AMS Timing Closure with Coside

Using timing-aware models and static analysis techniques

Guillermo Conde, Infineon Technologies Austria
09.11.2021

Table of contents

Table of contents

Introduction

Timing Closure

› process by which a logic design (sequential + combinatorial gates) is modified to meet its timing requirements.

Semi-custom designs using STA

› Timing requirements are translated into static timing constraints to the EDA tool.

› Models including accurate characterization

Timing Closure in AMS systems

› No timing requirements in place.

› Models do not include any timing characterization.

› Timing issues detected very late in development

Goals

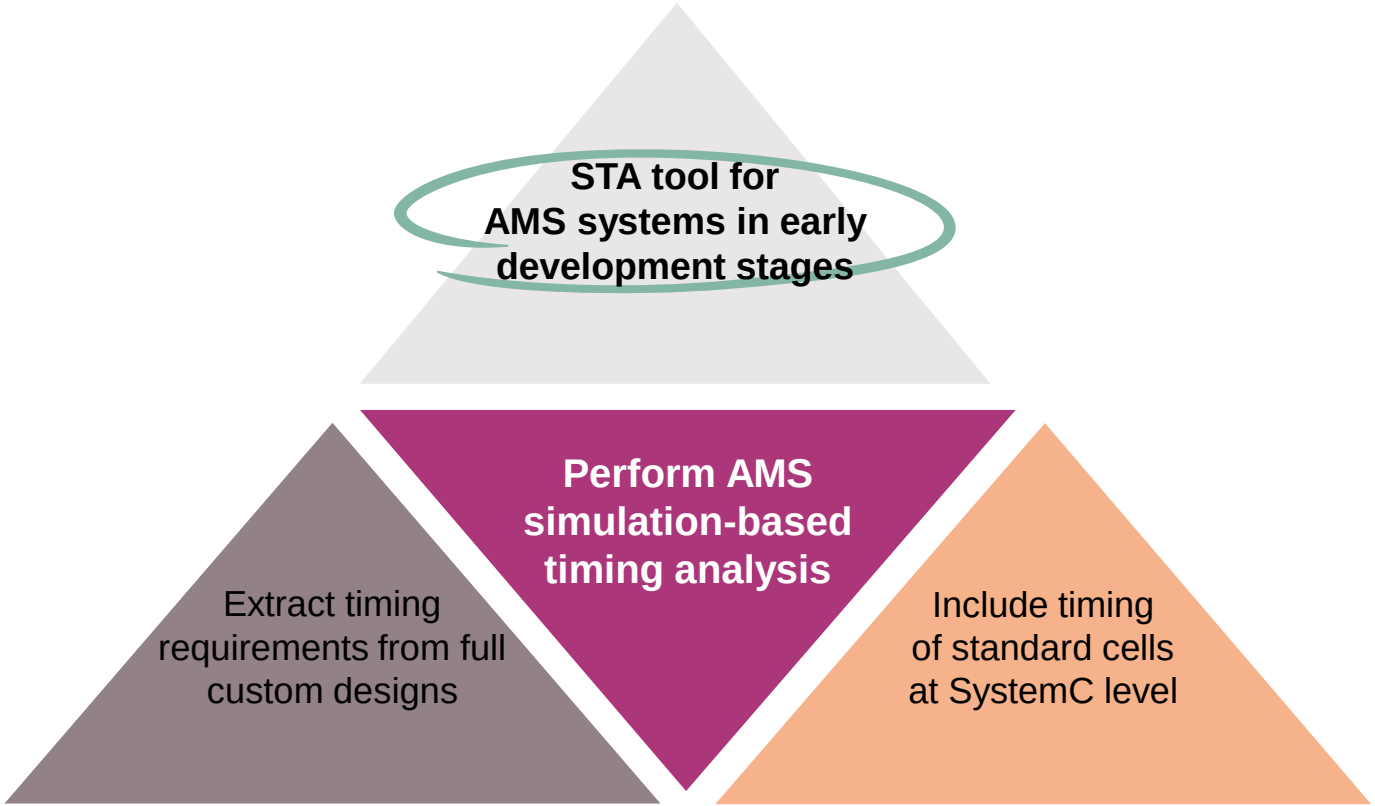
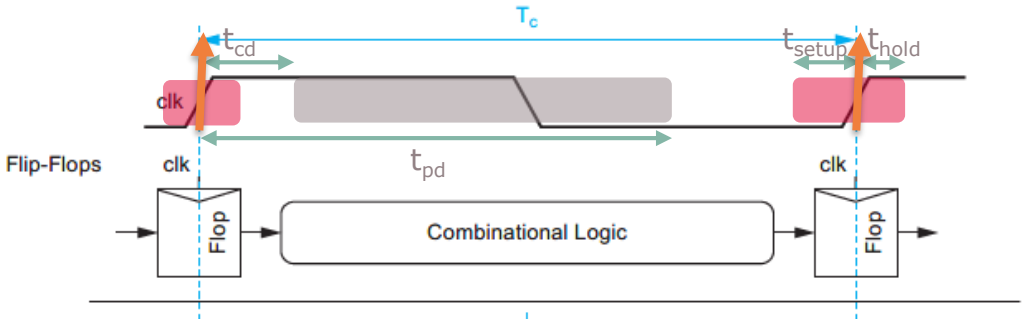


