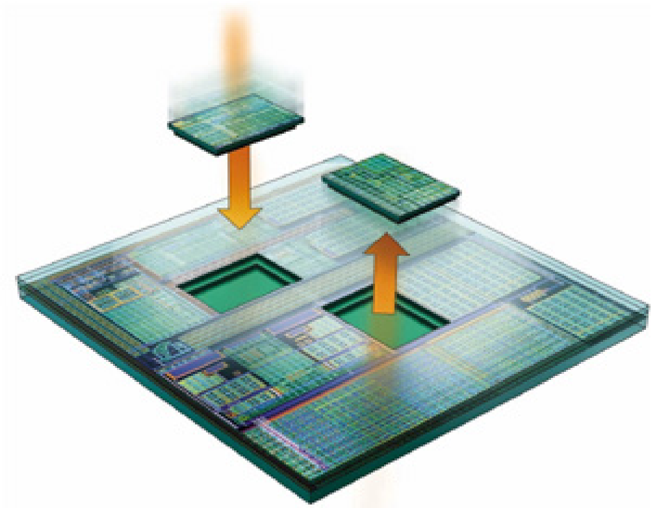
Make Analysis and Synthesis Compositional

Stepwise Refinement:

a. compose subsystems ✓
b. refine subsystems

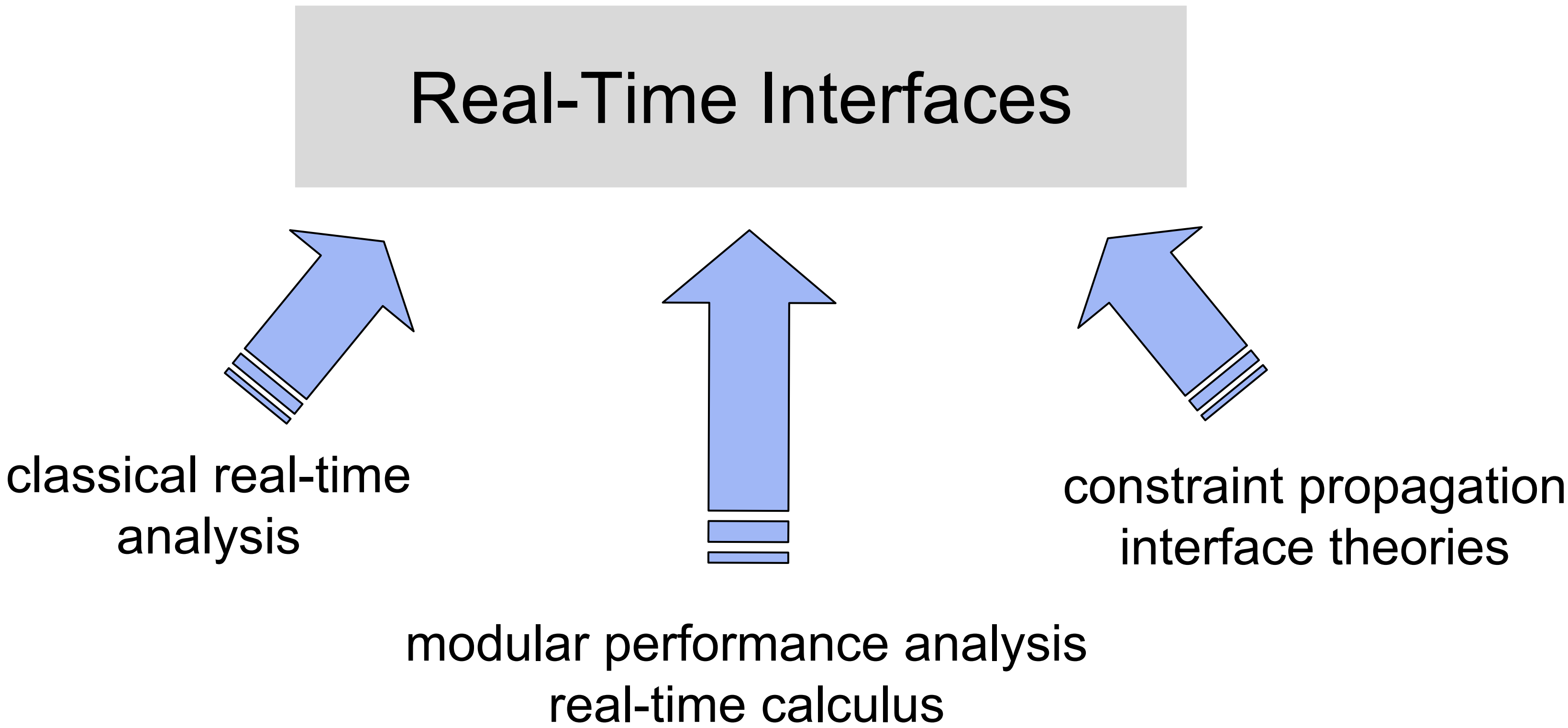
Adaptivity:

a. changes in environment
b. changes of requirements

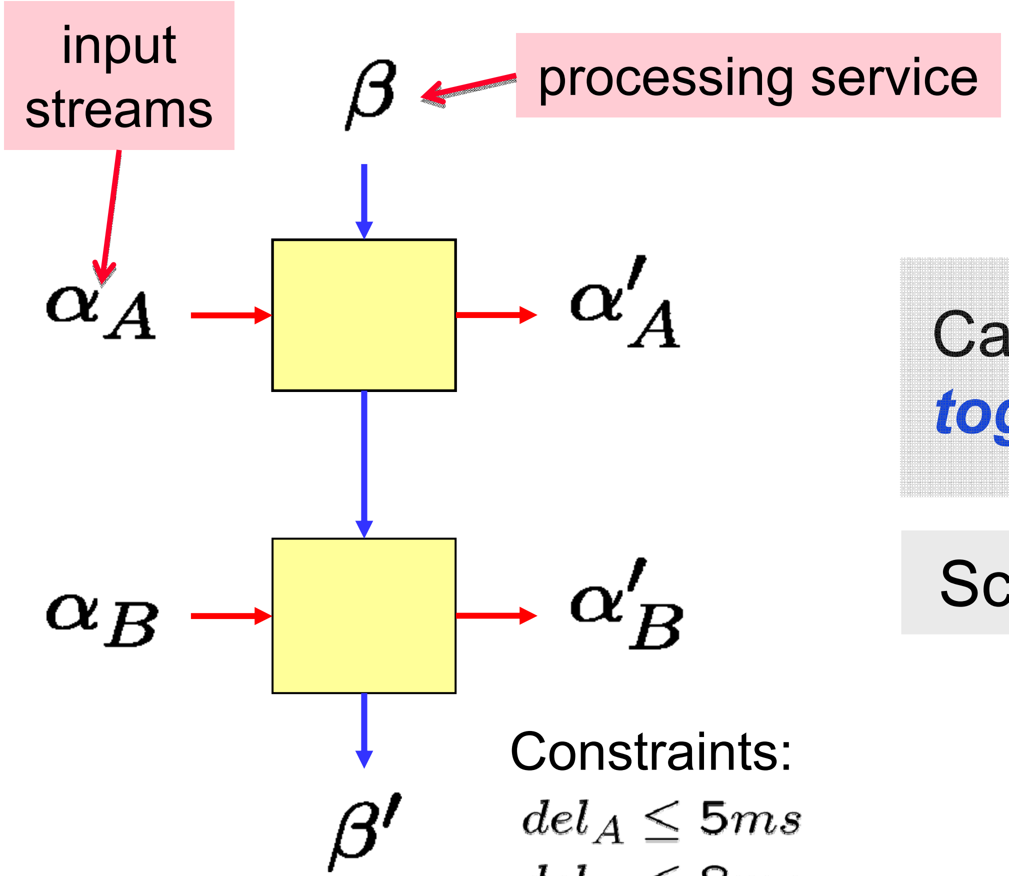


Concept

- Second concept: Real-Time Interfaces



Component-Based Analysis



Constraints:

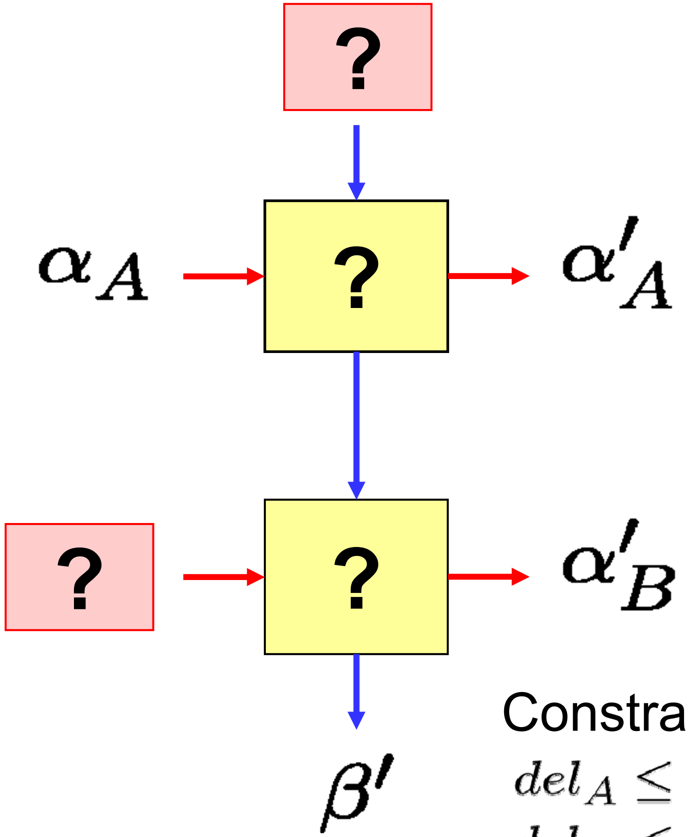
$del_A \leq 5ms$

$del_B \leq 8ms$

Can the components ***work together properly***?

Schedulable?

Interface-Based Design



Constraints:

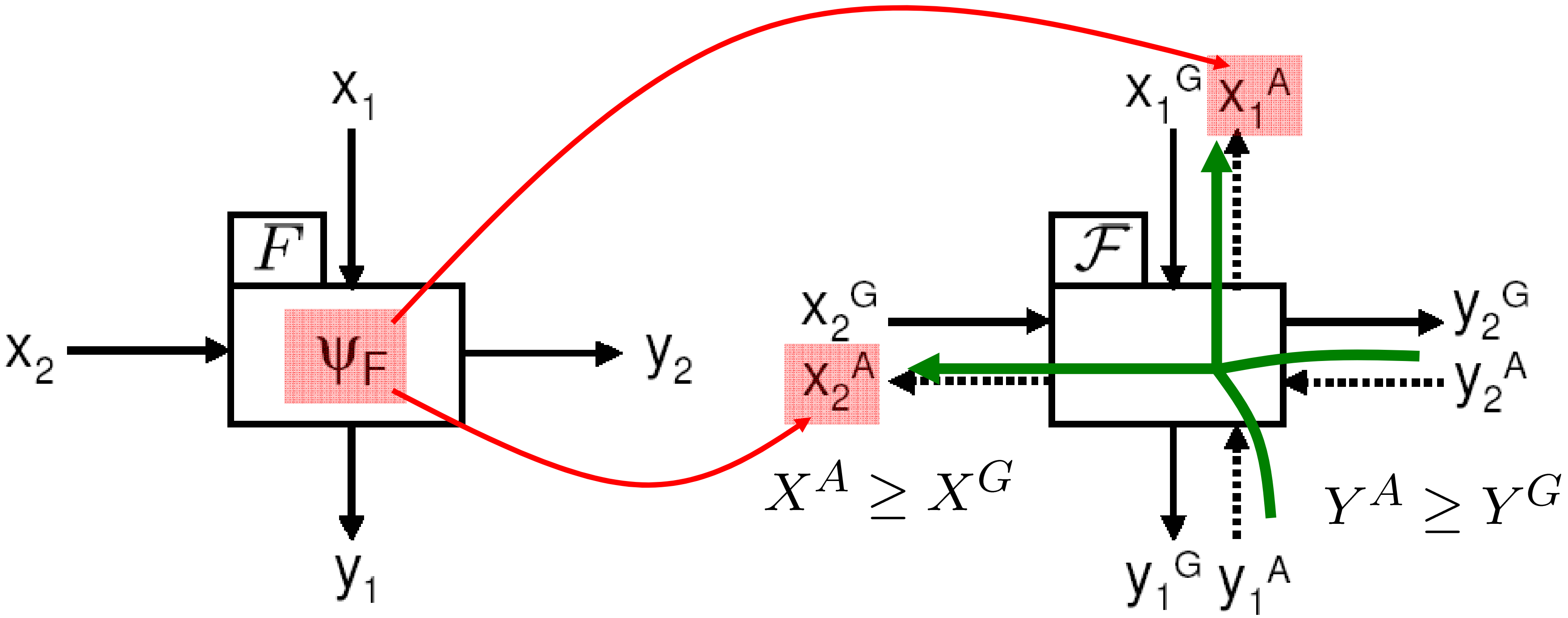
$del_A \leq 5ms$

$del_B \leq 8ms$

What are the ***assumptions*** towards the ***environment*** (inputs and/or requirements)?

What are the ***assumptions*** for the ***components*** (so that they can adapt)?

