REDOUBT - Reliable FPGA Datapath Design Using Control Techniques

Oana Boncalo

University Politehnica Timisoara

oana.boncalo@cs.upt.ro

December 4, 2019

Overview

Outline

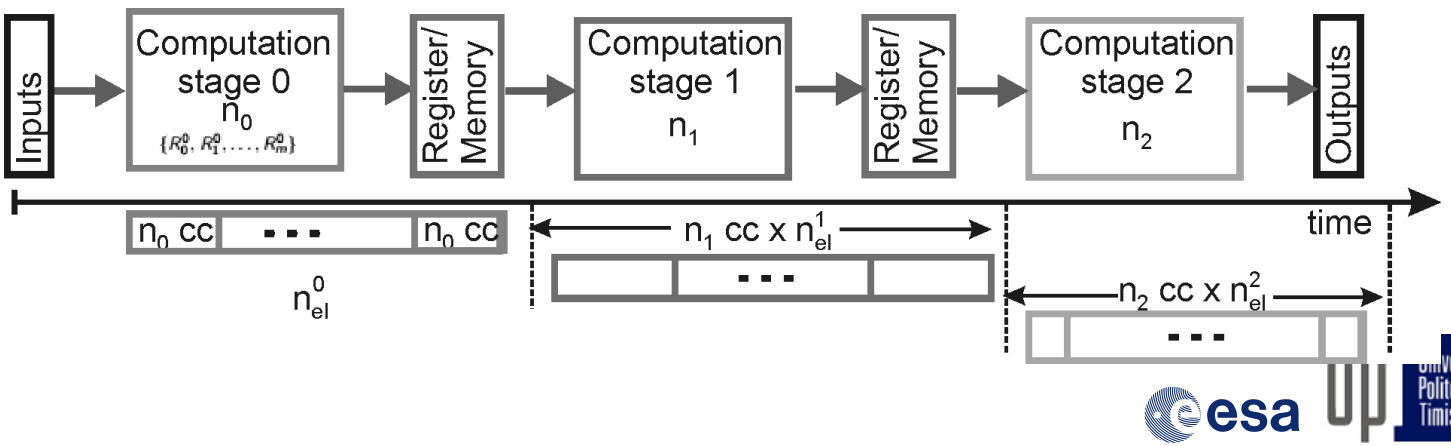
Overview

- ESA Innovation Triangle Initiative - Type A
- Duration : July 2018 - June 2019
- Consortium : University Politehnica Timisoara (Lead), Technical University of Cluj-Napoca (subcontractor)
- Project team : Oana Boncalo, Alexandru Amaricai (UPT), Zsofia Lendek (UTCN)
- Objective : Develop novel fault tolerant techniques for digital datapath circuits by applying control feedback loops
- Start - End TRL: 0 - 2
- Budget: 48K

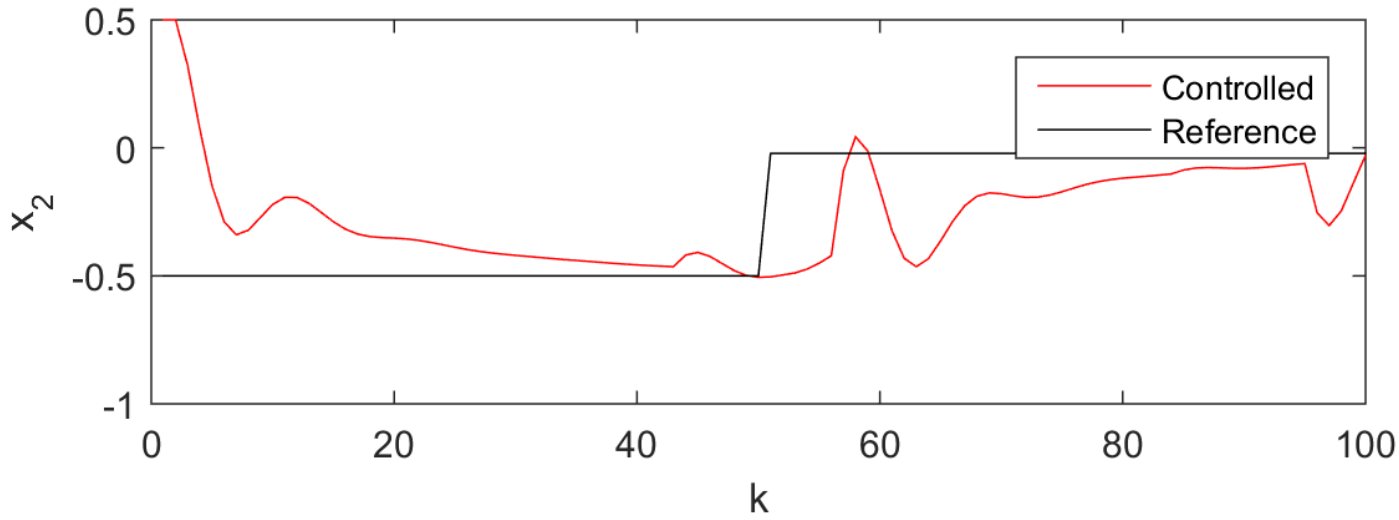
Outline

Target circuit

- Data-path units:
  - several processing stages built out of arithmetic operations: additions, multiply-add fused, multiplications
  - The partial output is stored to memory or inside registers and may consist of one or several data outputs;
- Each processing stage i consists of a sequence of registers $\{R_0^i, R_1^i, \ldots, R_{n_i}^i\}$, takes n_i clock cycles to compute one data output, and processes a total of n_{el}^i data values;

