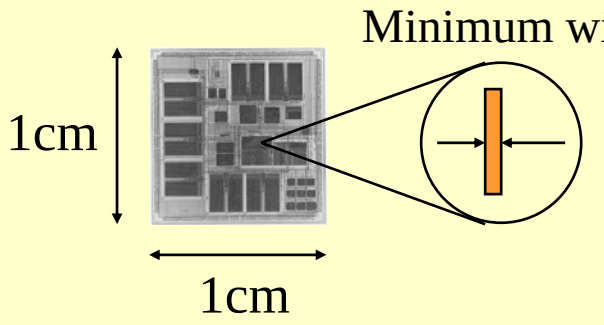
極微細LSIのタイミング設計

Timing Issues in Nanotechnology LSI

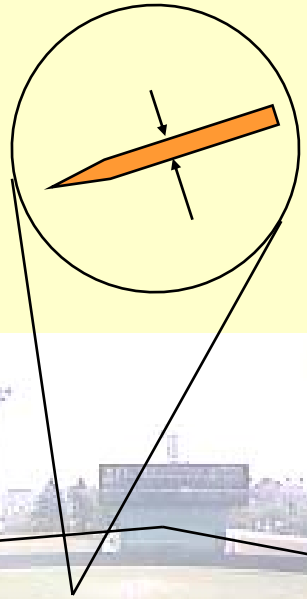
金子峰雄
北陸先端科学技術大学院大学 情報科学研究科

Mineo Kaneko
Information Science, JAIST

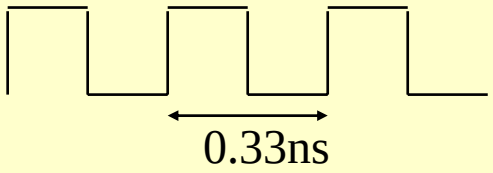
VLSI in the Year 2007



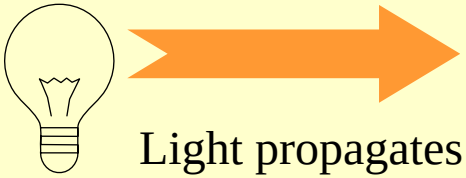
$$\frac{60\,[\mathrm{nm}]}{1\,[\mathrm{cm}]} = \frac{1\,[\mathrm{mm}]}{167\,[\mathrm{m}]}$$



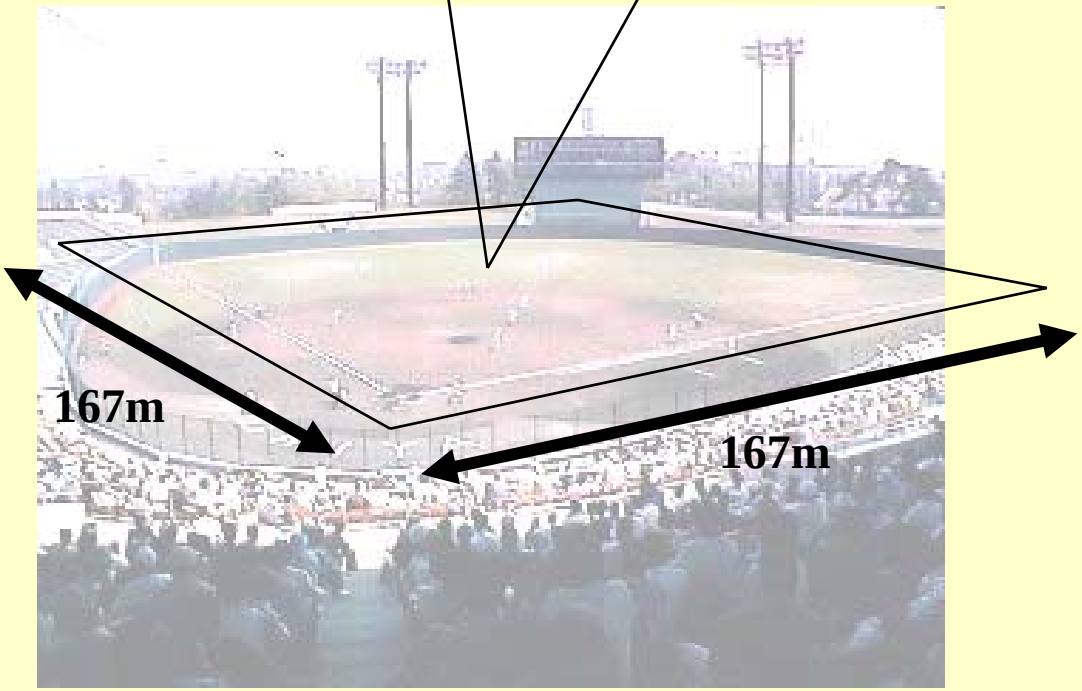
Clock frequency 3GHz



$$(3\times10^{8}\,[\mathrm{m}])\times(0.33\times10^{-9}\,[\mathrm{s}])$$
$$= 0.1\,[\mathrm{m}]$$

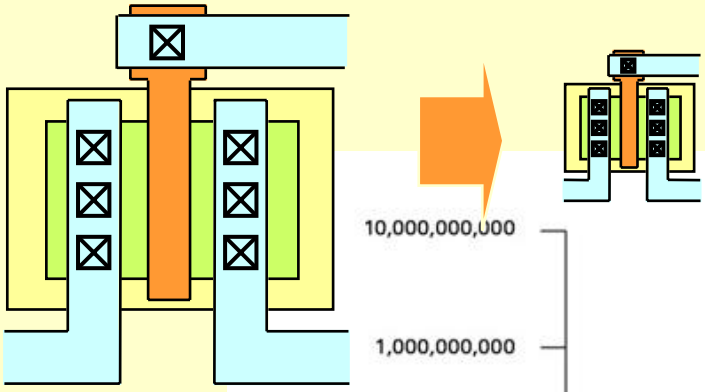


Light propagates 10cm in 0.33ns.



Arranging 1mm ϕ wire in Baseball ground

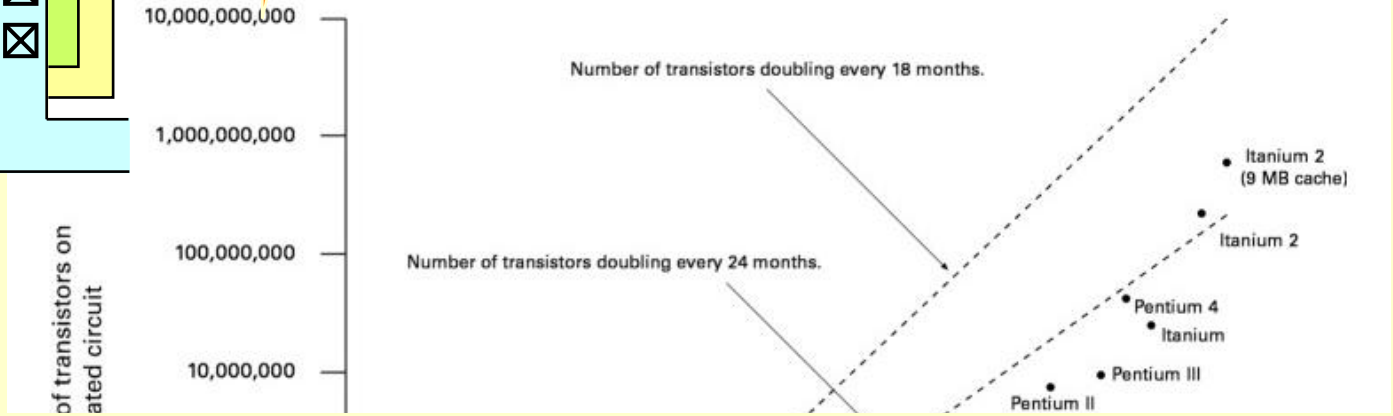
History of IC = History of Shrinking



Moore's Law (Gordon E. Moore)

Complexity grows double in every 18-24 months

"Cramming more components onto integrated circuits", Electronics Magazine 19 April 1965



Shrink → High Space-Density → More Transistors in a chip

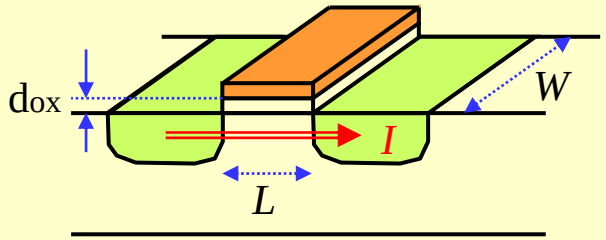
Shrink → Improved Tr. performance → High Speed IC

Further shrinking

→ Large propagation delay,
Inaccuracy in delay estimation,
Static and dynamic delay fluctuation

