UNIVERSIDADE FEDERAL DO RIO GRANDE DO SUL
INSTITUTO DE INFORMÁTICA
CURSO DE ENGENHARIA DE COMPUTAÇÃO

PEDRO HENRIQUE SALVADOR STEIN

Power Optimization Techniques for Advanced CPUs at Physical Implementation Level

Work presented in partial fulfillment of the requirements for the degree of Bachelor in Computer Engineering

Advisor: Prof. Dr. Paulo Butzen

Porto Alegre
2019

“You can't climb the ladder of success with your hands in your pockets”

— ARNOLD SCHWARZENEGGER

CONFIDENTIALITY DISCLAIMER

Considering that this graduation work is centered on the research developed at Arm® company during my double degree graduation internship in France, a confidentiality term of non-disclosure of information is stated on the next paragraph, honoring Arm® company interests and policies.

As Arm® company develops and ports Intellectual Property (IP) from its own and from third-parties, selling licenses to his customers, implies that most of the contained information in its products and related to its development and implementation is confidential and cannot be disclosed to who don't have the right of access. This page is dedicated as and will serve as a disclaimer relative to all the information contained in this work which had to be carefully selected and adapted to not infringe any of Arm® company confidentiality policies signed, protecting all its confidential data and any possibility of its undesired leakage. In this case, for this graduation work everything related to its subject that may contain any kind of information that cannot be disclosed and is specific or proprietary to the Arm® company products will be presented in the most generic possible way. In certain occasions some of the information might be even omitted. To summarize the real intent of this disclaimer page, the following guidelines were adopted in the development of this work to protect the interests described above.

- **Confidentiality Respected Guidelines:**

  ➔ NO project or product names

  ➔ NO raw figures, only relative ones

  ➔ NO architectural or internal details

  ➔ NO third-party information

  ➔ NO proprietary technological information

  ➔ NO internal "know-how" knowledge disclosure

ACKNOWLEDGEMENTS

First of all, I would like to thank Polytech Montpellier (Université Montpellier II) and UFRGS (Universidade Federal do Rio Grande do Sul) for the opportunity to pursuit my engineering double degree, making part of the BRAFITEC (BRAsil France Ingénieurs TEChnologie) exchange program, by the partnership between these two institutions, not forgetting to mention CAPES for the financing and distribution of the given scholarship.

Most important, I also would like to dedicate this work specially to my parents, as it would be impossible without them to fulfil these achievements, as they formed my character and values that molded me as the person that I am today.

All my friends and everybody else that supported this journey, in the pursuit of this goal, have as well my gratitude and appreciation. To finish, I want to acknowledge and thank Arm® company for the given opportunity and the possibility of making part of a great graduation internship, which certainly made me learn a lot through all its process and not evolve only as a person, but also as a human being.

Not forgetting to mention, I want to directly acknowledge and immensely thank my internship tutor Stephane Zonza. For all his support, care, patience, attention and tutorship offered during the period of my internship at Arm® company, as without his guidance, help and knowledge passed through, it wouldn't be possible to perform this work and had learned as much as I did during this period.

Also, I would like to thank greatly my advisor on the development of this work, Paulo Butzen, who helped me through all the process, giving me the proper guidance with a very good critical support, and always being available to advise me and answer all my questions related to the subject and proper work methodology.

RESUMO

Com o crescimento contínuo e pressão de mercado das exigências para sistemas embarcados, smartphones, tablets, microcontroladores e o recente ramo de Internet das Coisas (em inglês IoT), mais do que nunca o desenvolvimento de dispositivos de baixo consumo de energia tornaram-se obrigatórios e cruciais para que se obtenha sucesso na indústria de design de circuitos digitais. Em vista disso, ao desenvolver modernas e avançadas Unidades Centrais de Processamento (em inglês CPUs) que são capazes de atender a essas exigências da indústria e do mercado, consumo de energia é um dos parâmetros chave e mais críticos para que isso se torne uma possibilidade. Satisfazer requisitos de consumo de energia, frequência de operação e área de silício simultaneamente, é uma tarefa desafiadora que normalmente implica num compromisso de escolha entre a preservação de uma característica em detrimento de outra na concepção de Circuitos Integrados de Aplicação Específica (em inglês ASIC).

Este trabalho apresenta a aplicação de algumas técnicas de otimização de consumo de energia que podem ser adotadas no contexto de implementação física, com a ajuda de ferramentas avançadas de automação de projeto de circuitos digitais, em um CPU avançado para ser fisicamente implementado em tecnologia de processo de fabricação de 7 nm. A investigação e exploração das várias possibilidades e variação de parâmetros oferecidas por essas ferramentas podem levar a otimização em termos de Consumo de energia, Performance e Área (em inglês PPA), ou até para a descoberta de certas optimizações ou opções que não oferecem algum benefício, proporcionando apenas um alto aumento em tempo de processamento no fluxo de implementação. Características como o uso de diferentes opções de Tensão de Limiar de transistor (em inglês VT), de comprimento de canal a partir de múltiplas opções de células padrão (em inglês standard cells) e a opção de utilização de flip-flops de multíplos bits são exploradas nesse projeto de graduação. As técnicas referidas são avaliadas em termos de métricas, como consumo de energia, frequência de operação e área de silício, para diferentes casos de teste.