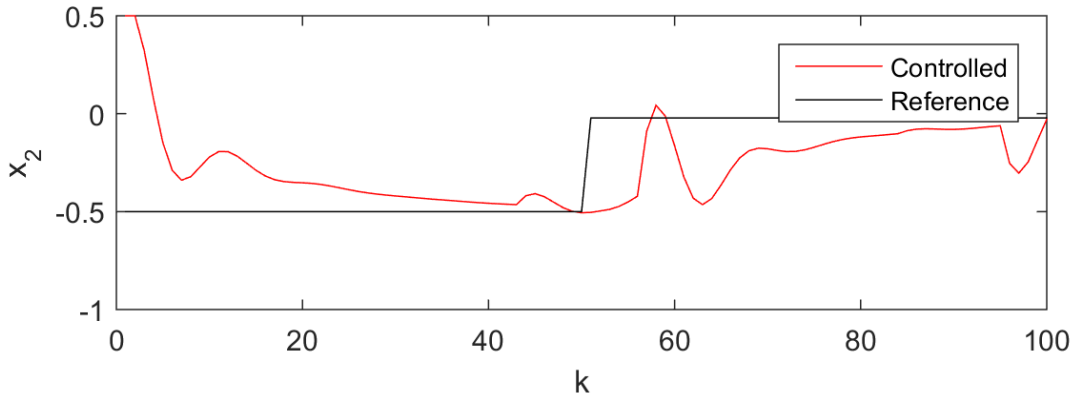


The idea



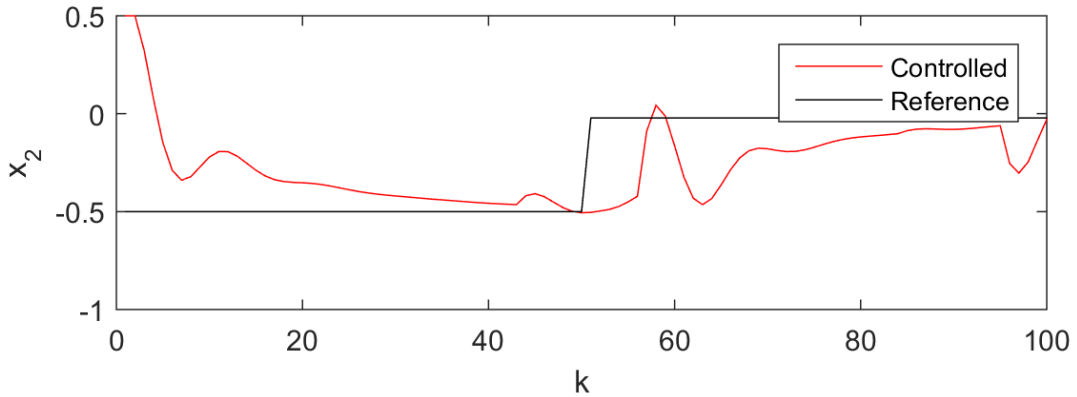
- Control engineering basic working principle – 3 building blocks:
  - the physical system modelled as a process
  - the reference
  - the controller that ensures that the process "follows" as closely as possible the reference
- Idea: include a control feedback loop that does the same for a circuit affected by probabilistic faults!

The idea



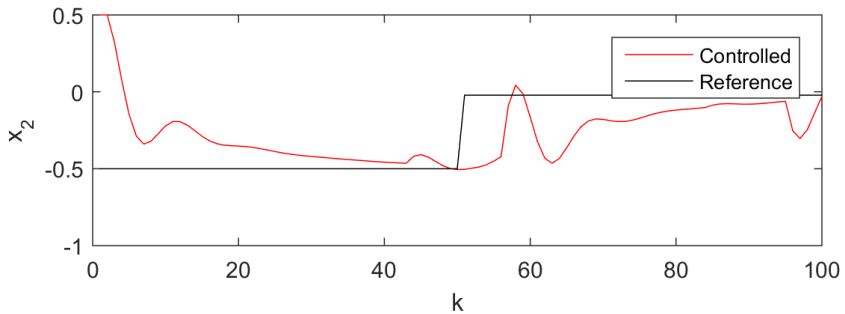
- Control engineering basic working principle – 3 building blocks:
  - the physical system modelled as a process → the digital circuit data-path
  - the reference → **?**
  - the controller that ensures that the process "follows" as closely as possible the reference → **?** needs to be designed somehow
- Note: control engineering mitigates the perturbation, such that the process successfully follows the reference (*i.e.* the system is stable).

The idea



- Note 1: control engineering does not target zero error! It reduces the error amplitude such that it is acceptable for the application, and it successfully follows the reference (*i.e.* the system is stable).
- Note 2: control engineering physical systems have large inertia. The controller adjusts its outputs in a window of time corresponding to several samples.
- Digital circuits are expected to deliver the correct, or almost correct output for every set of inputs!

The idea



- Approach is suitable for applications that tolerate small magnitude errors at the output!
- Digital circuits are expected to deliver the correct, or almost correct output for every set of inputs!
- $\rightarrow$ extend computation such that the correction controller manages to decrease the error amplitude at the expected level!