Electrical Aspect of VLSI

MOS Transistor



Current I, Voltage V, Capacitance C_g

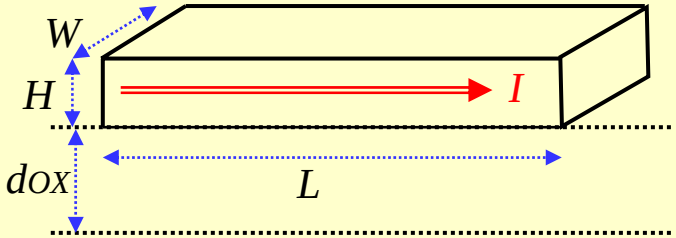
$$I \propto d_{ox}^{-1} \cdot W \cdot L^{-1} \cdot V^2$$

$$C_g \propto d_{ox}^{-1} \cdot W \cdot L$$

Switching delay; *delay*

$$delay \propto \frac{C_g \cdot V}{I} = \frac{d_{ox}^{-1} \cdot W \cdot L \cdot V}{d_{ox}^{-1} \cdot W \cdot L^{-1} \cdot V^2} = L^2 \cdot V^{-1}$$

Wire



Resistance R_W, Capacitance C_W

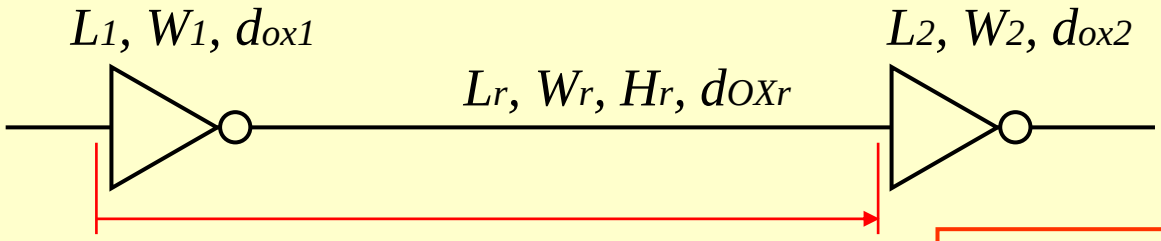
$$R_W \propto L \cdot H^{-1} \cdot W^{-1}$$

$$C_W \propto W \cdot L \cdot d_{OX}^{-1}$$

Propagation delay; *delay*

$$delay \propto \frac{C_W \cdot V}{I} = C_W \cdot R_W = \frac{W \cdot L^2}{H \cdot W \cdot d_{OX}}$$

Electrical Aspect of VLSI



Delay = Switching delay + Propagation de

Propagation delay is not improved by shrinking

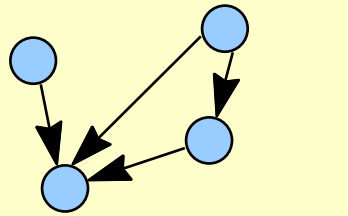
$$delay = \lambda \frac{\left(d_{OXr}^{-1} W_r L_r + d_{ox2}^{-1} W_2 L_2\right) L_1}{d_{ox1}^{-1} W_1 \cdot V} + \kappa \frac{L_r}{W_r H_r}\left(\frac{d_{OXr}^{-1} W_r L_r}{2} + d_{ox2}^{-1} W_2 L_2\right)$$

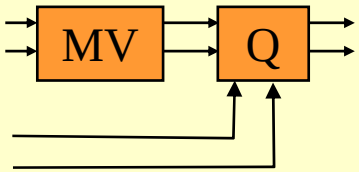
Elmore delay
(First Moment Model)

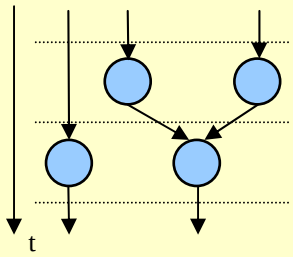
Difficulty in delay estimation: Need higher-order model
Various parasitic effects
Static/Dynamic delay fluctuation

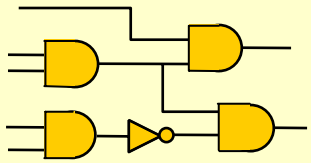
Static/Dynamic delay fluctuation: Fluctuations of chemical density and physical size in the fabrication process
Noise on supply voltage
Cross-talk noise

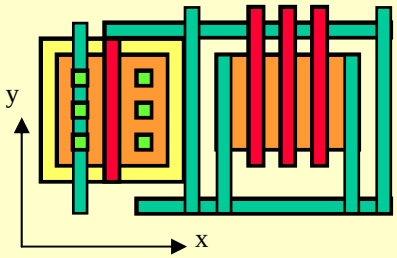
Top-down Hierarchical Design of VLSI











Specification

Algorithm-Level Design
- Data flow
- Control flow

System-Level Design
- HW/SW partitioning
- Memory architecture

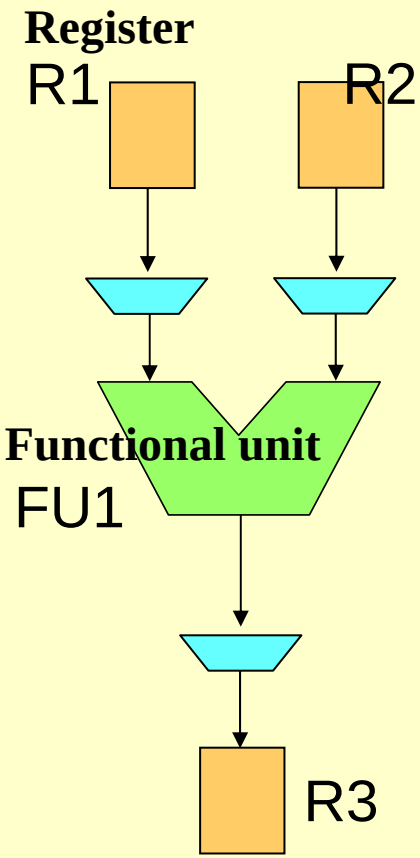
Register Transfer-Level Design
- Scheduling
- Binding
- Bus, multiplexer, net

Logic-Level Design
- Two-stage, Multi-stage
- State assignment
- Technology mapping

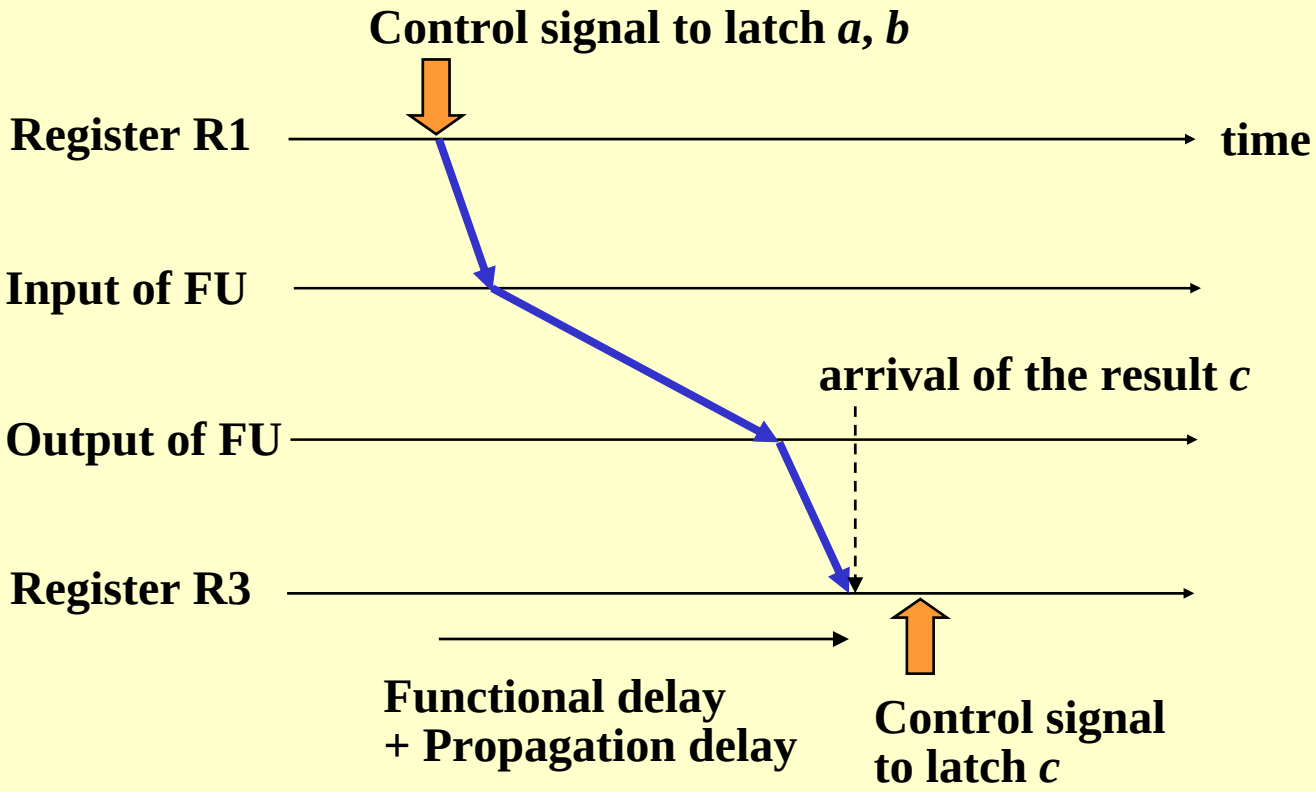
Physical-Level Design
- Transistor circuit
- Module generation
- Mask pattern design