Table of contents

Timing basics in digital systems

- Most common sequencing elements are latches and flip-flops



- Basic timing contraints and delays
- Worst case scenario known including PVT
- Systematic method to guarantee proper timing (STA)
- Systems insensitive to PVT variations once timing is guaranteed

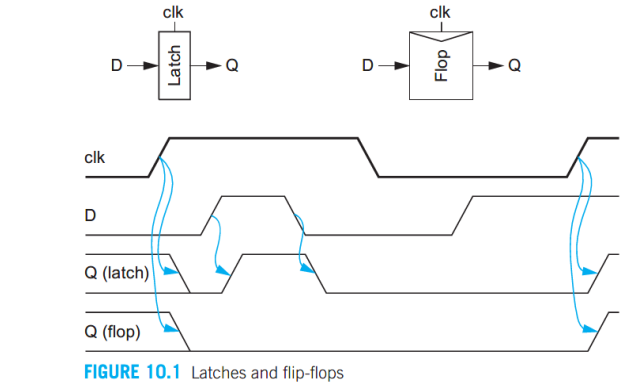
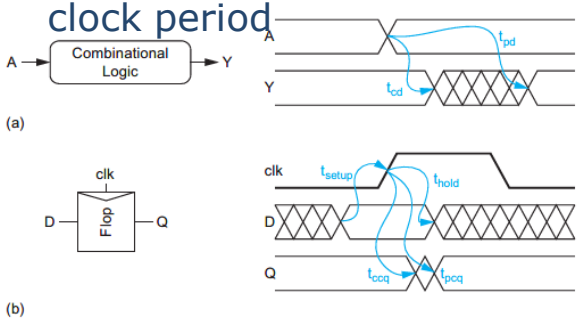