General considerations

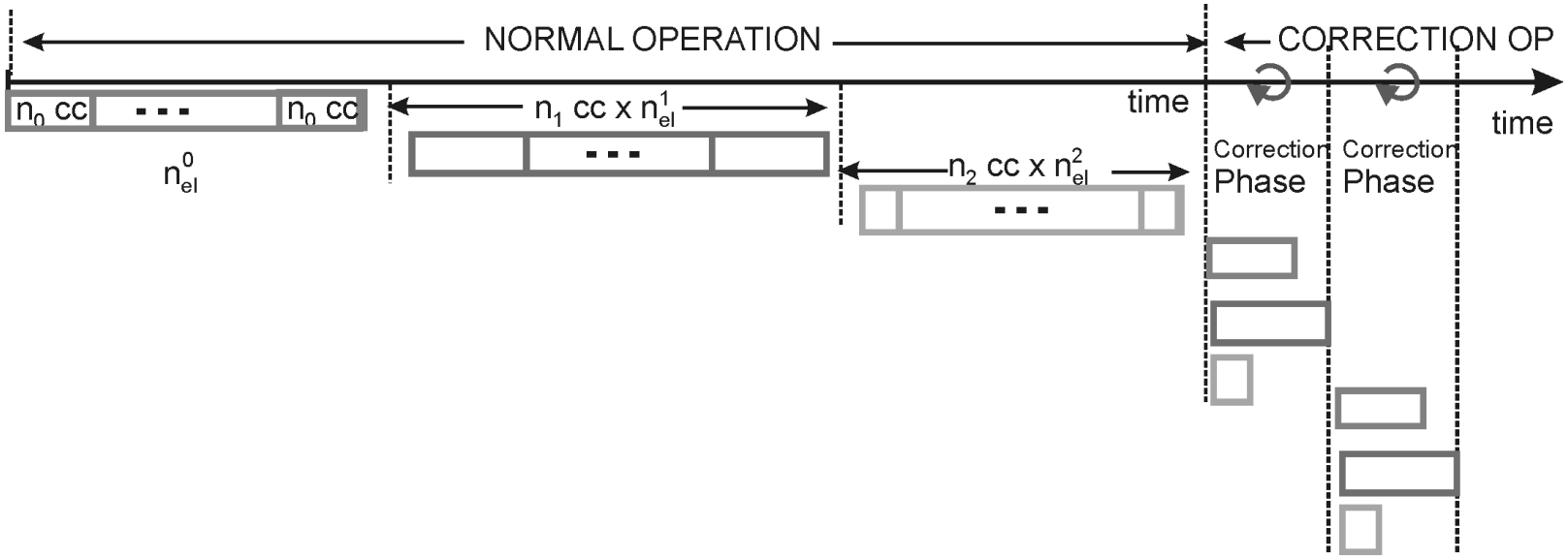
- Data-path circuit: several processing stages with intermediate results stored in registers or memory
- Fault model: probabilistic faults
- Fault mitigation: some sort of control loop that attenuates the error at the output until it becomes acceptable for the application at hand, or is zeroed
- Acceptable penalty (throughput): computation may be extended for a number of clock cycles under certain conditions
- Acceptable penalty (cost): less than triple modular redundancy

Outline

Design Problems

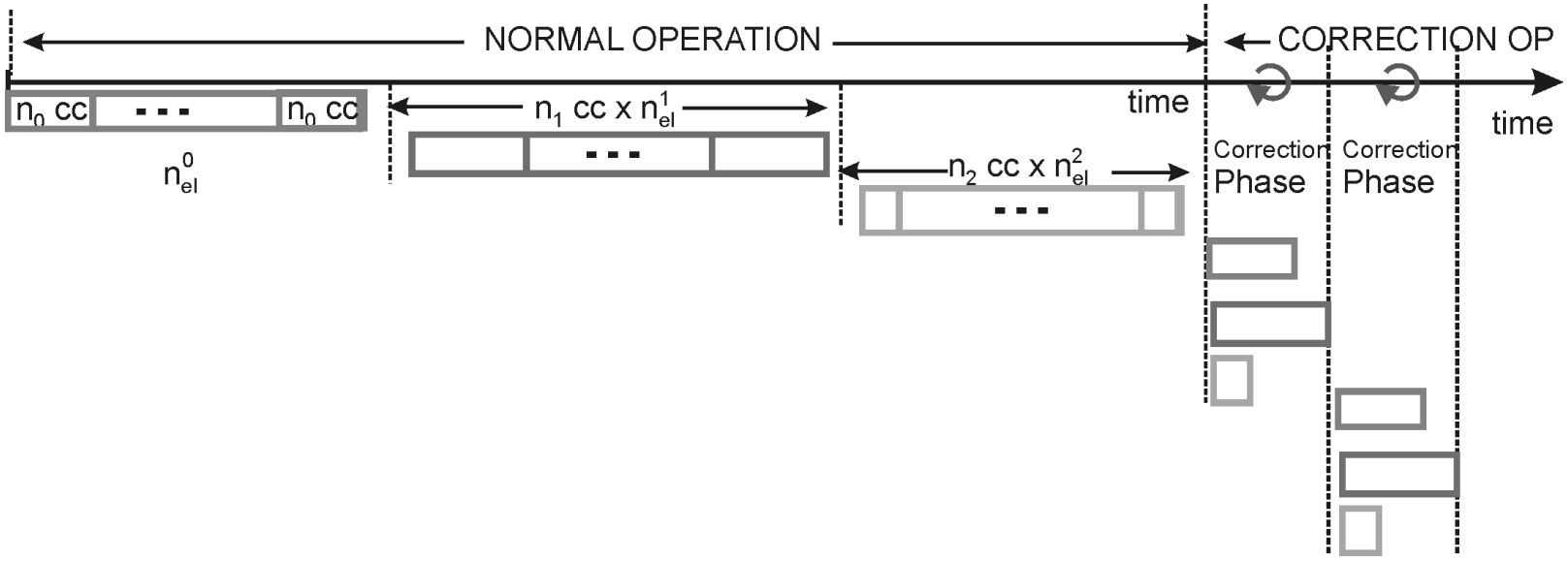
1. FT Digital Circuit operation
2. The correction control feedback loop:
   - the model of the digital circuit – non-linear
   - linearize the model of the digital circuit in order to be able to compute the linear correction controller
3. The reference