Mask pattern

Timing issues in Data-path



Execution of $c = a + b$

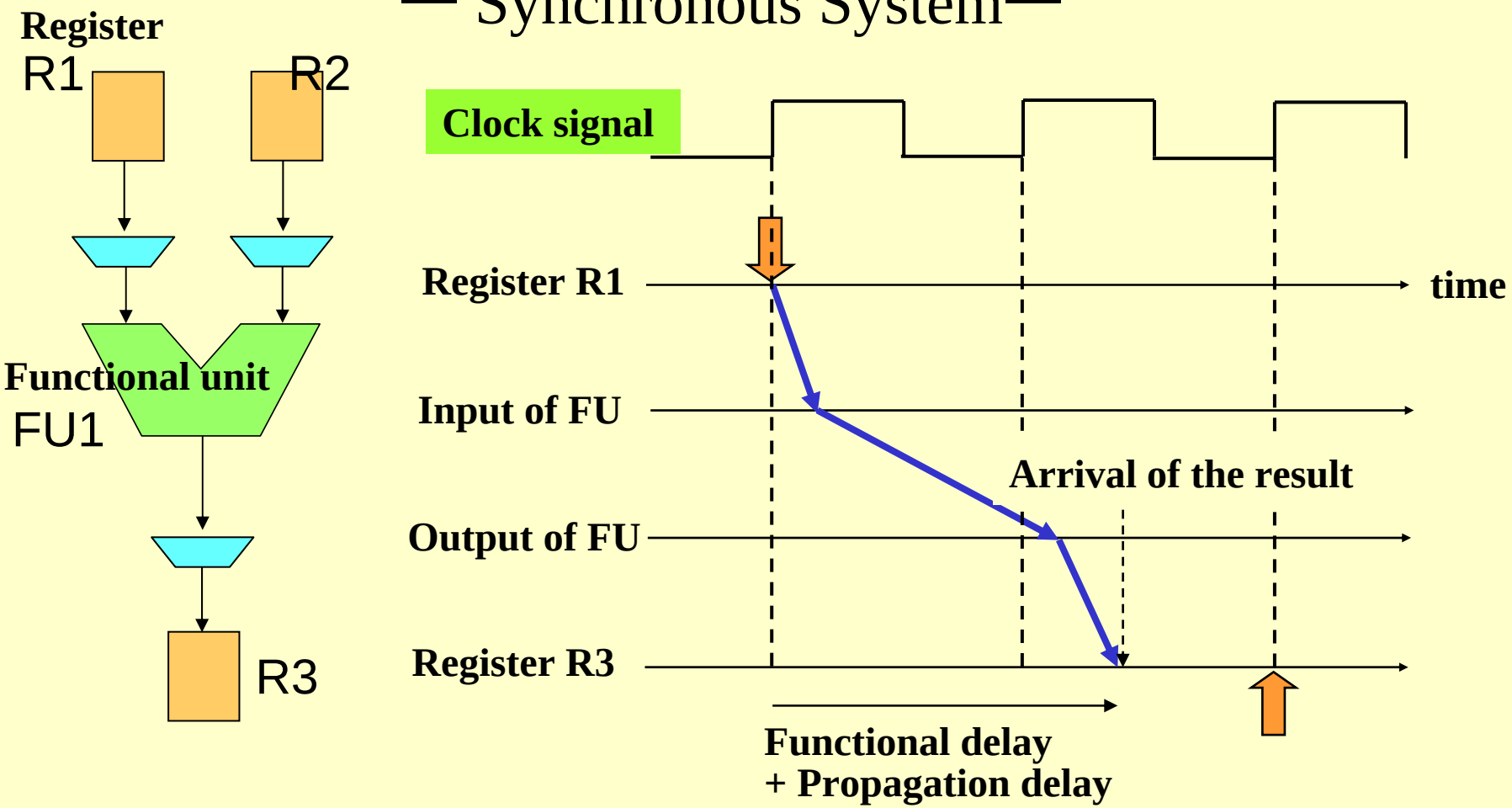


Timing of control signals (⇩, ⇧) determines data-path behavior

- Many operations share the same FU, many data share the same register.
- Various different delay values
- Delay may vary from its nominal value statically and dynamically.

Timing issues in Data-path

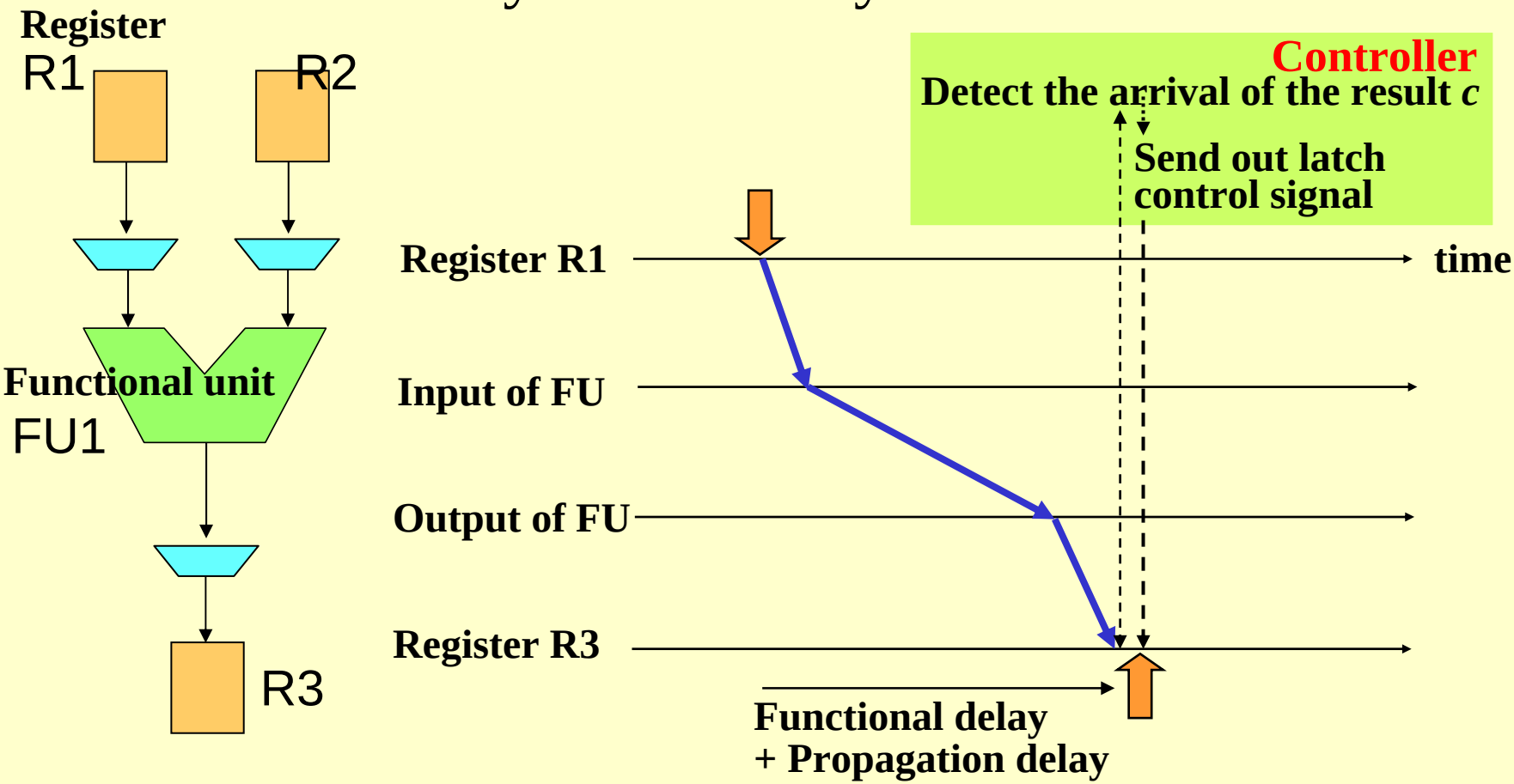
— Synchronous System —



- Design needs ``Delay estimation'' + ``Timing margin''
- Easy to implement as a circuit
- Worst-case estimation + Sufficient margin = Low performance

