Improving Synchronous Elastic Circuits: Token Cages and Half-Buffer Retiming

Mario R. Casu
Dipartimento di Elettronica, Politecnico di Torino, Italy
Outline

- Quick review of elastic circuits
- Motivations of this work
- Elastic Half-Buffer retiming
- Token Cages
- A realistic example
- Synthesis experiments
- Conclusions
Elasticity

- Elastic circuits offer a uniform methodology to deal with variability of delays:
  - Process, voltage, temperature variations
  - Variable latency computation (e.g. multi-cycle units)
  - Wire delays not accounted for in front-end design
  - Variable input rates
  - ...

- Elastic circuits work despite unpredictable arrival time of inputs
  - Can be formalized as “Latency-Insensitivity”
Elasticization of datapath

DATAPATH

CONTROL

wire or logic

valid

stop

Mario R. Casu, ASYNC’10
Elastic Protocol (SELF)

- Synchronous ELastic Flow (SELF) [Cortadella, DAC06]
  - TRANSFER, IDLE, RETRY
Elastic Protocol (SELF)

- Synchronous ELastic Flow (SELF) [Cortadella, DAC06]
  - TRANSFER, IDLE, RETRY
- Synchronous Interlocked Pipelines [Jacobson, ASYNC02]

Mario R. Casu, ASYNC’10
Elastic Buffer (EB)
Elastic Buffer (EB)
Elastic Buffer (EB) Elastic Half Buffer (EHB)

EHB CONTROLLERS

Valid_{in}  Stop_{in}  Valid_{out}  Stop_{out}
JOIN

Mario R. Casu, ASYNC’10
JOIN with Early Evaluation

2in/1out logic

X1
CK

X2
CK

Y1

Y2

W

Z

Mario R. Casu, ASYNC’10
JOIN with Early Evaluation

- Anti-tokens in Synch Elastic Circuits [Cortadella, DAC07]
- Previous works on early-evaluation or anti-tokens (async)
  - [Brej, IWLS03], [Reese, TCAD05], [Ampalam, ICCAD2006]
JOIN with Early Evaluation

DATAPATH
CONTROL

DUAL EHB CONTROLLERS

Mario R. Casu, ASYNC'10
Motivations of present work

1. Latency mismatch of join inputs
   - Case without early evaluation
   - The “Bubble Bounce” problem

2. Token preservation
   - Case with early evaluation
   - “An anti-token is generated only if a token is emitted at the same time”
   - The “useless (yet stopped) token” problem
The “Bubble Bounce” Problem

AND firing rule on C’s join controller
The “Bubble Bounce” Problem

- AND firing rule on C’s join controller

Issues:
- Long wires
- Throughput reduction
Which Solution?

- Buffer insertion

- Buffer sizing

Mario R. Casu, ASYNC’10
The “Useless Token” Problem

- Early Evaluation on C’s join controller
  - Processing input from B
  - Cannot be canceled since an anti-token is not generated
  - Useless yet stopped token!
  - Same throughput as with no early evaluation
The “Useless Token” Problem

- Do not enqueue it, cage it!
  - Just the validity bit gets caged (data killed immediately)
- …and when it’s safe, kill it!

Mario R. Casu, ASYNC’10
The “Useless Token” Problem

- Do not enqueue it, cage it!
  - And when it’s safe…kill it!
Making queues and cages…

- Buffer sizing
  - One place queue through elastic half-buffer (EHB) retiming

- Token cage
  - Simple two-states FSM (empty or full)
EHB retiming 2/2

DATAPATH

CONTROL

Join ctrl logic

stop

Mario R. Casu, ASYNC'10
Time After Re-Time

😊 Moving the latches ahead may increase the variability tolerance (*)
😊 Moving the latches ahead may cause a timing violation in a critical path (*)

(*) details in proceedings
Token Cages

Join controller with Early Evaluation

\[ P_1 \quad | \quad P_2 \]

valid\(^+\)_1
stop\(^+\)_1
valid\(^-\)_1
stop\(^-\)_1
valid\(^+\)_2
stop\(^+\)_2
valid\(^-\)_2
stop\(^-\)_2
valid+
stop+
valid-
stop-
Token Cages

Cage 1

valid+₁
stop+₁
valid-₁
stop-₁

P₁

P₂

Cage 2

valid+₂
stop+₂
valid-₂
stop-₂

Join controller with Early Evaluation

P₁
P₂

valid+
stop+
valid-
stop-
Cage 1

- Cage a useless token
  - $P_1 = 0$ (not processed token), $EE = 0$ (no early evaluation)

Mario R. Casu, ASYNC’10
Cage 1

- Cage-it!
  - \( ( \text{valid}^+ \text{and not } P_1 \text{and not } EE ) \text{ or full } \) and \( \text{stop}'^+_1 \)

Mario R. Casu, ASYNC’10
Kill-it!