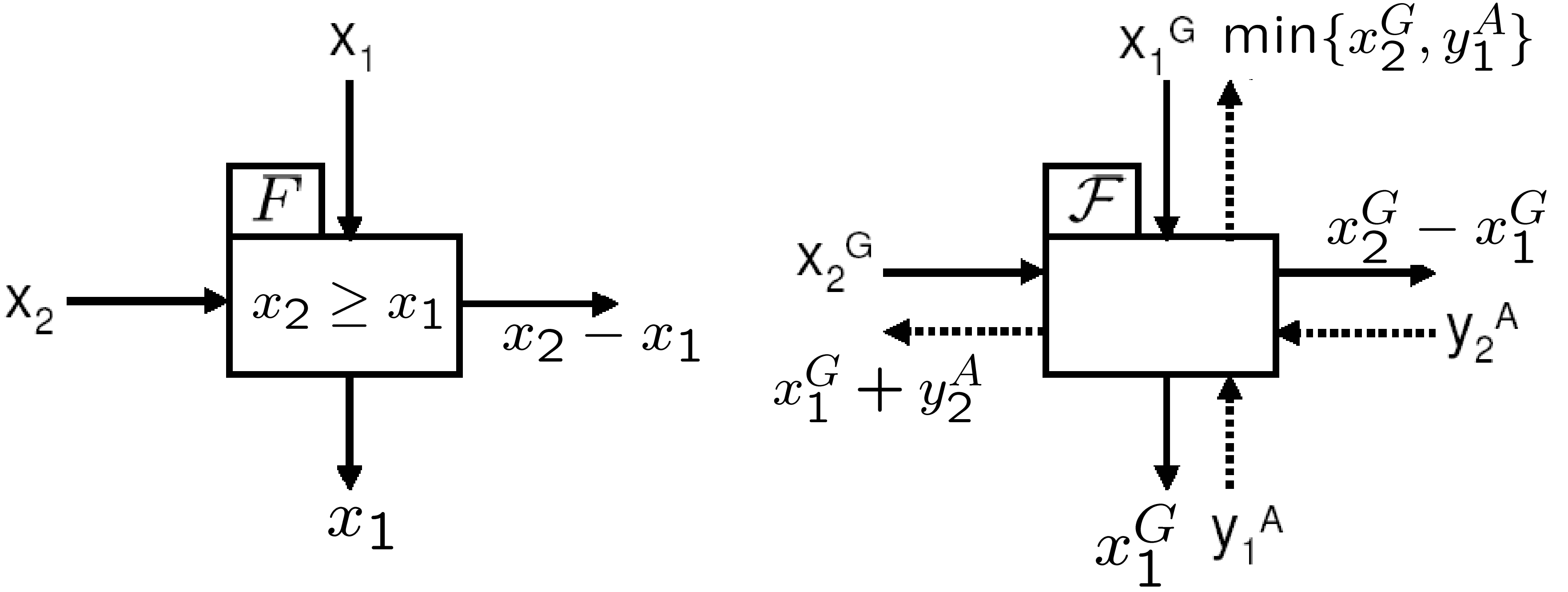
(Adaptive) Real-Time Interfaces



$$X^A = T^b(X^G, Y^A)$$

$$X^A \geq X^G \geq X \Rightarrow Y^A \geq Y^G \geq Y \wedge \Psi(X)$$

(Adaptive) Real-Time Interfaces



$$X^A = T^b(X^G, Y^A)$$

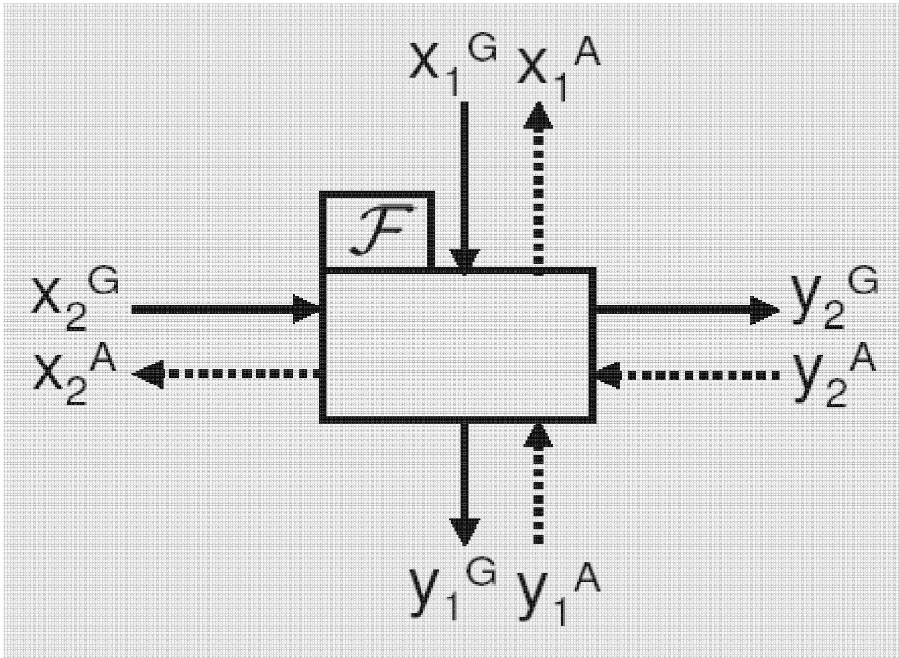
Questions …

- ***How can we determine the backward transfer function T_b ?***

- What is it good for ?

Backward Transfer Function

- Find as large as possible input assumes such that all constraints are satisfied:
  - predicate Ψ
  - output $Y^A \geq Y^G$



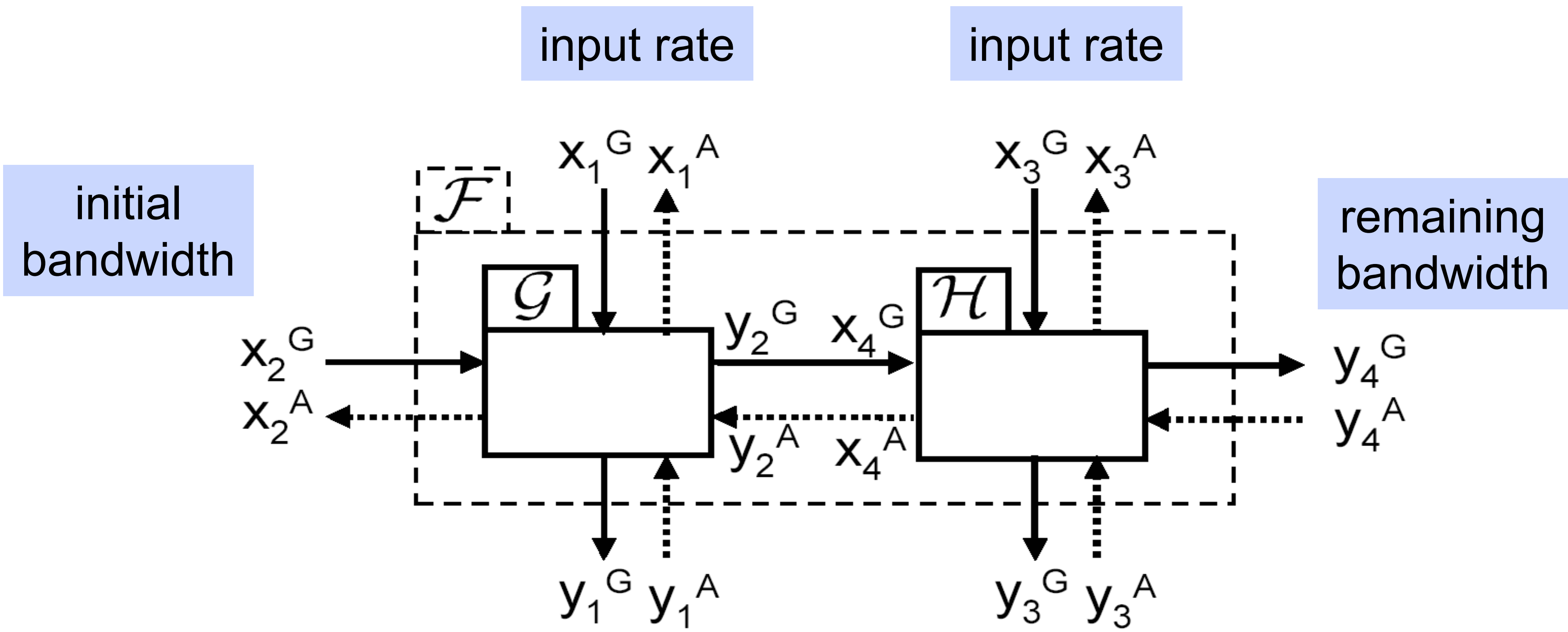
Questions …

- How can we determine the backward transfer function T_b?

- *What is it good for ?*

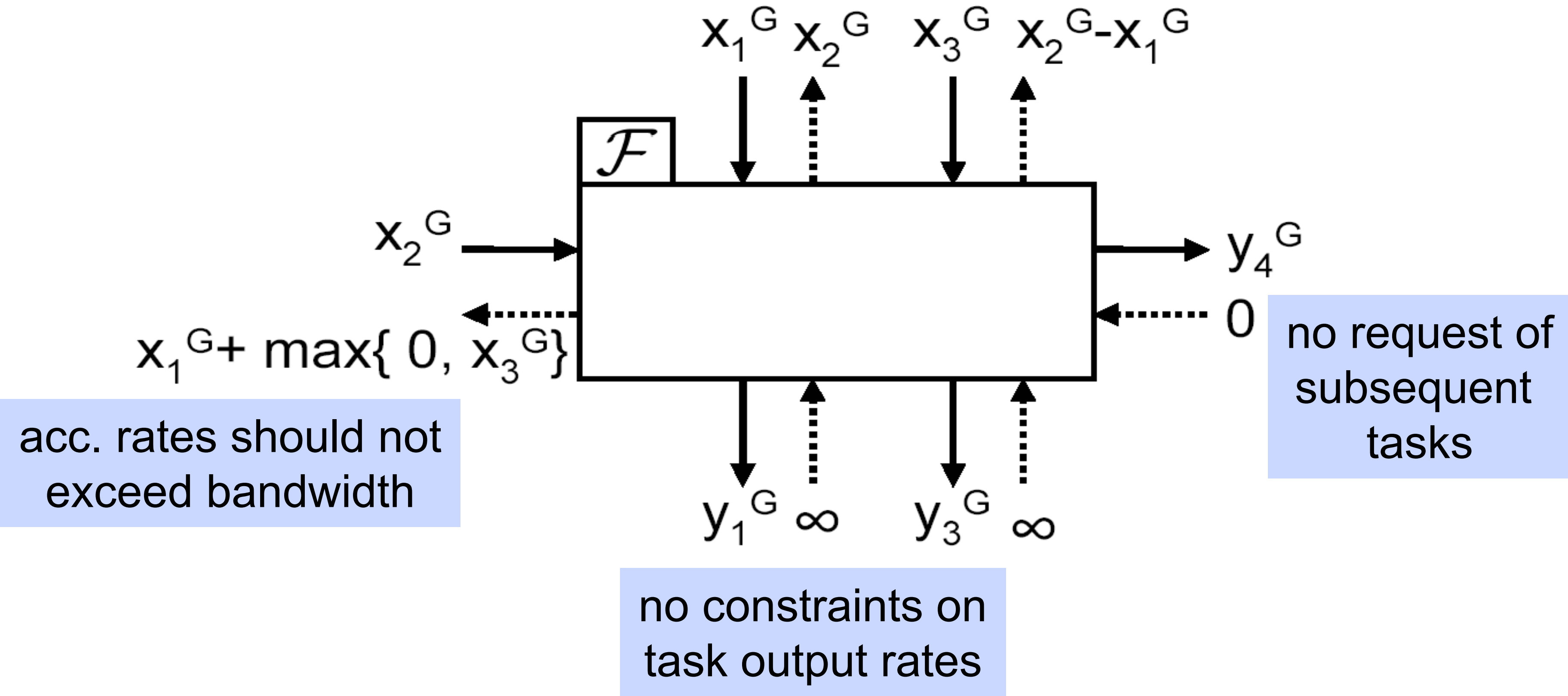
The Simple Example

- Two streams share a bus:



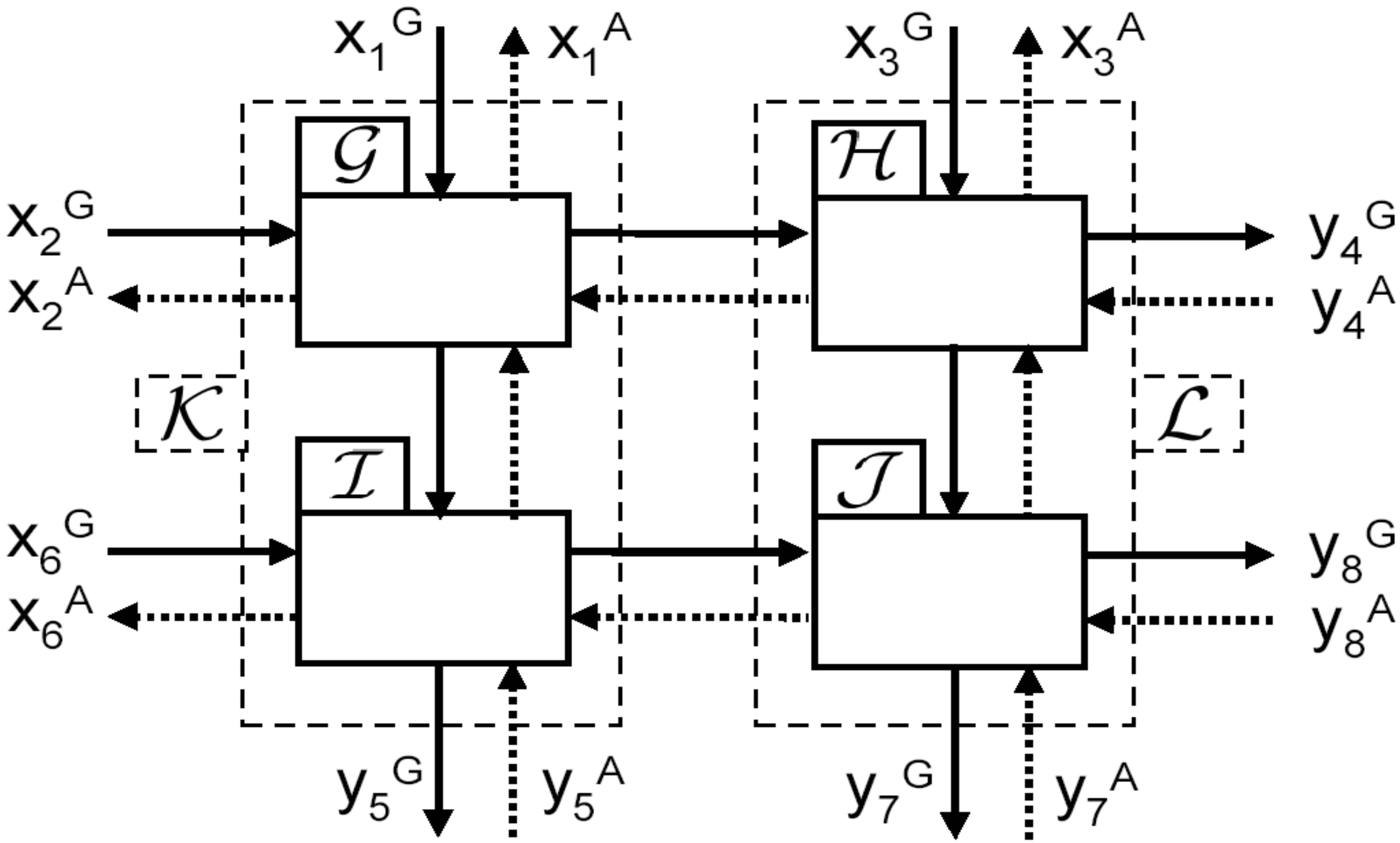
The Simple Example

- Connected to the ***weakest environment***:



The Simple Example

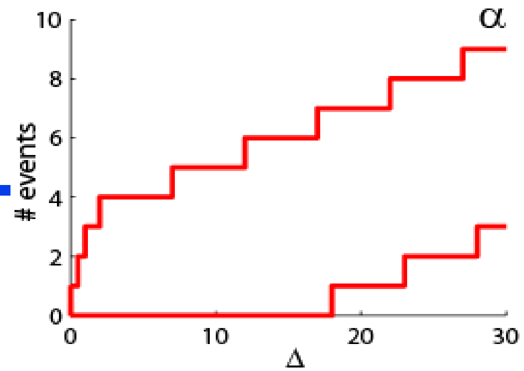
- Tasks going through two resources:



Obvious Question

Why such a complex construction for a simplistic constraint propagation ?

