Design for Low-Power IoT Systems:
Coarse-Grained Reconfigurable Acceleration Units

FRANCESCA PALUMBO

UNIVERSITÀ DEGLI STUDI DI SASSARI

Overview

- Motivations
  - What we need adaptation for
  - Triggers and Types
- Coarse-Grained Reconfigurable Systems
  - Computing Spectrum and Reconfigurable Systems Classification
  - Heterogeneous and Irregular Coarse-Grained Reconfigurable Accelerators
- Power Management
  - Issues and Strategies
  - Low-Power Coarse-Grained Reconfigurable Accelerators
- An FFT Example

Overview

- Motivations
  - What we need adaptation for
  - Triggers and Types
- Coarse-Grained Reconfigurable Systems
  - Computing Spectrum and Reconfigurable Systems Classification
  - Heterogeneous and Irregular Coarse-Grained Reconfigurable Accelerators
- Power Management
  - Issues and Strategies
  - Low-Power Coarse-Grained Reconfigurable Accelerators
- An FFT Example

Numbers: opportunity or issue?

> 7 billion

20 MWh/year

=

1,800 kg oil

http://www.gartner.com/newsroom/id/3598917

Numbers: opportunity or issue?

> 7 billion

20 MWh/year

=

1,800 kg oil

Designed by Freepik

> 1 billion smartphones

http://www.gartner.com/newsroom/id/3598917

Numbers: opportunity or issue?

> 7 billion

20 MWh/year

=

1,800 kg oil

Designed by Freepik

> 1 billion smartphones

8.4 billion connected things in 2017 (+31% wrt 2016)

20.4 billion by 2020

http://www.gartner.com/newsroom/id/3598917

Some examples …

Connectivity and real-time situation awareness are nowadays common in different scenarios.

Some examples …

Connectivity and real-time situation awareness are nowadays common in different scenarios.

SMART HEALTH:

distributed healthcare assistance to improve quality of life and active and healthy ageing, functionalities can be changed according to the daily tasks.

Some examples …

Connectivity and real-time situation awareness are nowadays common in different scenarios.

SMART HEALTH:
distributed healthcare assistance to improve quality of life and active and healthy ageing, functionalities can be changed according to the daily tasks.

SMART-SOCIETY:
increased building efficiency and comfort, i.e. lightning/air quality management can be adjusted to the room status.

Some examples …

Connectivity and real-time situation awareness are nowadays common in different scenarios.

SMART HEALTH:
distributed healthcare assistance to improve quality of life and active and healthy ageing, functionalities can be changed according to the daily tasks.

SMART-SOCIETY:
increased building efficiency and comfort, i.e. lightning/air quality management can be adjusted to the room status.

SMART-TRANSPORTATION:
autonomous electric vehicle, improved driver assistance and care. Path towards destinations may vary, even diverging from the optimal one, according to user preferences.

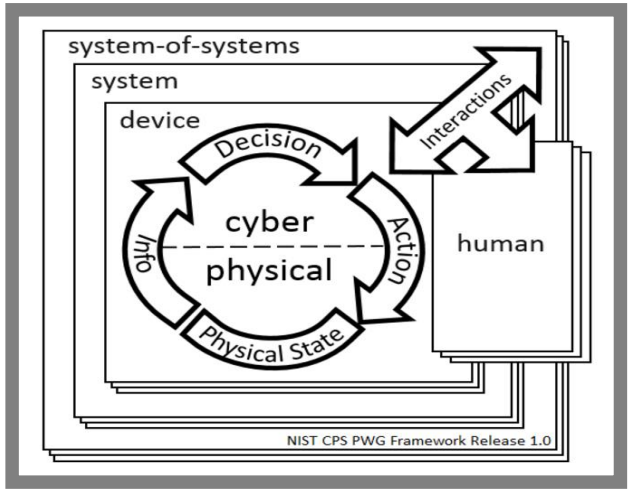
Reconfiguration: Recipe for Compromises

Modern digital devices (*real-time* and *ad-hoc*) are pervasive (*98%* of computers are *embedded*) and interconnected.

Reconfiguration: Recipe for Compromises

Modern digital devices (*real-time* and *ad-hoc*) are pervasive (*98%* of computers are *embedded*) and interconnected.

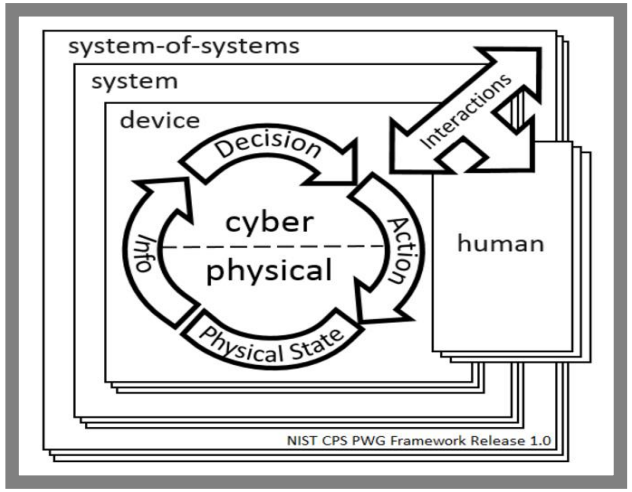
They may also present *sensing* and actuating *capabilities*, leading to the concept of CPS.



Reconfiguration: Recipe for Compromises

Modern digital devices (*real-time* and *ad-hoc*) are pervasive (*98%* of computers are *embedded*) and interconnected.

They may also present *sensing* and actuating *capabilities*, leading to the concept of CPS.

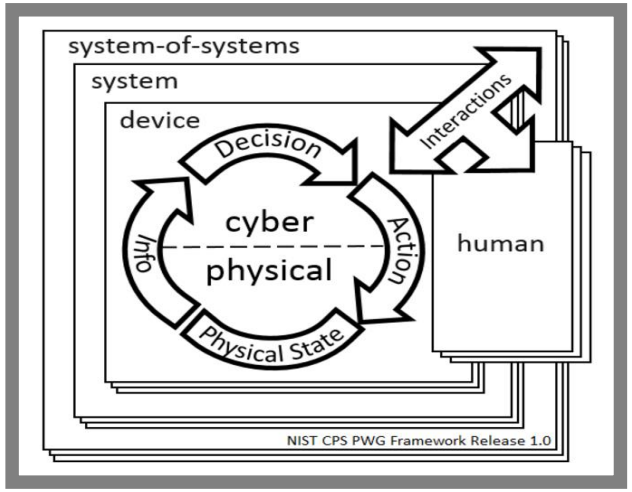


	Safety	Security	Certif.	Distrib.	HMI	Seamless	MPSoC	Energy
Automotive	x	x	x	x	x	x	x	
Aerospace	x	x	x	x	x		x	x
Healthcare	x	x	x	x	x	x	x	x
Consumer					x	x	x	

Reconfiguration: Recipe for Compromises

Modern digital devices (*real-time* and *ad-hoc*) are pervasive (*98%* of computers are *embedded*) and interconnected.

They may also present *sensing* and actuating *capabilities*, leading to the concept of CPS.



	Safety	Security	Certif.	Distrib.	HMI	Seamless	MPSoC	Energy
Automotive	x	x	x	x	x	x	x	
Aerospace	x	x	x	x	x		x	x
Healthcare	x	x	x	x	x	x	x	x
Consumer					x	x	x	

Reconfiguration may allow to optimally implement complex/demanding systems, managing ***numerous/conflicting requirements*** and a ***variety of functionalities***.

Triggers for Adaptation

Triggers for Adaptation

ENVIRONMENTAL AWARENESS:

Influence of the environment on the system, i.e. daylight vs. nocturnal, radiation level changes, etc.

Sensors are needed to interact with the environment and capture conditions variations.

Triggers for Adaptation

ENVIRONMENTAL AWARENESS:

Influence of the environment on the system, i.e. daylight vs. nocturnal, radiation level changes, etc.

Sensors are needed to interact with the environment and capture conditions variations.

USER-COMMANDED:

System-User interaction, i.e. user preferences, etc.

Proper human-machine interfaces are needed to enable interaction and capture commands.

Triggers for Adaptation

ENVIRONMENTAL AWARENESS:

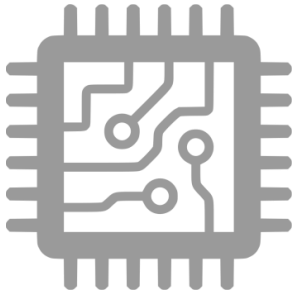
Influence of the environment on the system, i.e. daylight vs. nocturnal, radiation level changes, etc.

Sensors are needed to interact with the environment and capture conditions variations.

USER-COMMANDED:

System-User interaction, i.e. user preferences, etc.

Proper human-machine interfaces are needed to enable interaction and capture commands.



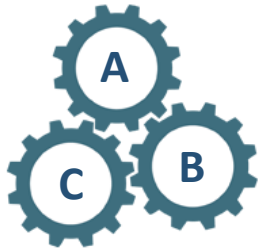
SELF-AWARENESS:

The internal status of the system varies while operating and may lead to reconfiguration needs, i.e. chip temperature variation, low battery.

Status monitors are needed to capture the status of the system.

Types of Adaptation

Types of Adaptation

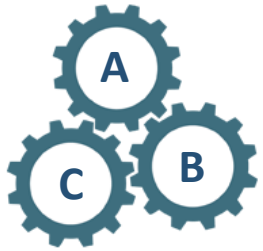


FUNCTIONALITY-ORIENTED:

To adapt functionality because the CPS mission changes, or the data being processed changes and adaptation is required.

It may be parametric (a constant changes) or fully functional (algorithm changes).

Types of Adaptation



FUNCTIONALITY-ORIENTED:

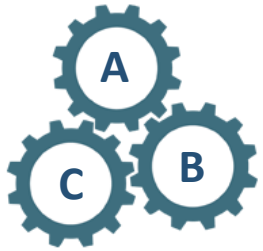
To adapt functionality because the CPS mission changes, or the data being processed changes and adaptation is required.

It may be parametric (a constant changes) or fully functional (algorithm changes).

NON-FUNCTIONAL REQUIREMENTS-ORIENTED:

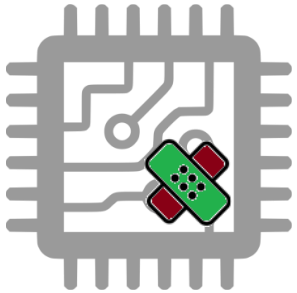
Functionality is fixed, but system requires adaptation to accommodate to changing requirements, i.e. execution time or energy consumption.

Types of Adaptation



FUNCTIONALITY-ORIENTED:
To adapt functionality because the CPS mission changes, or the data being processed changes and adaptation is required.
It may be parametric (a constant changes) or fully functional (algorithm changes).

NON-FUNCTIONAL REQUIREMENTS-ORIENTED:
Functionality is fixed, but system requires adaptation to accommodate to changing requirements, i.e. execution time or energy consumption.



REPAIR-ORIENTED:
For safety and reliability purposes, adaptation may be used in case of faults. Adaptation may add self-healing or self-repair features. e.g.: HW task migration for permanent faults, or scrubbing (continuous fault verification) and repair.

Overview

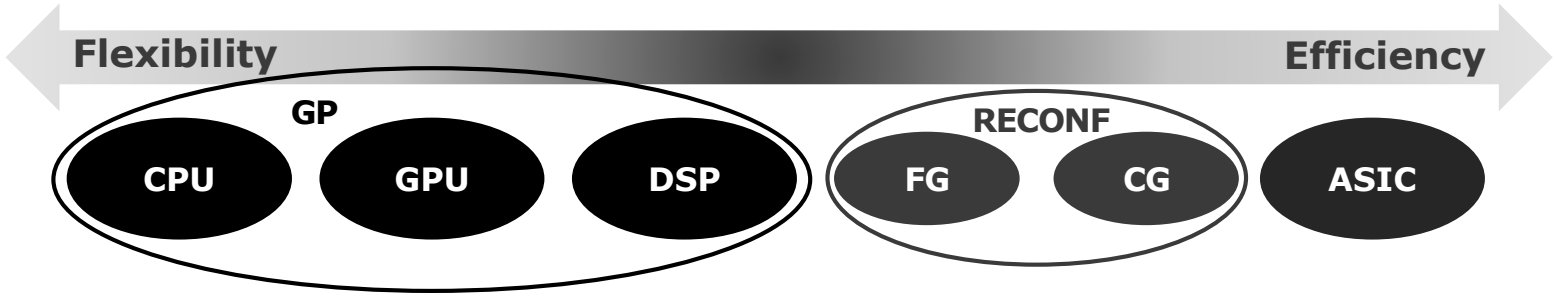
- Motivations
  - What we need adaptation for
  - Triggers and Types
- Coarse-Grained Reconfigurable Systems
  - Computing Spectrum and Reconfigurable Systems Classification
  - Heterogeneous and Irregular Coarse-Grained Reconfigurable Accelerators
- Power Management
  - Issues and Strategies
  - Low-Power Coarse-Grained Reconfigurable Accelerators
- An FFT Example

Computing Spectrum

Computing Spectrum

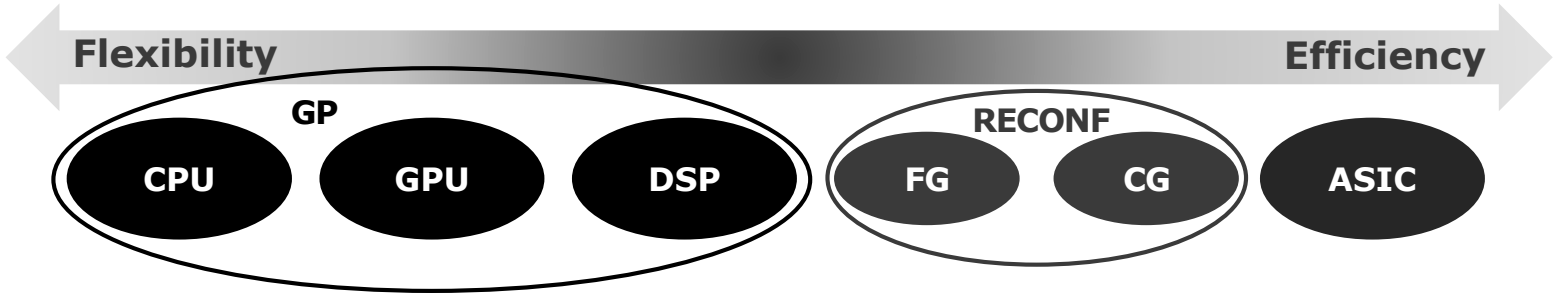
Computing Spectrum

Computing Spectrum



Reconfigurable computing provides a **trade-off** between execution **efficiency** typical of ASICs and **flexibility** mainly exhibited by GP devices.

Computing Spectrum



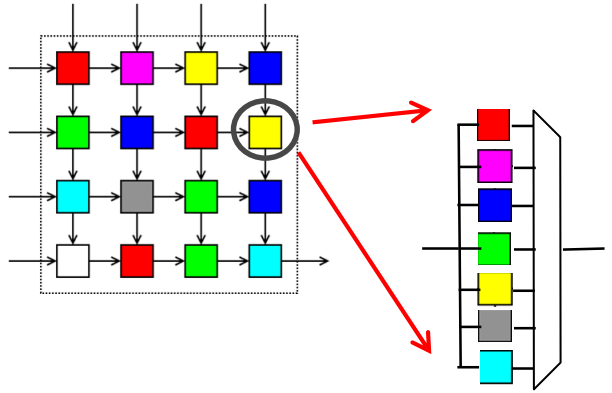
Reconfigurable computing provides a **trade-off** between execution **efficiency** typical of ASICs and **flexibility** mainly exhibited by GP devices.