Digital circuit operation



- Extend computation in case errors are detected – activate correction controller loop
- Limit latency increase by:
  - pipeline stages execution rewind
  - parallel block execution
  - concept of correction execution phase - largest latency – $max n_i$ cc over all stages

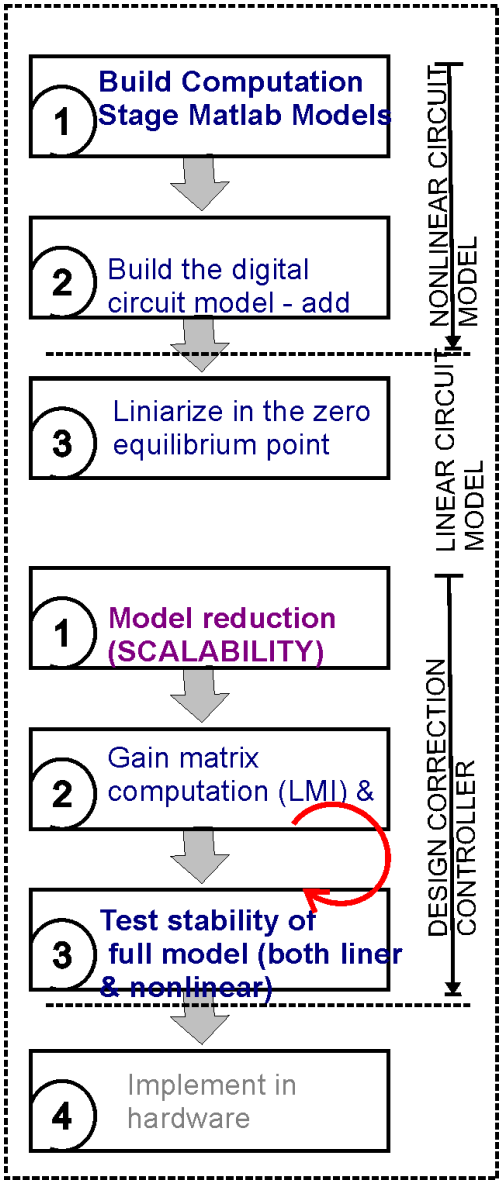
Digital circuit operation



- At each cc the correction controller adds some correction offsets to designated registers of the blocks rewinded such that the error magnitude is reduced.
- Limit cost increase by:
  - simple controller $\rightarrow$ linear
  - short response time

Outline

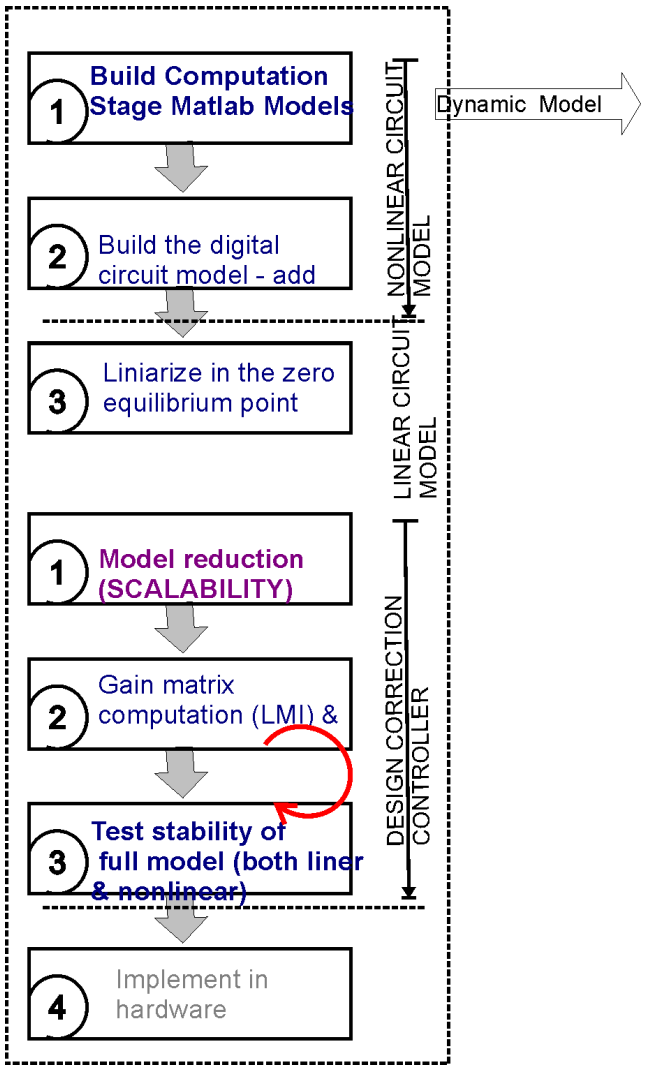
Correction Controller Design Steps



Steps for correction controller model design and test:

1. Build Dynamic circuit Model (Matlab)
2. Design linear correction controller
3. Test stability of model in different linearization points
4. Implement in hardware