Palavras-chave: CPU. ASIC. Baixo Consumo de Energia. Otimização de Consumo de Energia. EDA. PPA. Fluxo de Implementação.

Power Optimization Techniques for Advanced CPUs at Physical Implementation Level

ABSTRACT

With the continuous growth and customer push to the requirements of embedded systems, smartphones, tablets, microcontrollers and the recent IoT (Internet of Things) market, more than ever the development of power efficient devices became a must, and highly crucial to achieve success in the digital design industry. Taking this into account, when developing modern advanced CPUs that are able to follow those industry requirements, power consumption is one of its key and most critical parameters. Meeting power consumption, operating frequency and silicon area, is a challenging task that usually implies in many trade-offs in the conception of an ASIC, as the strive for maximal power efficiency while offering good performance in small silicon area.

This work presents the application of some power optimization techniques that can be performed on the context of physical implementation level with the help of advanced EDA tools', in a modern advanced CPU design to be physically implemented in a 7nm process technology node. The investigation and exploration of the various possibilities and parameters variations offered by these tools can lead to PPA improvements, or even to the discovery of features or optimizations that doesn't offer any improvements at the expense of considerable increase in processing runtime in the implementation flow. Characteristics like the use of different VT and channel length from multiple standard cell technology options and a multi bit flip-flop merging feature are addressed in this practical research work. The referred techniques are evaluated in terms of the collection of metrics, such as, power consumption, operating frequency, and silicon area, for different test cases.

Keywords: CPU. ASIC. Low Power. Power Optimization. EDA. PPA. Implementation Flow.

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS AND ACRONYMS

ASIC	Application Specific Integrated Circuit
CISC	Complex Instruction Set Computer
CMOS	Complementary Metal Oxide Semiconductor
CPU	Central Processing Unit
CTS	Clock Tree Synthesis
DFF	D-type Flip Flop
DRC	Design Rules Check
EDA	Electronic Design Automation
FinFET	Fin Field Effect Transistor
GLS	Gate Level Simulation
GLS	Gate Level Simulation
HVT	High Voltage Threshold
IoT	Internet of Things
IP	Intellectual Property
LVS	Layout Versus Schematic
LVT	Low Voltage Threshold
MIPS	Microprocessor without Interlocked Pipelined Stages
MOSFET	Metal-Oxide Semiconductor Field-Effect Transistor
PPA	Power, Performance and Area
QoR	Quality of Results
RISC	Reduced Instruction Set Computer
RTL	Register Transfer Level
SoC	System On Chip
STA	Static Timing Analysis
SVT	Standard Voltage Threshold
UFRGS	Universidade Federal do Rio Grande do Sul
VLSI	Very Large Scale Integration
VT	Voltage Threshold

CONTENTS

1 INTRODUCTION

With the continuous growth and customer push to the requirements of embedded systems, like smartphones, tablets, microcontrollers and the recent IoT market, more than ever the development of power efficient devices became a must, and highly crucial to achieve success in this industry. Arm® CPUs, for example, which are known for adopting a RISC (Reduced Instruction Set Computer) architecture approach when being designed, already strive for maximal power efficiency while offering good performance in small silicon area. However, the current market standards and customer expectations for those devices are so high nowadays that the constant research and seek for further improvements in those characteristics became mandatory, as even when achieved by a small margin, can impact in a big difference on the final product.

Taking this into account, when developing modern advanced CPUs that are able to follow those industry requirements and exigences, power consumption is one of its key and most critical parameters. Meeting power consumption, operating frequency and silicon area, is a challenging task that usually implies in many trade-offs in the conception of an ASIC (Application-Specific Integrated Circuit).

When designers describe an architecture at RTL (Register Transfer Level), they are already concerned somehow with the impacts in power consumption and performance, intrinsically coming from the way they design and write its code. However, there are more complex and specific effects in those metrics generated by the way the physical implementation flow for a design is executed and on the manufacturing of a chip. Effects that can be linked to the specificity of a foundry technology and the algorithms used by the EDA tools in the different steps of this process.

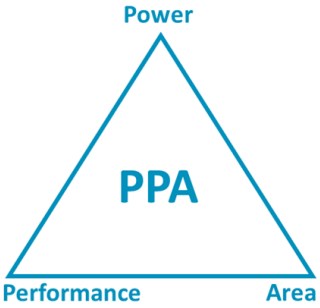
During the physical implementation flow of an ASIC, the current very advanced EDA (Electronic Design Automation) tools available in the industry can offer various possibilities and optimization features that can impact directly in terms of power consumption, performance and area in the chips willing to be conceived. Thus, the investigation of power optimization techniques by the different possible parameter variations and modifications that can be applied in the physical implementation flow, is considered a vital step to achieve excellence in this domain.