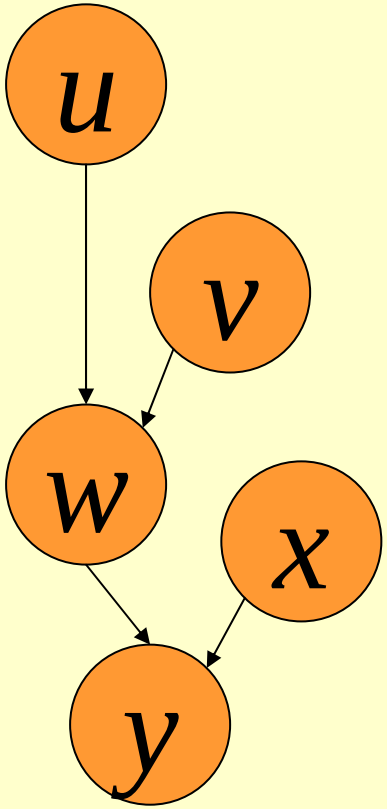
Timing issues in Data-path

— Asynchronous System —

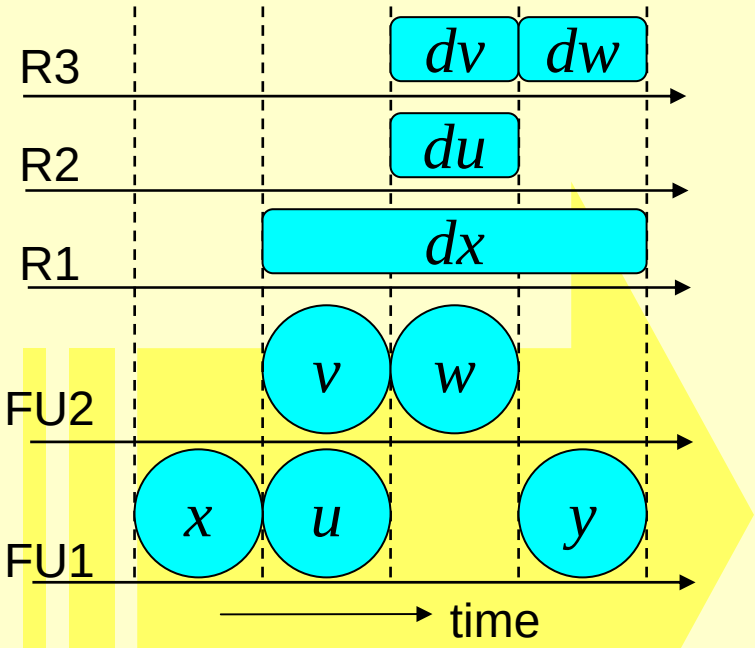


- No delay-estimation, no timing-margin
- Tolerance to a large range of delay fluctuation
- Large area (circuit) overhead in detecting-circuitry

What is High-Level Synthesis?



Behavioral description of an application algorithm

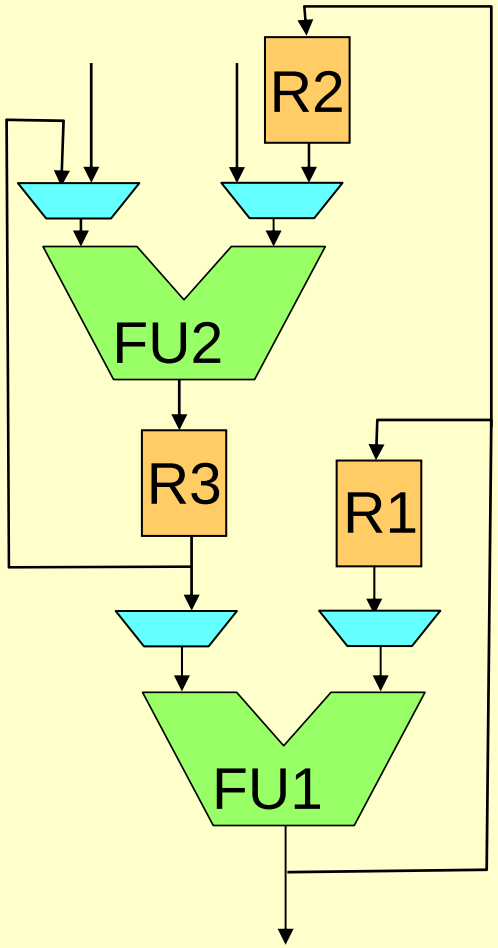


Scheduling;

determines the start time of each operation

Resource Binding;

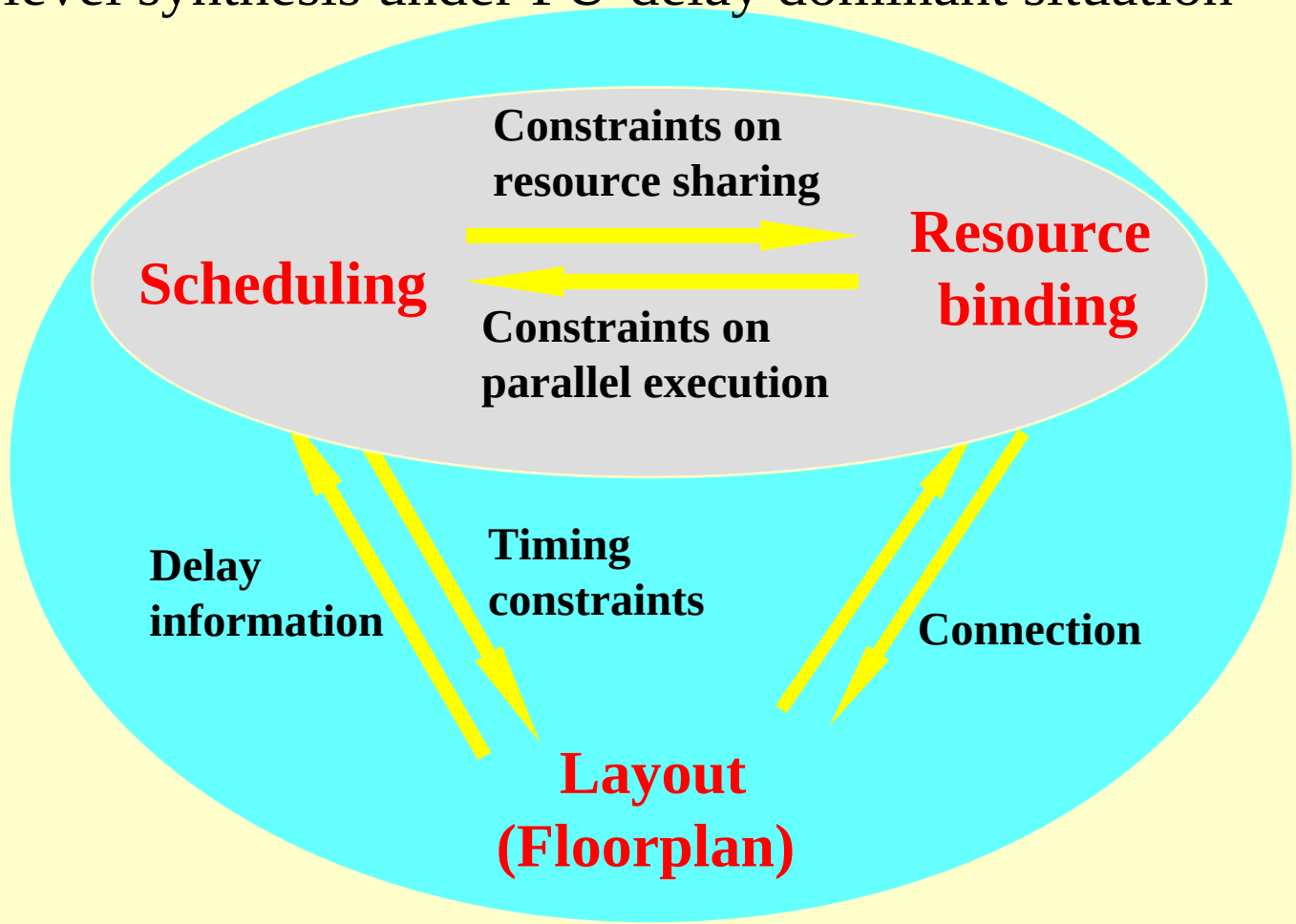
assigns each operation to one of available functional units, and assigns each data to one of available registers



Data-path part
＋Control part

A. New Approach to High-Level Synthesis

High-level synthesis under FU-delay dominant situation

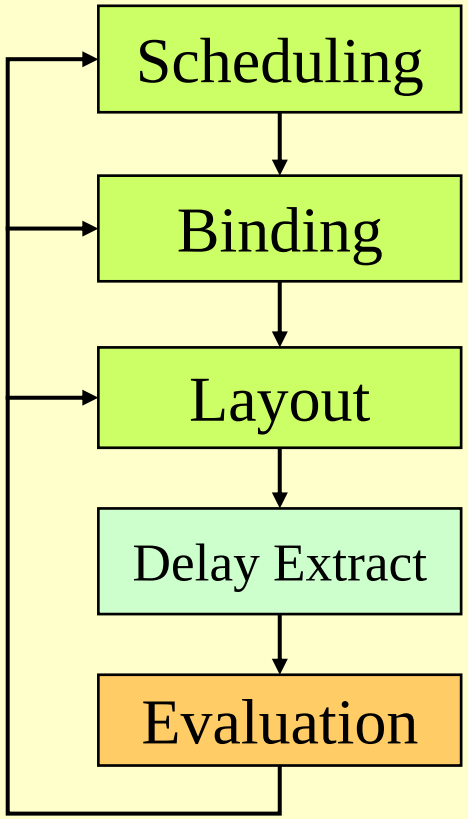


Wire-delay aware high-level synthesis:

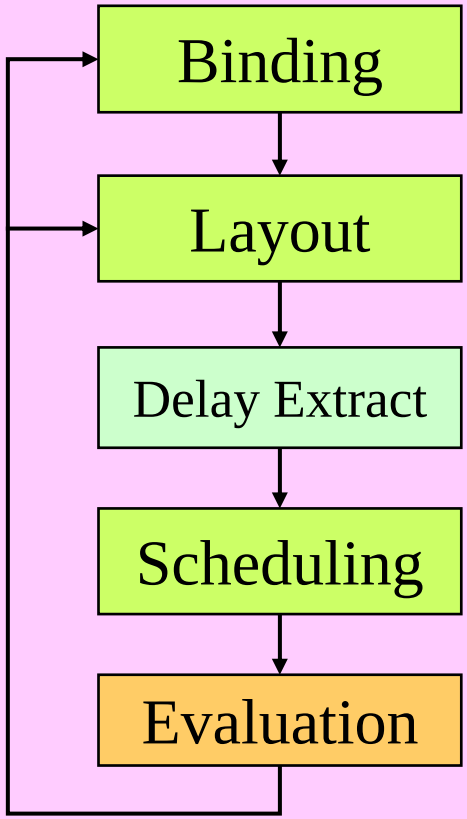
A. New Approach to High-Level Synthesis

Wire-delay aware high-level synthesis:

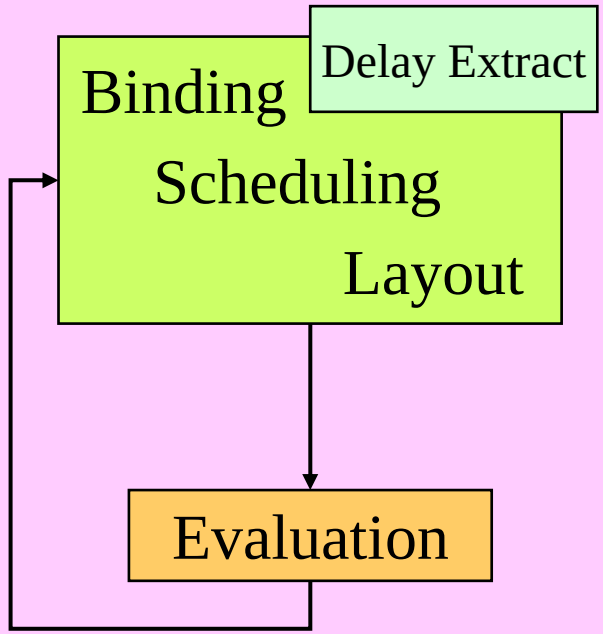
Scheduling-Centric



Binding-Centric

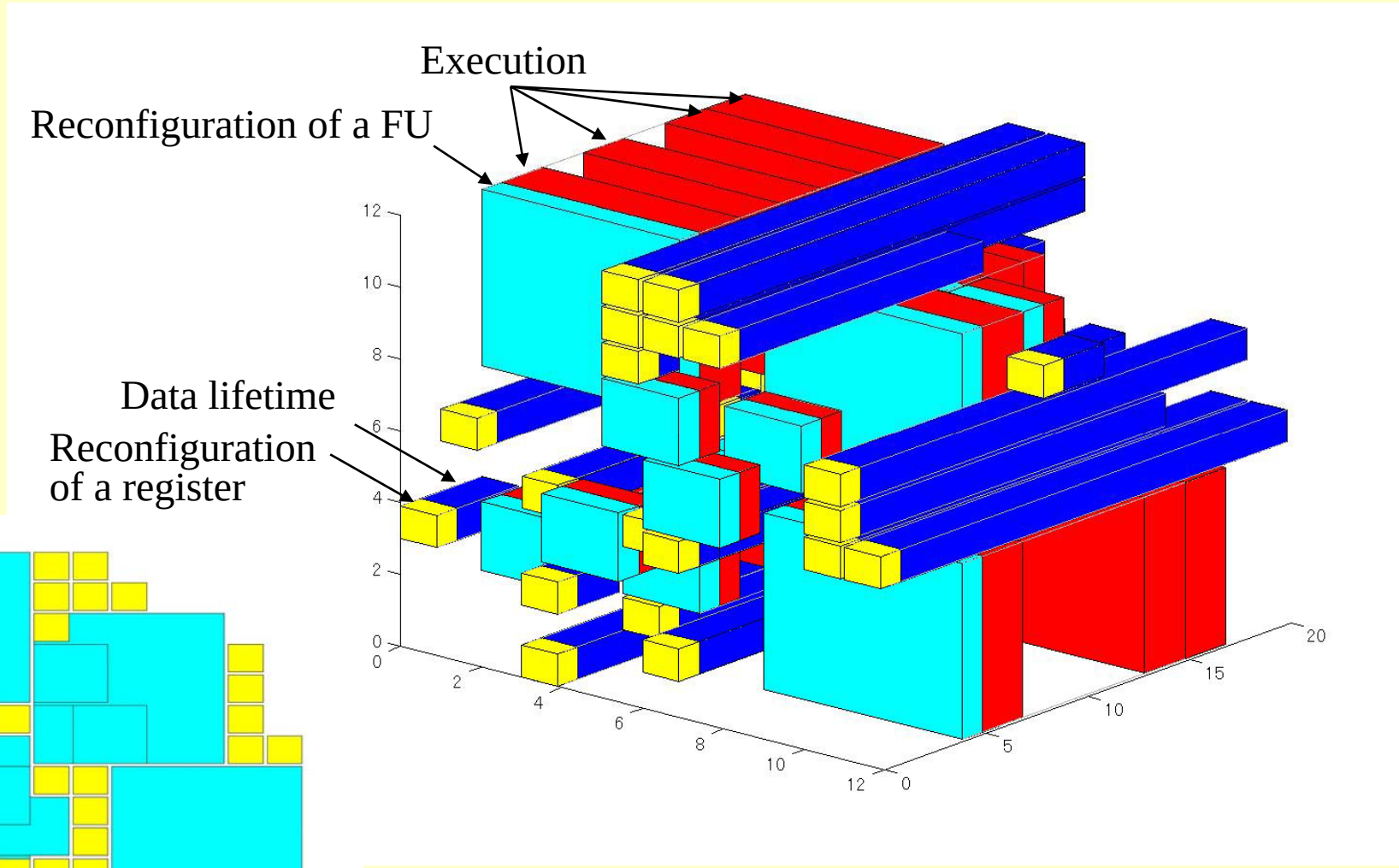


3D-Approach



A. 3D-Approach High-Level Synthesis

Application to Dynamically Reconfigurable LSI



A. 3D-Approach High-Level Synthesis

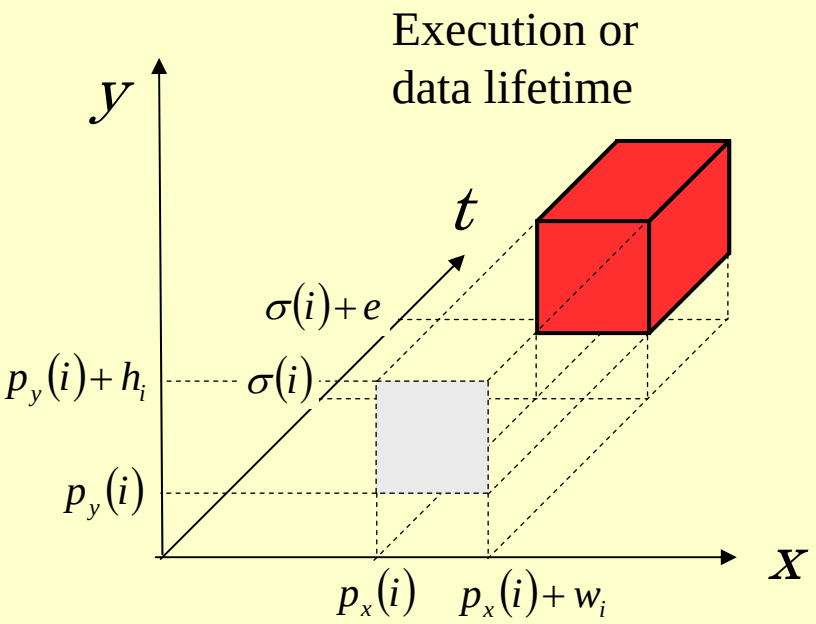
Computation algorithm to be implemented (Dependence Graph)