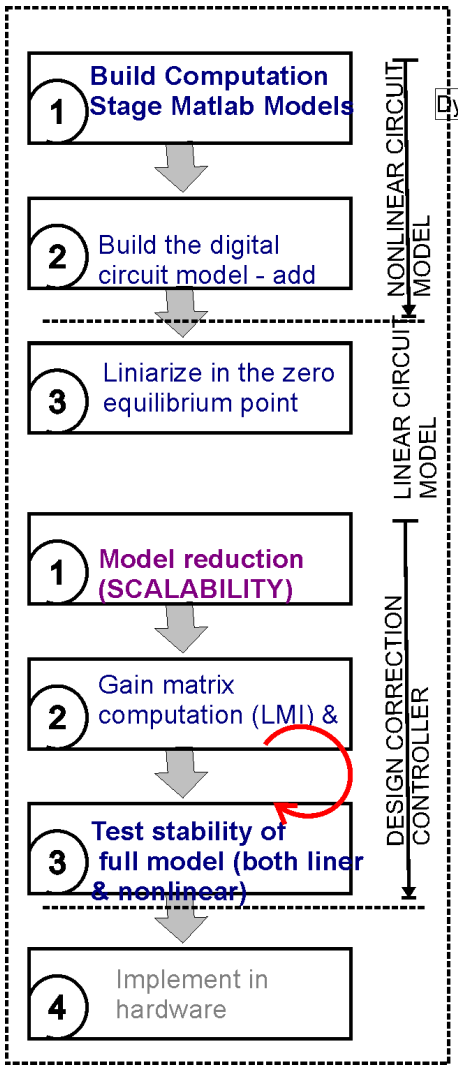
Correction Controller Design Steps: Dynamic Model



Requirements:

- Discrete time model: clock;
- General structure independent of the underlying processing stage;
- Fixed point representation;
- Registers may contain partial results (*e.g.* multiply accumulate), hence final results are available at well determined moments in time;

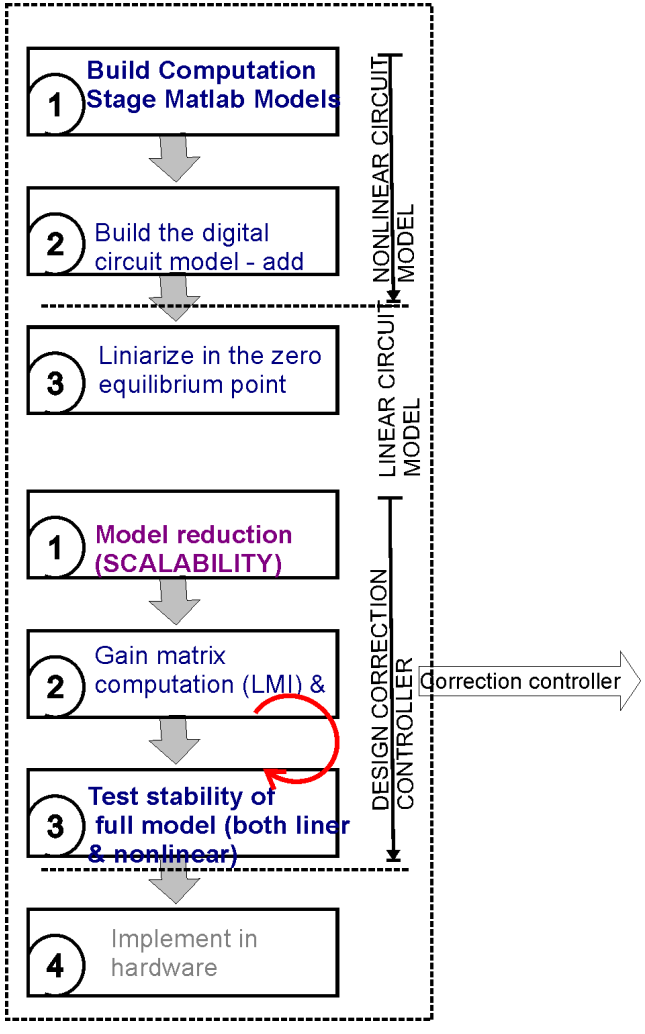
Correction Controller Design Steps: Dynamic Model



Steps for obtainig the dynamic model of a circuit processing stage:

- describe RTL operations for computing an output data element
- Assign model state variables to:
  - partial register computations
  - output registers
- Rewrite as:
  $x(k+1) = Q\kappa(x(k)) + Ax(k) + a$,
  a -input vector, A - additions, Q - multiplications
- build the overall circuit model

Correction Controller Design Steps: Correction controller Gain matrix



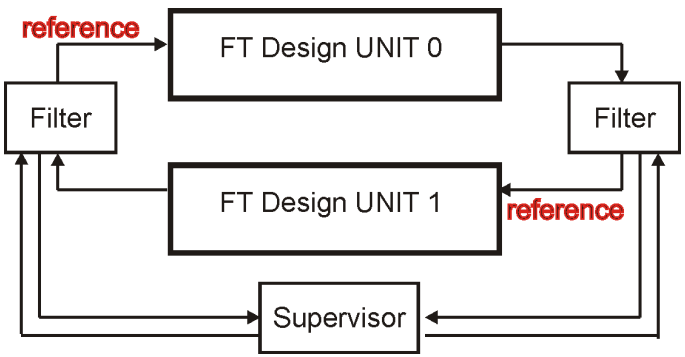
FT circuit model:

- Dynamic model with correction controller control: $x(k+1) = Q\kappa(x(k)) + Ax(k) + a + Bu^c(k)$
- B selects the correction offset for each register
- $u^c(k) = -K(x(k) - x^d)$, K gain matrix, $x(k) - x^d$ difference between the reference and the circuit output
- gain matrix – solve the linear matrix inequality conditions for the reduced size model

Outline

The Reference problem

- The considered approach for designing a control feedback loop assumes the existence of a golden reference;
- Proposed approach: duplicate the FT design and connect the 2 instances in reaction:
  - Filter – performs a processing of the data outputs to serve as reference to limit potential errors to propagate between instances;
  - Supervisor – the control part deciding entering the error correction mode, and its termination;