FIGURE 10.1 Latches and flip-flops

- Digital flow rules: Synchronous systems based on FFs and constant clock period



Semi-custom design flow

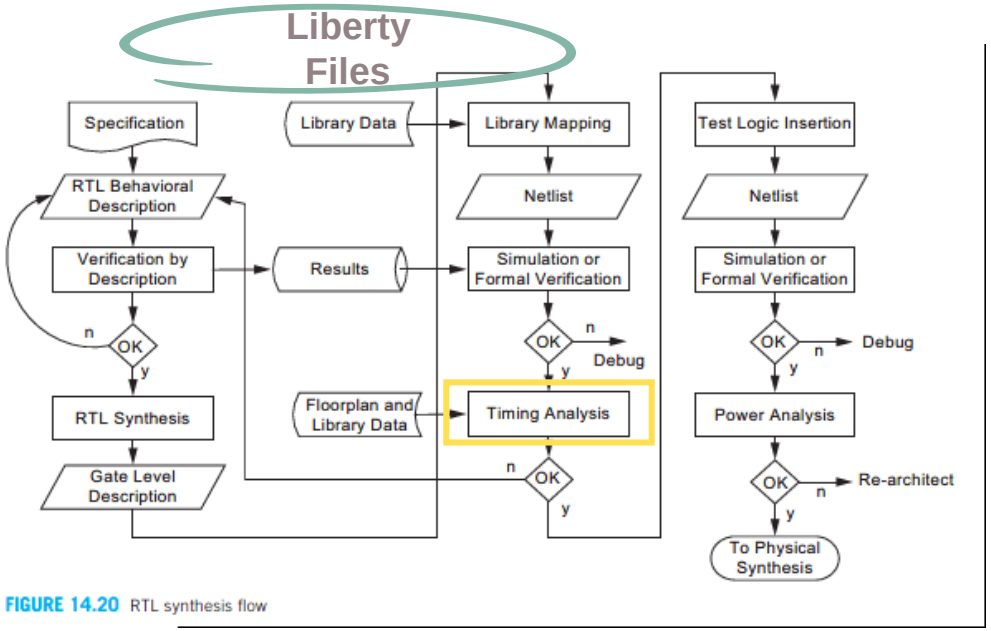


FIGURE 14.20 RTL synthesis flow

Source: CMOS VLSI Design, Neil H.E Weste

STA evaluates automatically all timing paths

Intrinsic gate delay derived from the library

Loads are either estimated statistically or derived from the floorplanning.

Table of contents

Full-custom design flow

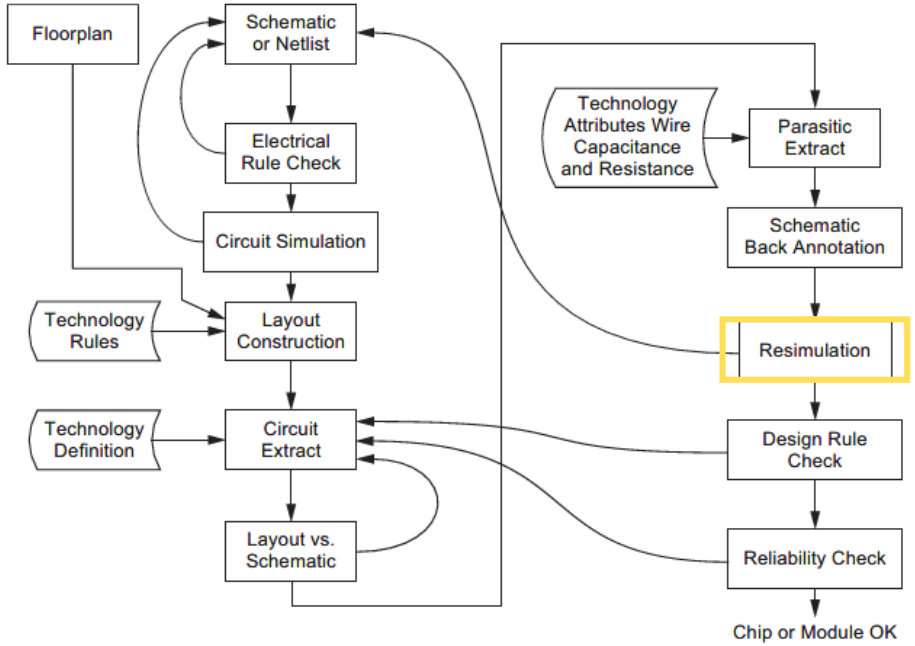


FIGURE 14.25 Mixed-signal or custom-design flow

Source: CMOS VLSI Design, Neil H.E Weste

Extracted simulation at system level late in the development phase

No timing analysis tools available

Potential failing scenarios have to be identified in advanced

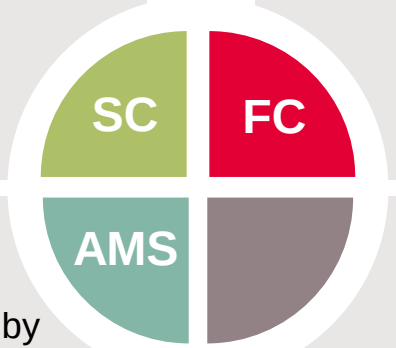
AMS system components

Semi-custom logic (digital part)

- Logic gates
- Sequential gates
  - Flip-flops

Full-custom Logic (in the analog part)

- Logic gates
- Sequential gates
  - Flip-flops
  - Latches
  - Special cells



AMS System

- Analog part
  - Full-custom logic: Timing provided by Liberty files
  - Analog Blocks: Timing has to be characterized
- Digital Part: Timing closure determined by STA