- not ( stop’+1 and ( full or ( valid+1 and not P1 and not EE ) ) )

Mario R. Casu, ASYNC’10
Putting it all together...

- A simple yet close to real-life example
  - Datapath of a simple pipelined processor
- Elastic control
  - Join with early evaluation
- Effect on throughput of
  - Buffer insertion
  - Buffer sizing (EHB retiming)
  - EHB retiming and Token cages
Putting it all together…

- Simple processor with 4 types of arithmetic operations
  - ADD, MUL, MAC, AAC
Simple processor with 4 types of arithmetic operations
- ADD, MUL, MAC, AAC

Three domains each under a different EB controller
- IFD + RF, EXE, MEM

Mario R. Casu, ASYNC’10
Simple processor with 4 types of arithmetic operations
- ADD, MUL, MAC, AAC

Three domains each under a different EB controller
- IFD + RF, EXE, MEM
Datapath made elastic

Abstract Model with Tokens

Mario R. Casu, ASYNC’10
Breaking the critical path

True...But what about throughput?

...Just add the bubble and everything works!

Mario R. Casu, ASYNC’10
Join with early evaluation

- Processing fast input
  - Throughput: 1

- Processing slow input
  - Useless token problem
  - Throughput: $\frac{1}{2}$

- Buffer insertion
Join with early evaluation

- Processing fast input
  - Throughput: 1

- Processing slow input
  - Useless token problem
  - Throughput: \( \frac{1}{2} \)

- Buffer insertion
  - Throughput: \( \frac{3}{4} \) (always)
  - Can’t reach max throughput!

Mario R. Casu, ASYNC’10
Join with early evaluation

- Processing fast input
  - Throughput: 1

- Processing slow input
  - Useless token problem
  - Throughput: $\frac{1}{2}$

- EHB retiming
Join with early evaluation

- **Processing fast input**
  - Throughput: 1

- **Processing slow input**
  - Useless token problem
  - Throughput: \( \frac{1}{2} \)

- **EHB retiming**
  - Processing fast input
    - Throughput: 1
  - Processing slow input
    - Throughput: \( \frac{2}{3} \)

Mario R. Casu, ASYNC’10
Join with early evaluation

- Processing fast input
  - Throughput: 1

- Processing slow input
  - Useless token problem
  - Throughput: $\frac{1}{2}$

- EHB retiming + cage
  - Processing fast input
    - Throughput: 1
  - Processing slow input
    - Throughput: $\frac{3}{4}$

Mario R. Casu, ASYNC’10
Synthesis & Mapping exercise

- **Reference designs:**
  - 2in/1out and 2in/2out controllers w/ and w/o early evaluation [Cortadella, DAC06][Cortadella, DAC07]

- **Comparisons with new designs:**
  - 2in/1out and 2i/2out … + EHB retiming
  - 2in/1out and 2i/2out … + token cages
  - 2in/1out and 2i/2out … + retiming & cages

- **CMOS 45nm 1.1V technology target**
  - area, dynamic and leakage power evaluated

Mario R. Casu, ASYNC’10
Overhead w.r.t. EB

- **Area**
- **Dynamic Power**
- **Leakage Power**

Reference designs

Mario R. Casu, ASYNC’10
Conclusions

- Two innovations in synchronous elastic circuits
  1. Elastic Half Buffer retiming as a smart way to create by-passable input queues (buffer sizing)
  2. Token cages help improve throughput by discarding some useless tokens in case of early evaluation

- Caveat
  - Revise timing constraints after latch retiming

- Costs
  - Area and power of control logic increase
  - Usually much lower than datapath area/power
Contacts

Mario R. Casu
VLSI Lab, Dipartimento di Elettronica
Politecnico di Torino
mario.casu@polito.it

www.vlsilab.polito.it    www.polito.it