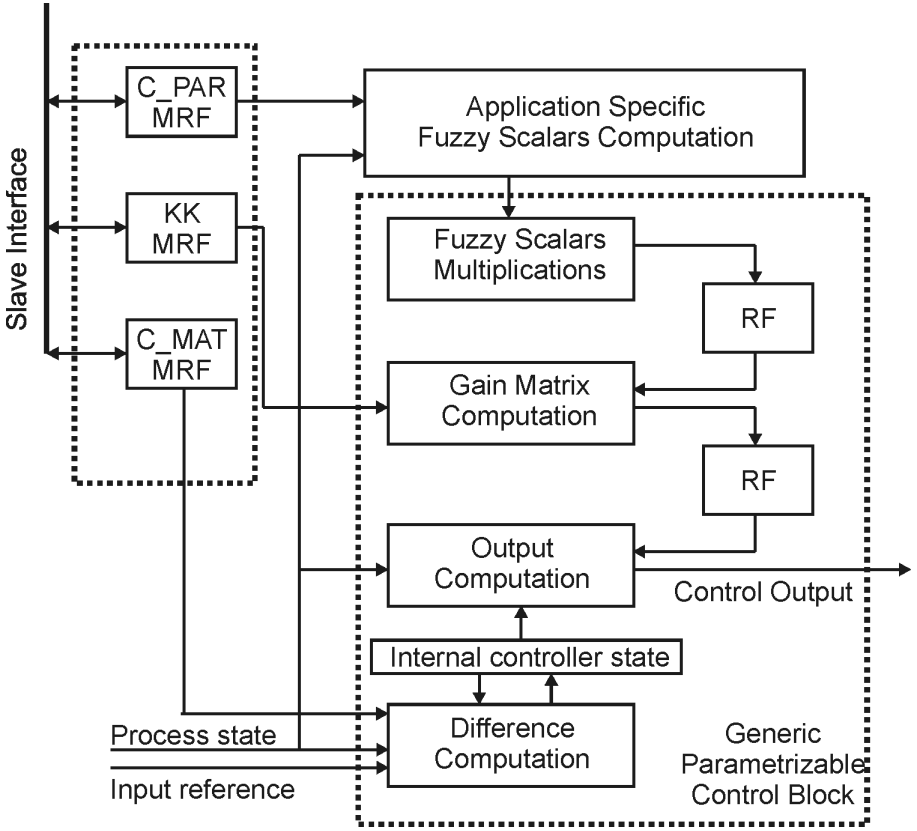
Outline

Use case - 2 DOF pipeline

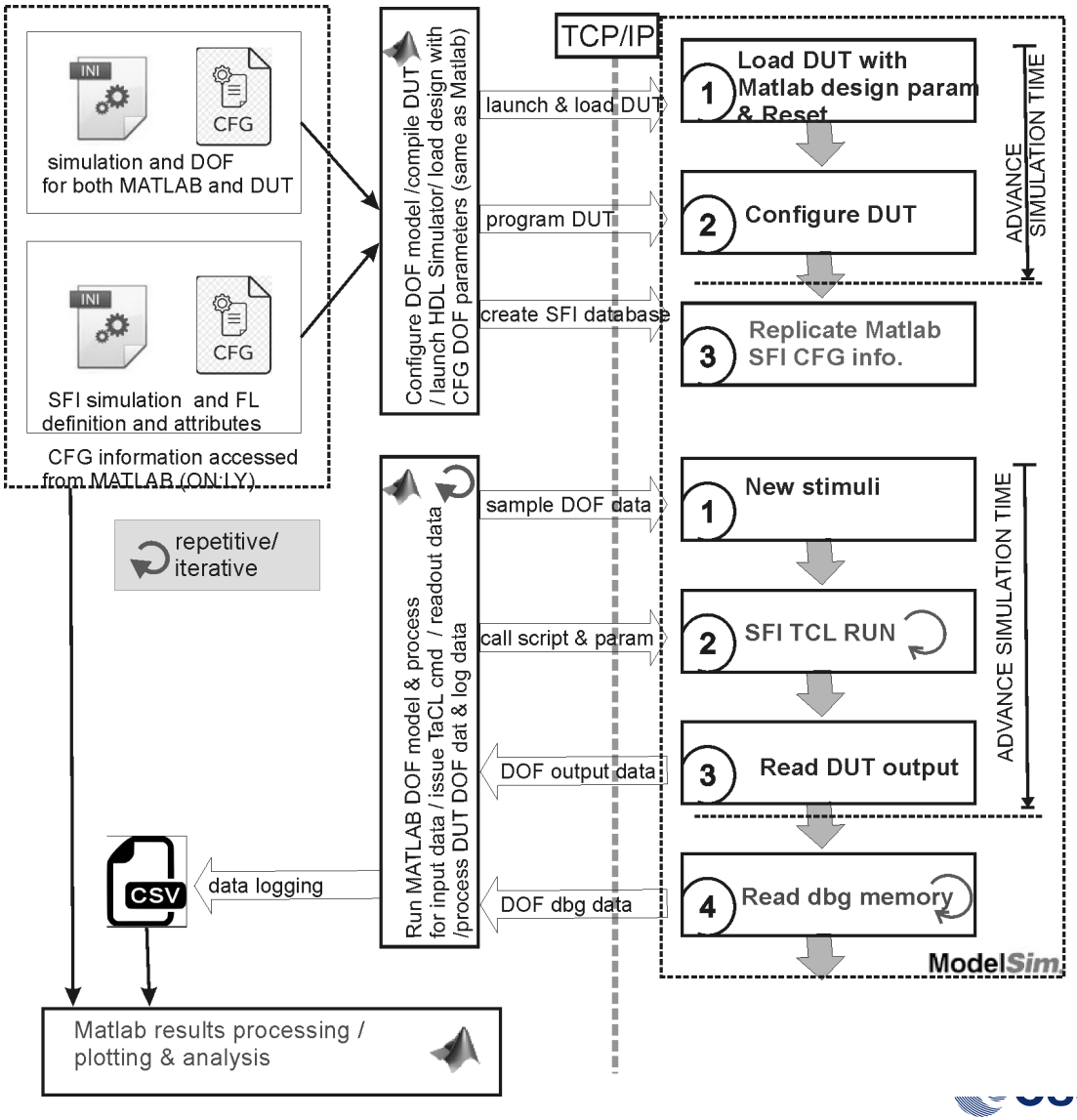
- Fuzzy controller for 2-DOF robot arm
- Based on measured angles and angular velocities - 4 value process state vector $x(k)$ -, we apply a set of 2 torques - 2 value control vector $u(k)$ - in order to follow a reference - 2 value vector y_r
- The 2 DOF employs the following computations:
  - Trigonometric function evaluation - based on 2nd and 3rd order Taylor series
  - Computation of fuzzy scalars - multiply-accumulate operations
  - Fuzzy scalars products
  - Gain matrix computation - accumulation of scalar-matrix multiplications
  - Control vector computation - matrix-vector multiplication
  - Difference computation
- Fixed point implementation - 24 bits (16 bits fractional part)