	Fine-Grained (FG) bit-level	Coarse-Grained (CG) word-level
flexibility	☺	😐
speed	😐	☺
memory	☹	😐

Virtual vs. Dynamic & Partial

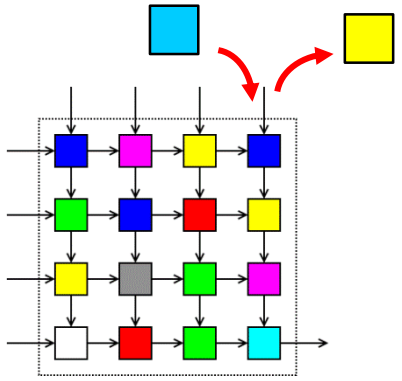
VRC → Virtual Reconfigurable Circuits

- High reconfiguration speed
- Lower operation speed (mux and size)
- Higher Area Overhead
- Technology independent (ASIC or FPGA)



DPR → Dynamic and Partial Reconfiguration

- Lower reconfiguration speeds
- Better operation speed (no mux/less logic)
- Better Resource Utilization (no dark logic)
- Higher Flexibility and Scalability
- Technology dependent (FPGA)



Dataflow Model of Computation

COMPUTING PARADIGM:

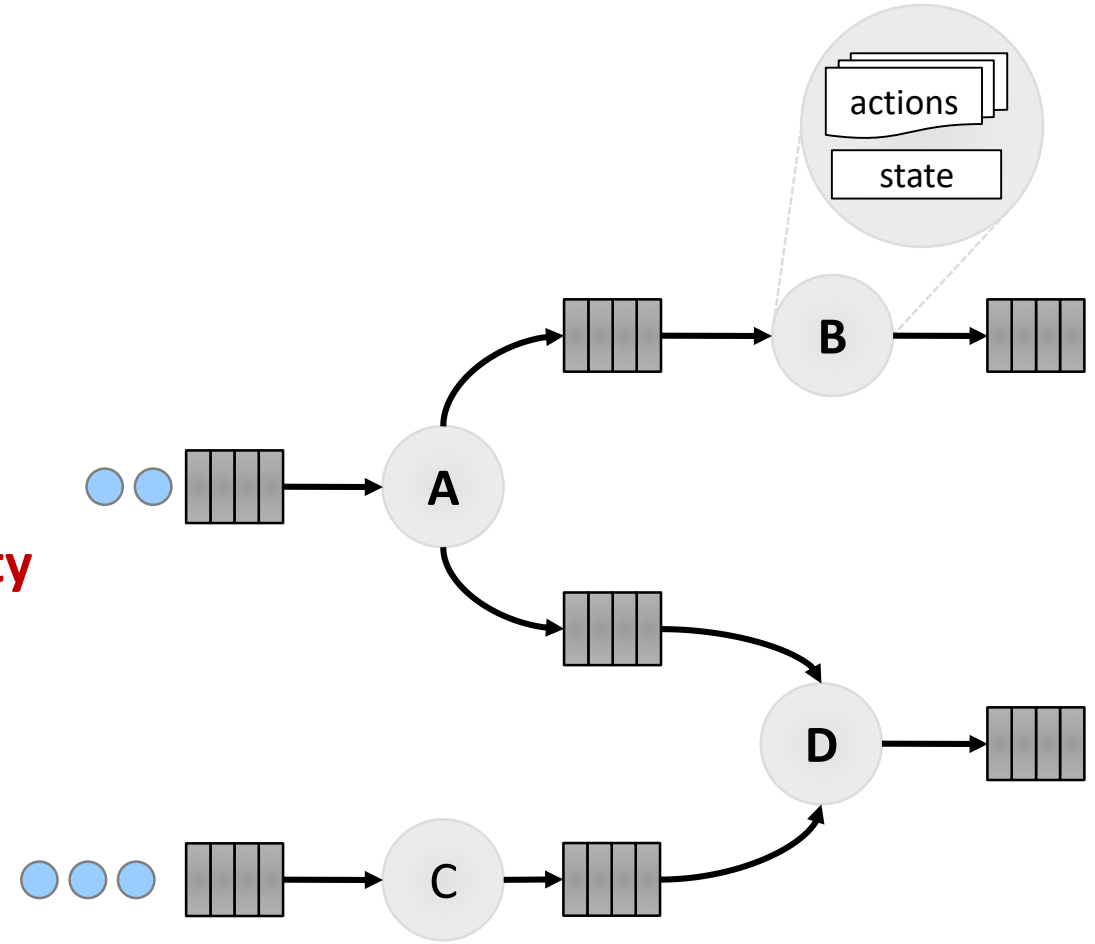
- directed flow graph of **actors** (functional units)
- communication with **tokens** (packets of data) exchange through dedicated channels

PECULIARITIES:

- explicit intrinsic application **parallelism**
- modularity favours model **re-usability/adaptivity**

EXTERNAL INTERFACE:

- I/O **ports number**
- I/O **ports depth**
- I/O ports tokens **burst**



Dataflow Model of Computation

COMPUTING PARADIGM:

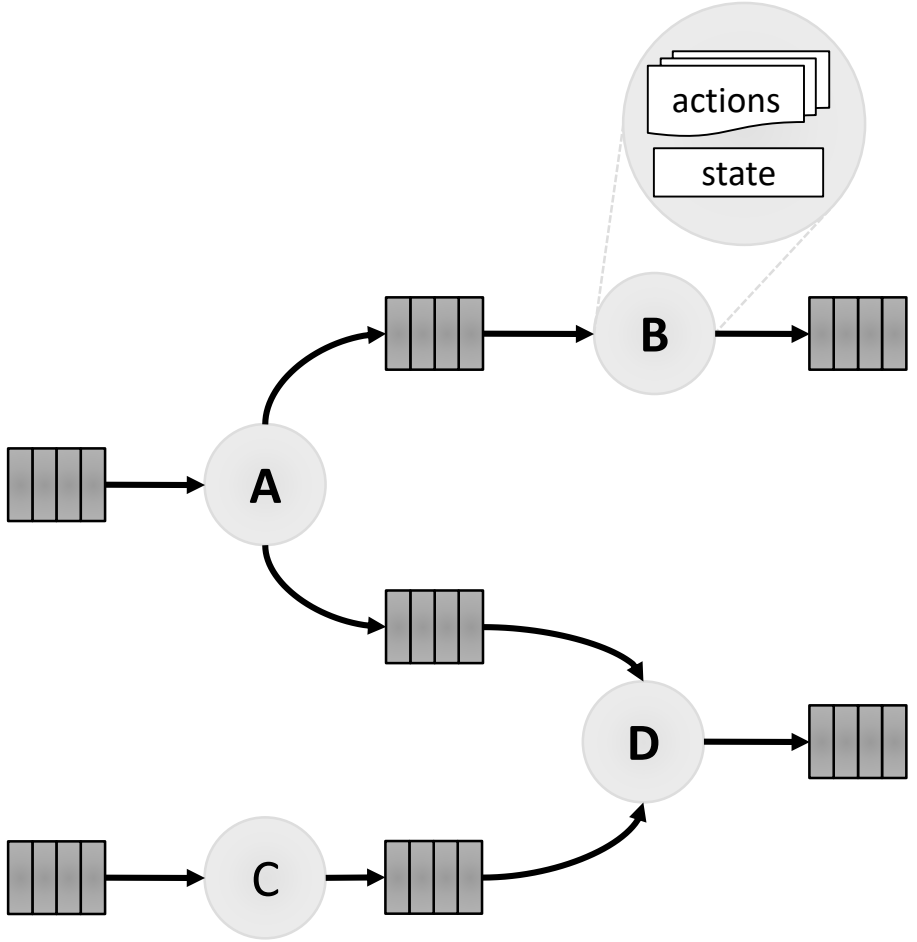
- directed flow graph of **actors** (functional units)
- communication with **tokens** (packets of data) exchange through dedicated channels

PECULIARITIES:

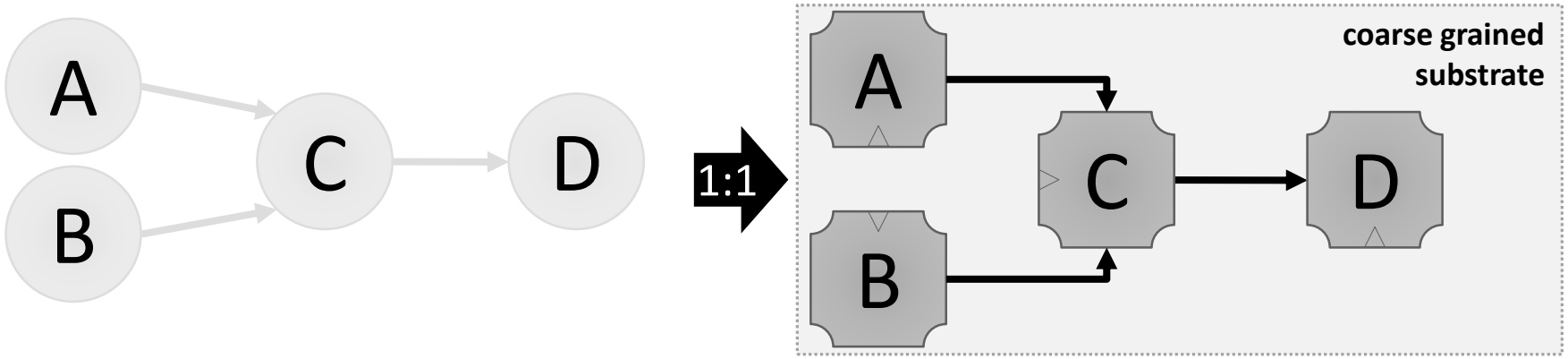
- explicit intrinsic application **parallelism**
- modularity favours model **re-usability/adaptivity**

EXTERNAL INTERFACE:

- I/O **ports number**
- I/O **ports depth**
- I/O ports tokens **burst**

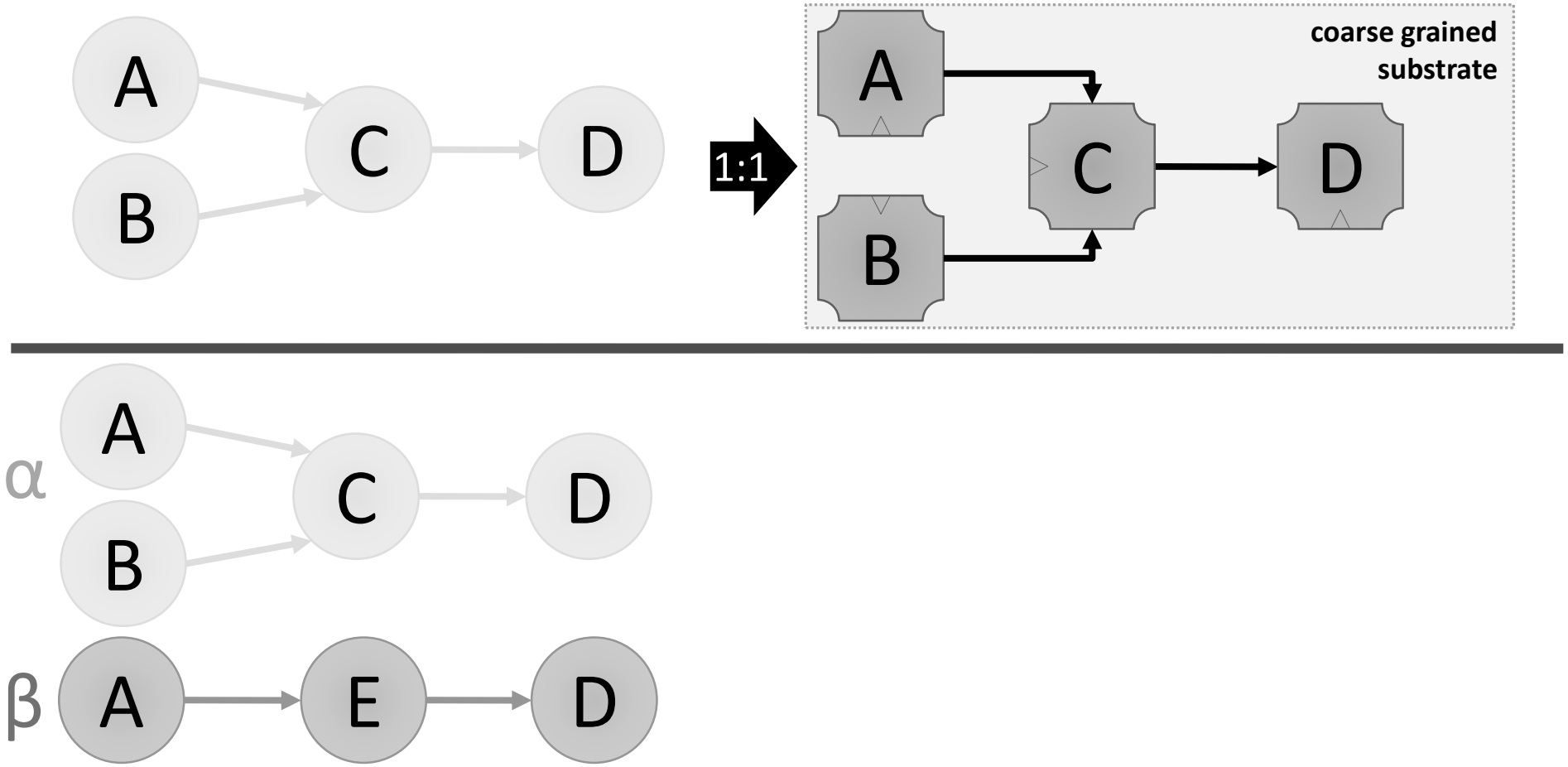


Heterogeneous and Irregular: approach



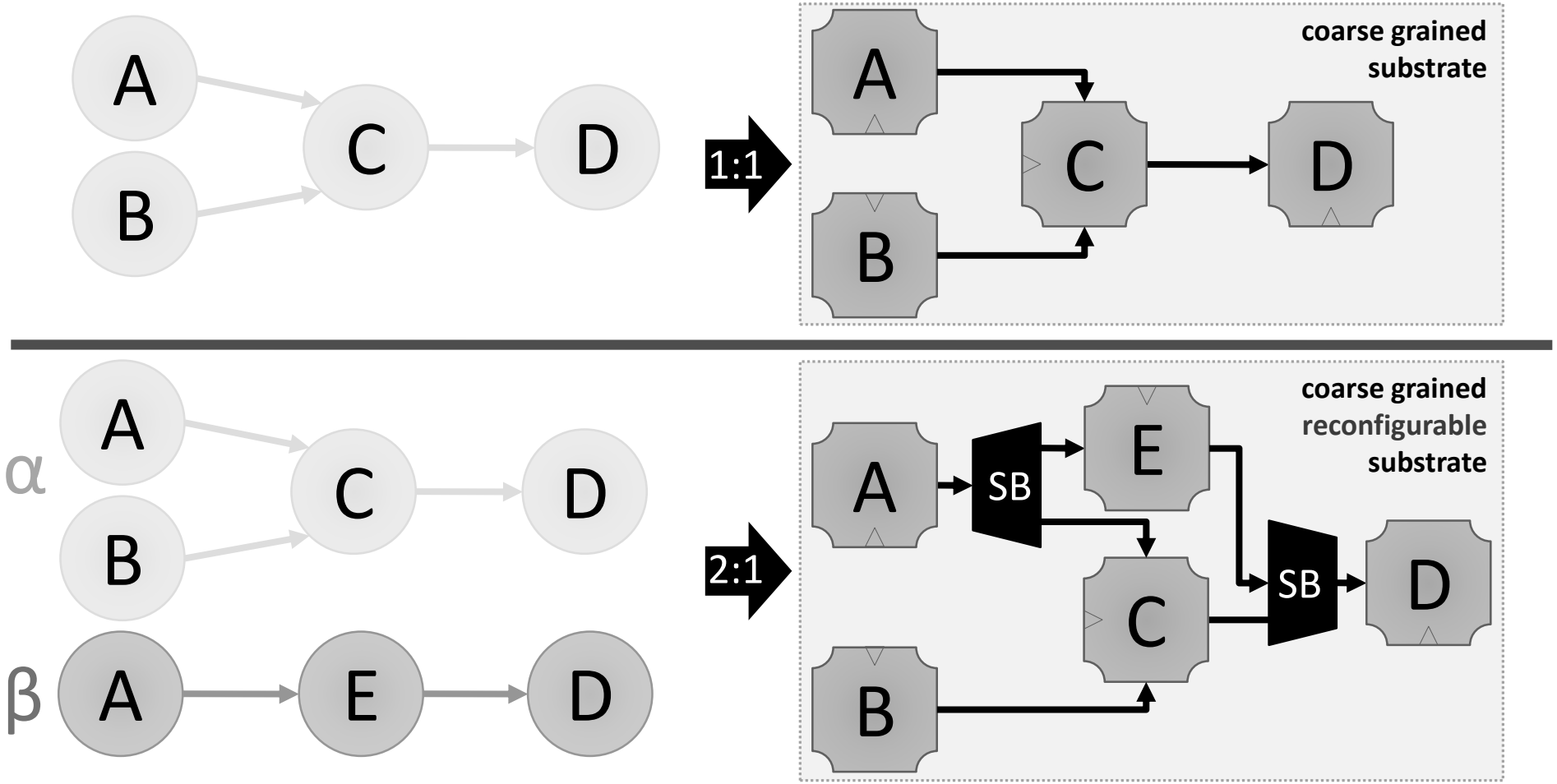
http://sites.unica.it/rpct/

Heterogeneous and Irregular: approach



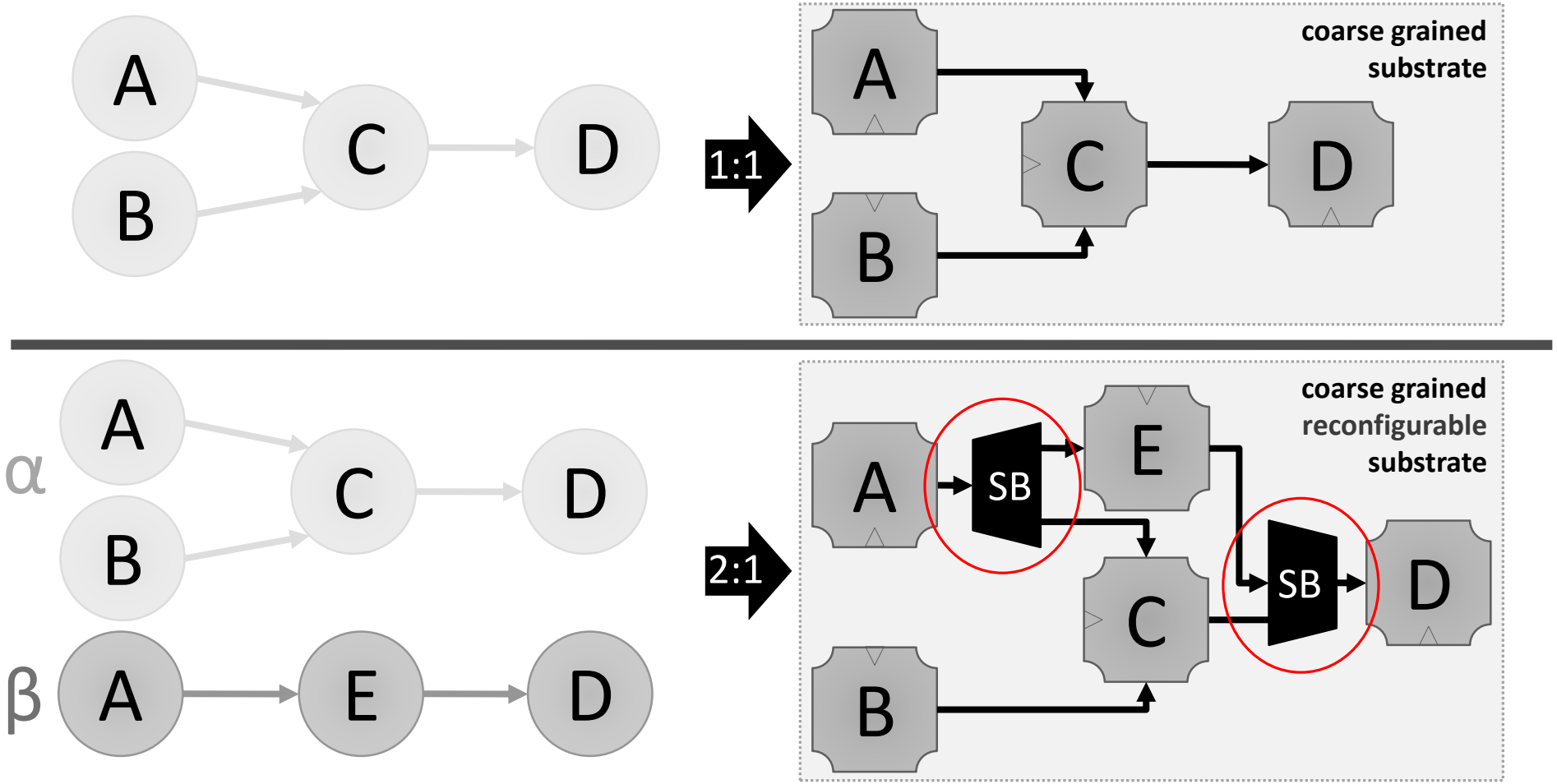
http://sites.unica.it/rpct/

Heterogeneous and Irregular: approach



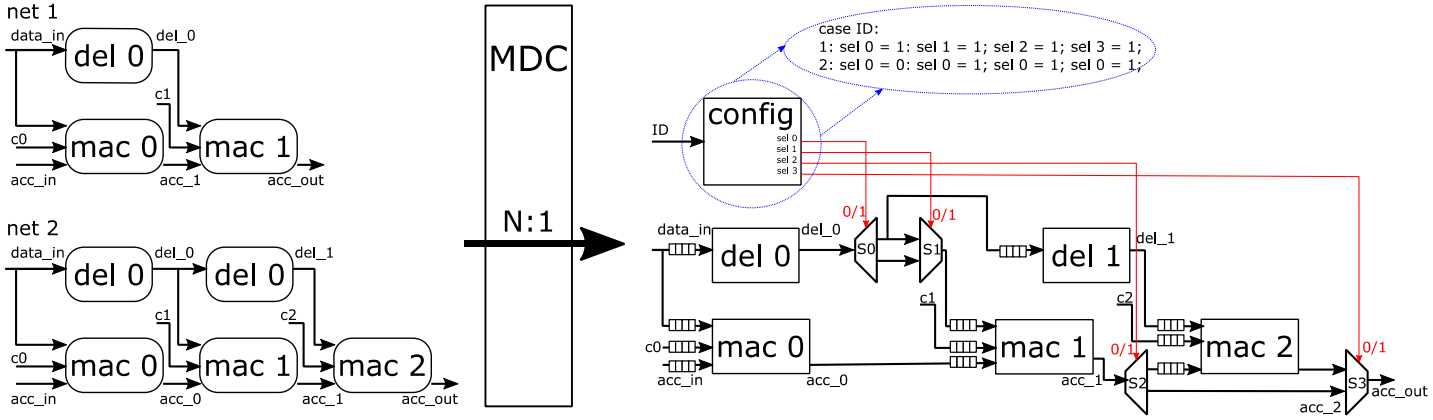
http://sites.unica.it/rpct/

Heterogeneous and Irregular: approach



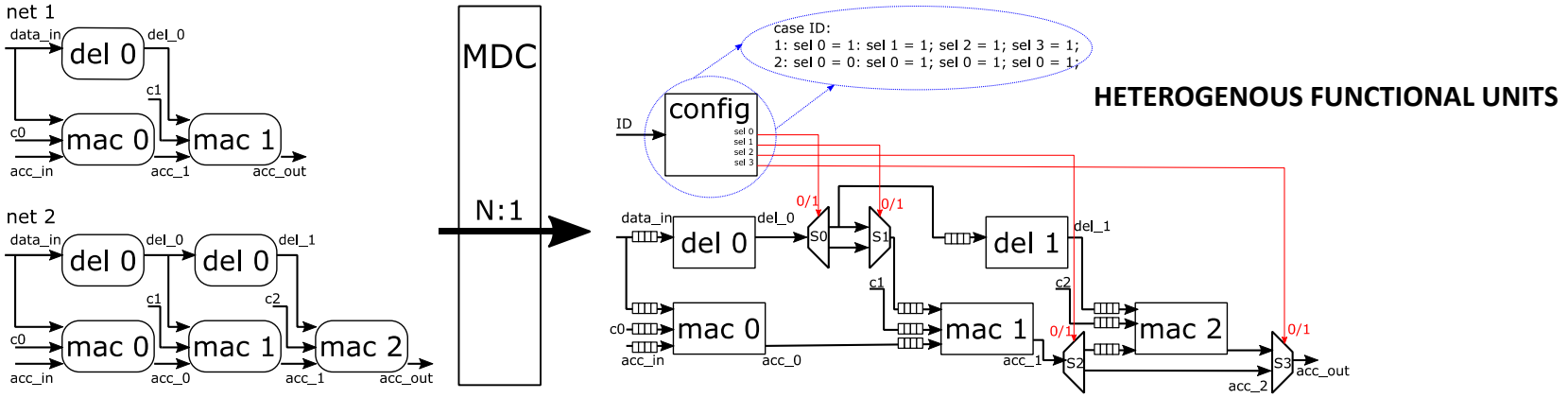
http://sites.unica.it/rpct/

Heterogeneous and Irregular: MDC framework



Multi-Dataflow Composer (MDC) tool: Dataflow 2 HW tool

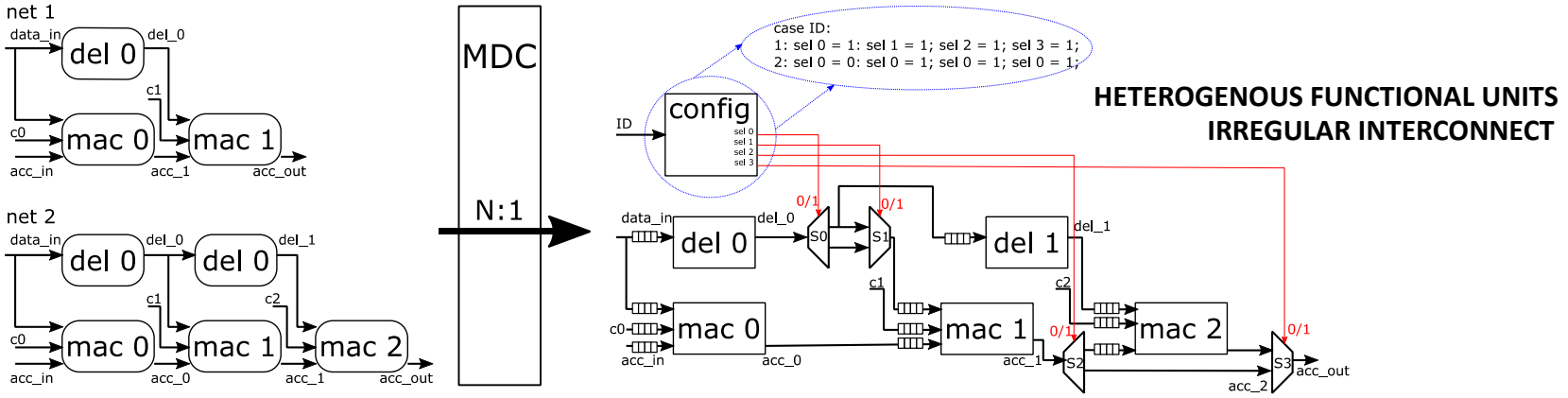
Heterogeneous and Irregular: MDC framework



Multi-Dataflow Composer (MDC) tool: Dataflow 2 HW tool

- Given N input dataflow specification, it generates the HDL Coarse-Grain (CG) reconfigurable substrate

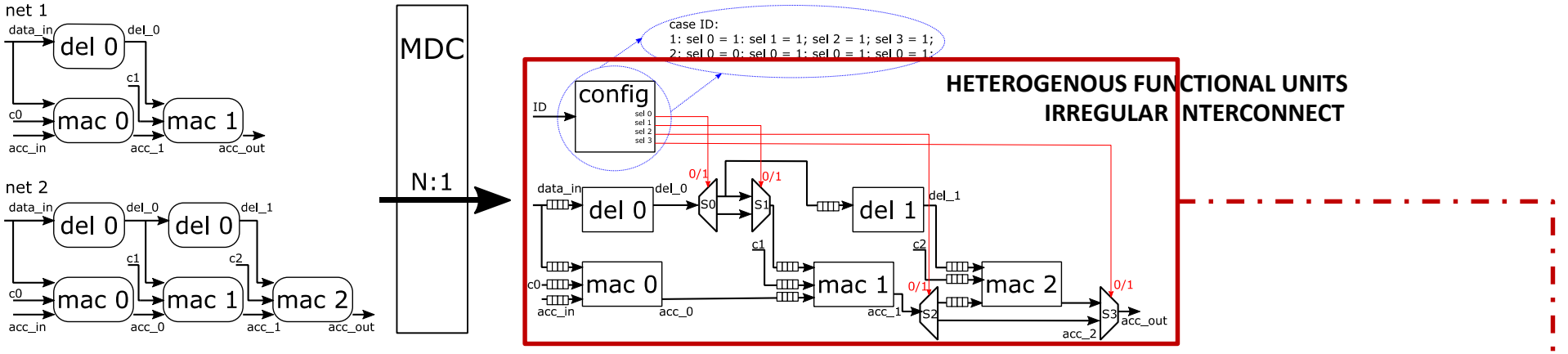
Heterogeneous and Irregular: MDC framework



Multi-Dataflow Composer (MDC) tool: Dataflow 2 HW tool

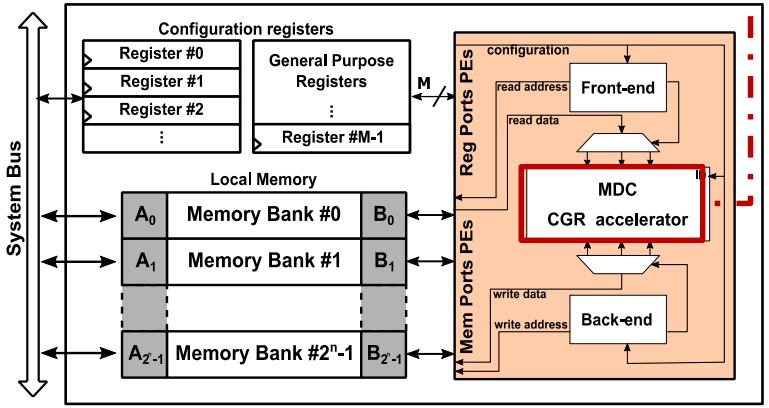
- Given N input dataflow specification, it generates the HDL Coarse-Grain (CG) reconfigurable substrate
- Handles programmability, by defining switching and configuration logic

Heterogeneous and Irregular: MDC framework



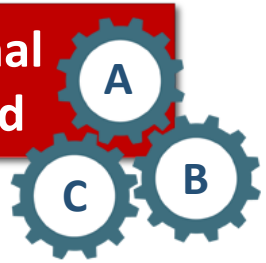
Multi-Dataflow Composer (MDC) tool: Dataflow 2 HW tool

- Given N input dataflow specification, it generates the HDL Coarse-Grain (CG) reconfigurable substrate
- Handles programmability, by defining switching and configuration logic
- Deploy Xilinx-compliant IP blocks, plus their drivers