$DG = (O, D, A)$ O, D : Set of operations, and set of data

A : Dependency

$e : O \rightarrow N$; Operation delay

Sizes of functional units and registers



For each operation/data

$(p_x(i), p_y(i), \sigma(i))$

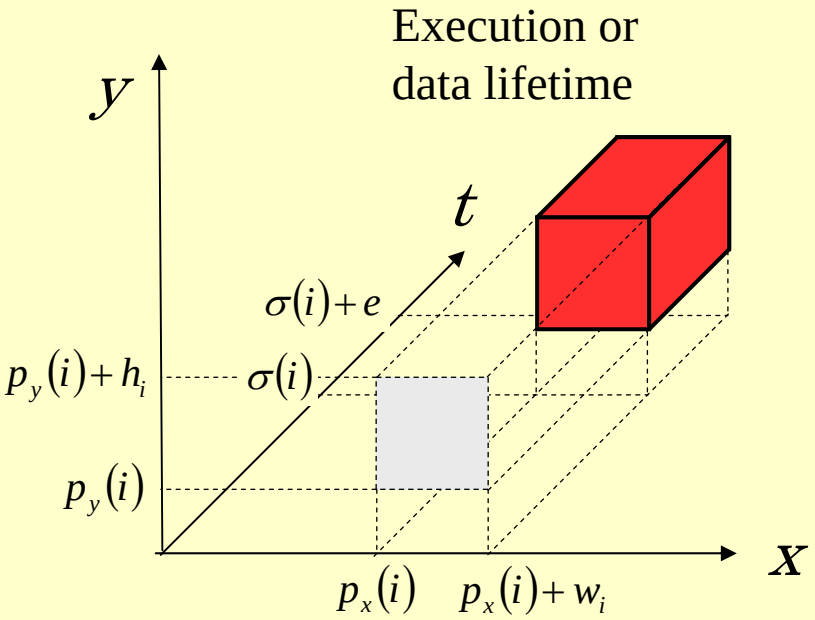
$p_x(i) \in N, p_y(i) \in N, \sigma(i) \in N$

Naive solution space $N^{3(|O|+|D|)}$

Need to check conflicts

Need to check timing constraints

A. 3D-Approach High-Level Synthesis



Constrained Sequence-Quintuple

5-tuple (Γ_1, Γ_2, Γ_3, Γ_4, Γ_5)

- Each of Γ_1, Γ_2, Γ_3, Γ_4 is a permutation of elements in O, D
- Γ_5 is a permutation of elements in O (a topological order w.r.t. DG)

- (Γ_1, Γ_2, Γ_3, Γ_4, Γ_5) represents relative spatial relation in *x*-*y*-*t* space.
- *O((|O|+|D|)^2)* computation-time algorithm to compute $\{(p_x(i), p_y(i), \sigma(i)) \mid i \in O \cup D\}$ which has the minimum layout area and the minimum makespan among all solutions satisfying the spatial relation specified by the code.
- The size of the solution space $((|O|+|D|)!)^5$

A. New Approach to High-Level Synthesis

Basic Theory

- Condition for feasible binding
- Efficient solution space for 3D-Approach to High-Level Synthesis

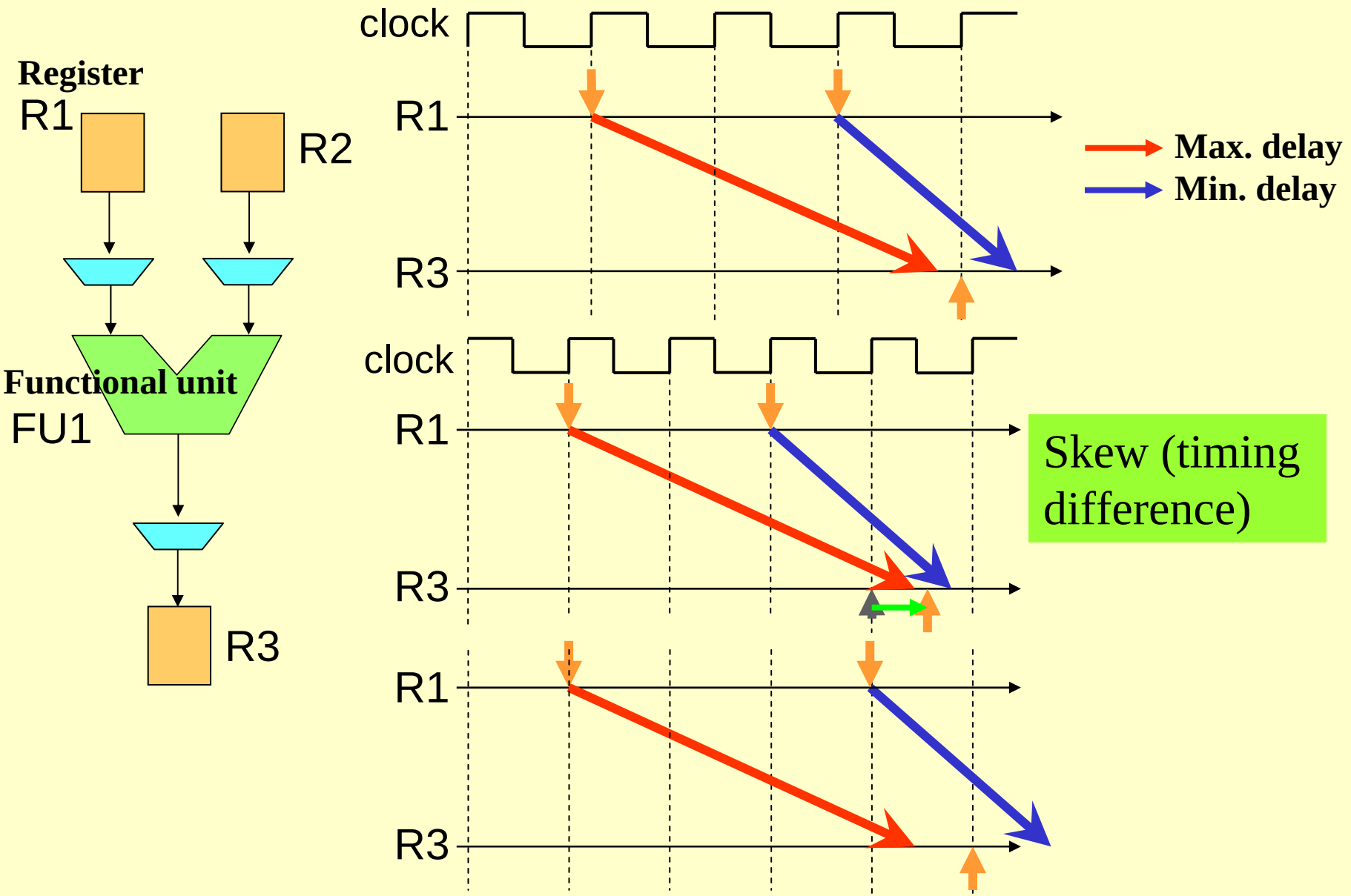
Elementary Technology

- Binding constrained scheduling
- Data-path layout, performance estimation

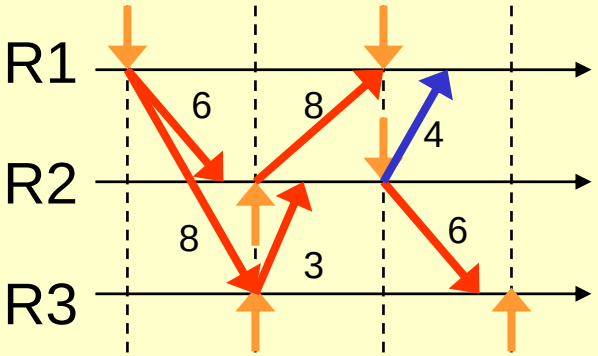
Synthesis System

- Synthesis system considering wire delay
- Synthesis system for reconfigurable systems
- Synthesis system considering control skew
- Synthesis system for asynchronous systems

B. Design Considering Skew

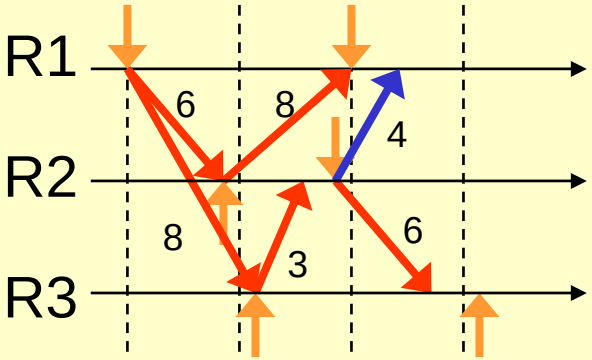


B. Schedule and Skew Optimization



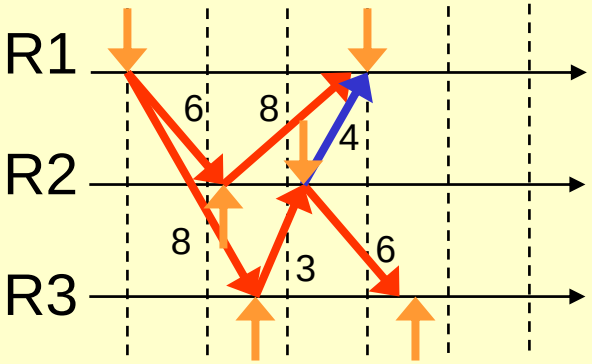
Optimum schedule under zero-skew

Minimum clock period＝8
Schedule length＝3
Total computation time ＝ 8×3+0 ＝ 24



Applying skew optimization
→skew values $(\tau_{r_1}, \tau_{r2}, \tau_{r3}) = (0,-1,1)$

Minimum clock period＝7
Schedule length＝3
Total computation time ＝7×3+1＝22



Simultaneous schedule and skew optimization
→skew value $(\tau_{r_1}, \tau_{r2}, \tau_{r3}) = (0,1,3)$

