ScuDo
Scuola di Dottorato ~ Doctoral School
WHAT YOU ARE, TAKES YOU FAR

36th Cycle

An optimization flow for high-performance designs with single-clock mixed-polarity latches and flip-flops

Filippo Minnella

Supervisor: Prof. Luciano Lavagno

Research context and motivation

- Sequential circuits use Flip-Flops or Latches for data storage. Latches can be used in error-resilient applications, work at lower supply voltages, reduce power consumption and can increase operating frequency;
- The actual techniques to transform Flip-Flop based circuits in Latch based ones are mainly characterized by pulsed latches, which use small duty cycles produced with specific cells, or by different clocks, that differ in terms of duty cycle and phase;
- The pulsed latches techniques rely on pulse generators that take as input the clock that should be fed to the sequential element and modify it to reduce the duty cycle;
- The multiple clock-based techniques need different clocks which must be generated and fed to the circuit;
- Both need specific hardware which causes area overhead and could be difficult to implement, especially in the case of pulsed latches;

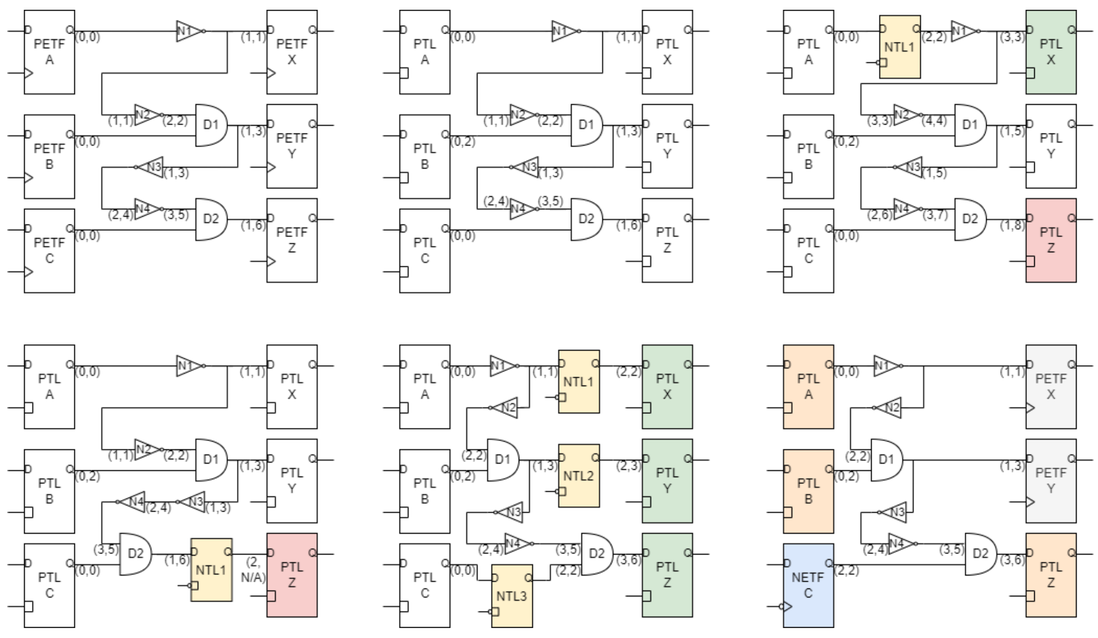
Addressed research questions/problems

Problems:
- Latch-based methods to increase operating frequency, such as time borrowing, rely on tight hold constraints that are difficult to satisfy using traditional backend optimizations, such as clock skew and retiming;
- The latch-based circuits use multiple clocks or pulse generators which cause area overhead and have a difficult implementation flow;
- The state-of-the-art approaches are evaluated only on a limited set of benchmarks running at a reduced set of frequencies;
- Most of the proposed techniques are evaluated only at synthesis level without analyzing the possible drawbacks of physical implementation;

Questions:
- Is it possible to obtain a single-clock methodology able to cope with the tight hold constraints generated by latch-based circuits?
- Is it possible to keep a large duty cycle in order to exploit the full potential of time borrowing, while increasing circuit performance?

Novel contributions

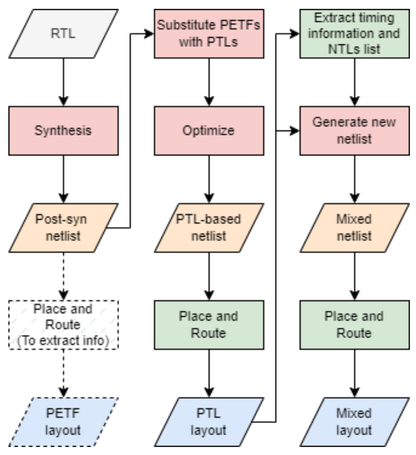
- **Mix & Latch** is a conversion algorithm for Flip-Flops based netlists that is able to **increase the operating frequency** of synchronous digital circuits, using **a single clock-tree and a mixed distribution of Positive-Negative Edge-Triggered Flops** (PETF/NETF) and **Positive-Negative Level-Transparent Latches** (PTL/NTL);
- The distribution of the sequential elements is obtained through the analysis of a PTL-based layout and of the critical paths, in terms of hold violations, and selecting the set of edges where to place the group of NTLs able to delay short paths. Given the position of the elements, it is possible to merge them into NETF/PETF.