MDC-based Reconfiguration: Adaptation Types

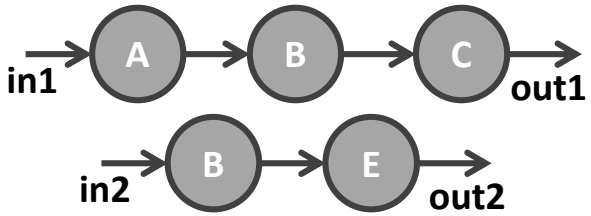
Functional Oriented

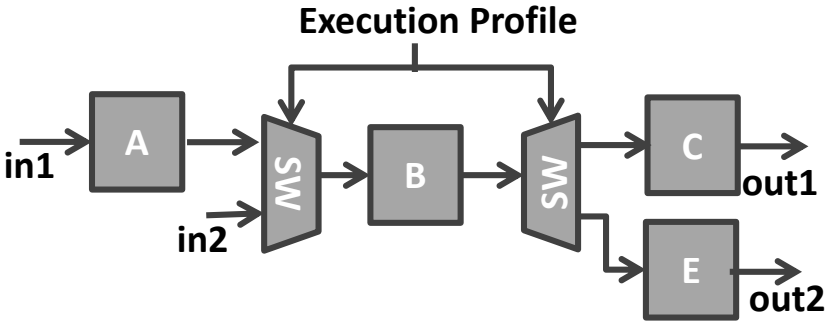


Non-Functional Oriented

MDC-based Reconfiguration: Adaptation Types

Functional Oriented

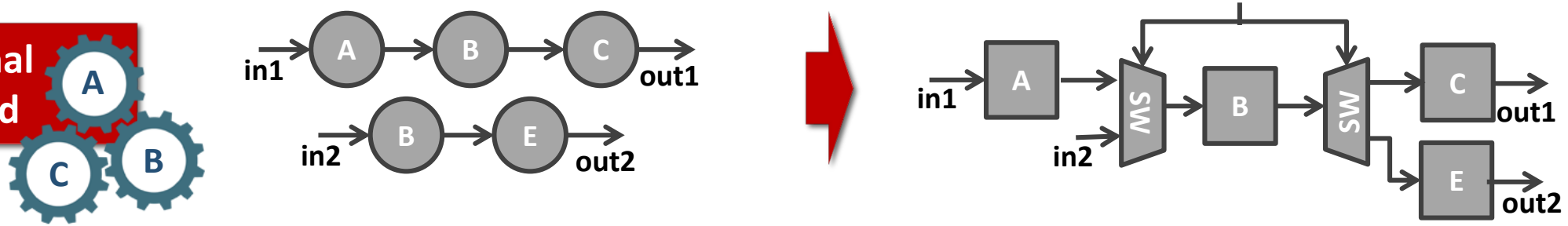




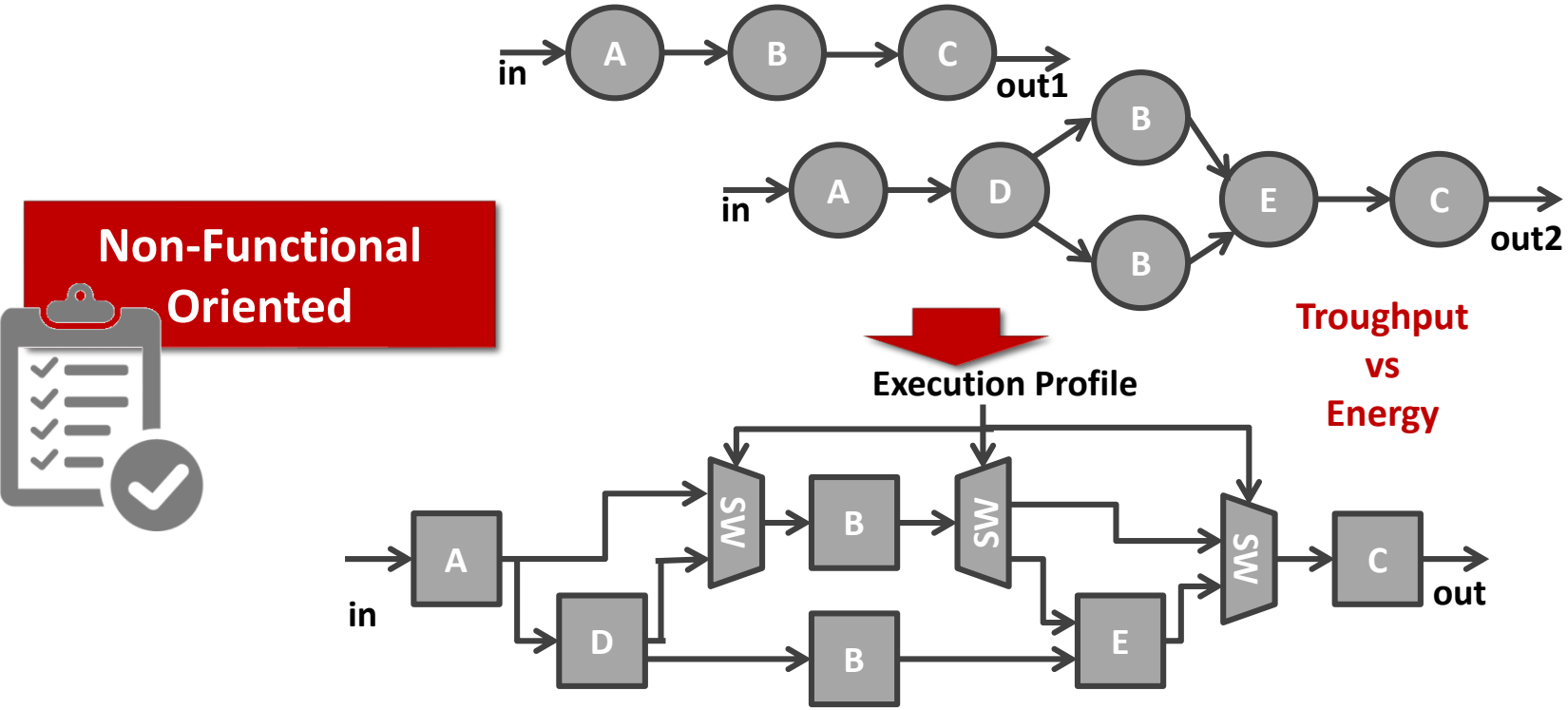
Non-Functional Oriented

MDC-based Reconfiguration: Adaptation Types

Functional Oriented

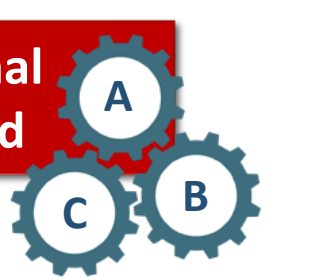


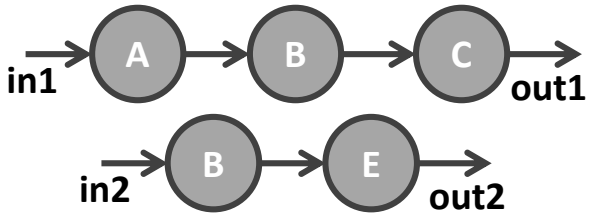
Non-Functional Oriented

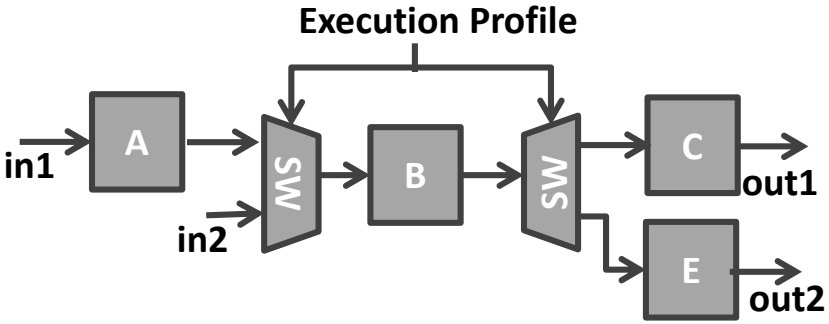


MDC-based Reconfiguration: Adaptation Types

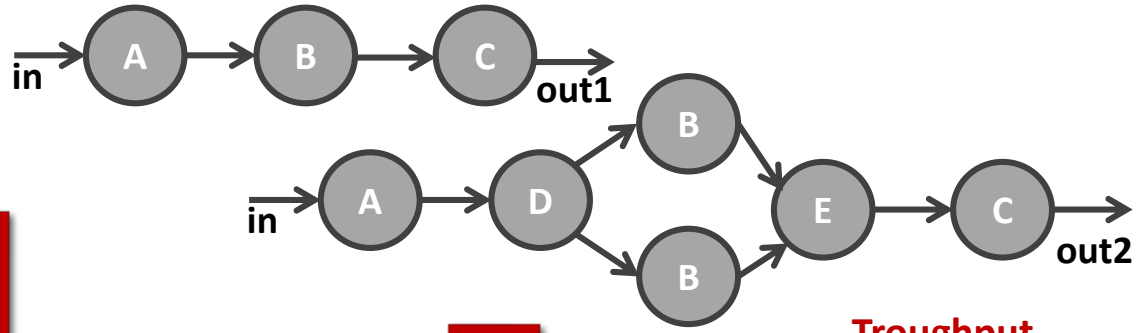
Functional Oriented

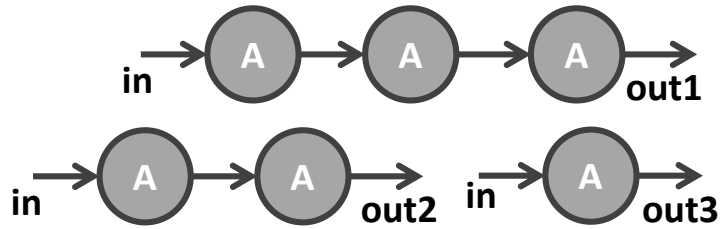




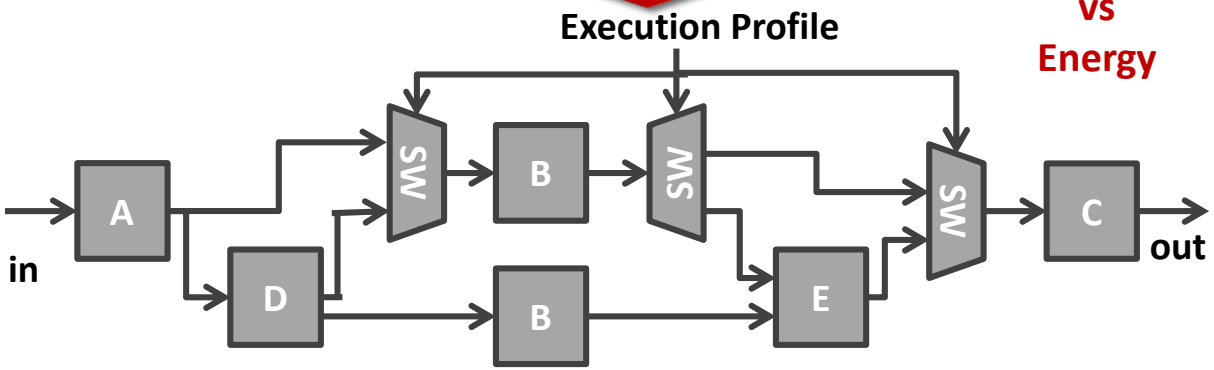


Non-Functional Oriented

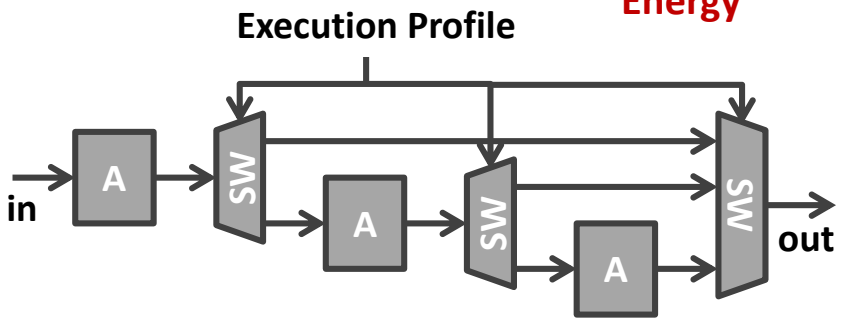




Troughput vs Energy



QoS Vs. Energy



Overview

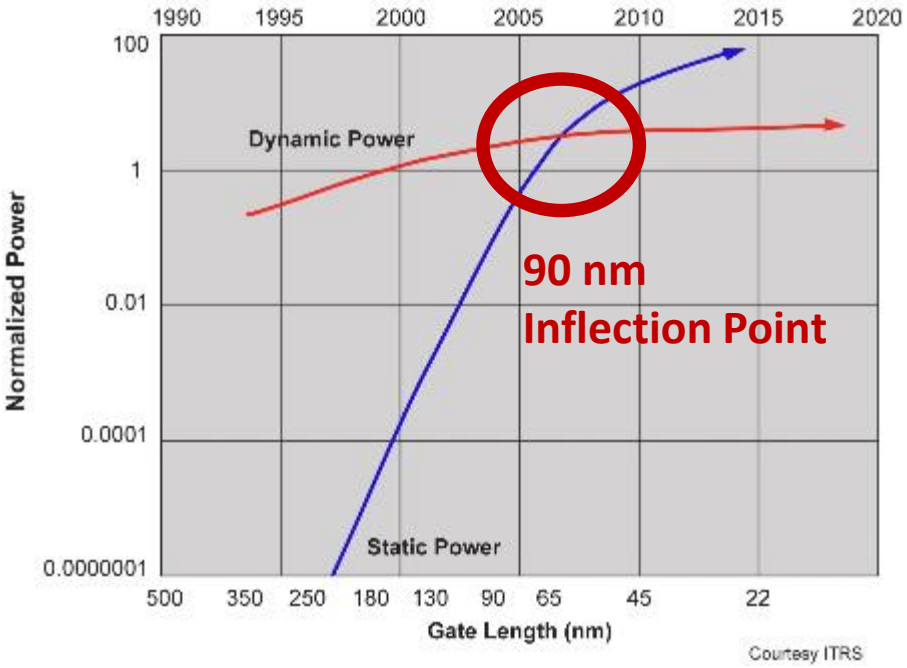
- Motivations
  - What we need adaptation for
  - Triggers and Types
- Coarse-Grained Reconfigurable Systems
  - Computing Spectrum and Reconfigurable Systems Classification
  - Heterogeneous and Irregular Coarse-Grained Reconfigurable Accelerators
- Power Management
  - Issues and Strategies
  - Low-Power Coarse-Grained Reconfigurable Accelerators
- An FFT Example

The Power Issue

Power consumption =
Dynamic power + Static power

Dynamic: activity dependent

- *Short-circuit*: when the output line of a transistor is switching, there is a period of time when both the PMOS and the NMOS transistors are on (I·V·f)
- *Switching power*: due to the charging and discharging of the load capacitance when logic transitions occur (determined by the formula C·V2·f).



Static: not activity dependent, but due to leakage currents.

- Do not depend on switching and operating frequency.
- As transistors get smaller, channel lengths become shorter and leakage currents increase.

The Power Issue: Textbook Techniques

Dynamic	***Static***
Clock gating	Multi-threshold dev.
Variable frequency	Power gating
Voltage islands	Back (substrate) bias
Variable power supply	Multi-oxide devices
Multi power supply	SOI CMOS
DVFS	

- ***DESIGN***: Multi Vt, Clock gating, Power gating, Multi Vdd, DVFS.
- ***PROCESS***: Multi Vt, PD SOI, FD SOI, FinFet, Body Bias, Multi oxide devices, Minimize capacitance by custom design.

MANAGEABLE AT SYSTEM LEVEL

- ***ARCHITECTURE***: power-constrained DSE, hw customization and IP specific techniques, parallelism, mapping.

INTRINSICALLY SYSTEM LEVEL

Clock-Based Techniques

Reduce frequency whenever you can.
Stop the clock when the component is not active.

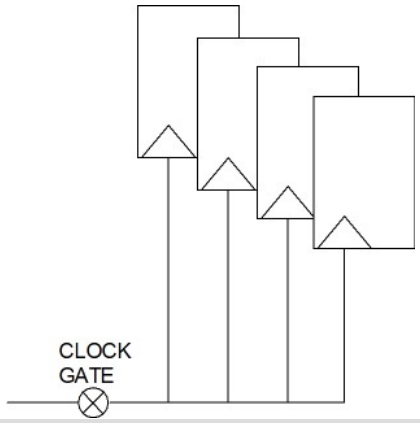
- Power consumed by flip-flops.
- Power consumed by combinatorial logic driven by registers.
- Power consumed by the clock buffer tree.

Clock-Based Techniques

Reduce frequency whenever you can.
Stop the clock when the component is not active.

- Power consumed by flip-flops.
- Power consumed by combinatorial logic driven by registers.
- Power consumed by the clock buffer tree.

Fine-Grained

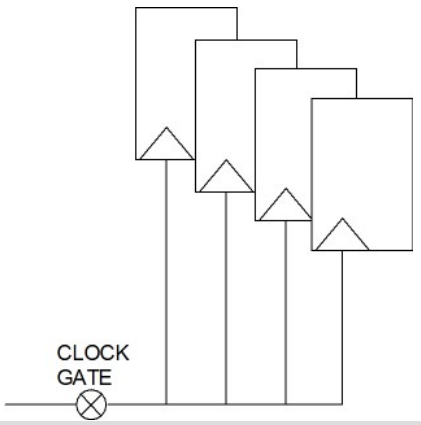


Clock-Based Techniques

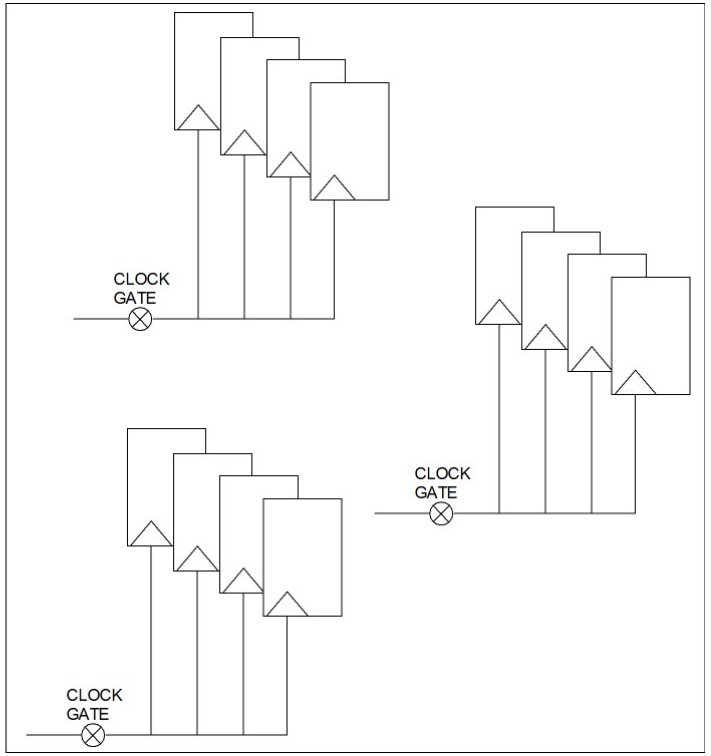
Reduce frequency whenever you can.
Stop the clock when the component is not active.

- Power consumed by flip-flops.
- Power consumed by combinatorial logic driven by registers.
- Power consumed by the clock buffer tree.

Fine-Grained



Coarse-Grained



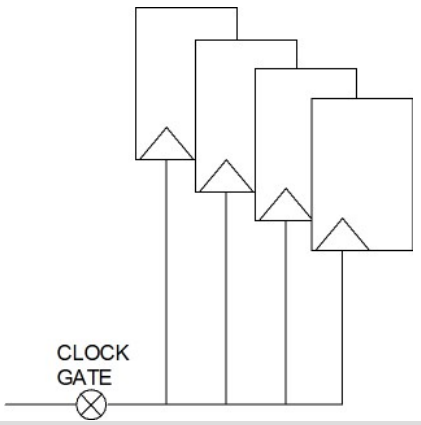
Clock-Based Techniques

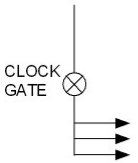
Reduce frequency whenever you can.
Stop the clock when the component is not active.

- Power consumed by flip-flops.
- Power consumed by combinatorial logic driven by registers.
- Power consumed by the clock buffer tree.

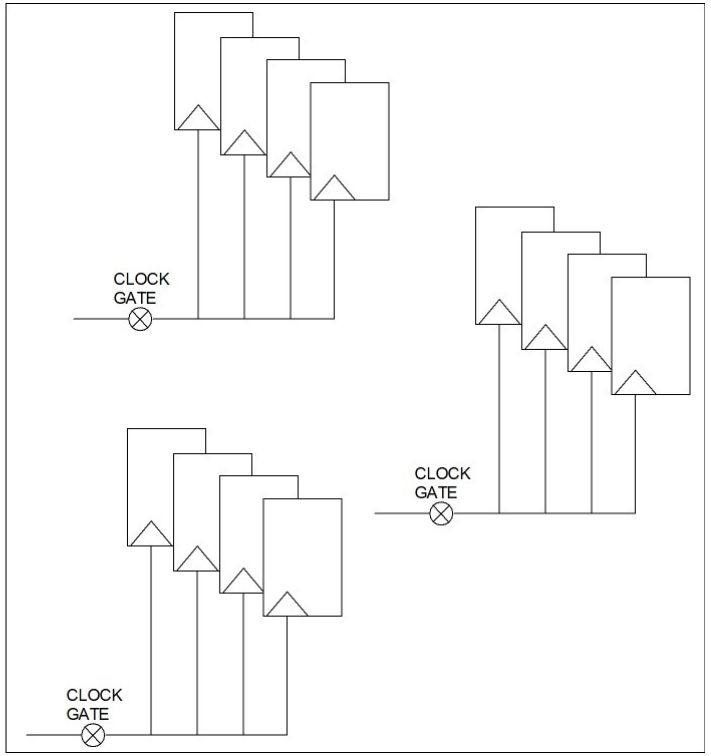
50% less dynamic power

Fine-Grained





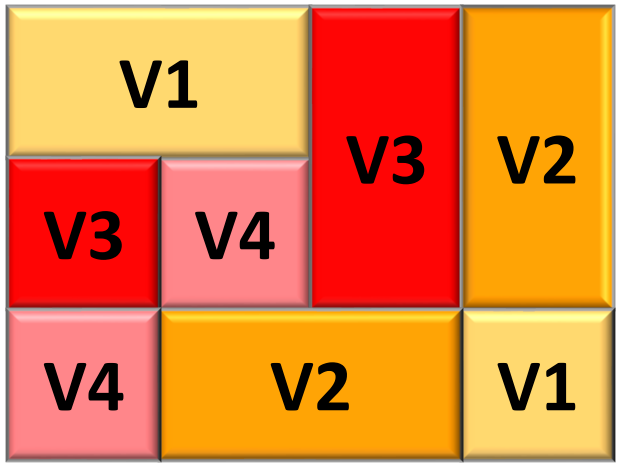
Coarse-Grained



Power-Based Techniques

Reduce the voltage level of a power island whenever you can.
Switch it off when it is not active.

- Power-aware partitioning.