Minimum clock period＝5
Schedule length＝3
Total computation time＝5×3+3＝18

B. Skew-aware High-Level Synthesis

Basic Theory

- Computational Complexity:
  Fixed Schedule, Optimize Skew → P
  Simultaneous Schedule and Skew Optimization
  (even if the execution order is fixed) → NP-hard

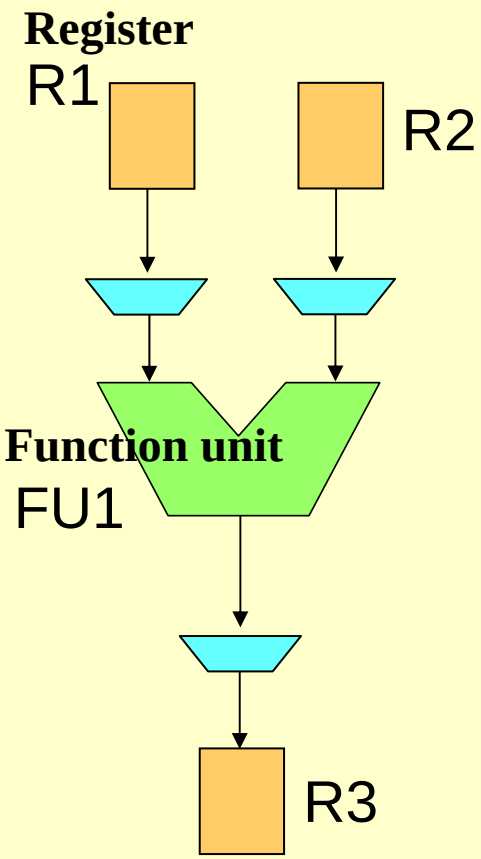
Elementary Technology

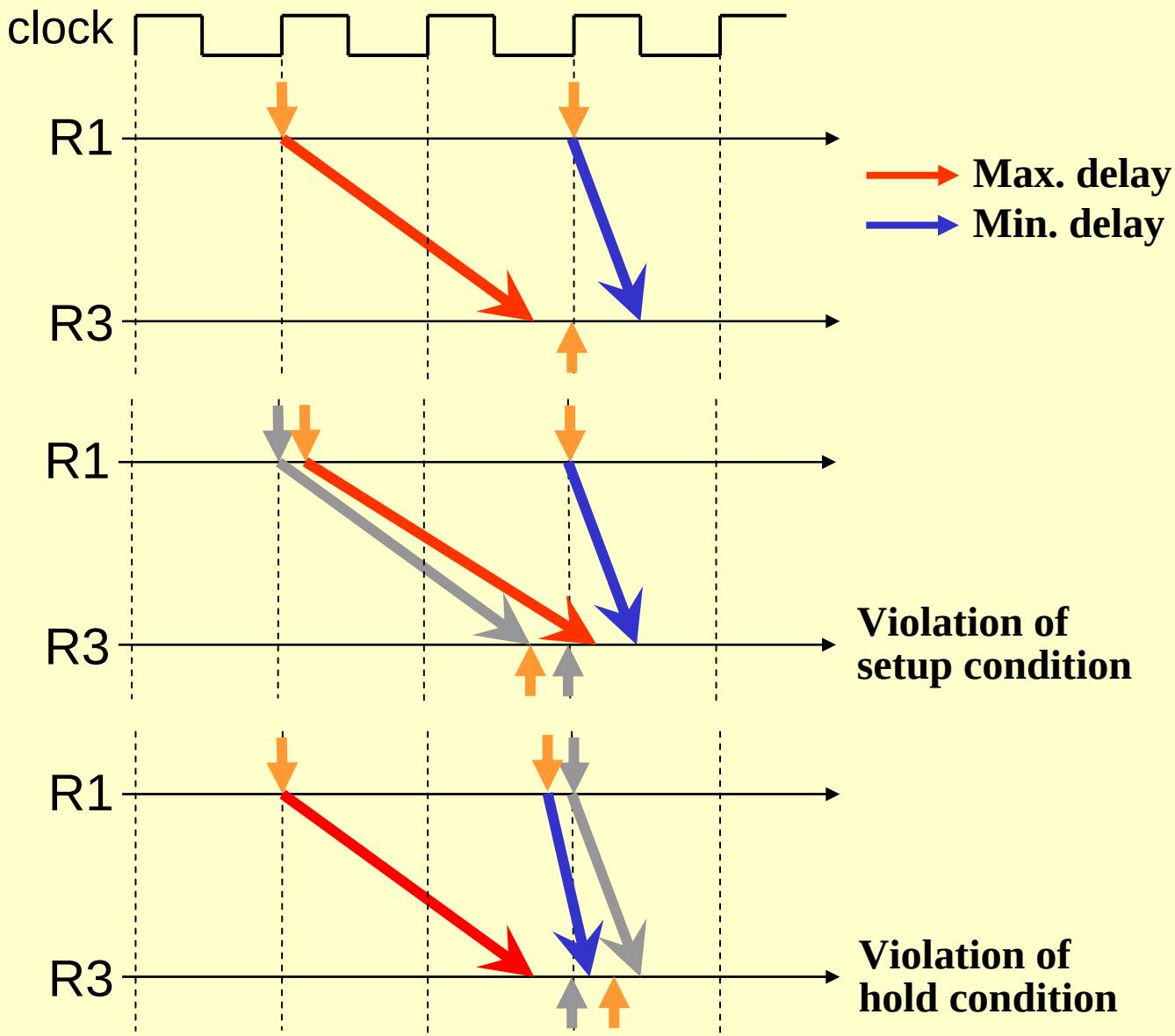
- Exact algorithm to compute optimum skew
- Heuristic algorithm for simultaneous schedule and skew optimization

Synthesis System

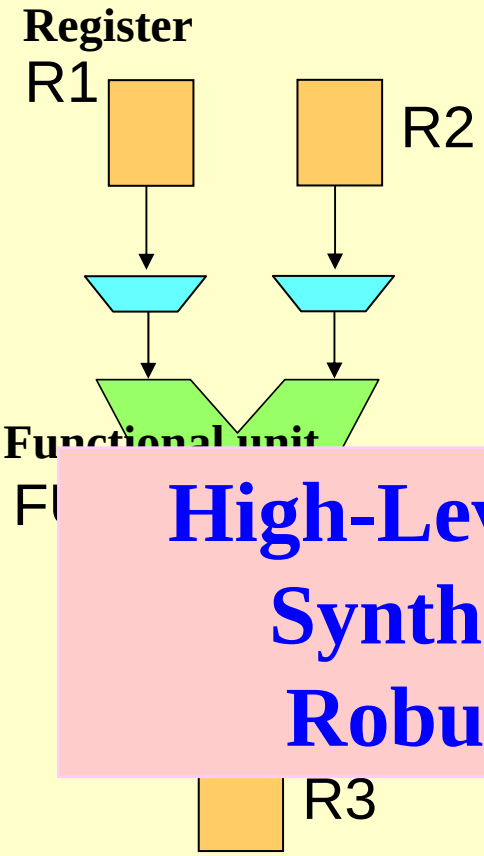
- Binding-centric approach/3D approach to skew-aware data-path synthesis