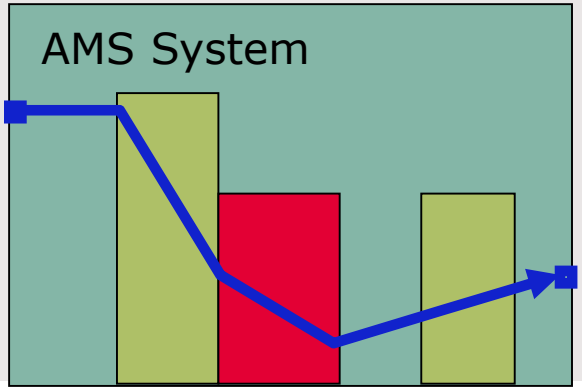


Table of contents

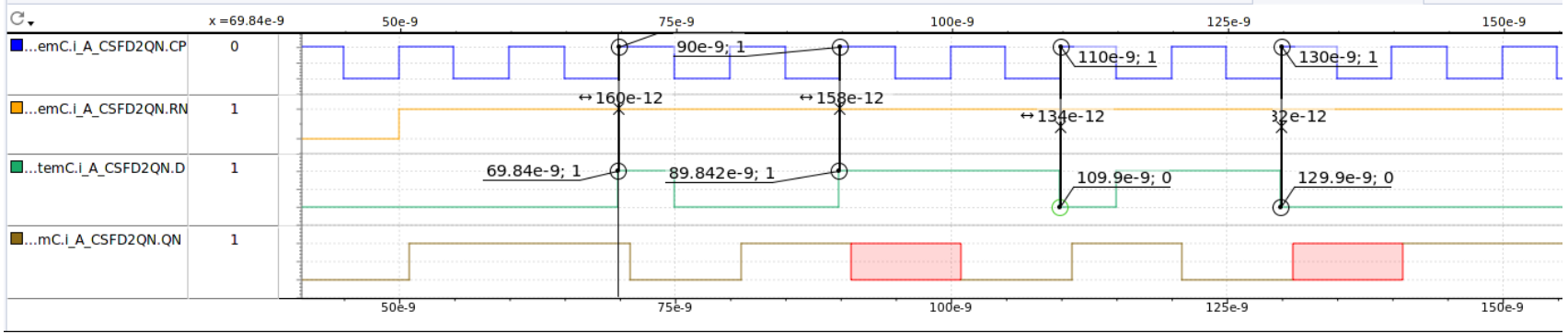
Timing-aware models

- System C models with 4-value data types
- Delay included
- Time stamps taken at every event
- Automatic checks on
  - Setup and hold time
  - Reset recovery and removal
  - Minimum Pulse Width

Delay Path [ps]	Load Capacitance [fF]				
	5	10	25	50	250
CP ⇑ ⇒ QN ⇓	403	422	473	552	1.18e+03
CP ⇑ ⇒ QN ⇑	443	463	519	613	1.35e+03
RN ⇓ ⇒ QN ⇑	266	286	343	436	1.18e+03

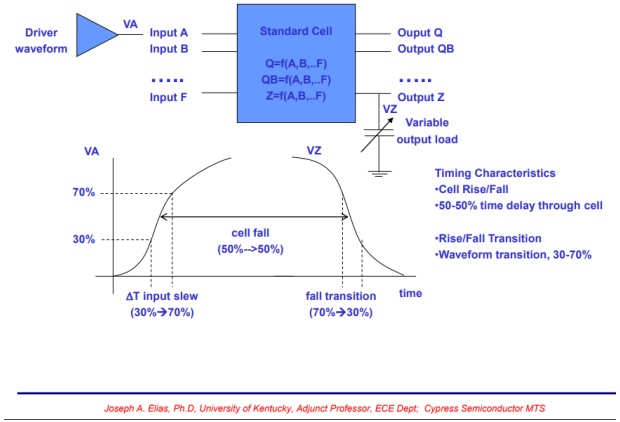
Check	Constraint [ps]
	typ
D ⇓ setup CP ⇑	133
D ⇓ hold CP ⇑	14.9
D ⇑ setup CP ⇑	159
D ⇑ hold CP ⇑	24.2
RN ⇑ recovery CP ⇑	-261
RN ⇑ removal CP ⇑	263