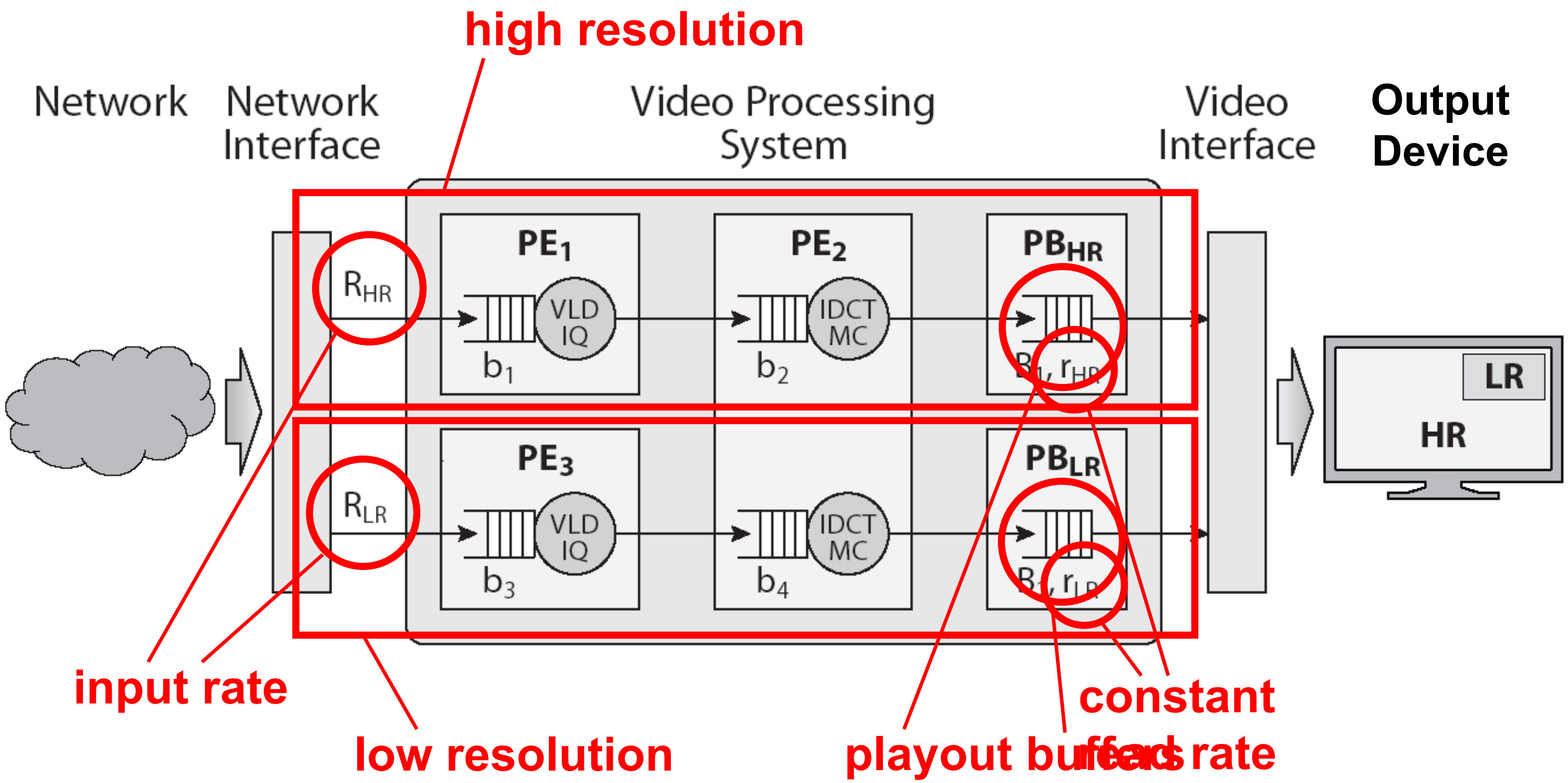
Application Scenarios



- Well known abstractions in Real-Time Analysis are not simple numbers but tupels or functions:
  - (period, jitter, burst, delay, buffer space)
  - demand bound and resource bound functions
  - arrival and service curves
- Leads to
  - ***Hierarchical and compositional analysis*** of distributed real-time systems
  - Analysis of ***hierarchical scheduling schemes***
  - ***Adaptivity*** (on-line and off-line) .

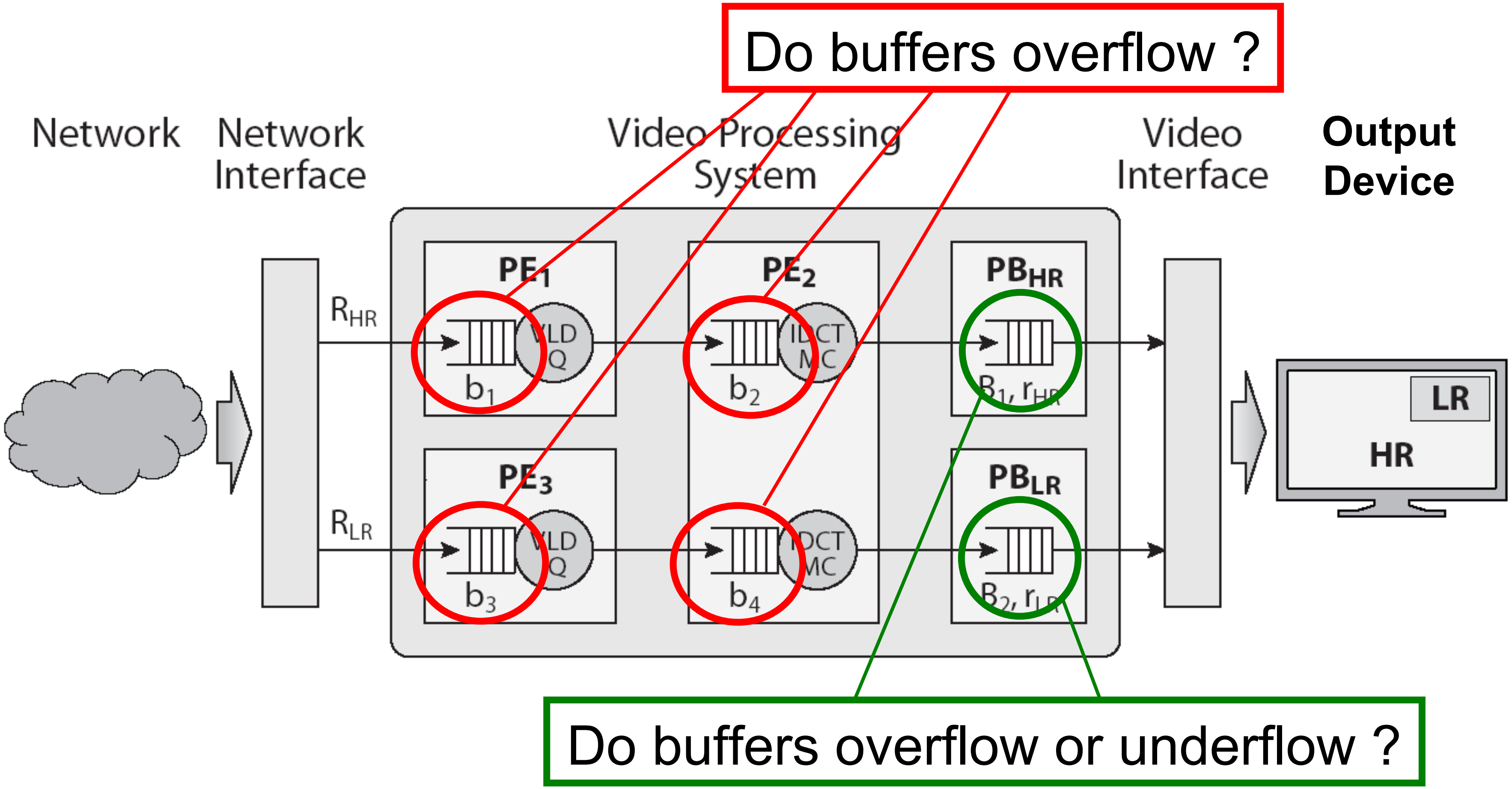
Application Scenario

- MPEG-2 decoding: Picture-in-picture application



Application Scenario

- ***Analysis Questions***:

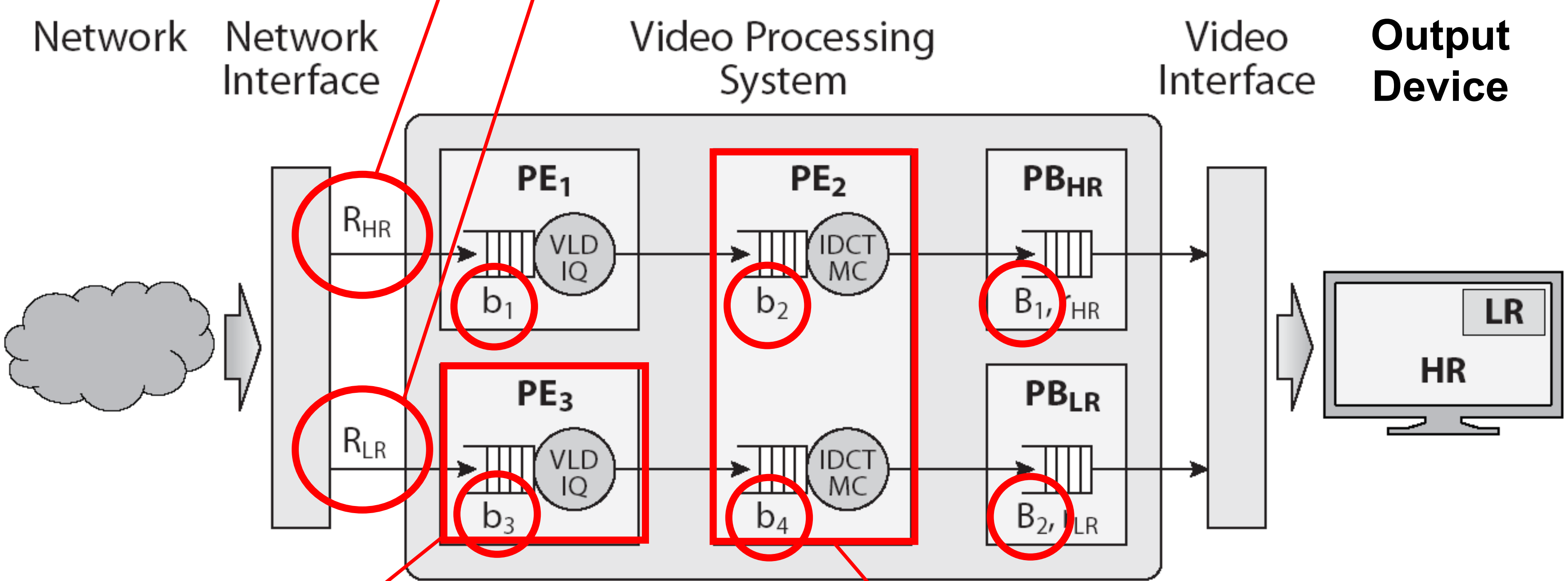


Application Scenario

- ***Design Questions***:

What are feasible input rates ?

While respecting all (buffer) constraints!



What are feasible processor speeds ?

What are feasible scheduling policies ?

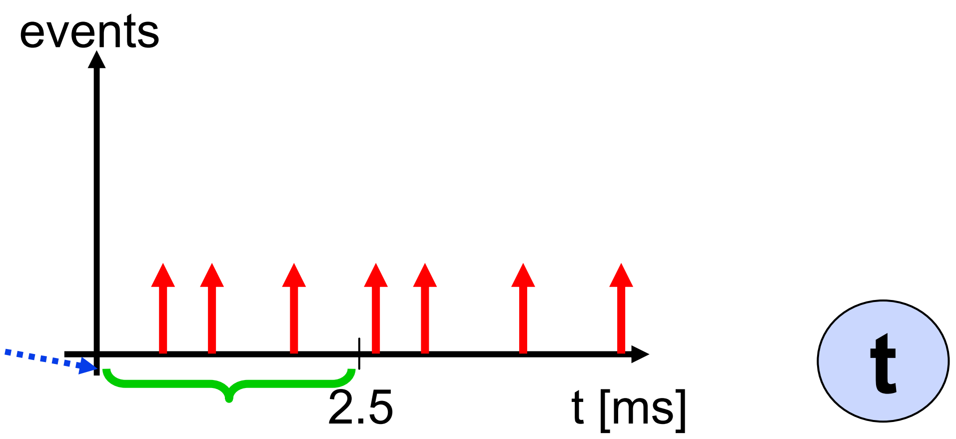
Related Work

- ***Rate Analysis***
  - [Bini, Natale 05], [Mathur, Dasdan, Gupta 98]
  - [Liu, Chakraborty, Marculescu 06]
- ***Network Calculus***
  - [Cruz 91], [Boudec, Thiran 01]
- ***Real-Time Calculus, Modular Performance Analysis***
  - [Thiele 00], [Chakraborty, Kuenzli, Thiele 03]
- ***Interface Theory, Real-Time Interfaces***
  - [Henzinger, de Alfaro 01], [Wandeler, Thiele 06]