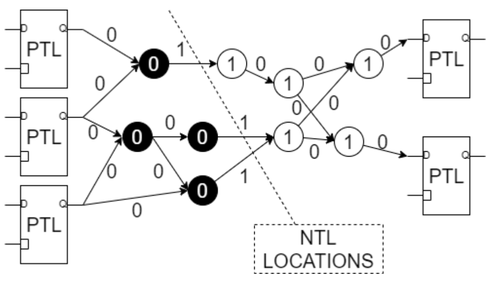
Submitted and published works

Adopted methodologies

- A two-step implementation flow is used to obtain a working layout for an optimized version of the flip-flop based netlist. This flow is fully based on commercial EDA tools and exploits useful skew both to obtain baseline circuits and to produce our own results.
To perform the experiments needed to validate this flow, we use Design Compiler from Synopsys for synthesis and Innovus from Cadence for placement and routing and to extract timing info.



- A graph-based data representation with several edge attributes is extracted from the PTL-based post-layout netlist to represent timing and positional data. The solution to the NTL allocation problem is the optimal set of edges which can produce a bipartition of the input circuit graph with respect to the quantity of sequential resources added to the circuit and the timing constraints.



- We use an ILP formulation with an area oriented objective function and timing-oriented constraints. The solution is the set of edges corresponding to the NTL allocation.
Post-layout timing data are used as input to the ILP model. Incremental placement and routing are performed to obtain the final layout.

Algorithm 2 Integer linear programming (ILP) Model

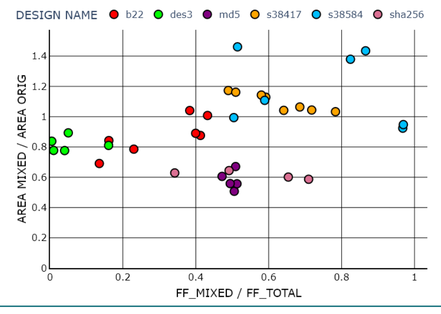
Inputs: TG, SPG, *PERC*, **Variable:** X

$E \leftarrow TG(E)$
$E_{\text{red}} \leftarrow SPG(E)$
$DE_p, D^{\text{ptl}}_p, D^{p}_{\text{ptl}} \leftarrow TG(ATTR)$
$SP_p \leftarrow SPG(ATTR)$

$$\text{minimize} \quad \sum_{\forall e \in E_{\text{red}}} f_g(e, SP_p, X) \tag{4}$$
$$\text{subject to} \quad \forall e \in E_{\text{red}}, f_g(e, DE_p, X) \geq 0 \tag{5}$$
$$\forall e \in E_{\text{red}}, f_g(e, D^{p}_{\text{ptl}}, X) \leq PERC \cdot T \tag{6}$$
$$\forall e \in E_{\text{red}}, f_g(e, D^{\text{ptl}}_{p}, X) \leq PERC \cdot T \tag{7}$$
$$\forall e \in E, R(e, X) \geq 0 \tag{8}$$

- A validation of our algorithm on circuits from different benchmarks (CEP, ISCAS89, ITC99) is used to evaluate the post-layout frequency improvement. To evaluate the performance improvement and the area overhead, we tested the algorithm on the same circuits at different frequencies.

DESIGN	$f_{\max}$ (GHz) ORIG	$f_{\max}$ (GHz) MIXED	ORIG *PETF*	MIXED *PETF*	MIXED *NETF*	MIXED *PTL*	MIXED *NTL*
aes_192	*[1]	0.33	9382	0	9153	229	530
b22	0.48	0.67	613	77	13	523	1206
des3	0.67	1.00	199	32	1	166	189
md5	0.44	0.67	269	71	61	137	519
s38584	0.48	0.53	1275	3	1240	32	548
s38417	0.48	0.67	1564	207	823	534	869
sha256	0.56	0.67	1040	502	284	254	579



Future work

- Extend the benchmarks in order to validate our techniques on other circuit topologies;
- Reformulate the ILP as max-flow min-cut problem in order to improve algorithm scalability with respect to the size of the circuits;
- Understand the performance of the algorithm against retiming methods offered by commercial tools or using our technique in addition to them;
- Fine-tuning algorithm hyperparameters to improve the results of the heuristics;
- Analyze the impact on the circuit power-consumption;
- Implement a modified version of the algorithm for FPGA platforms;
- Study the DFT methodologies that can be used for the newly generated circuits;

List of attended classes

- 01UJBRV – Adversarial training of neural networks (03/06/2021, 3 credits)
- 01UJRIU – Computing Paradigms for Error-Tolerant Applications (26/07/2021, 5 credits)
- 01QTEIU – Data mining concepts and algorithms (01/02/2021, 4 credits)
- 01TVUQW – Embedded Electronic Systems for AI/ML (05/02/2021, 6 credits)
- 01UNRRV – Entrepreneurship and start-up creation (02/07/2021, 8 credits)
- 03QTIIU – Mimetic Learning (22/12/2020, 4 credits)
- 01MNFIU – Parallel and distributed computing (26/07/2021, 5 credits)
- 01DCURV – Principle of digital image processing and technologies (22/07/2022, 5 credits)
- 01QORRV – Writing Scientific Papers in English (18/02/2021, 3 credits)

Politecnico di Torino
1859

PhD program in
Electrical, Electronics and Communications Engineering