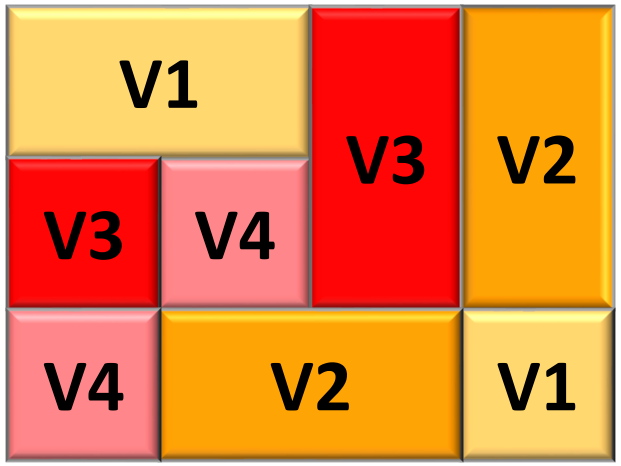


Static Voltage Scaling (SVS)

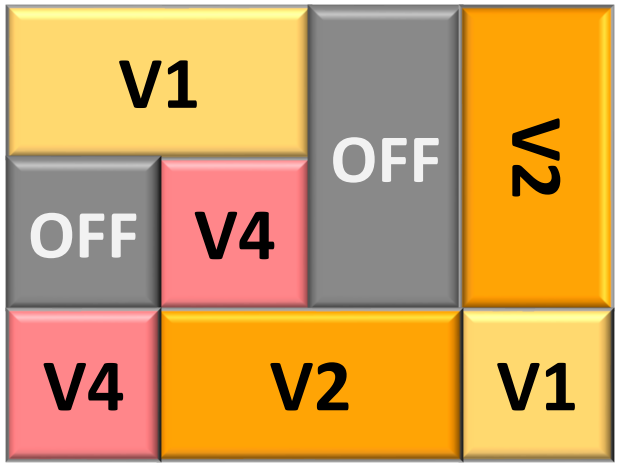
Power-Based Techniques

Reduce the voltage level of a power island whenever you can.
Switch it off when it is not active.

- Power-aware partitioning.
- Switching-off power island brings local leakage to zero.



Static Voltage Scaling (SVS)

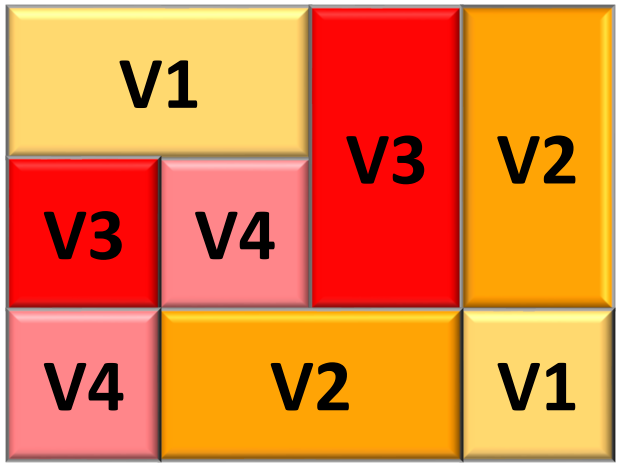


SVS with Power Shut Off (PSO)

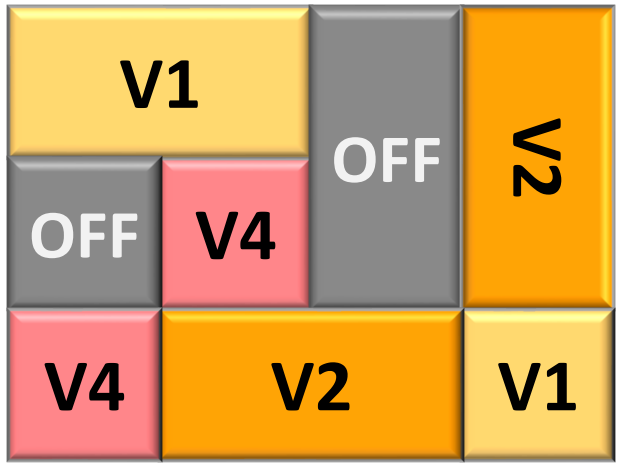
Power-Based Techniques

Reduce the voltage level of a power island whenever you can.
Switch it off when it is not active.

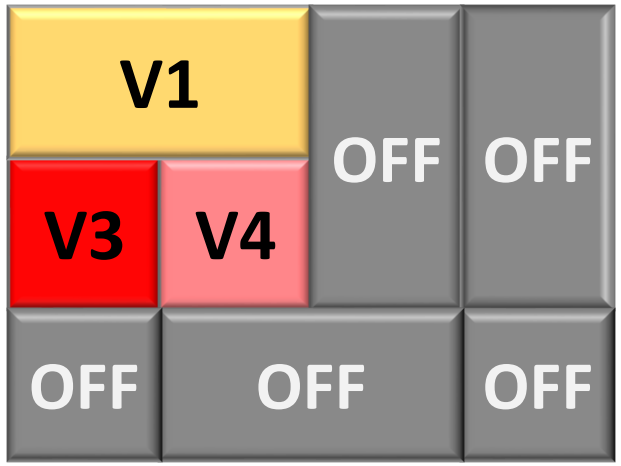
- Power-aware partitioning.
- Switching-off power island brings local leakage to zero.
- Modes of operation -> power down all the idle chip regions.



Static Voltage Scaling (SVS)



SVS with Power Shut Off (PSO)

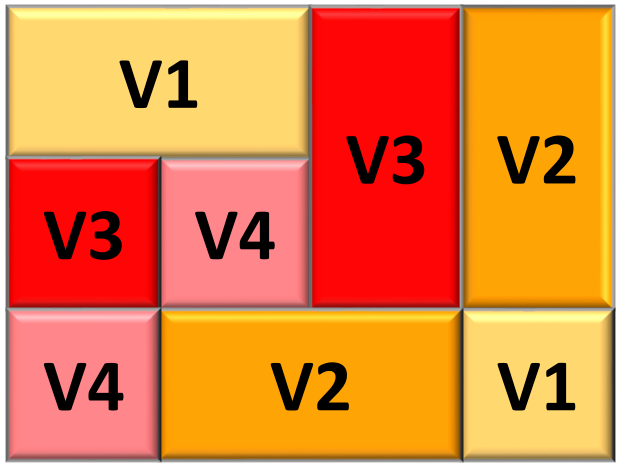


SVS with PSO and Power Modes

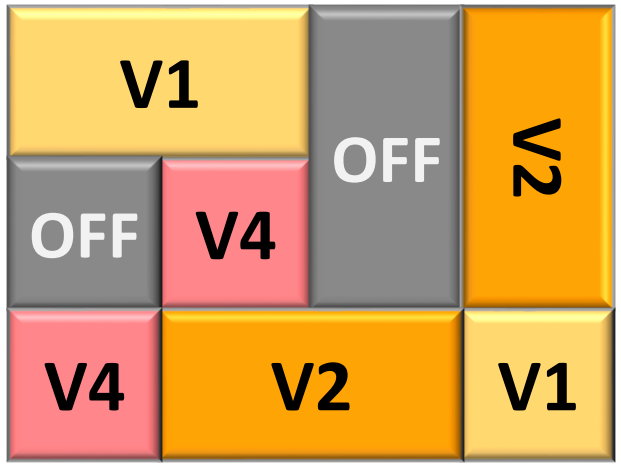
Power-Based Techniques

Reduce the voltage level of a power island whenever you can.
Switch it off when it is not active.

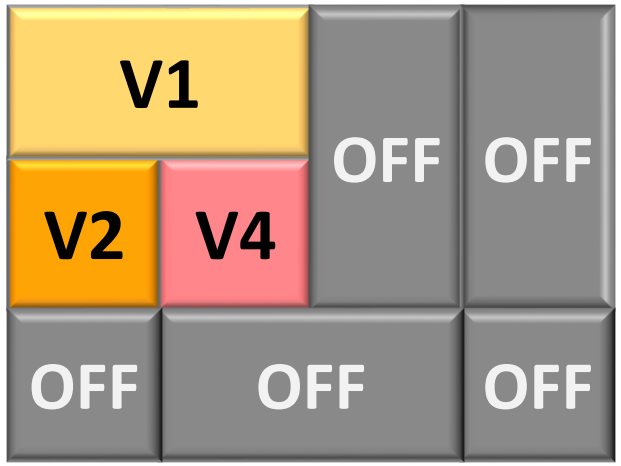
- Power-aware partitioning.
- Switching-off power island brings local leakage to zero.
- Modes of operation -> power down all the idle chip regions.



Static Voltage Scaling (SVS)



SVS with Power Shut Off (PSO)



SVS with PSO and Power Modes

Clock-/Power-Based Techniques: Overhead

Switch off the clock, applicable to both ASIC and FPGA.

- @ASIC: simple and gates.
- @FPGA: dedicated vendor specific cells.

Clock-/Power-Based Techniques: Overhead

Switch off the clock, applicable to both ASIC and FPGA.

- @ASIC: simple and gates.
- @FPGA: dedicated vendor specific cells.

Switch off the power supply, ASIC only (partially FPGA).

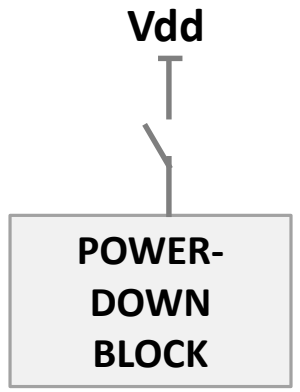
Clock-/Power-Based Techniques: Overhead

Switch off the clock, applicable to both ASIC and FPGA.

- @ASIC: simple and gates.
- @FPGA: dedicated vendor specific cells.

Switch off the power supply, ASIC only (partially FPGA).

- *Sleep transistors*: to switch on and off power supply.



Power Switch-Off Cell

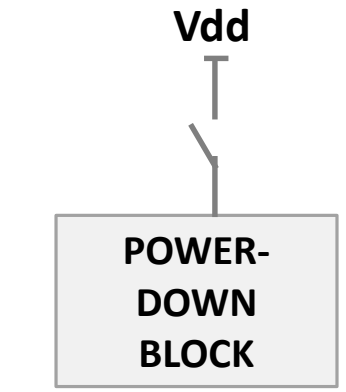
Clock-/Power-Based Techniques: Overhead

Switch off the clock, applicable to both ASIC and FPGA.

- @ASIC: simple and gates.
- @FPGA: dedicated vendor specific cells.

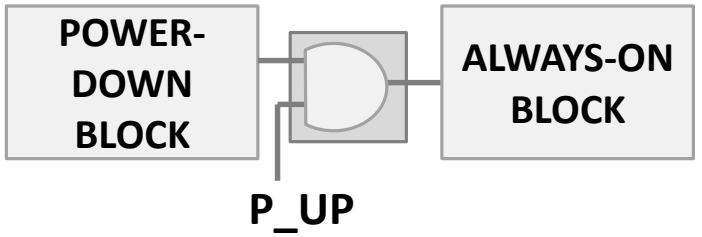
Switch off the power supply, ASIC only (partially FPGA).

- *Sleep transistors*: to switch on and off power supply.
- *Isolation logic*: to avoid the transmission of spurious signals from gated regions to normally-on cells.



Power Switch-Off Cell

Isolation Cell



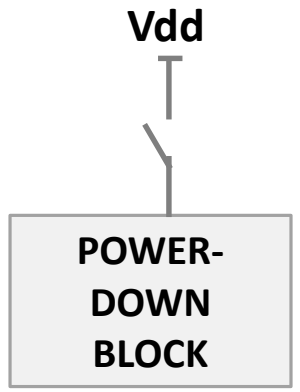
Clock-/Power-Based Techniques: Overhead

Switch off the clock, applicable to both ASIC and FPGA.

- @ASIC: simple and gates.
- @FPGA: dedicated vendor specific cells.

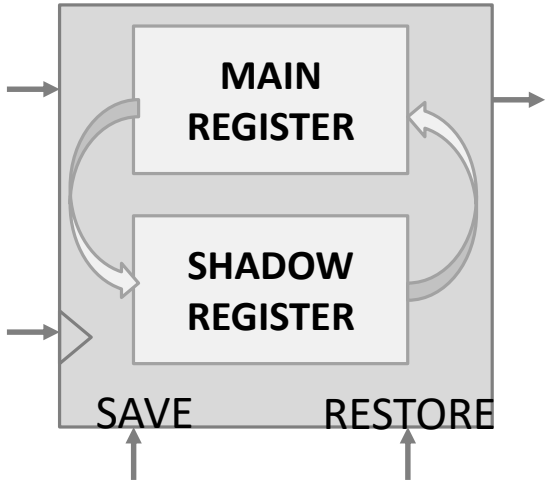
Switch off the power supply, ASIC only (partially FPGA).

- *Sleep transistors*: to switch on and off power supply.
- *Isolation logic*: to avoid the transmission of spurious signals from gated regions to normally-on cells.
- *Retention logic*: to maintain the internal state of gated regions.

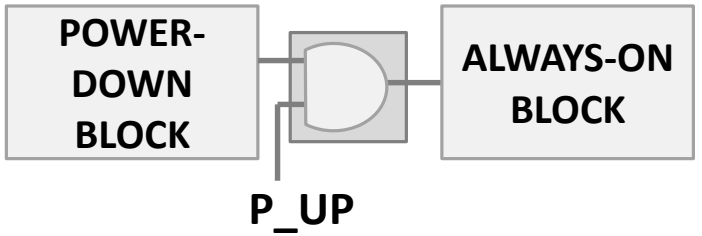


Power Switch-Off Cell

Retention Register



Isolation Cell



Clock-/Power-Based Techniques: Overhead

Switch off the clock, applicable to both ASIC and FPGA.

- @ASIC: simple and gates.
- @FPGA: dedicated vendor specific cells.

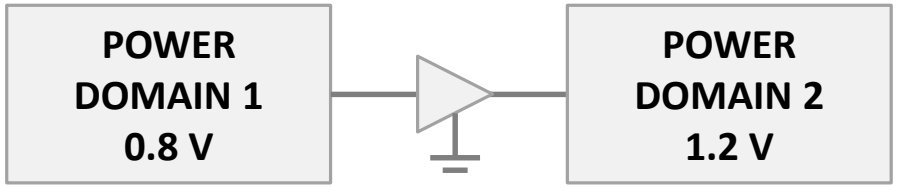
Switch off the power supply, ASIC only (partially FPGA).

- *Sleep transistors*: to switch on and off power supply.
- *Isolation logic*: to avoid the transmission of spurious signals from gated regions to normally-on cells.
- *Retention logic*: to maintain the internal state of gated regions.

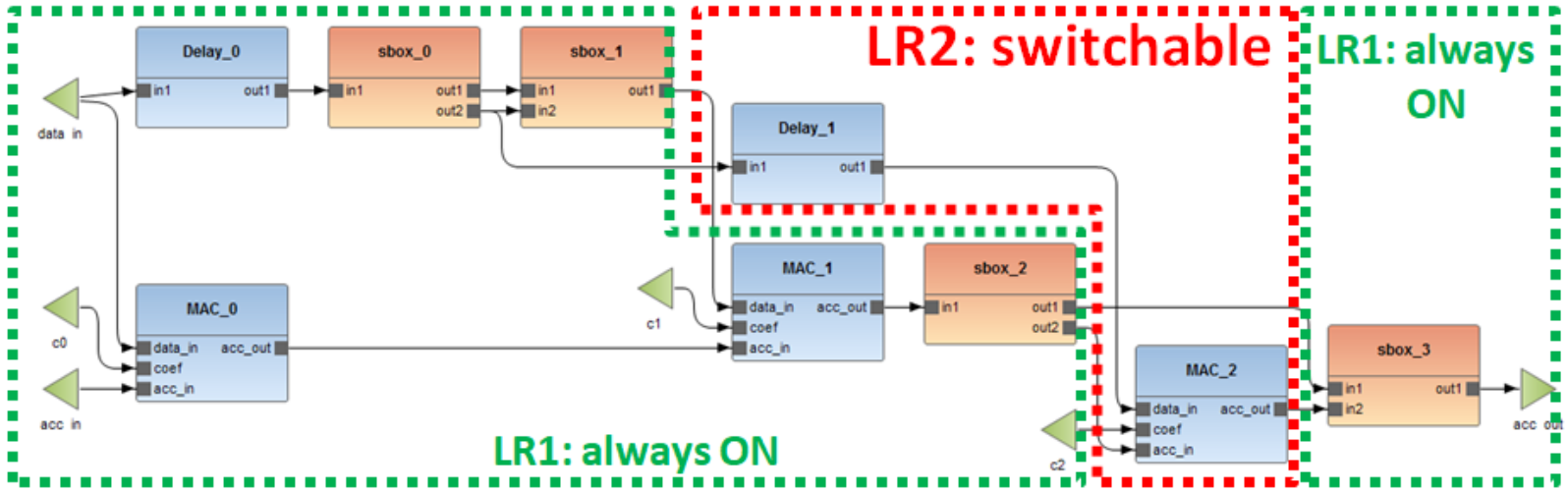
Vary mode changing Vdd, ASIC only.

- *Level shifters*: to pass signals between portions of the design that operate on different voltages.