From Event Streams to Arrival Curves

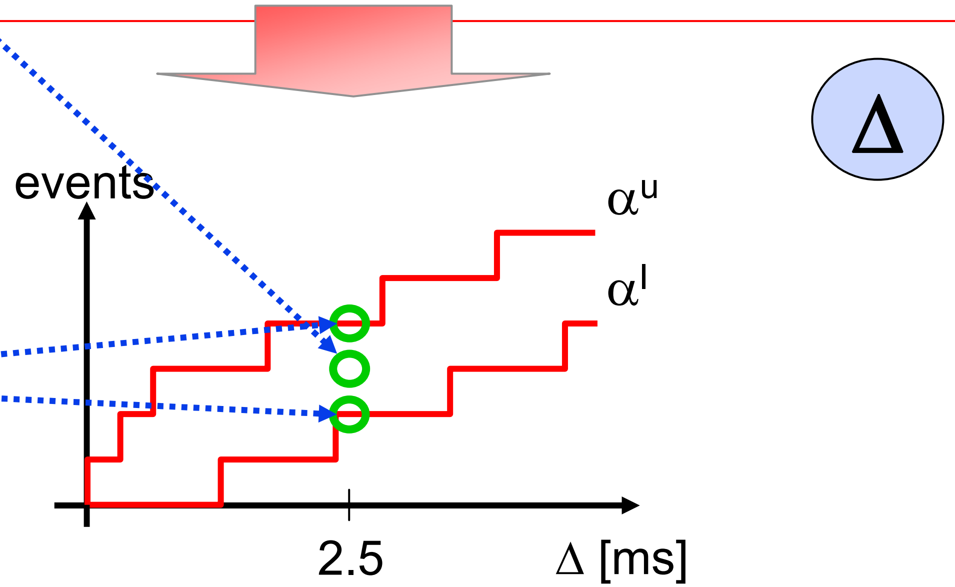
Event Stream

number of events in
in t=[0 .. 2.5] ms

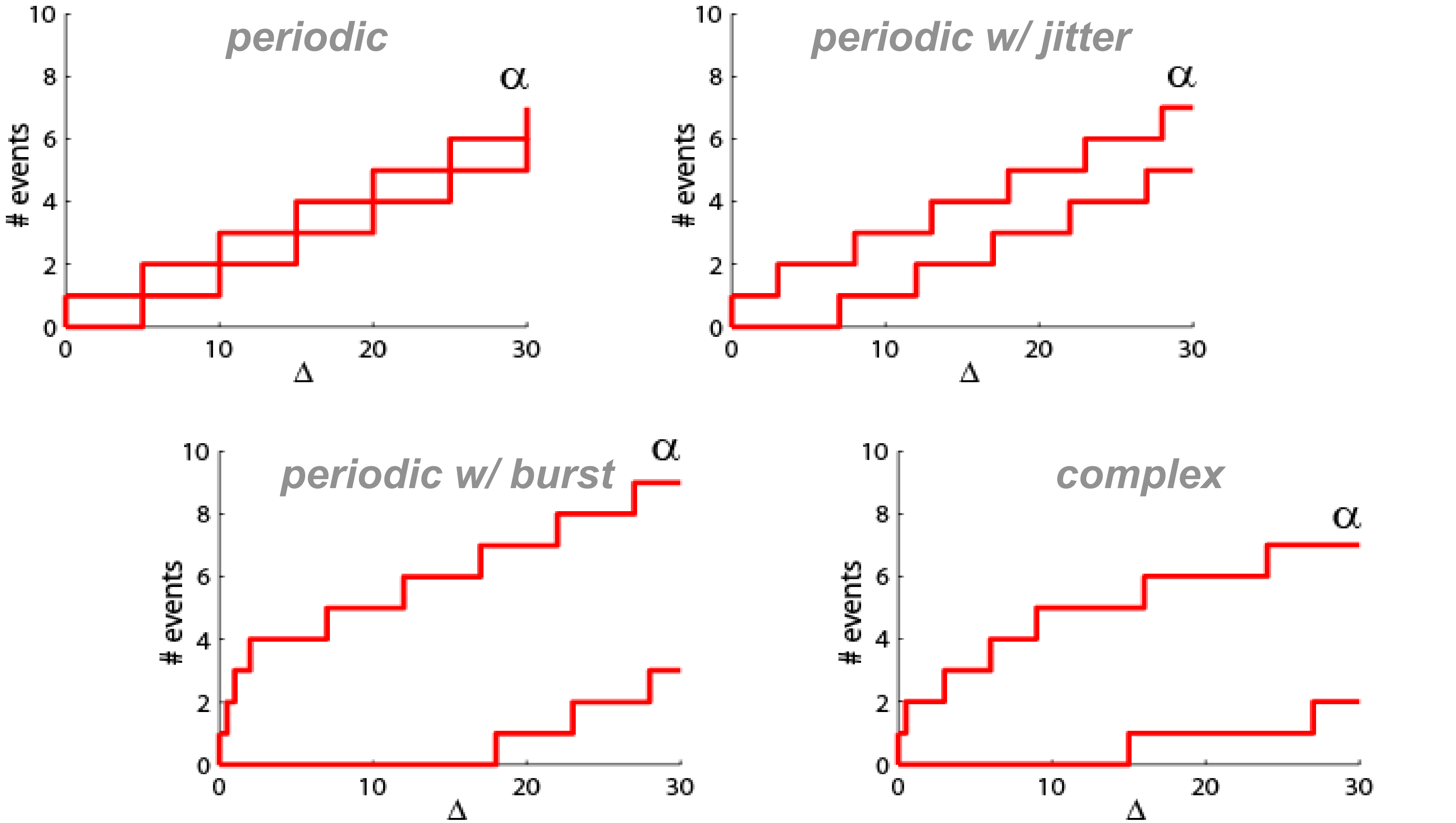


Arrival Curves $\alpha = [\alpha^l, \alpha^u]$

maximum / minimum
arriving events in *any interval* of length 2.5 ms



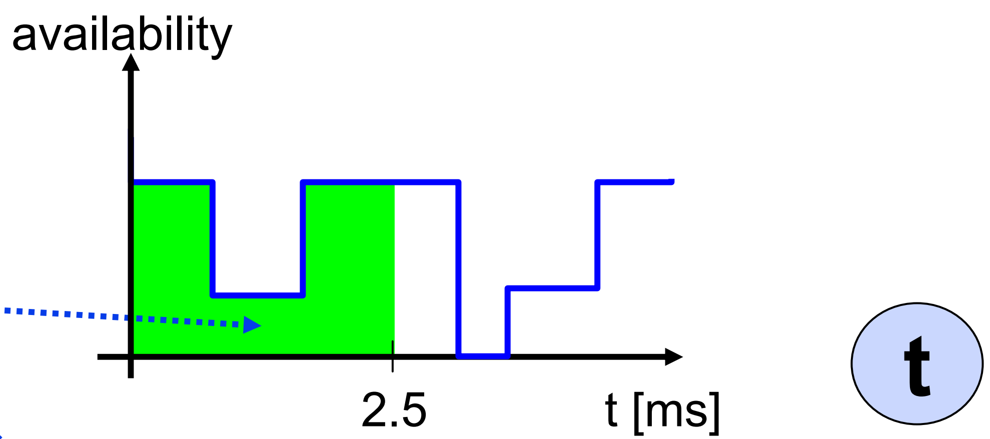
Load Model - Examples



From Resources to Service Curves

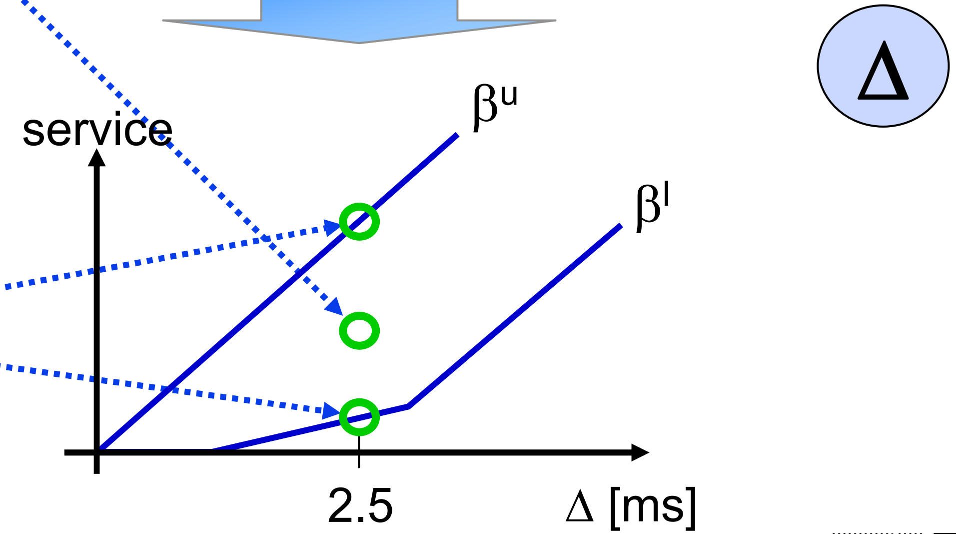
Resource Availability

available service
in t=[0 .. 2.5] ms

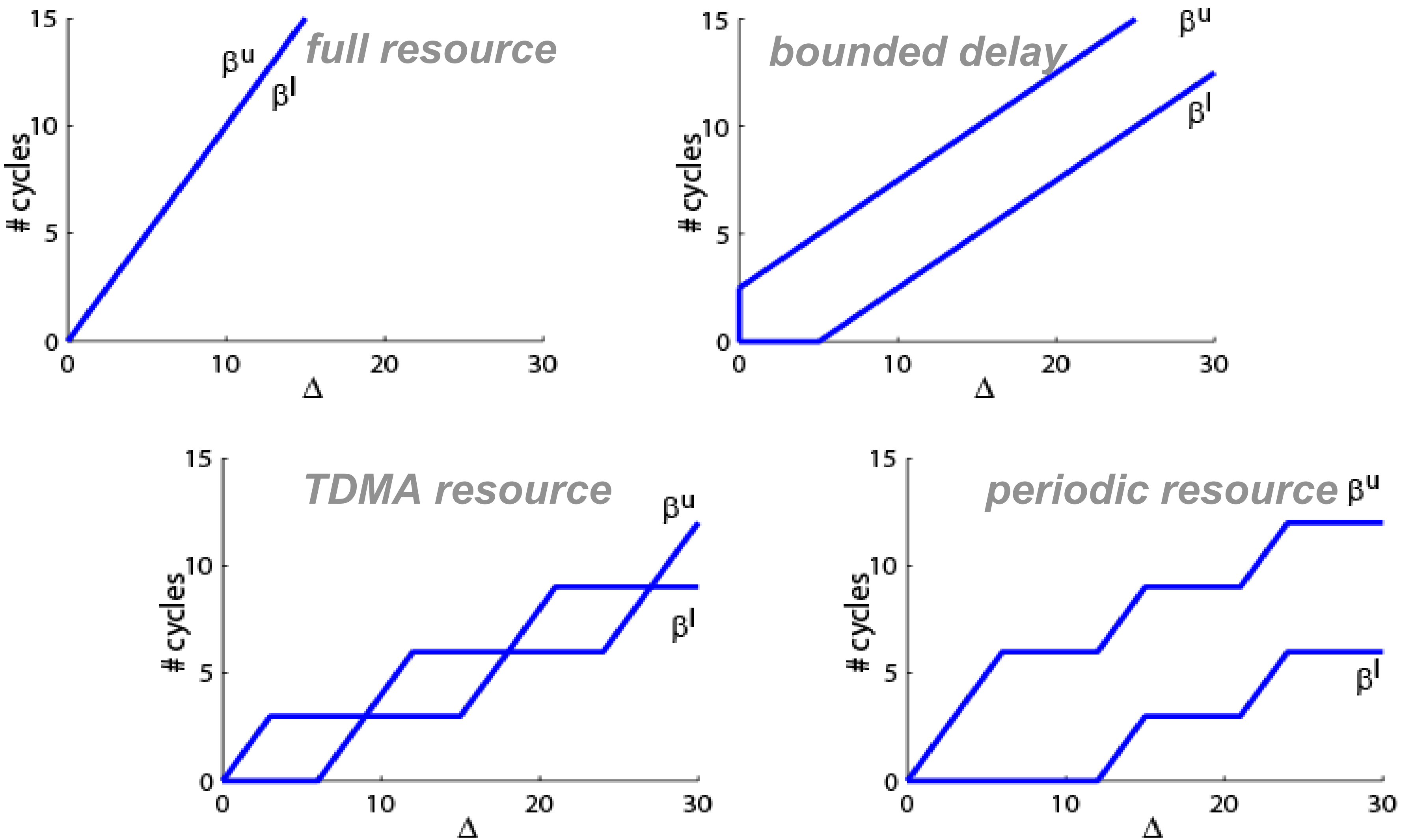


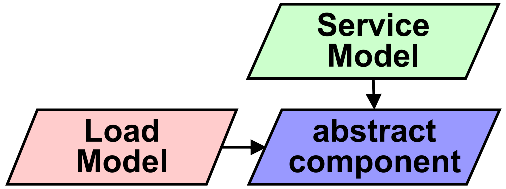
Service Curves $\beta = [\beta^l, \beta^u]$

maximum/minimum
available service in *any interval* of length 2.5 ms



Service Model - Examples

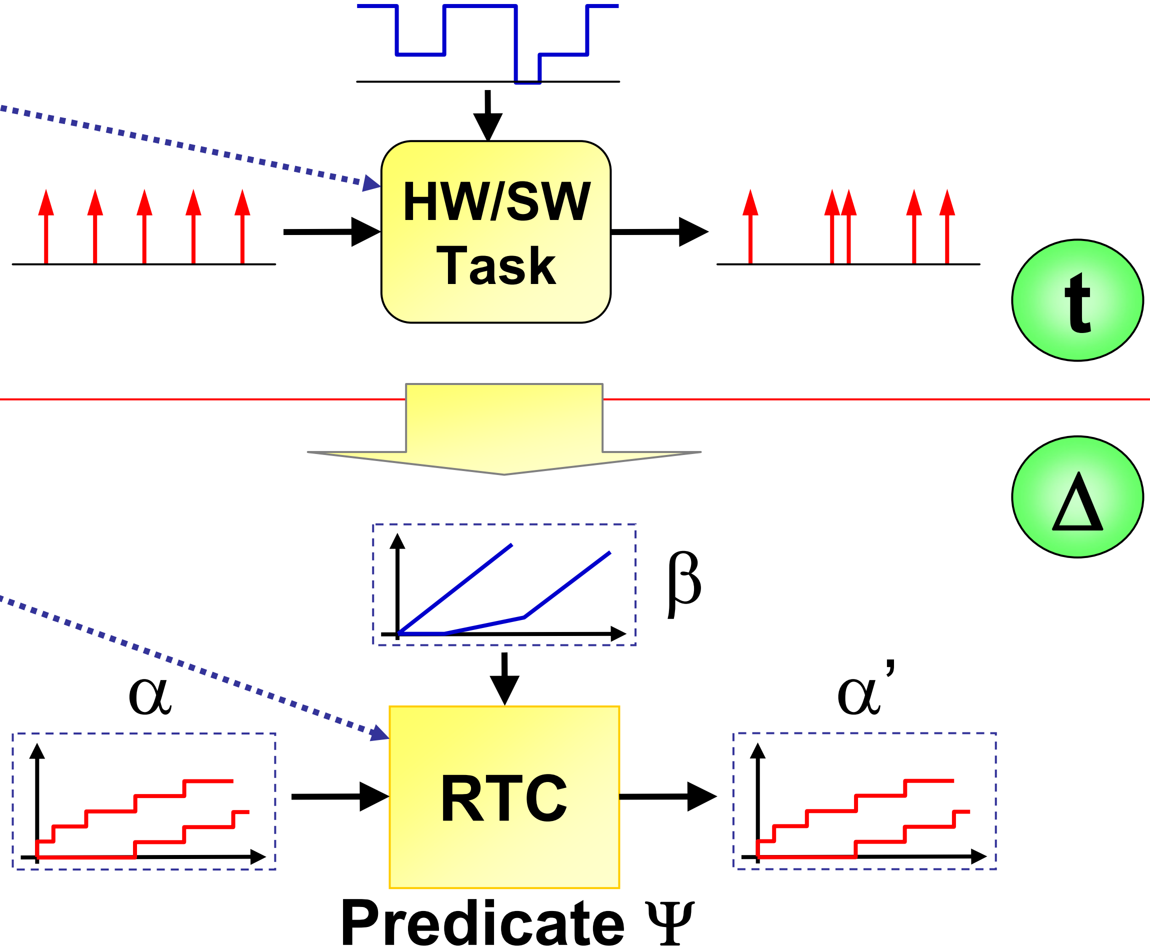




Processing Model (HW/SW)