MPW	value [ps]
CP (L)	173
CP (H)	156
RN (L)	274



Liberty Files

- Open Source ASCII format to specify:
  - PVT Characterization
  - Area
  - **Timing**
  - Power
  - Noise
- Cell delays/constraints depending on:
  - Input slew
  - Output load

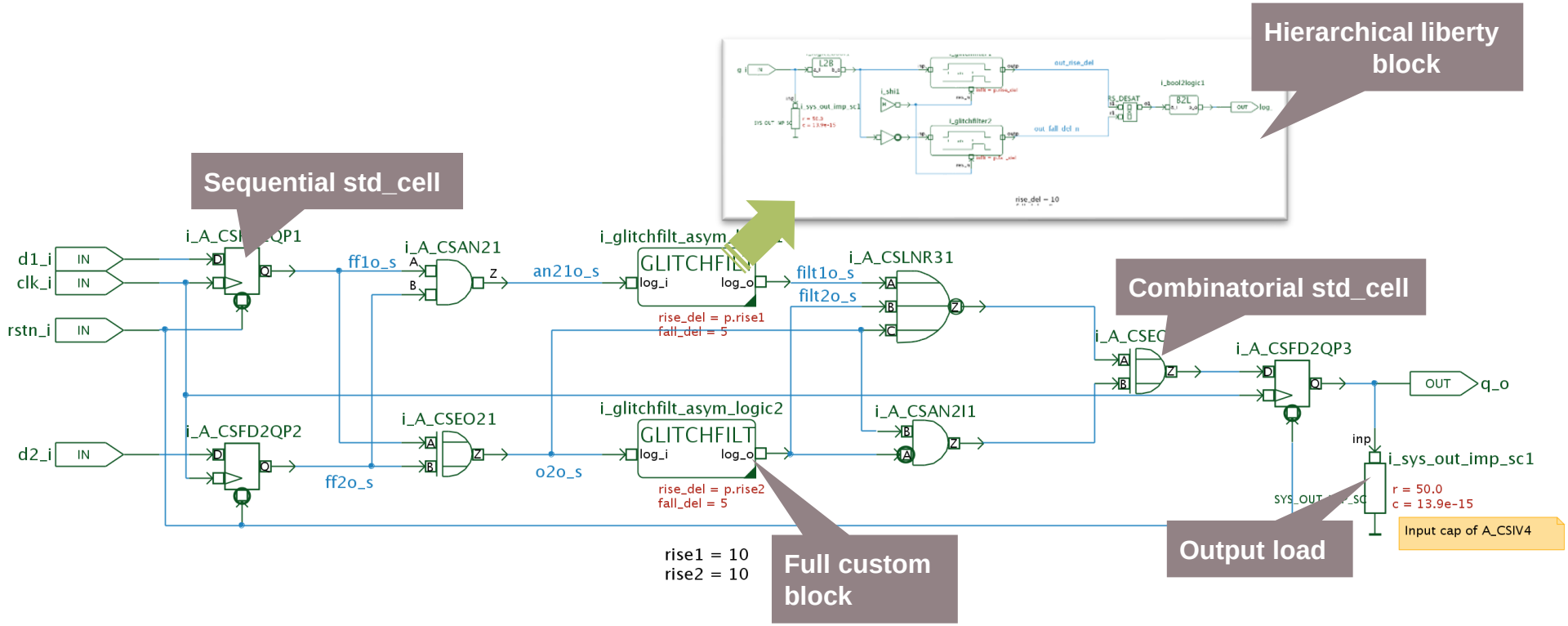



```
pin(QN) {
        direction           : output;
        capacitance         : 0;
        max_capacitance     : 0.256;
        min_capacitance     : 0.001;
        related_ground_pin  : VSS;
        related_power_pin   : VDD;
        function            : "IQN";
        power_down_function : "!VDD + VSS";
        /* characterization range(QN) = [1E-15, 256E-15]F */
        /* characterization range(CP) = [5e-11, 1.2e-08]s */
        /* characterization range(RN) = [5e-11, 1.2e-08]s */
        timing() {
                related_pin         : "CP";
                timing_type         : "rising_edge";
                cell_fall(slp_load) {
                        index_1("  0.001000  ,  0.002000  ,  0.004000  ,  0.008000  ,  0.016000  ,  0.032000  ,  0.064000  ,  0.128000  ,  0.256000  ");
                        index_2("  0.001000  ,  0.002000  ,  0.004000  ,  0.008000  ,  0.016000  ,  0.032000  ,  0.064000  ,  0.128000  ,  0.256000  ");