Level Shifters

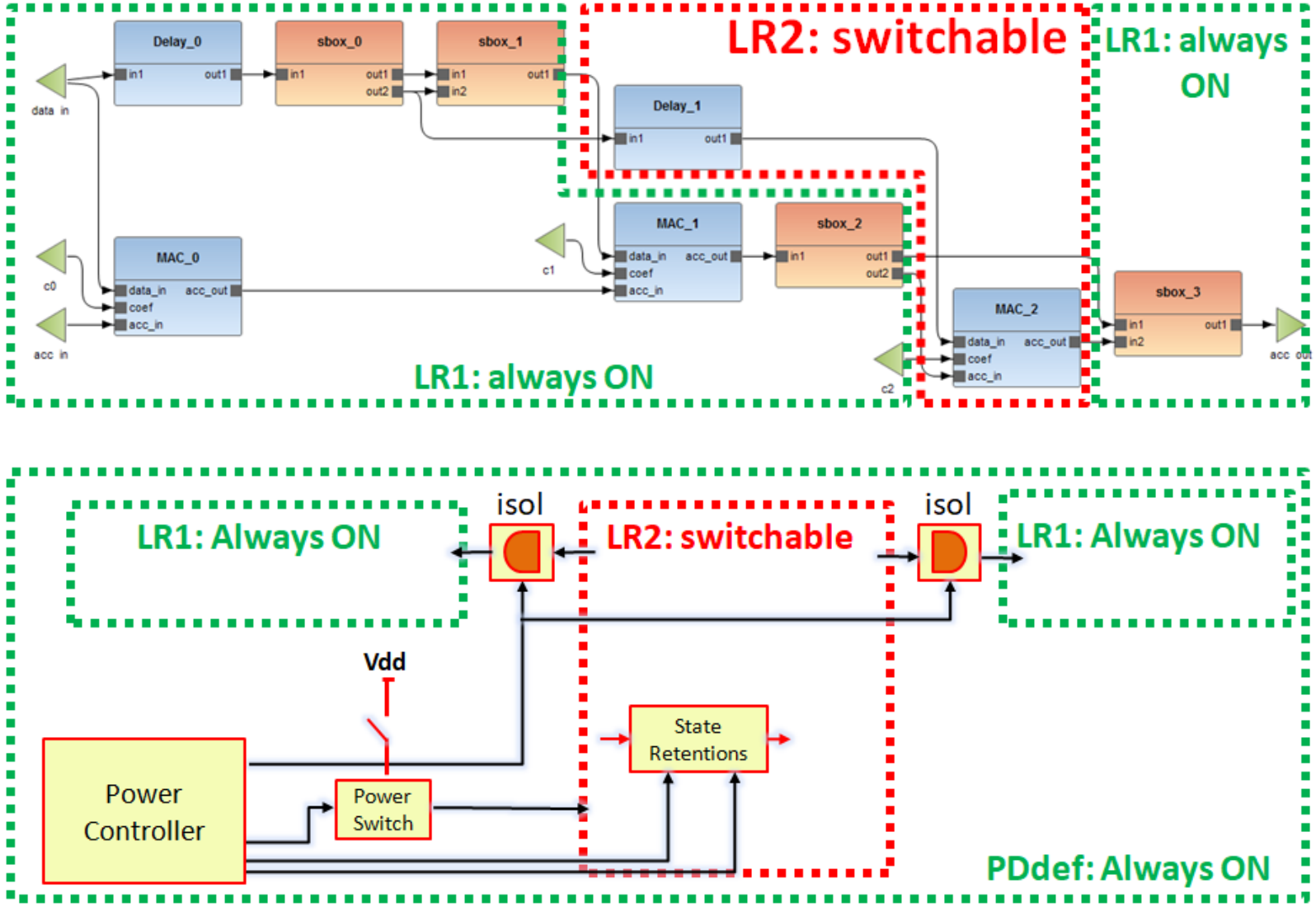


PSO Example



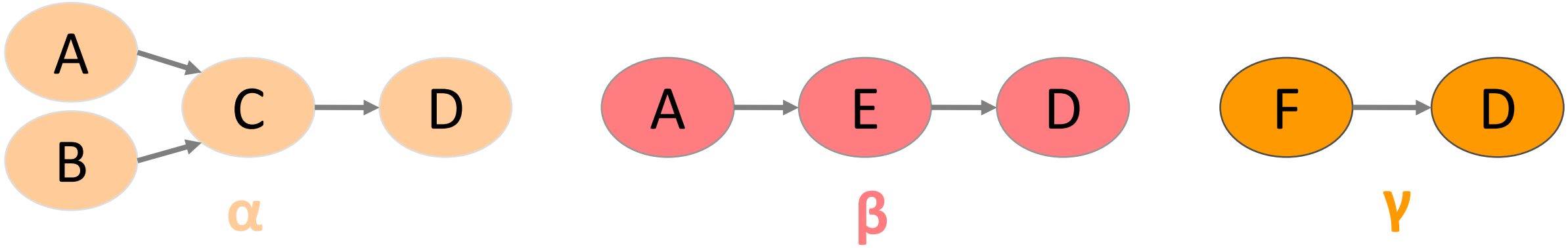
- Reconfigurable Filter, the depth of the filter can vary.
- 2 different Logic Regions (LR), one always on and one switchable

PSO Example

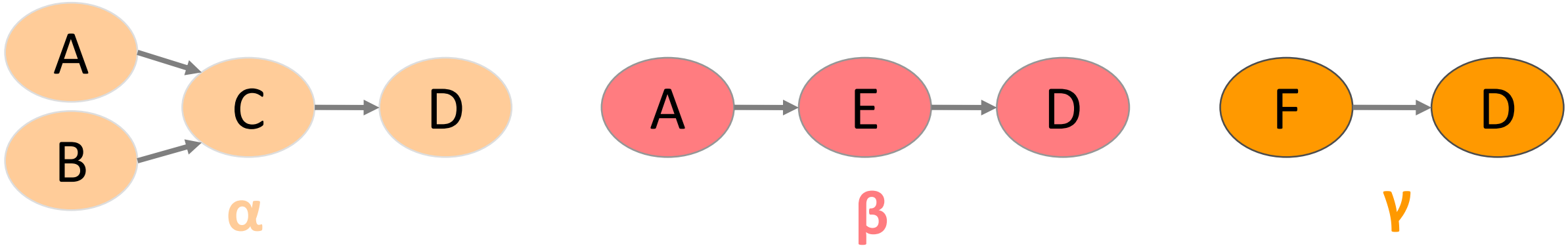


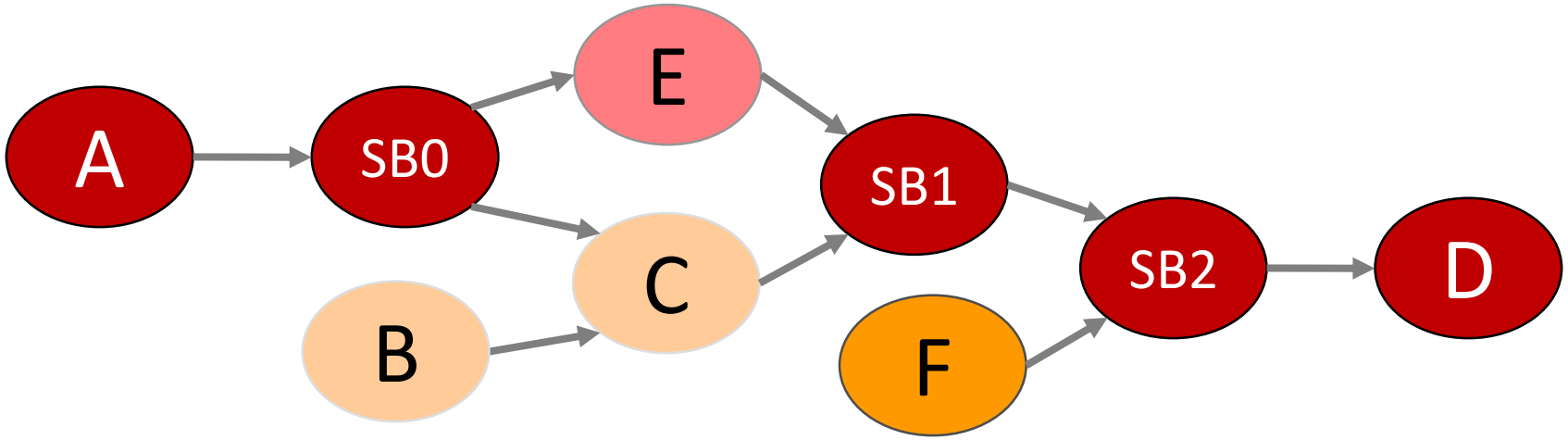
- Reconfigurable Filter, the depth of the filter can vary.
- 2 different Logic Regions (LR), one always on and one switchable
- Switchable LR needs the insertion of
  - isolation, retention and power switch cells;
  - one power controller to handle the control signals [1*shut-o + 1*isol. + 2*reten.]

Heterogeneous and Irregular CGR: power issue



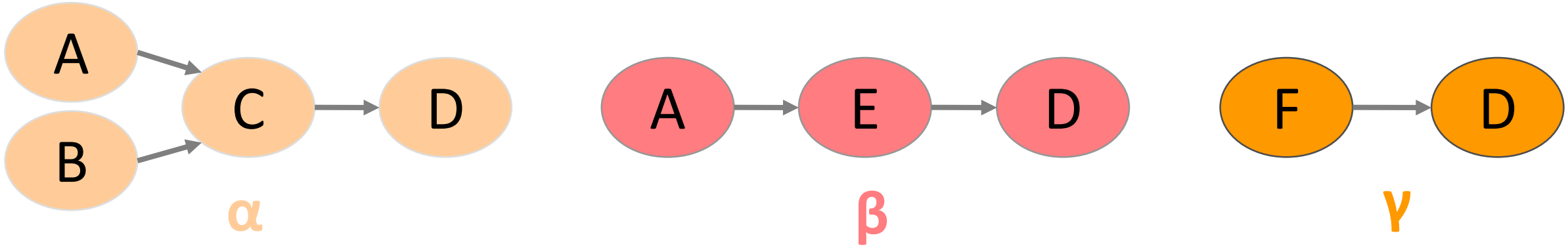
Heterogeneous and Irregular CGR: power issue

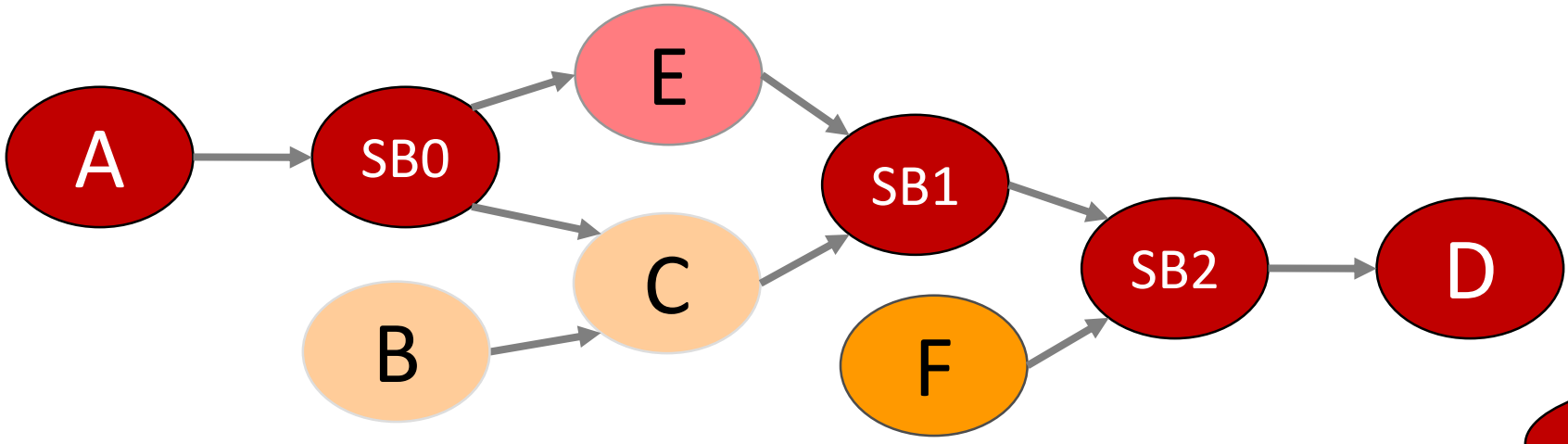




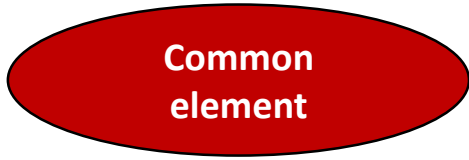
Multi-Dataflow Graph

Heterogeneous and Irregular CGR: power issue

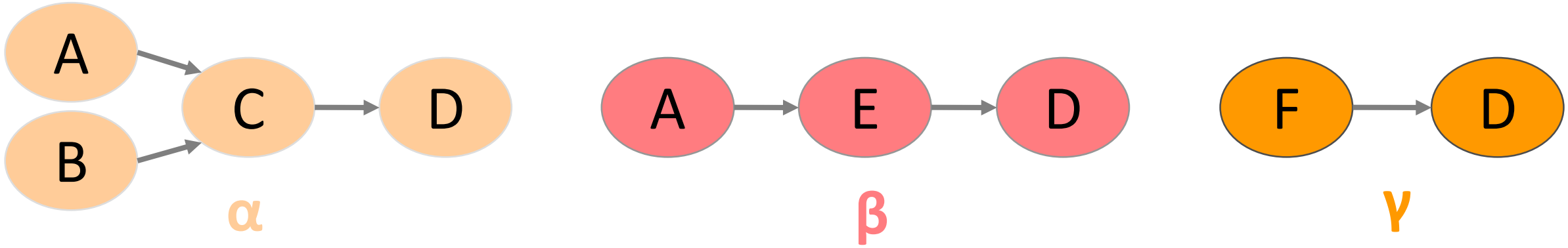




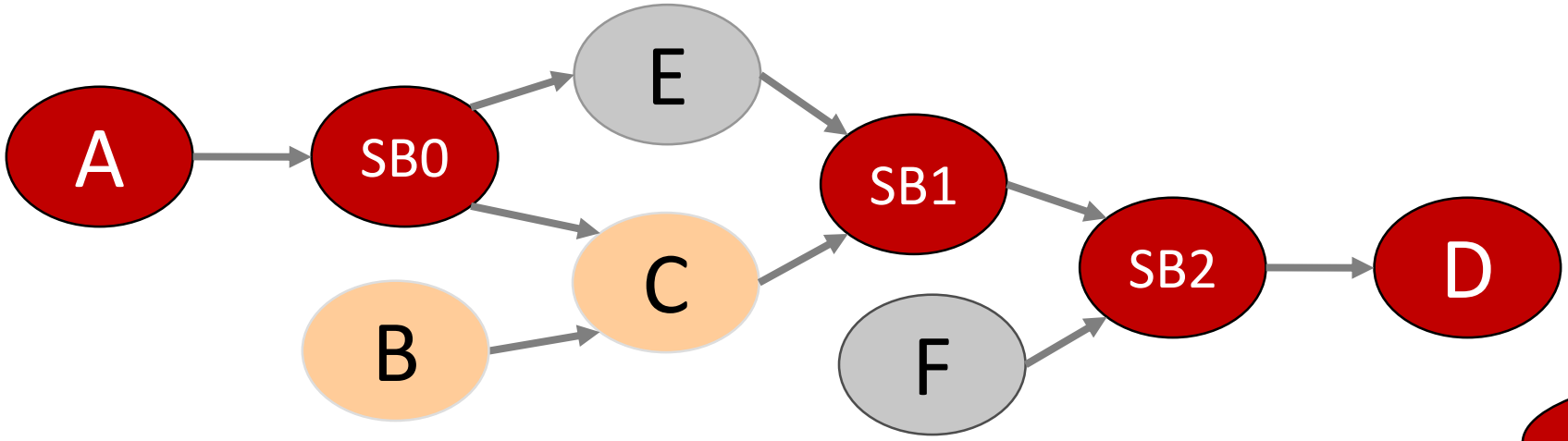
Multi-Dataflow Graph



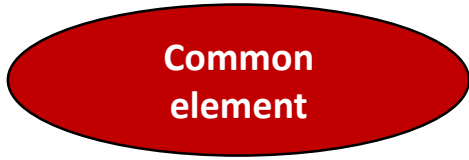
Heterogeneous and Irregular CGR: power issue



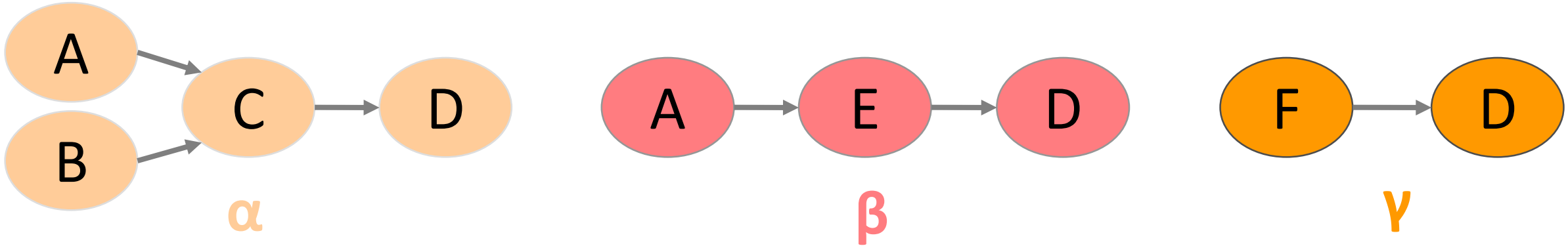
α execution: E and F, being not involved, are wasting power!