HW/SW Components

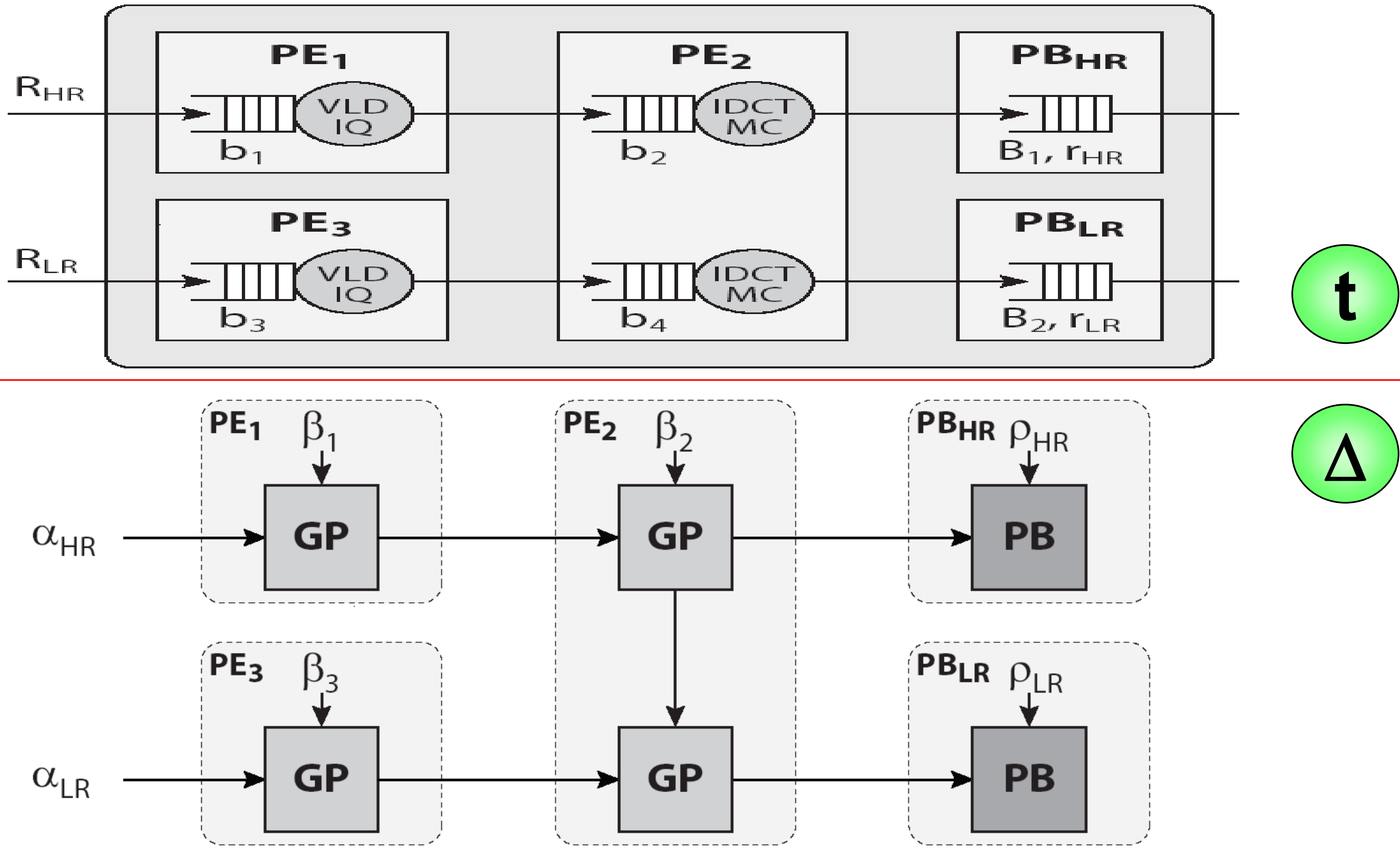
Processing semantics and functionality of HW/SW tasks



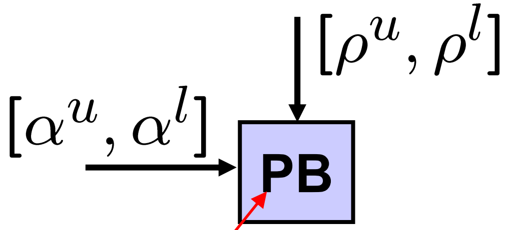
Abstract Components

$$\alpha'(\Delta) = f(\alpha, \beta)$$

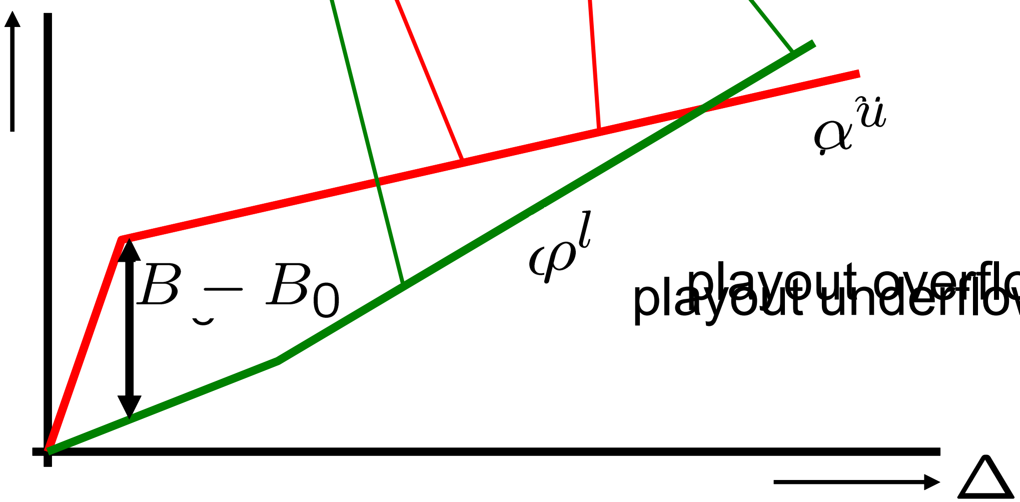
Complete System Composition



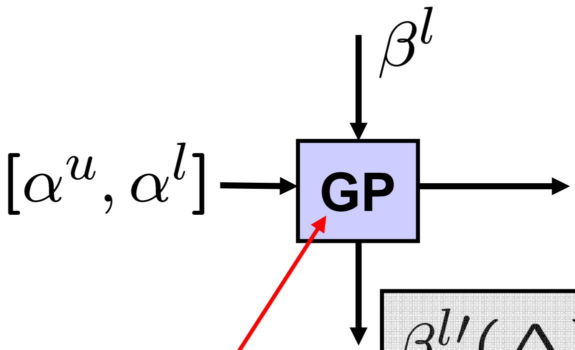
Transfer Functions and Predicates



$$\Psi := (\forall\Delta : \alpha^u(\Delta) \leq \rho^l(\Delta) + B - B_0) \wedge$$
$$(\forall\Delta : \alpha^l(\Delta) \geq \rho^u(\Delta) - B_0)$$



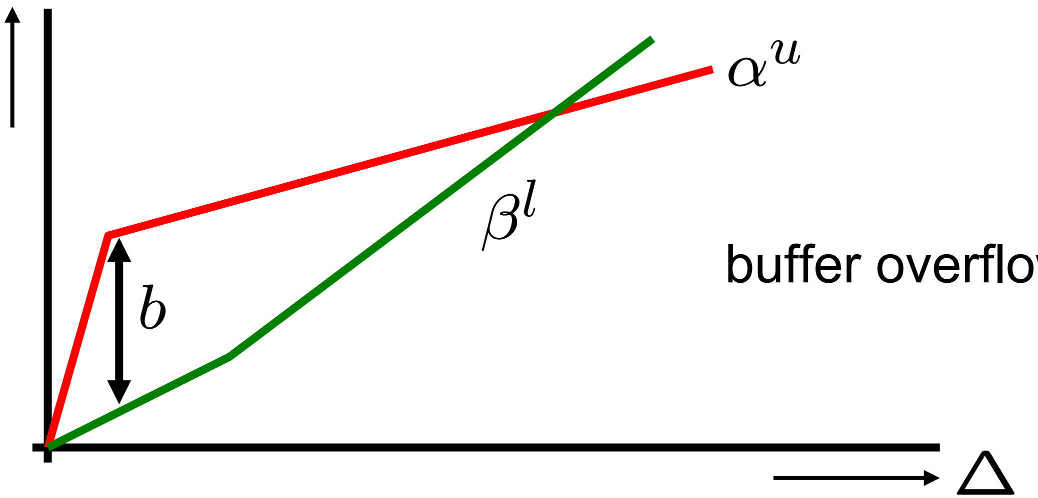
Transfer Functions and Predicates



$$[\alpha^{u\prime}, \alpha^{l\prime}] = [\alpha^u \oslash \beta^l, \alpha^l \oplus \beta^l]$$

$$\beta^{l\prime}(\Delta) = \sup_{0 \le \lambda \le \Delta}\{\beta^l(\lambda) - \alpha^u(\lambda)\}$$

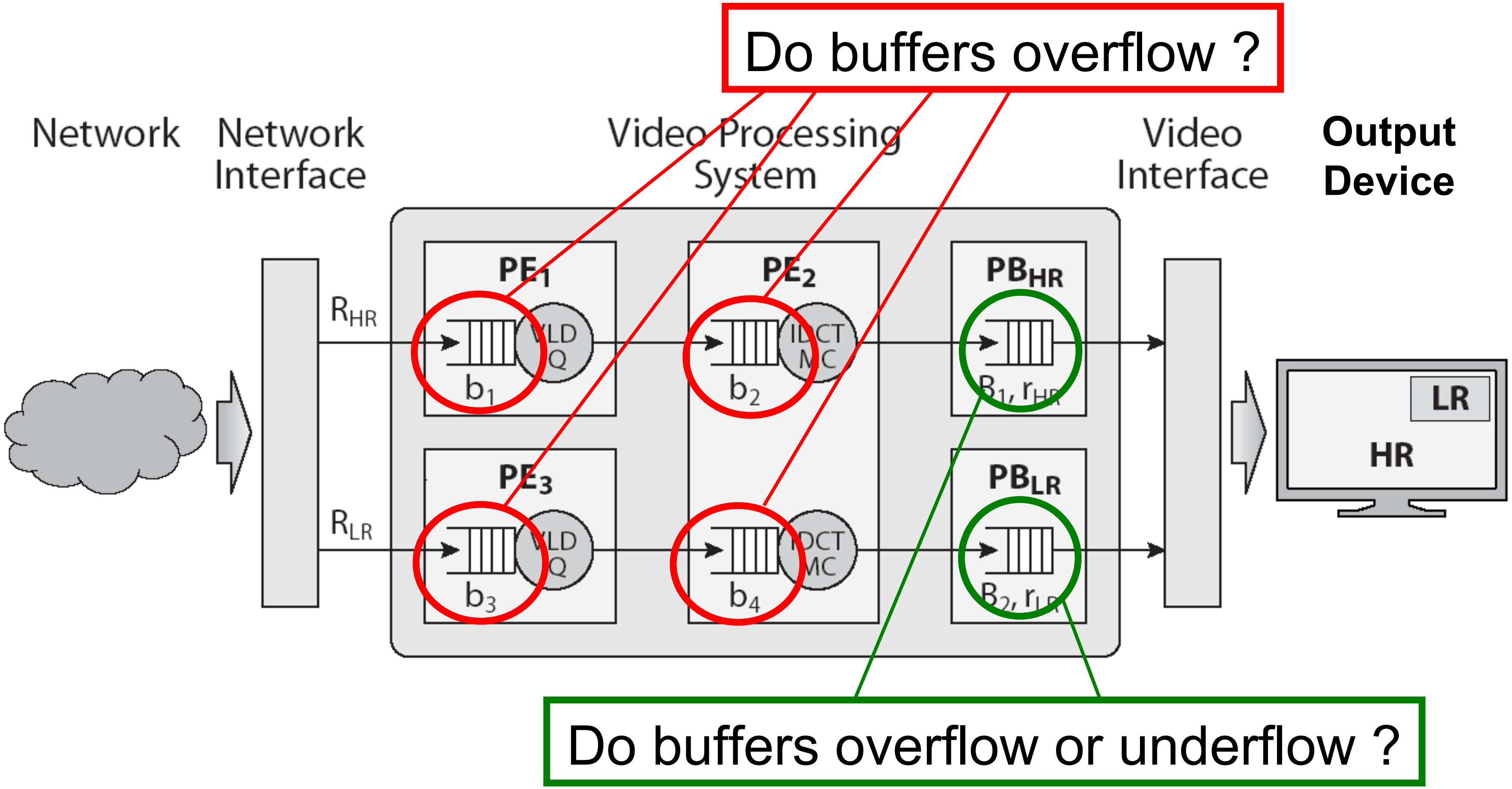
$$\Psi := (\forall \Delta : \alpha^u(\Delta) \le \beta^l(\Delta) + b)$$



buffer overflow ?