                        values( "  0.384626  ,  0.389638  ,  0.398695  ,  0.414765  ,  0.443268  ,  0.495594  ,  0.596684  ,  0.797313  ,  1.198060  ", \
                                "  0.398938  ,  0.403951  ,  0.413006  ,  0.429078  ,  0.457581  ,  0.509906  ,  0.610997  ,  0.811625  ,  1.212370  ", \
                                "  0.419452  ,  0.424466  ,  0.433522  ,  0.449594  ,  0.478096  ,  0.530421  ,  0.631512  ,  0.832140  ,  1.23288  ", \
                                "  0.448812  ,  0.453825  ,  0.462881  ,  0.478952  ,  0.507455  ,  0.559781  ,  0.660873  ,  0.861499  ,  1.26226  ", \
                                "  0.501400  ,  0.506413  ,  0.515470  ,  0.531541  ,  0.560044  ,  0.612372  ,  0.713465  ,  0.914095  ,  1.31484  ", \
                                "  0.549201  ,  0.554214  ,  0.563274  ,  0.579345  ,  0.607850  ,  0.660180  ,  0.761274  ,  0.961908  ,  1.36264  ", \
                                "  0.596858  ,  0.601872  ,  0.610933  ,  0.627006  ,  0.655513  ,  0.707849  ,  0.808947  ,  1.00958  ,  1.41034  ", \
                                "  0.622917  ,  0.627935  ,  0.636996  ,  0.653072  ,  0.681586  ,  0.733927  ,  0.835032  ,  1.03567  ,  1.43642  ", \
                                "  0.621233  ,  0.626252  ,  0.635313  ,  0.651391  ,  0.679906  ,  0.732251  ,  0.833359  ,  1.034  ,  1.43474  ");
                }
```