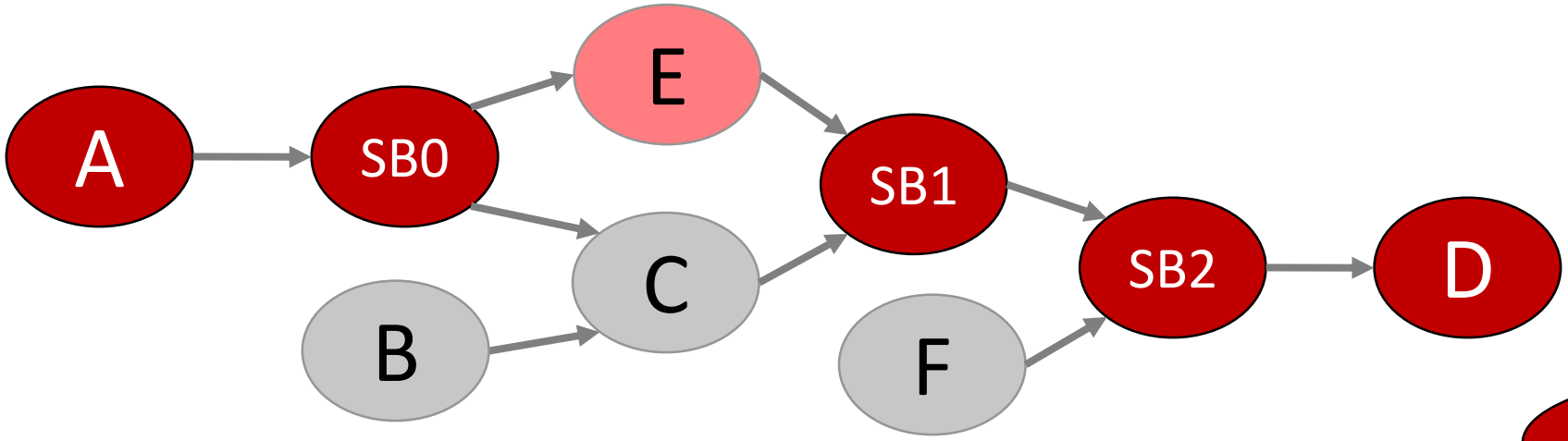
Multi-Dataflow Graph



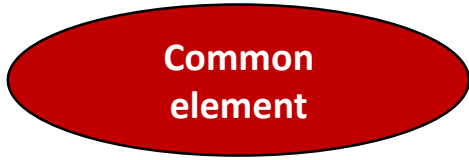
Heterogeneous and Irregular CGR: power issue



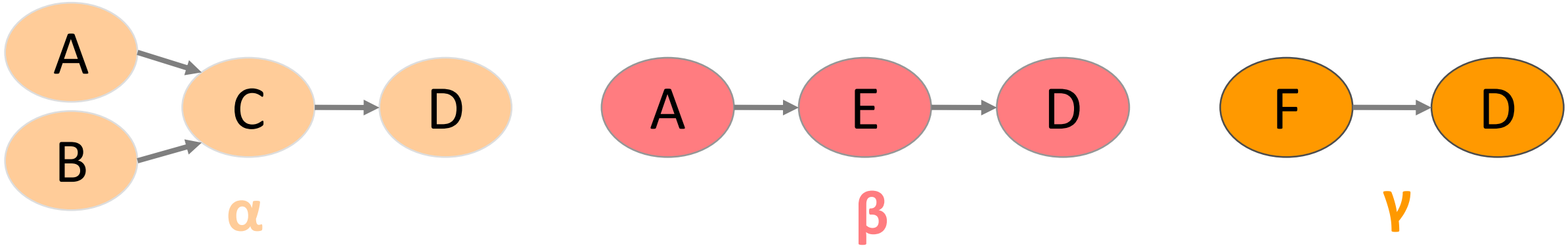
β execution: B C and D, being not involved, are wasting power!



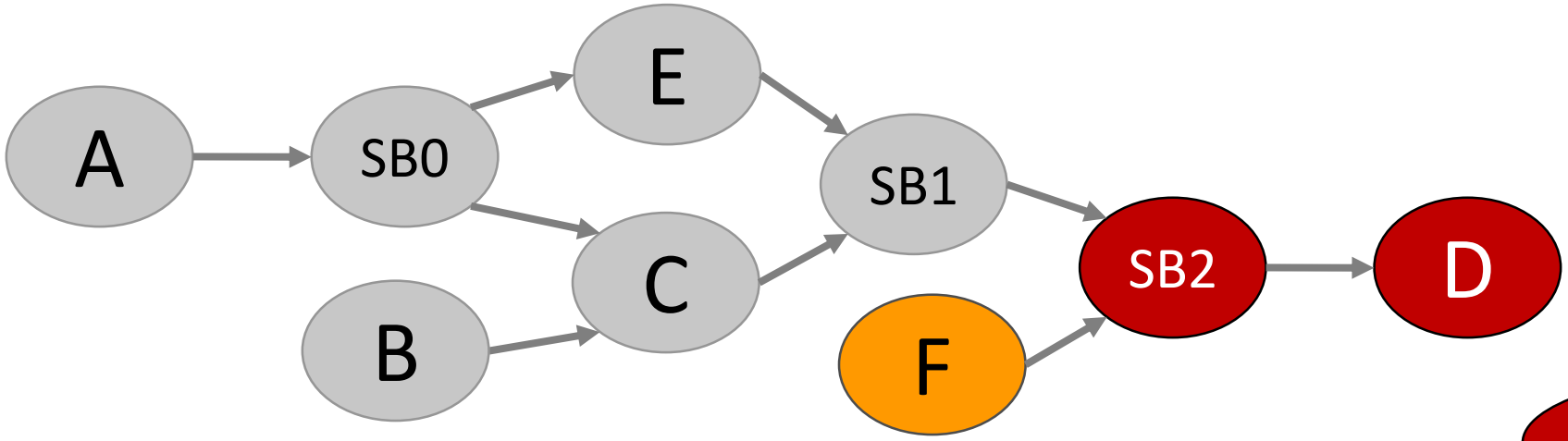
Multi-Dataflow Graph



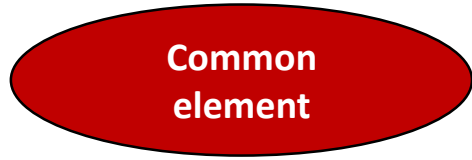
Heterogeneous and Irregular CGR: power issue



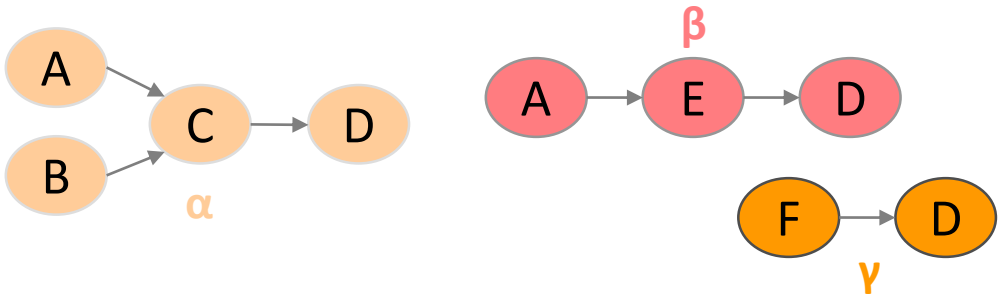
γ execution: B C and D, being not involved, are wasting power!



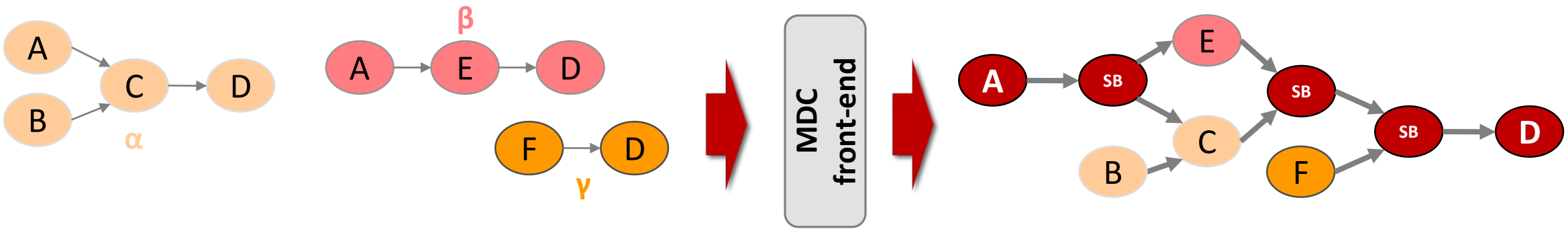
Multi-Dataflow Graph



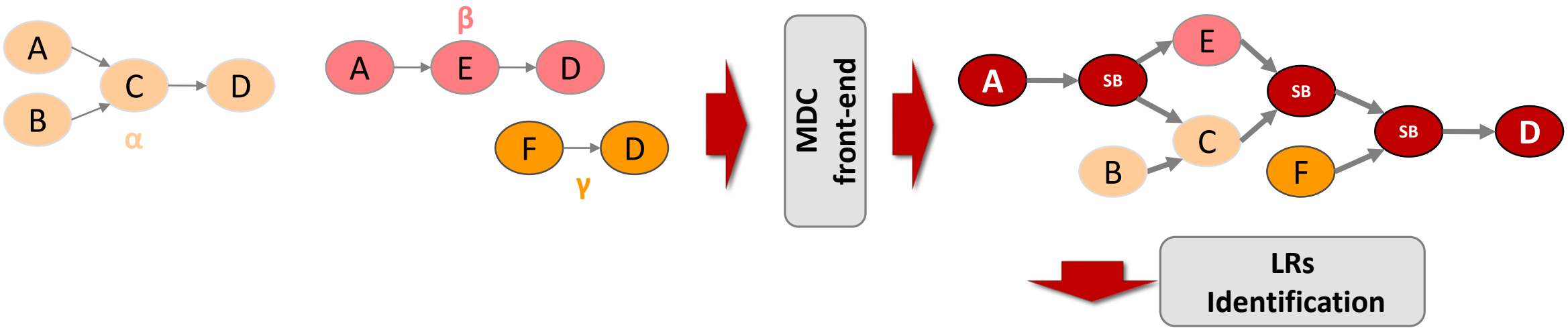
Heterogeneous and Irregular CGR: MDC approach