2 DOF - Pipeline

- Trigonometric functions approximation
- Fuzzy scalar computation
- Fuzzy scalar multiplication
- Final gain matrix computation
- Control output computation
- Difference computation



Matlab execution environment – verification & SFI



2 DOF - Implementation cost

- Implementation results - Virtex-7 FPGA
- Cost - 1948 slice, 34 DSP
- Number of registers in the processing datapath : 34, 24 bit

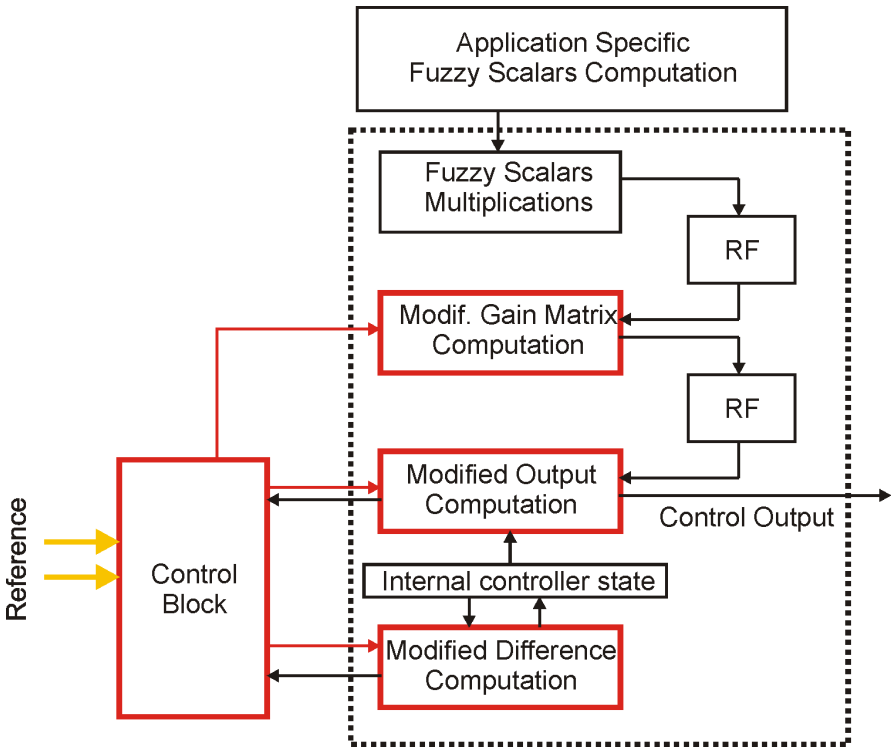
Fault Tolerant 2-DOF

- Applied the methodology
- Correction factors are applied to 3 out of 5 blocks
  - Final gain matrix computation
  - Control output computation
  - Difference computation
- Number of "corrected" registers: 8 (out of 34)
- Correction phase: latency (in clock cycles) for computing one element of the final gain matrix
- Correction gains are constants:

Register	Gains			
REG_OC_ACCUM	−0.0103	−0.0092	0.0000	0.0000
REG_OC_PROD	−0.0017	−0.0022	0.0000	0.0000