Application Scenario

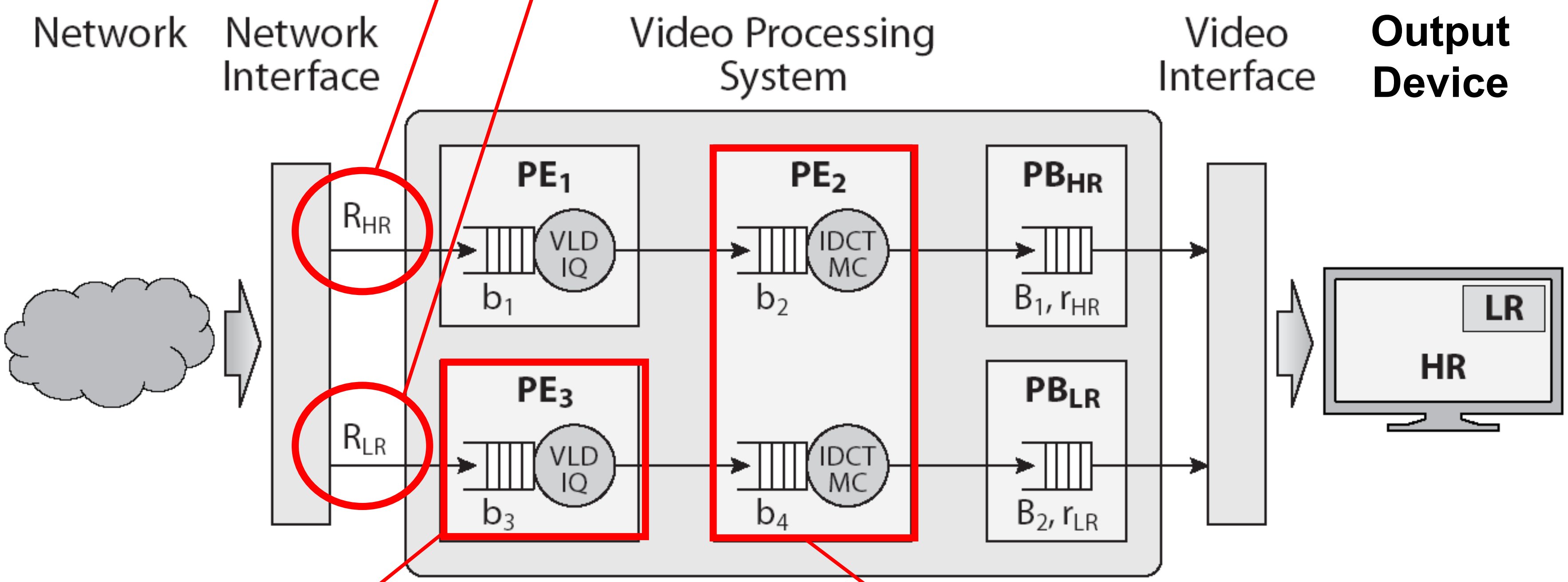
- Analysis Questions: ✓



Application Scenario

- ***Design Questions***:

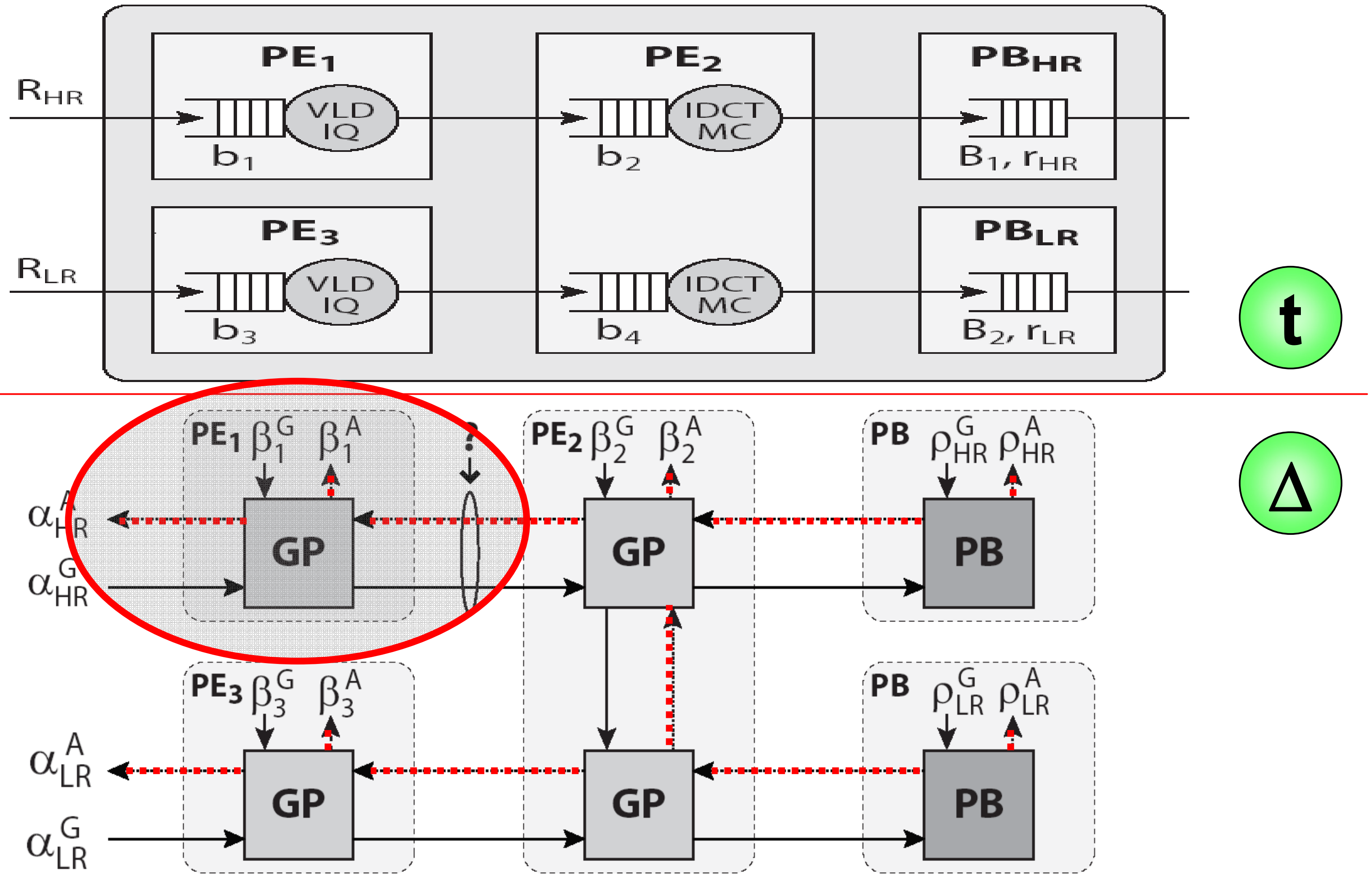
What are feasible input rates ?



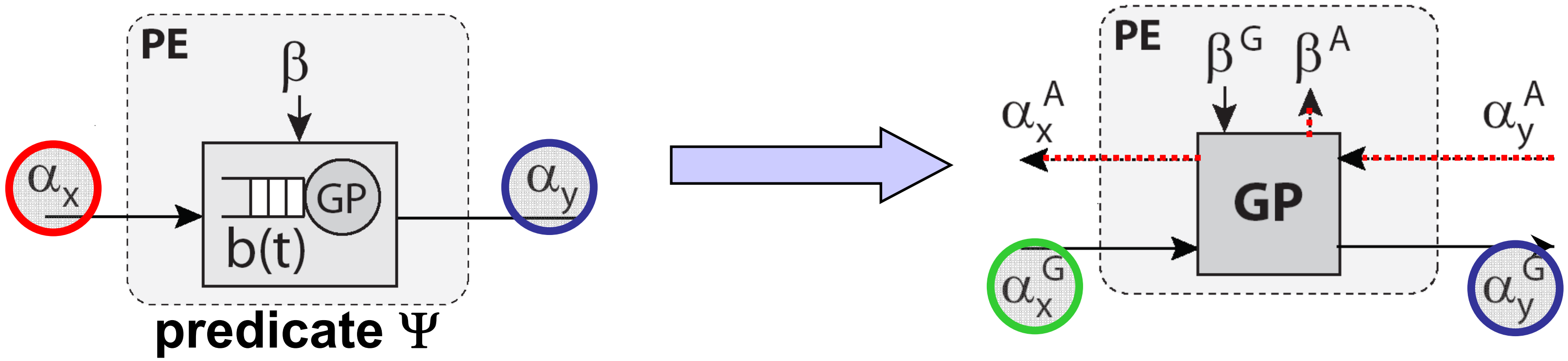
What are feasible processor speeds ?

What are feasible scheduling policies ?

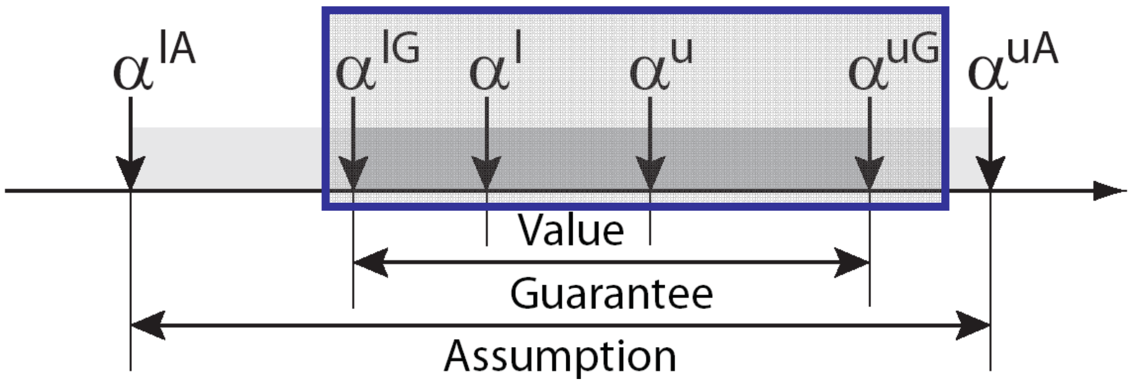
Complete System Composition



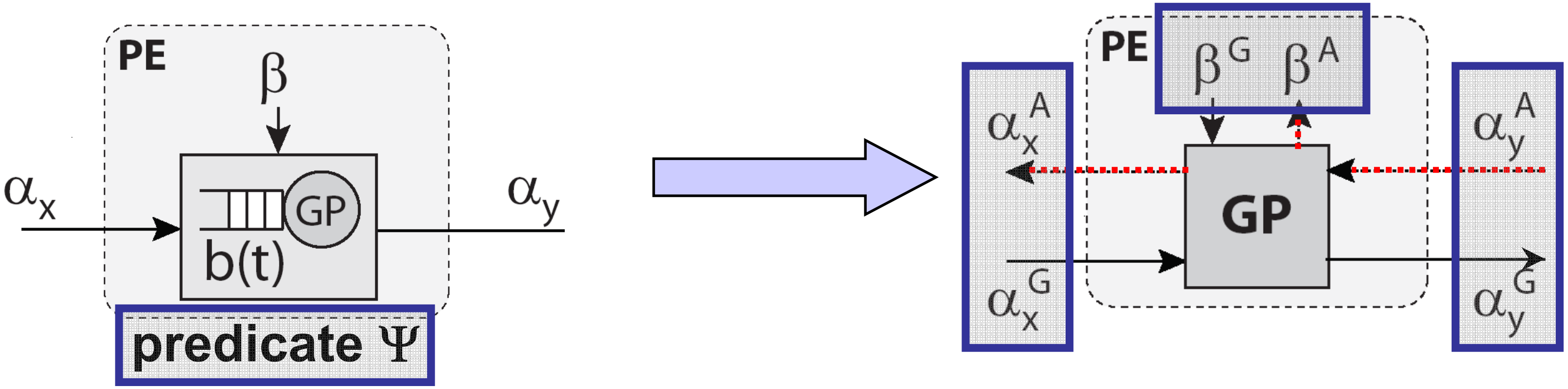
From Components to Rate Interfaces



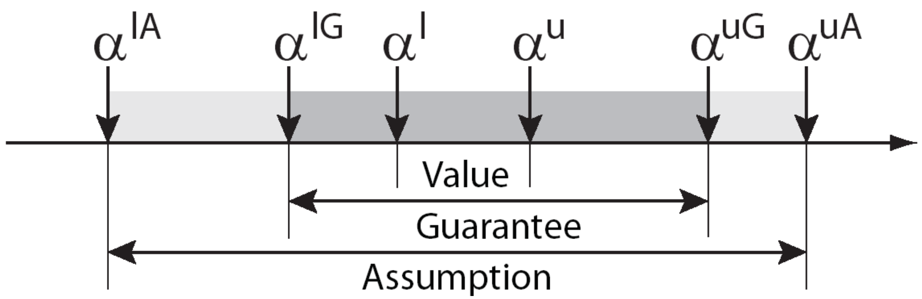
1. The environment guarantees that $\alpha_x \subseteq \alpha_x^G$ and $\beta \subseteq \beta^G$.
2. Then, GP guarantees that $\alpha_y \subseteq \alpha_y^G$.