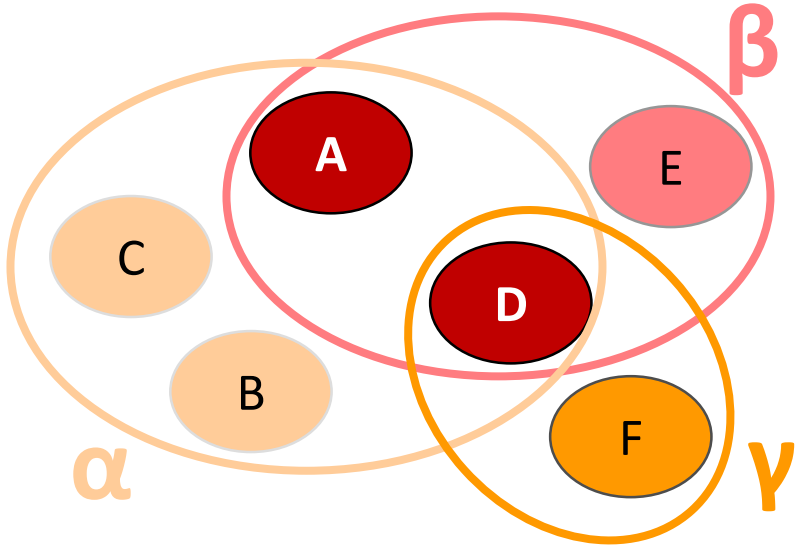


Heterogeneous and Irregular CGR: MDC approach

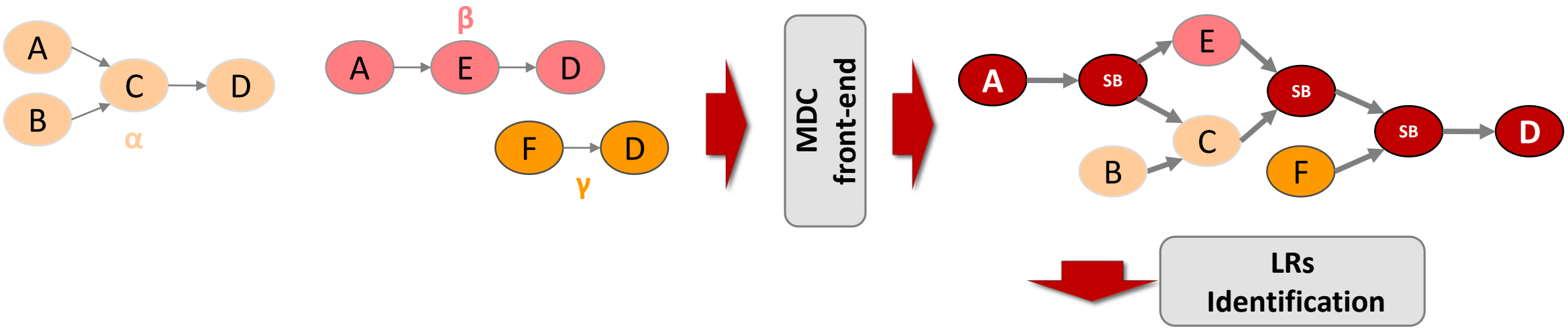


Heterogeneous and Irregular CGR: MDC approach

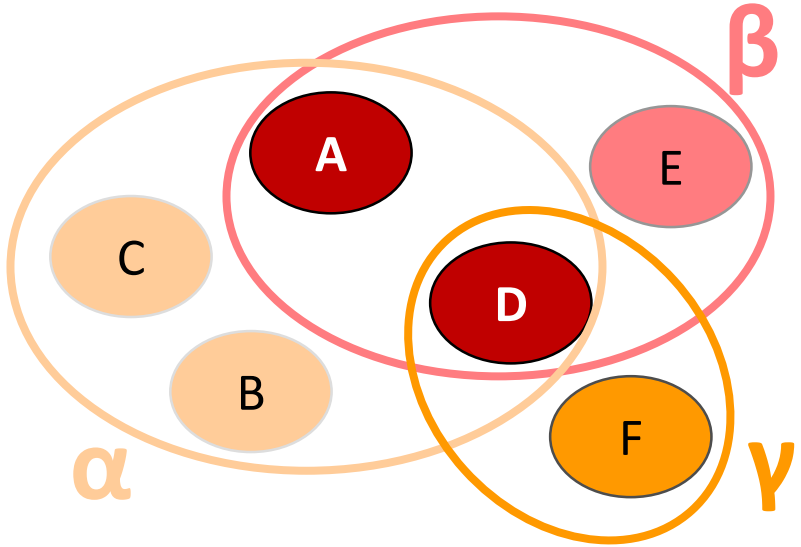




Heterogeneous and Irregular CGR: MDC approach



LR	1	2	3	4	5
actors	A	B,C	D	E	F
α	1	1	1	0	0
β	1	0	1	1	0
γ	0	0	1	0	1

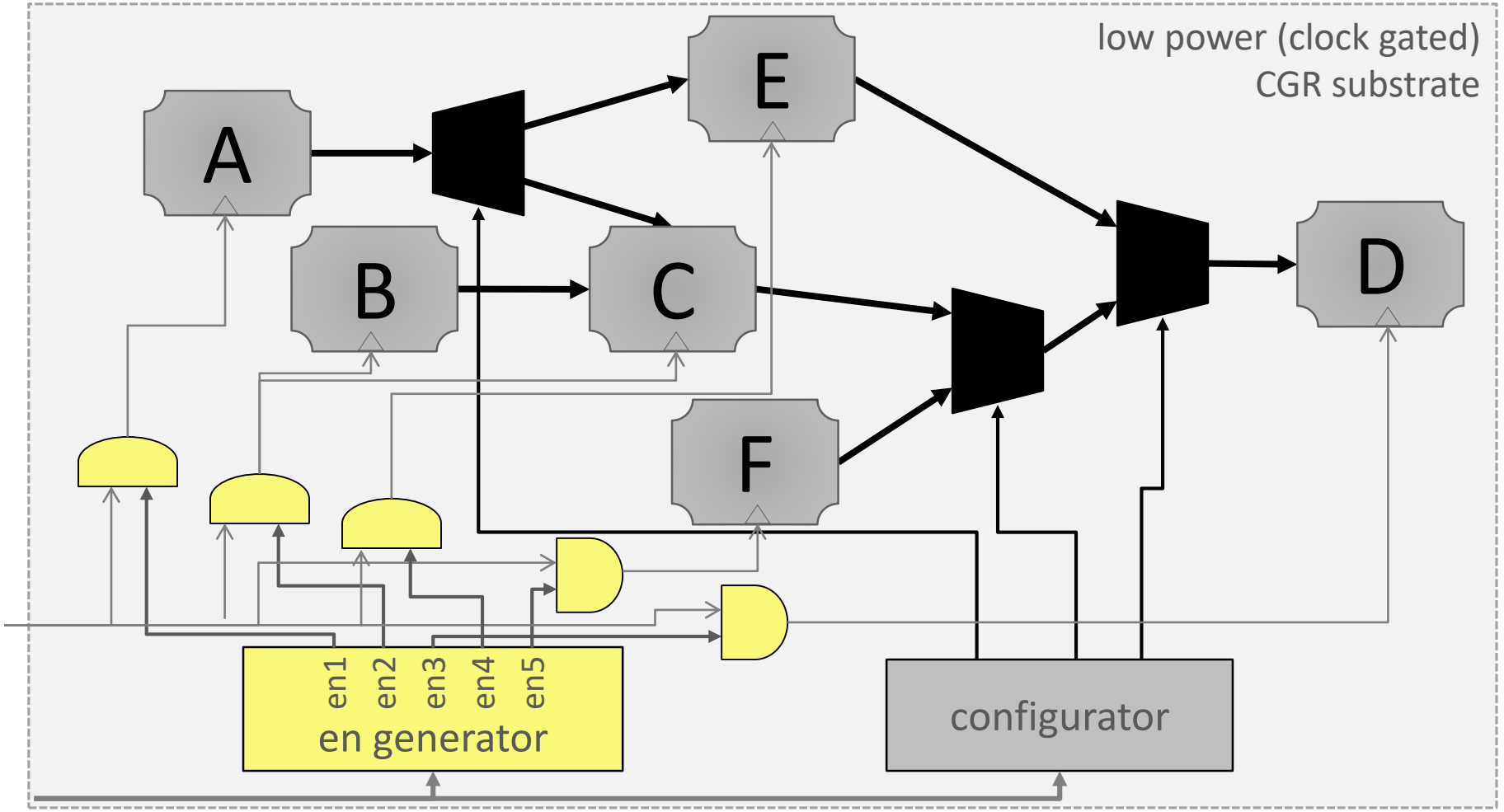


AUTOMATED DYNAMIC POWER MANAGEMENT

LR	actors	α	β	γ
1	A	1	1	0
2	B,C	1	0	0
3	D	1	1	1
4	E	0	1	0
5	F	1	0	1

MDC back-end

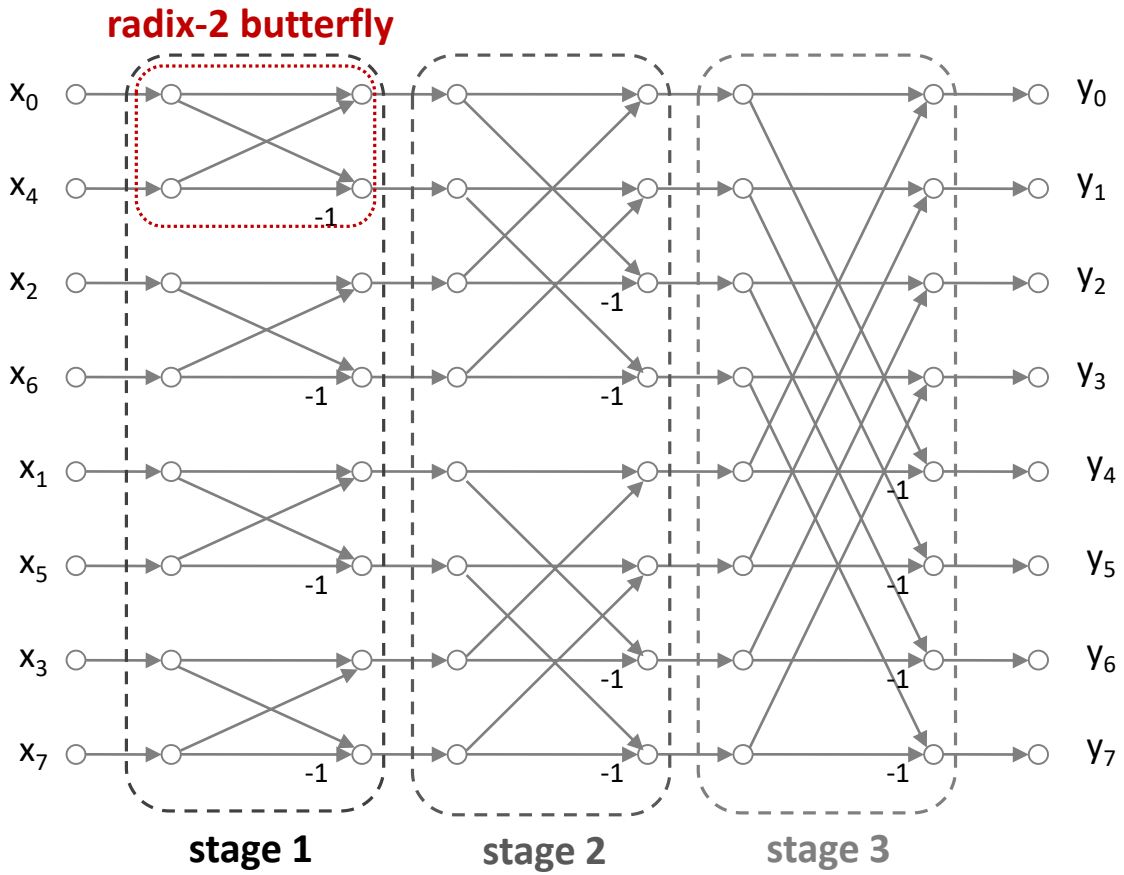
clk



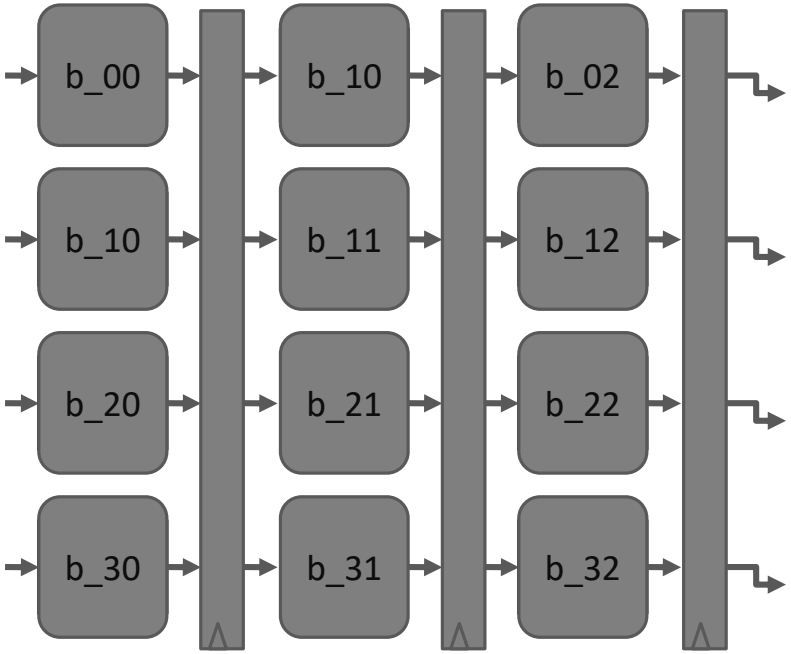
Overview

- Motivations
  - What we need adaptation for
  - Triggers and Types
- Coarse-Grained Reconfigurable Systems
  - Computing Spectrum and Reconfigurable Systems Classification
  - Heterogeneous and Irregular Coarse-Grained Reconfigurable Accelerators
- Power Management
  - Issues and Strategies
  - Low-Power Coarse-Grained Reconfigurable Accelerators
- An FFT Example

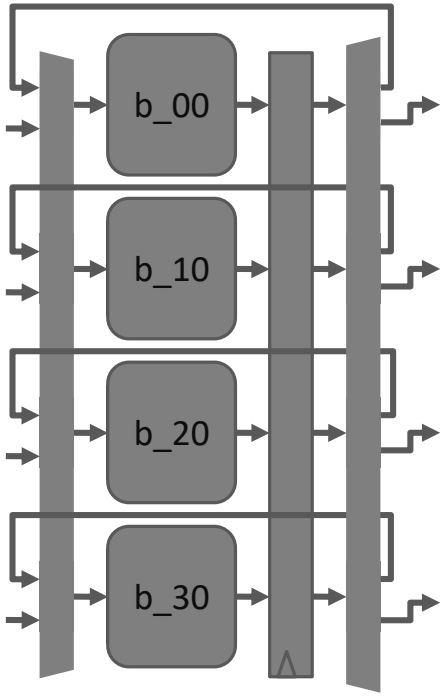
CG Reconfigurable FFT Design



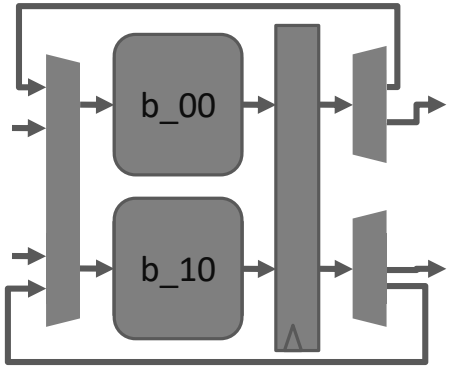
CG Reconfigurable FFT Design



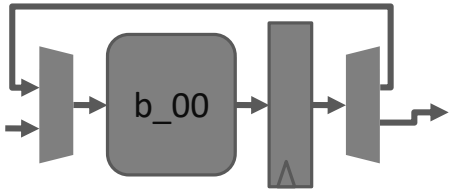
12 butterflies



4 butterflies

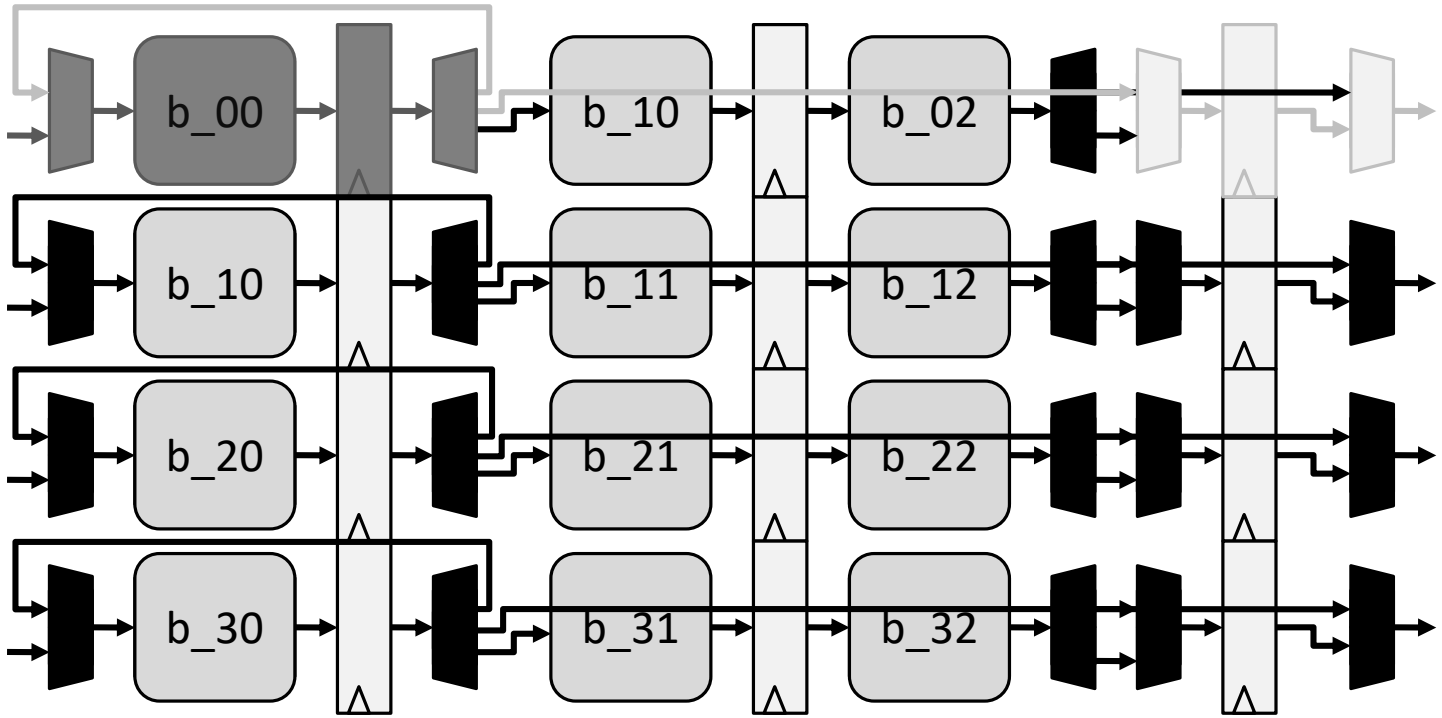


2 butterflies



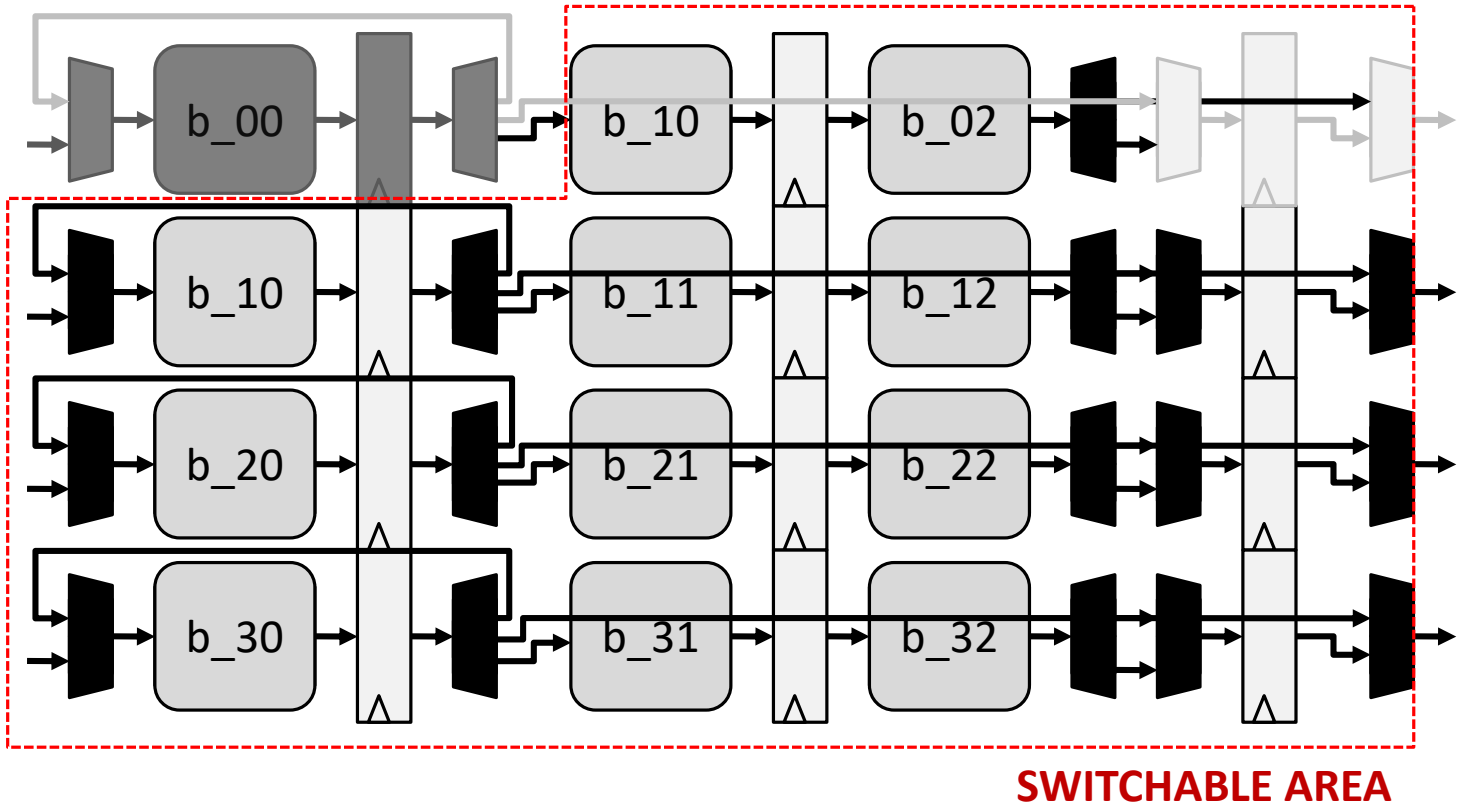
1 butterfly

Low-Power CG Reconfigurable FFT Design

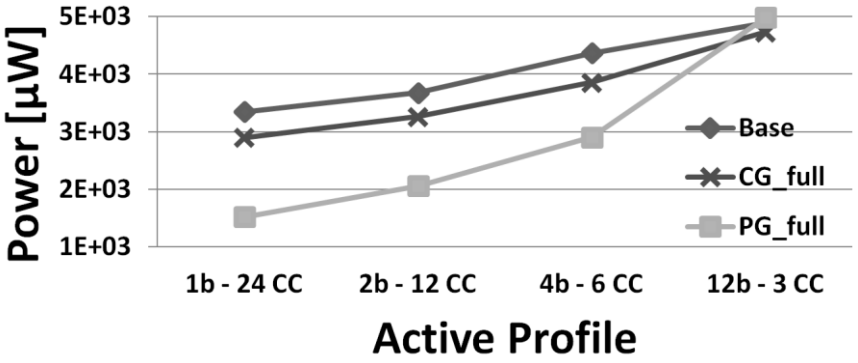


SWITCHABLE AREA

Low-Power CG Reconfigurable FFT Design



Low-Power CG Reconfigurable FFT: 90nm ASIC



FFT: power vs throughput

Dynamic trade-off management

	Area [GE]	vs Base%
Base	885575	---
CG_full	885871	0.03
PG_full	926835	4.66

FFT: Area

On ASIC MDC offers automatic implementation of power-gated and clock-gated designs

Intelligent system DEsign and Application (IDEA) @ UNISS

Luca Pulina
Associate Professor

Francesca Palumbo
Assistant Professor

Arthur Bit-Monnot
Postdoctoral Researcher

Laura Pandolfo
Postdoctoral Researcher

Claudio Rubattu
Researcher Assistant

Simone Vuotto
Researcher Assistant

Monica Marini
Communication and Administration

The MDC Group – UNISS + UNICA team

UNIVERSITY OF SASSARI

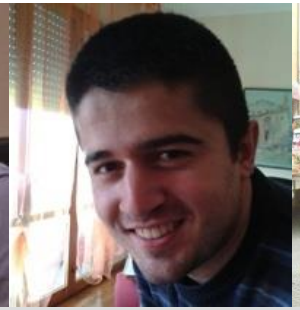

UNIVERSITY OF CAGLIARI




The MDC Group – UNISS + UNICA team

UNIVERSITY OF SASSARI

UNIVERSITY OF CAGLIARI



Thanks to …

EU Commission for funding the **CERBERO** (*Cross-layer modEl-based fRamework for multi-oBjective dEsign of Reconfigurable systems in unceRtain hybRid envirOnments*) project as part of the **H2020 Programme** under grant agreement No 732105.

Coordinator:

Michal Masin (IBM), michaelm@il.ibm.com

Scientific Coordinator:

Francesca Palumbo (UniSS), fpalumbo@uniss.it

Innovation Manager:

Katiuscia Zedda (Abinsula), katiuscia.zedda@abinsula.com

Dissemination-Communication Manager:

Francesco Regazzoni (USI), francesco.regazzoni@usi.ch

www.cerbero-h2020.eu

info@cerbero-h2020.eu

@CERBERO_h2020

CERBERO