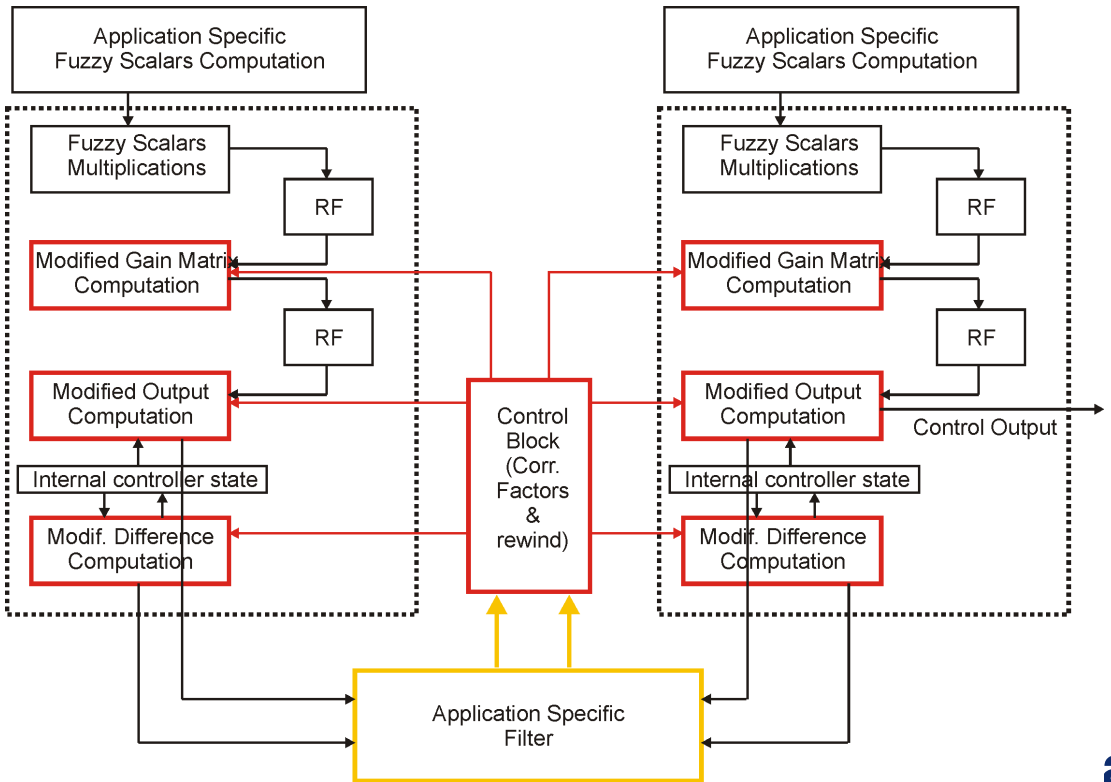
Fault Tolerant 2-DOF

- The original architecture requires minimum changes:
  - Control signal generation redesigned - for rewind
  - Modified data-path for adding the correction offset factors
- Control block:
  - Difference between reference and output value
  - Multiplication with correction gains



Fault Tolerant 2-DOF - with reference

- Duplication of the FT 2-DOF design and connect the 2 instances in reaction;
- Application specific Filter – range validation filter that does a check that outputs are in the desired value range

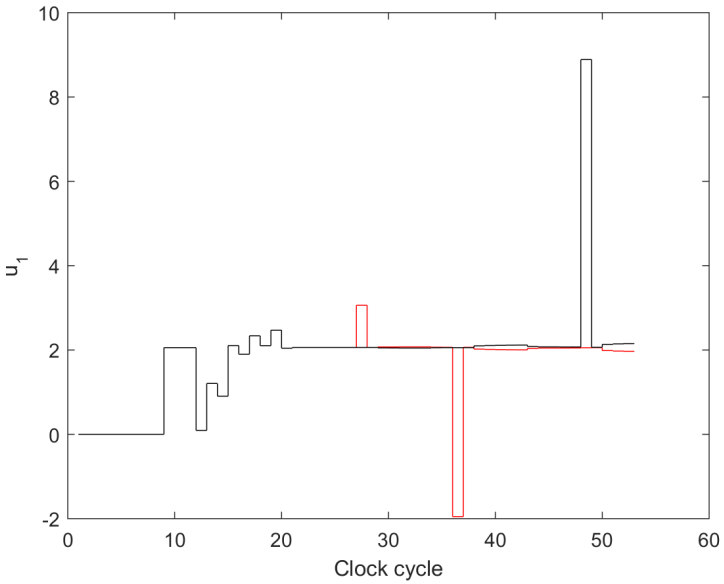


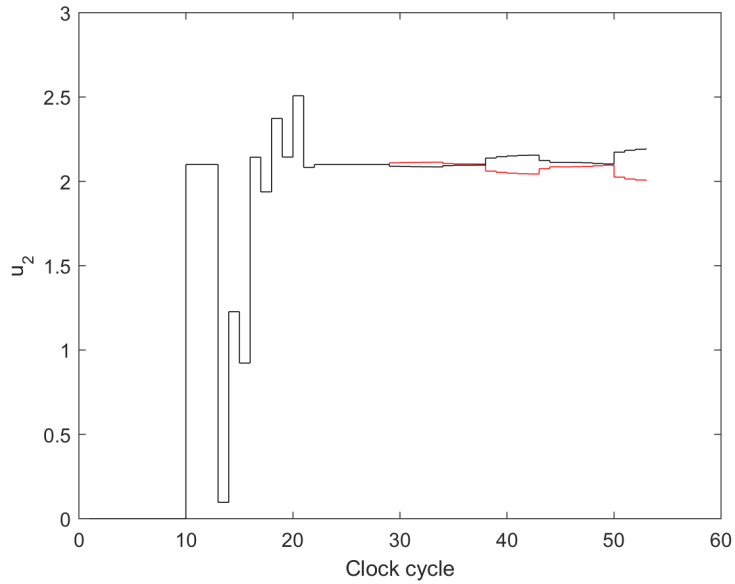
Reliability analysis

- Fault probability 10^{-5}
- TMR and the 2-DOF with correction controller (CC)
- Output error gaussian distributions

Method	μ	Σ
CC	0.000841282044353	0.002051307070919
TMR	0.010365724617813	3.952222463580792

Reliability analysis - simulation





Implementation Results

- The feedback correction control loop adds and overhead of around 1.3x, while the circuit duplication solution has an overhead of 2.7x with respect to the baseline circuit
  - Multiplication with the correction constants is performed using conventional multiplication – in FPGA implemented with DSP blocks –, and not optimized constant based multiplication

Design	Slices	Slice LUT	Slice Registers	DSP Blocks	Frequency [MHz]
Baseline	1948	3033	5181	34	142
FT with gold reference	2257	3782	5290	46	100
FT	3465	6637	7265	92	100

Outline

Conclusions & Future work

- We have analysed and validated the control based approach for improved fault tolerance of datapath processing pipelines:
  - Correction controller design
  - The operation of the correction process
  - Proposed a solution for the reference problem
- Future work:
  - Applications: - image processing pipeline
  - Reference problem - using reduced precision replicas

Thank you!