Delay Data

Delay Path [ps]	Load Capacitance [fF]				
	5	10	25	50	250
CP ⇑ ⇒ QN ⇓	403	422	473	552	1.18e+03
CP ⇑ ⇒ QN ⇑	443	463	519	613	1.35e+03
RN ⇓ ⇒ QN ⇑	266	286	343	436	1.18e+03

Liberty blocks



Liberty flowchart

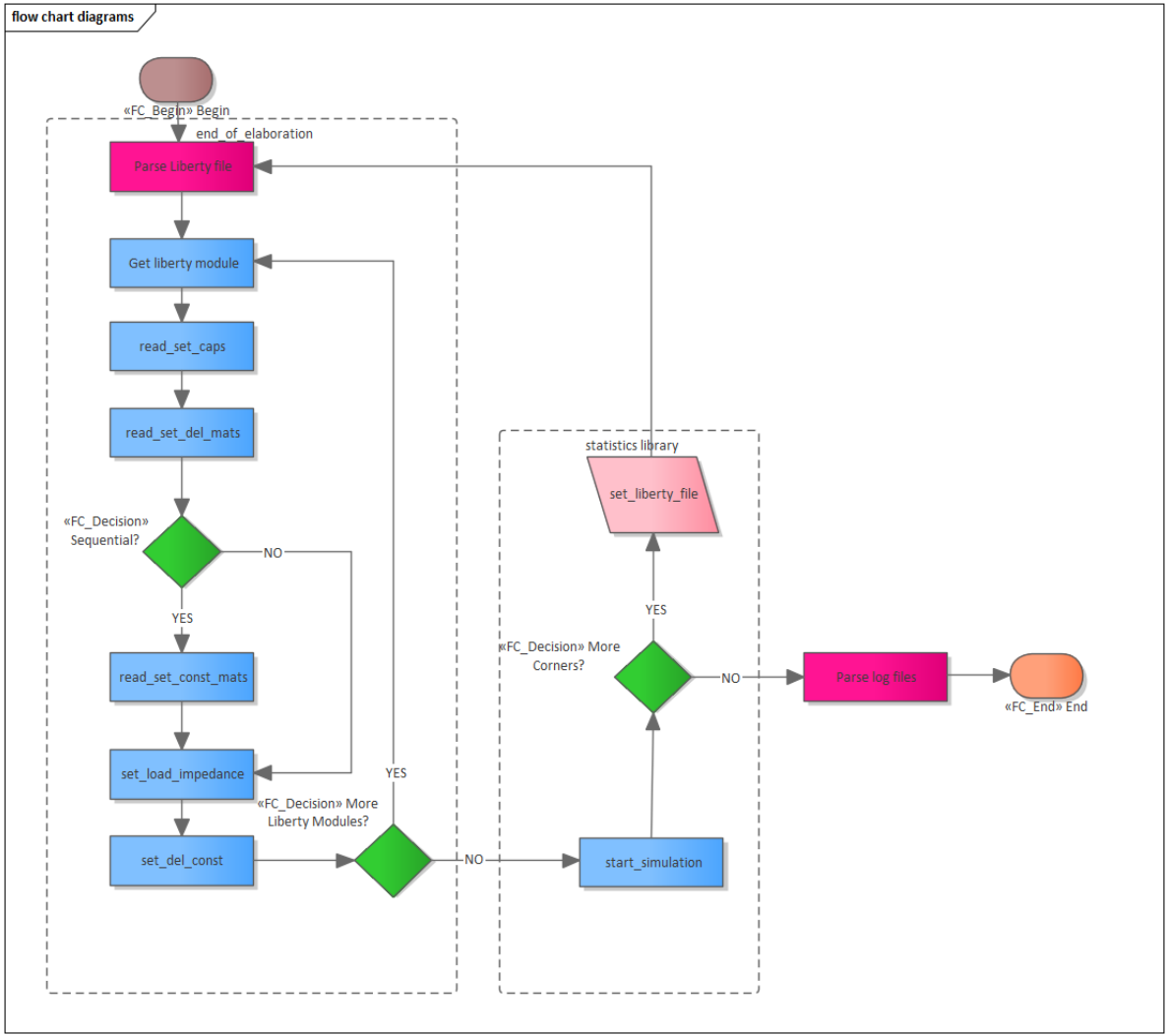


Table of contents

Achievements

Automatic timing calculation

› Assignment of delay and constraints directly from Liberty files into the Liberty blocks

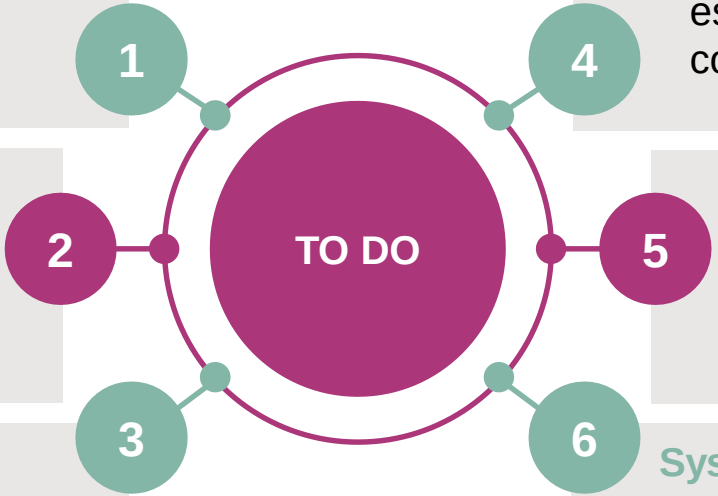
AMS timing-aware simulations

› Automatic detection of scenarios with timing violations

Multi-corner simulations

› Timing-aware simulations over corners via statistical library

Future Work



Formal Analysis
› Analytical approach to achieve timing closure

Static Analysis
› Graphical representation of timing parameters

Area estimation
› Use of static analysis for area estimation

Power analysis
› Static anlysis to estimate power consumption

Random constraint stimuli
› Increase coverage by applying UVM techniques

SystemC assertions
› Alternative way of performing automatic checks

Part of your life. Part of tomorrow.