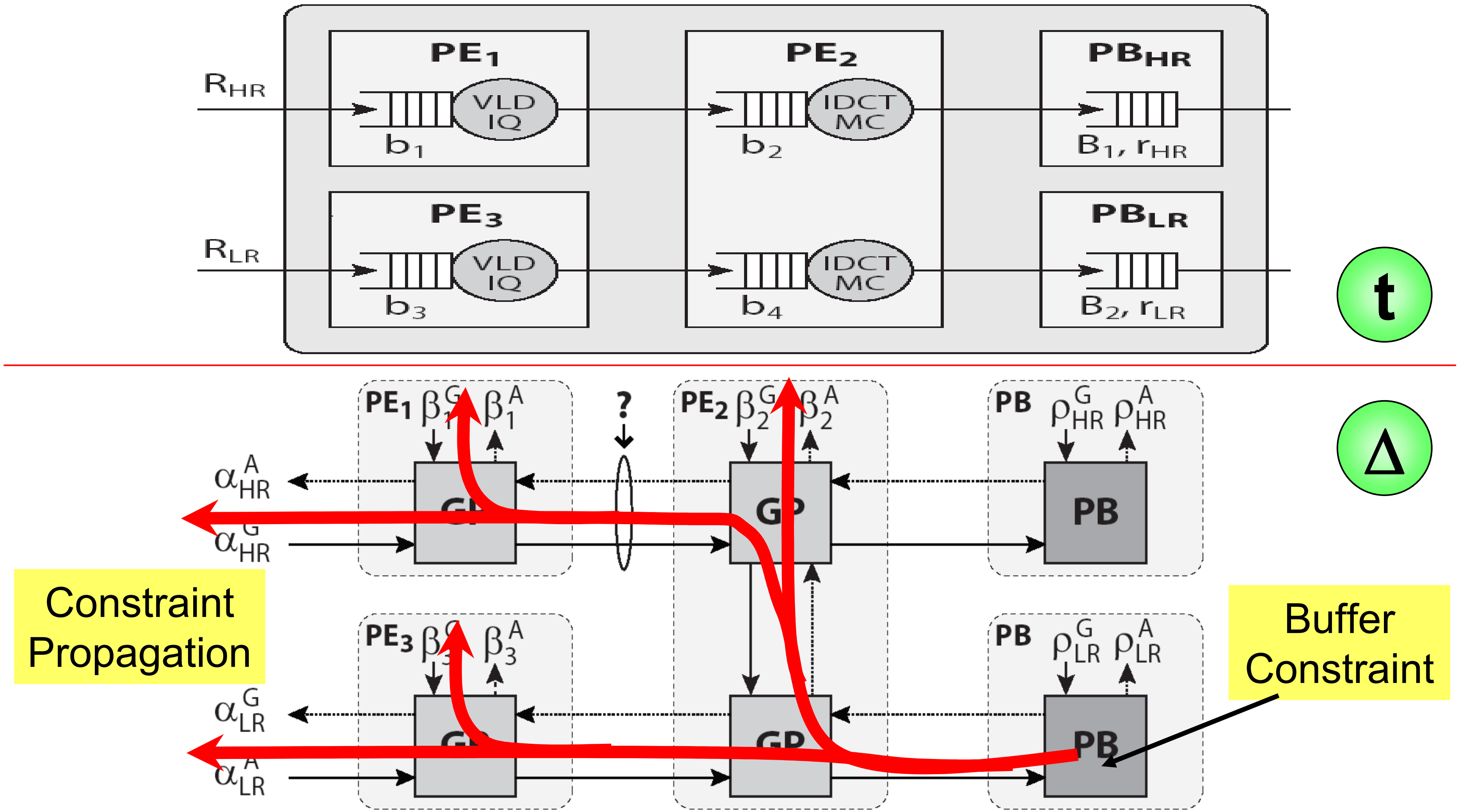
From Components to Rate Interfaces



1. Input guarantees match input assumes ($\alpha_x^G \subseteq \alpha_x^A \quad \beta^G \subseteq \beta^A$).
2. Then, predicate Ψ is satisfied.
3. Then, output guarantees match output assumes ($\alpha_y^G \subseteq \alpha_y^A$).

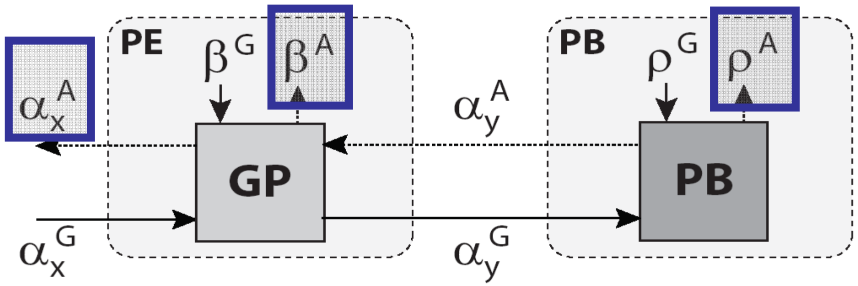


What is this good for ?



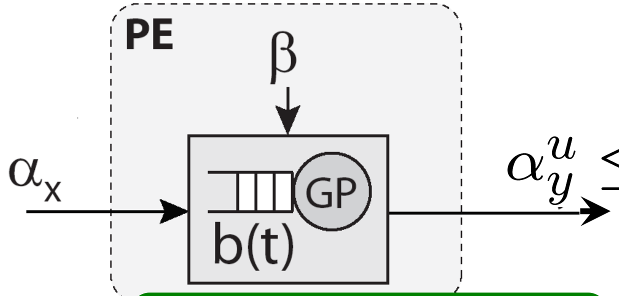
How do we compute the input assumes ?

- How do we compute the input assumes ?
  - 'Inverses' of (forward) analysis functions (in terms of partially ordered set) [Thiele et.al. EMSOFT 06].
  - Forward functions are based on max-+ algebra; therefore, assumes can be computed explicitly.



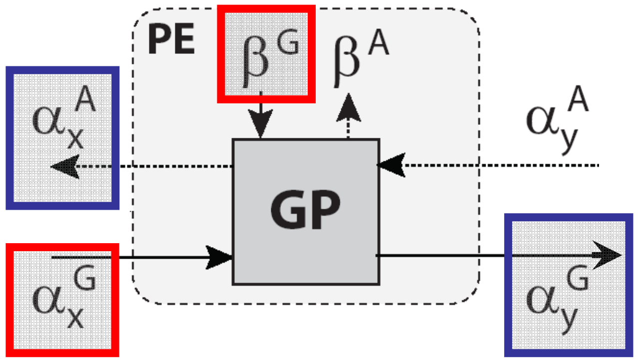
$$\alpha_x^{lA} = (\rho^{uG} - B_0) \oslash \beta^{lG}$$
$$\alpha_x^{uA} = \min\{\beta^{lG} \otimes (\rho^{lG} + B_{max} - B_0), \beta^{lG} + b_{max}\}$$
$$\beta^{lA} = \max\{(\rho^{uG} - B_0) \oslash \alpha_x^{lG}, \alpha_x^{uG} \oslash (\rho^{lG} + B_{max} - B_0), \alpha_x^{uG} - b_{max}\}$$
$$\rho^{lA} = (\alpha_x^{uG} \oslash \beta^{lG}) - (B_{max} - B_0)$$
$$\rho^{uA} = (\alpha_x^{lG} \otimes \beta^{lG}) + B_0$$

Backward Transformation



$\alpha_y^u \le \alpha_x^u \oslash \beta^l$

$\Psi : \alpha^u \le \beta^l + b_{max}$



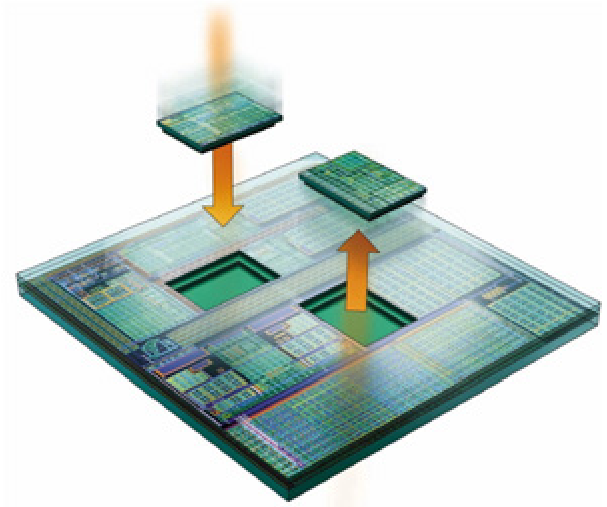
$$\alpha_y^{uG} = \alpha_x^{uG} \oslash \beta^{lG}$$

$$\alpha_x^{uA} = \min\{\beta^{lG} \otimes \alpha_y^{uA}, \beta^{lG} + b_{max}\}$$

Component-Based Analysis and Design

Analysis:

Infer system properties from subsystem properties.



Component-Based Analysis:

Check compatibility of component-interfaces.

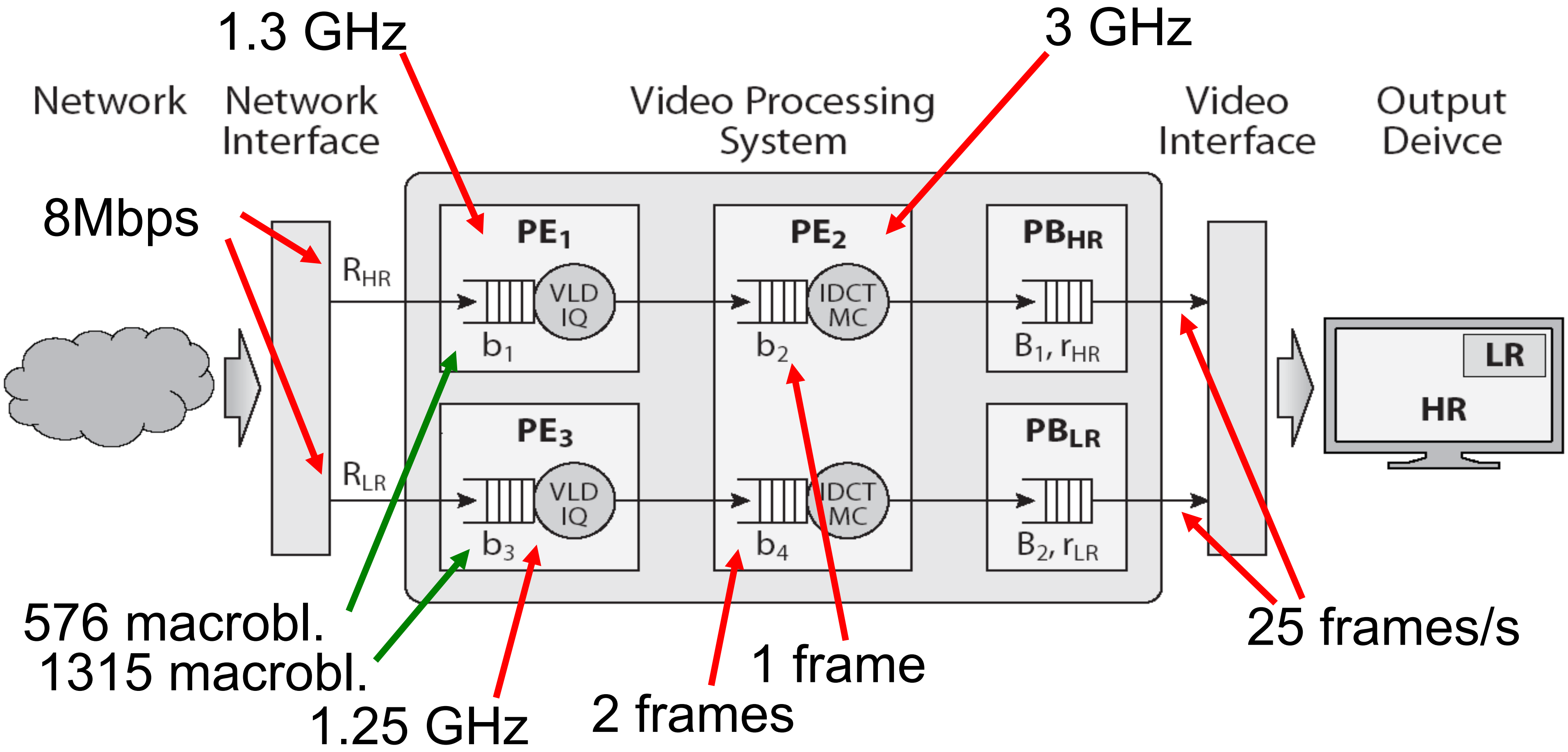
Design:

Build a system from subsystems while meeting requirements.

✓

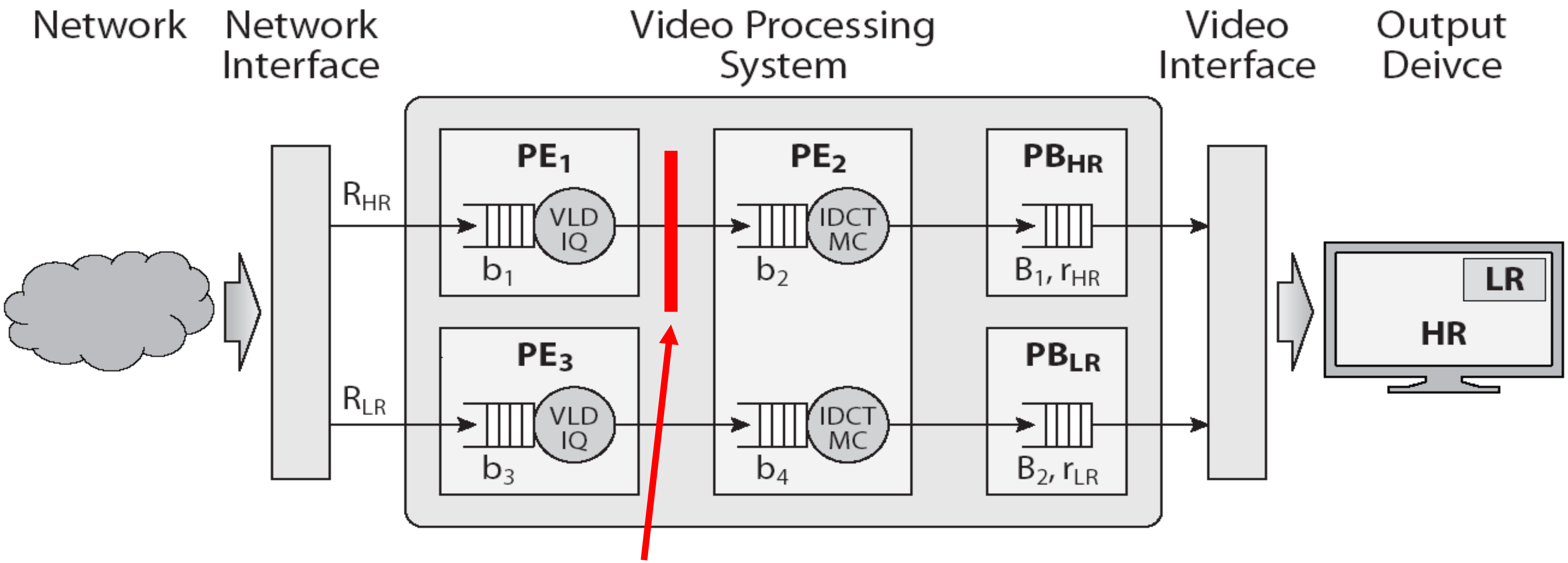
Scenario

- Parameters: SimpleScalar Simulation



Scenario

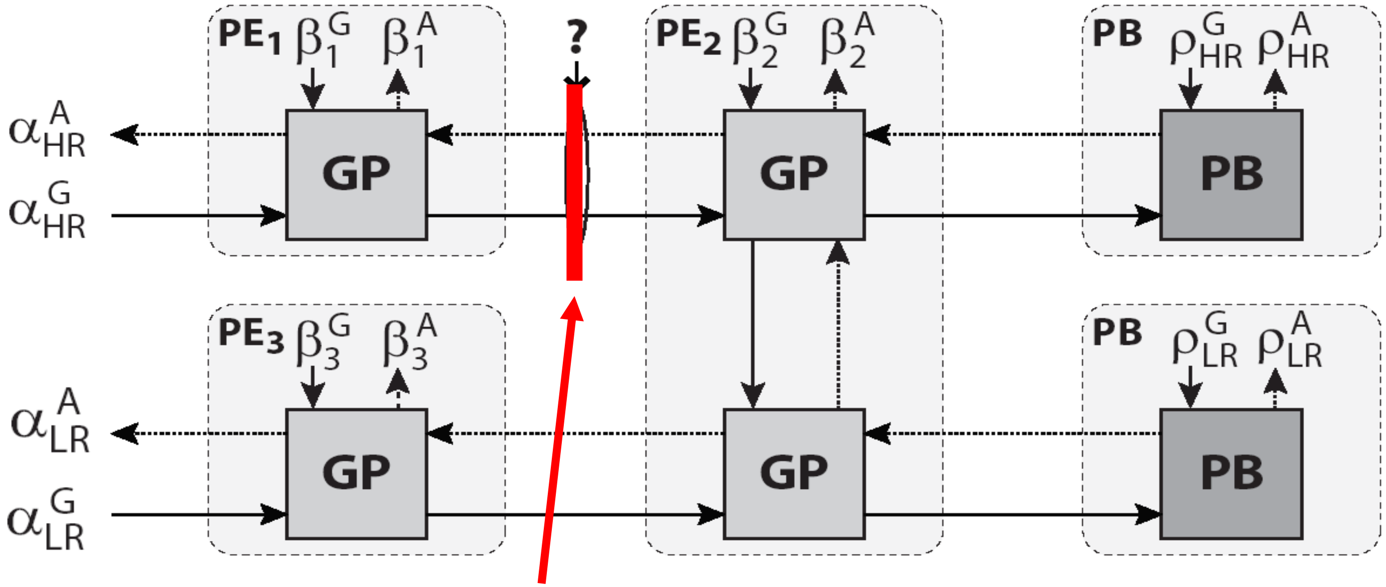
Is PE_1 compatible with the rest of the system?
What are the requirements towards PE_1?