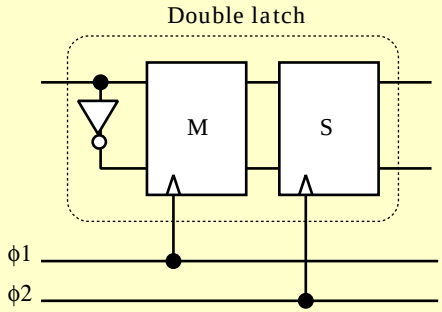
C. Delay Fluctuation

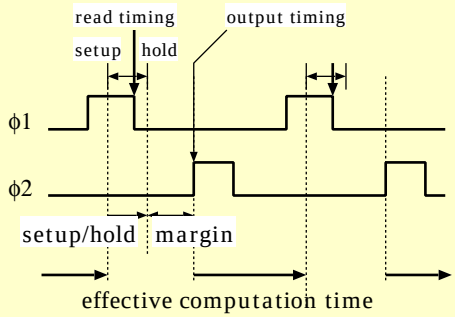


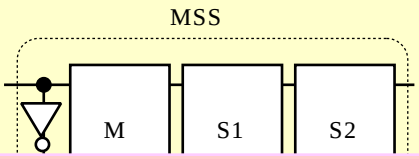


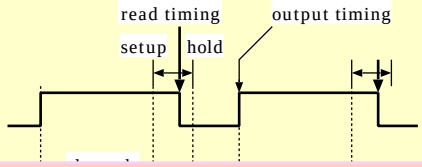
C. Delay Fluctuation



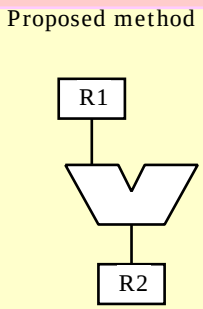


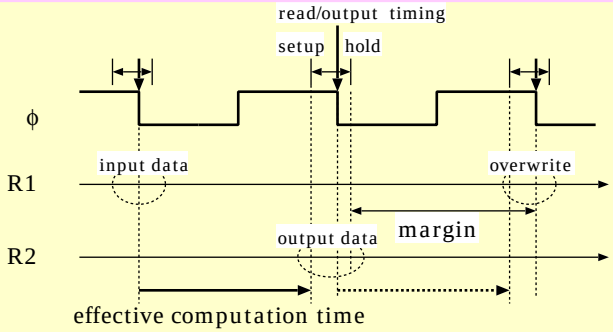




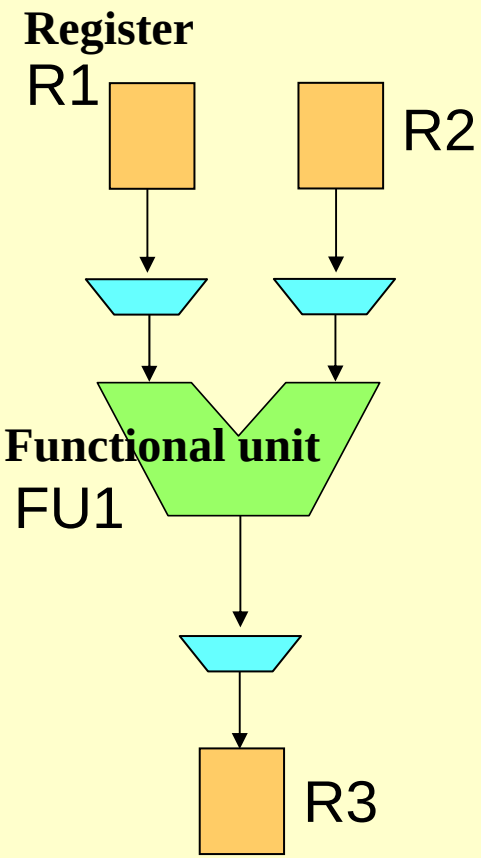


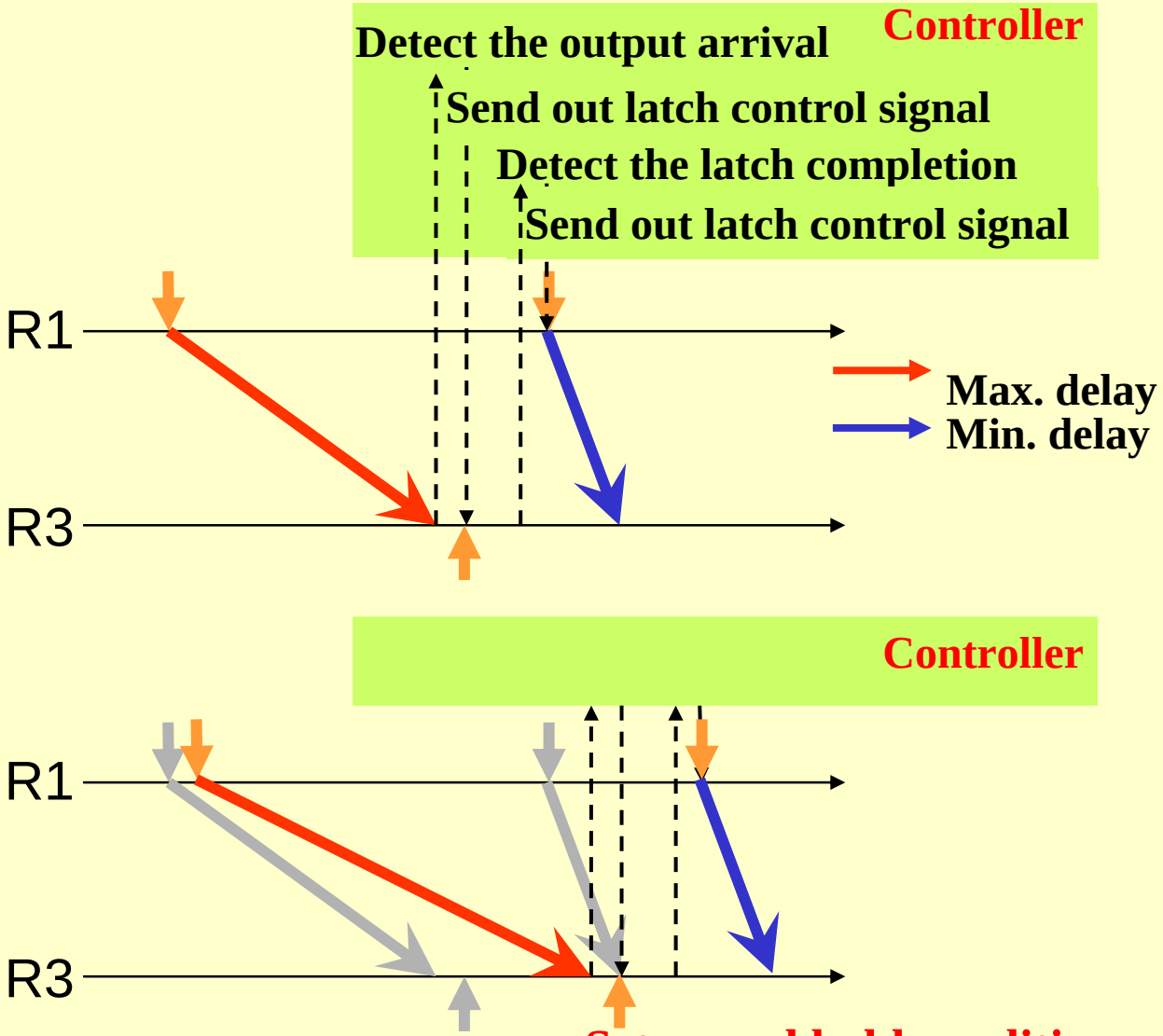
High-Level, Logic-Level, Circuit-Level Synthesis for VLSI which has the Robustness to Delay Fluctuation



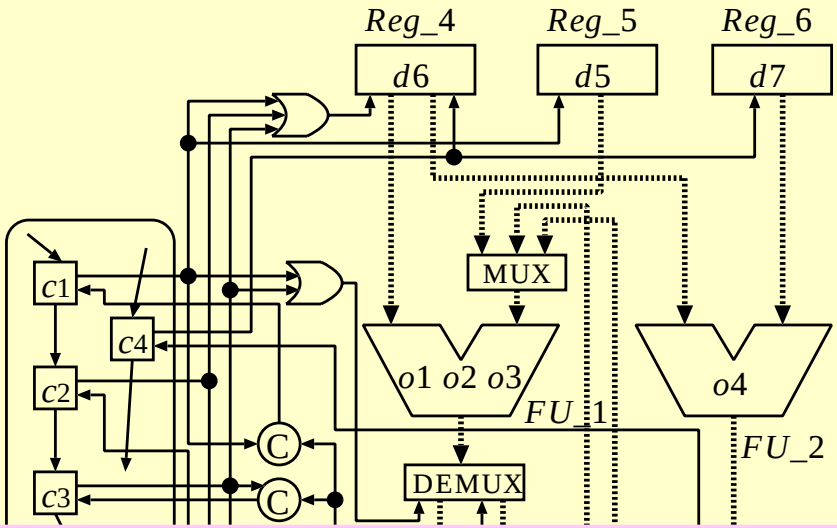


D. Asynchronous System



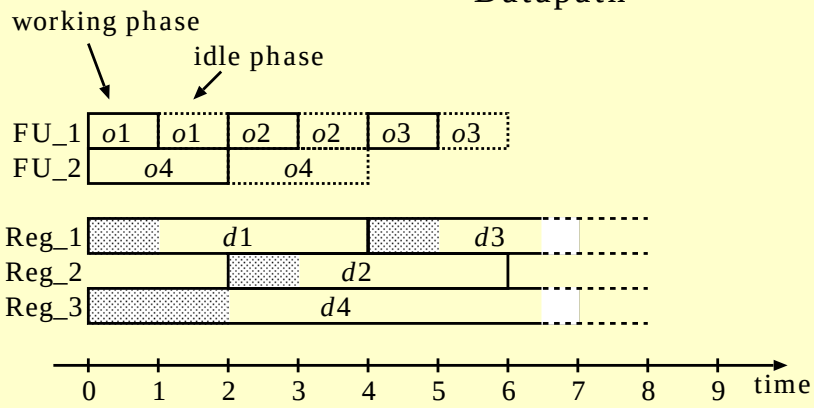


D. Asynchronous System



High-Level, Logic-Level, Circuit-Level Synthesis for Asynchronous System

Datapath



VLSI: a core device for reliable e-society

High speed, low power

- **Propagation delay and power consumption on signal/clock wires**
- **Static/dynamic delay fluctuation**

➡ **Considering layout in high-level design**

➡ **Robustness, tolerance, insensitiveness to delay fluctuation**

Large scale, system on chip

- **Huge size of optimization problems**

➡ **Design methodologies to break through the design crisis**

➡ **Efficient algorithms for huge size of problems**

Reliability

- **Complex design constraints, a large number of design variables**
- **Increasing circuit complexity, low voltage, chemical/physical fluctuation**

➡ **Reliable design: Reliable EDA tools, 100% automation**

➡ **Reliable chip: VLSI test, fault-tolerance**

High performance/Low power/Reliable System on Chip