Are the interfaces compatible ?

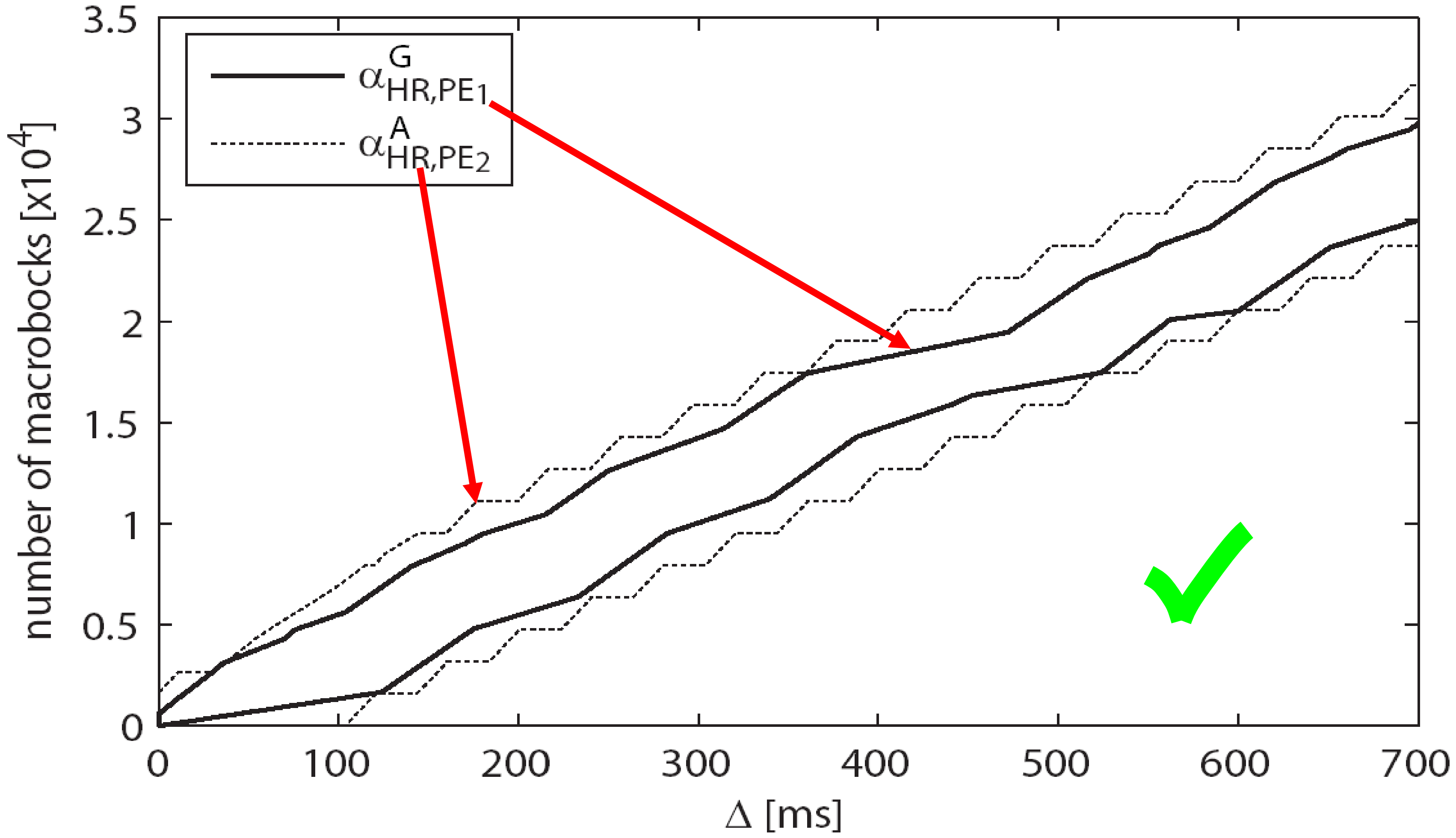
Scenario

Is PE_1 compatible with the rest of the system?
What are the requirements towards PE_1?



Are the interfaces compatible ?

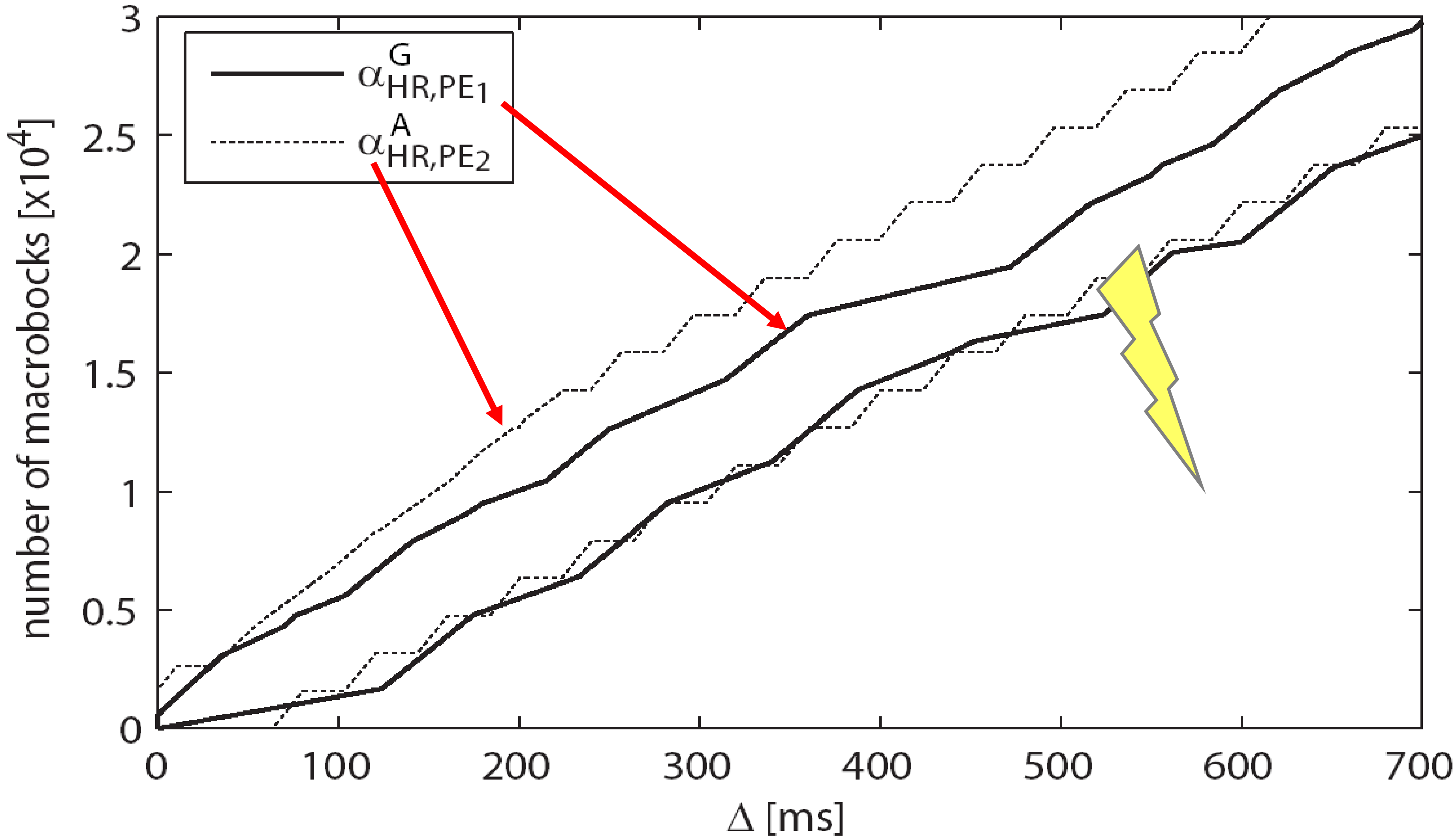
Experiment 1



ax = 6

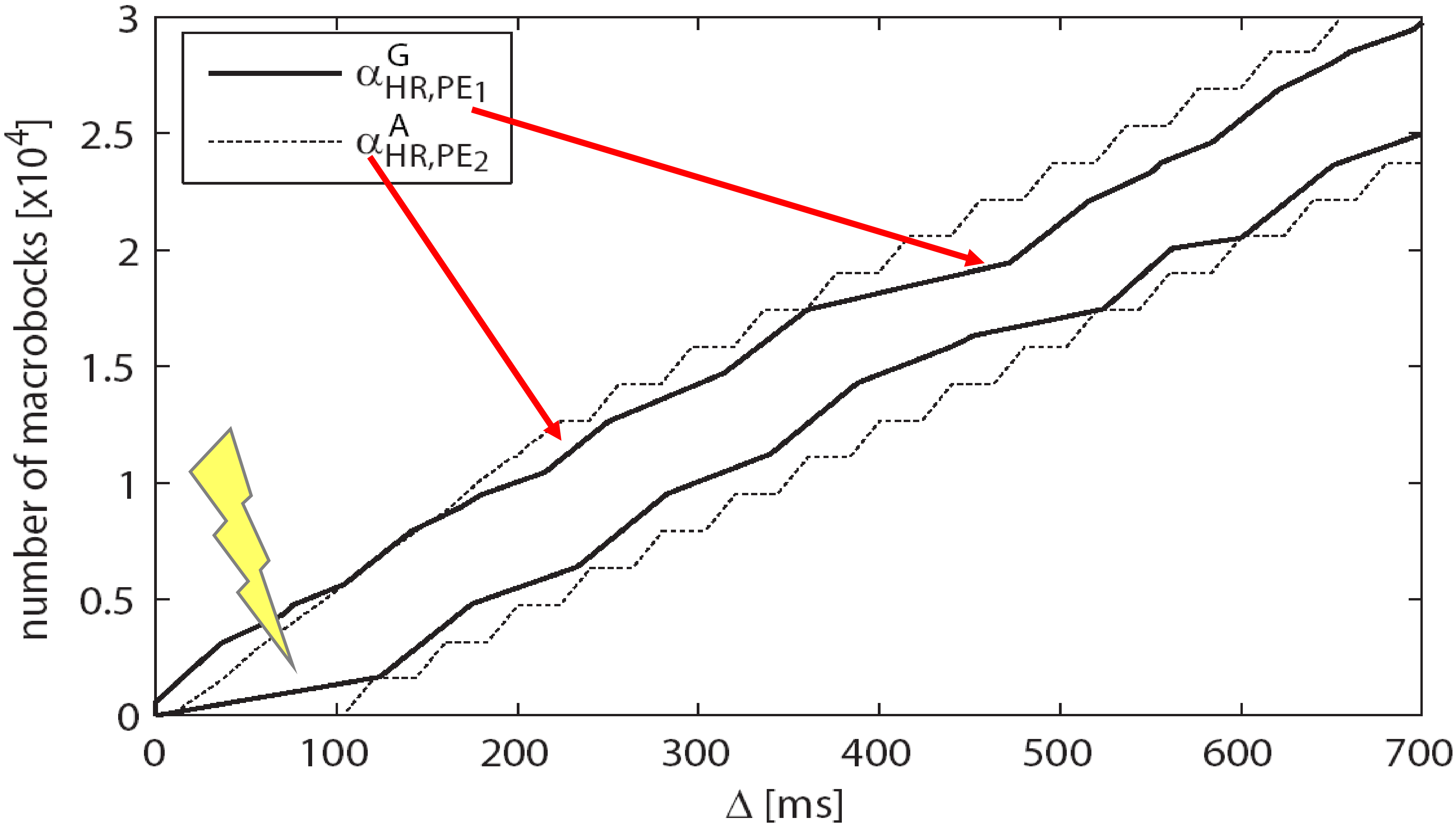
) = 3

Experiment 2

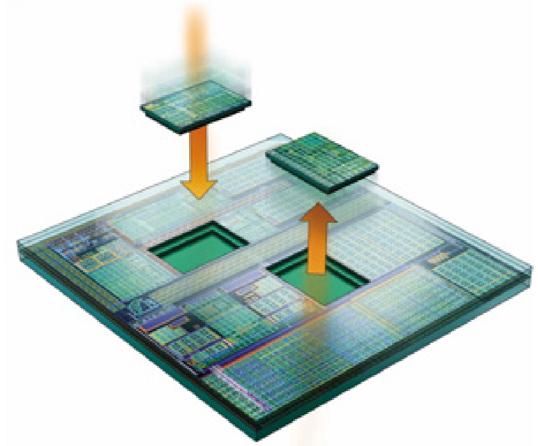


6
2
: 6
3

Experiment 3



Concluding Remarks



Analysis and Design

Component-based approach

Rate Interfaces

Check compatibility of component interfaces

Implementation

MPA toolbox
http://www.mpa.ethz.ch

Acknowledgement

E. Wandeler, N. Stoimenov, S. Chakraborty

Scope:

EDF, servers, TDMA, GPS

Constraints:

Delay, buffer, processing, communication, input rates