To pursuit these ideals, the subject of this work was dimensioned as the research and seek for improvements of the referred metrics on the described context above. This work was

developed as the research realized during the period of my double degree graduation internship at the Arm® France office, located in the Sophia Antipolis technology park.

Historically what is known by Power, Performance and Area (PPA), turned to be the main variables looked after on chip design. Meeting power consumption, operating frequency and silicon area, usually is a challenging task that implies in many trade-offs. The constant investigation of power optimization techniques that can be applied during the physical implementation flow of an ASIC, by the various possibilities offered by advanced EDA tools, is a vital step to achieve excellence in this domain. The following figure represents the balance that the term PPA stands for:

Figure 1.1: Power, Performance and Area (PPA)



Source: The Author

The main goal of the work and research realized can be summarized on the following scenario:

Starting from an modern and current Arm® fixed CPU architecture using an standard physical implementation flow, in a 7-nanometer process technology node, the investigation of different power optimization techniques that can be applied along with the help of advanced EDA tools, while meeting requirements in frequency and area, deciding which techniques can deliver the best results in a specific context.

The contributions and results from the research performed in this context have the intent to the constant optimization and better efficiency of the physical implementation flow of an ASIC, in order to collect the best possible PPA figures to be showcased to customers on the market when trying to sell an IP.

Power optimization techniques based in the use of standard cells with multiplate options for VT (Voltage Threshold) and channel length, as the use of a multi bit flip flop merging feature and dynamic power reduction feature after routing are discussed in this work.

This work is structured in a way that initially the main ideas of the subject will be presented. On the following chapters, the ideas and aspects that englobes the subject are present in theoretical form. Starting from a quick overview about the generic ASIC flow in the digital circuit design industry, technology and power consumption concepts, to the presentation of the power optimization techniques applied on the practical research work. On the chapter three the work methodology applied during the research along with the metrics used to evaluate the results are also presented. Finally, the collected results are presented along with its conclusions. At the very end, an appendix contemplates Arm® company, with special acknowledgements and a short description.

2 BACKGROUND

In this chapter theoretical aspects about the ASIC physical implementation flow will be presented generically, with the concepts of front-end and back-end and its different steps. In addition, transistor technology aspects and power consumption concepts will be presented to the reader.

2.1 ASIC IMPLEMENTATION FLOW

In the industry of digital circuit design, the road to the designing of an ASIC (Application Specific Integrated Circuit) to its manufacturing nowadays became a very solid and mature process, which can be represented as cycle composed of various steps that are somehow standardized in their ideas (WESTE, 2010). Starting from an initial product specification, to a high-level abstraction hardware description at RTL (Register-Transfer Level) and ending as a physical implemented chip. EDA Tools can make this whole process very efficient and convenient from an automation and versatility point of view, as every step can be written in the form of a script.

For the implementation of an ASIC, usually the process is devised in two groups, usually named as Front-End and Back-End and can generically described as following:

- **Front-End:**

Contains the steps of what is known as synthesis, which can be described as the transformation or conversion of a high-level abstraction hardware description at RTL (usually written in Verilog language), into a gate-level description, outputting a netlist, which contains all components and its interconnections (WESTE, 2010). This is achieved using an EDA Synthesis tool.

The synthesis is usually divided in three steps:

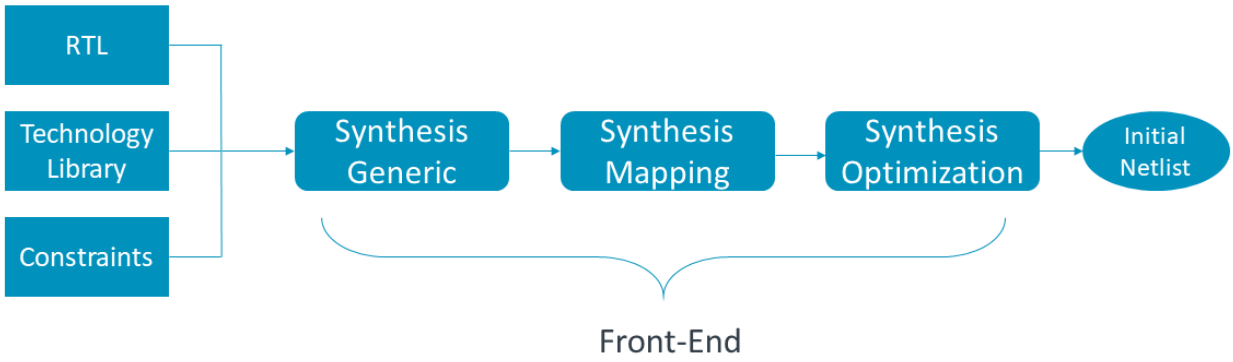
  - **Generic Synthesis:** Transforms a RTL description to a gate-level description, outputting a netlist using generic standard cells, from given constraints and performing some structural optimizations.

- **Synthesis Mapping:** From the netlist of the previous step, maps the generic standard cells to a specific real foundry technology of standard cells. From the technology library, includes information like cell size, timing and power consumption. This mapping usually is realized along with optimizations to recover power and area, while maintaining timing.

- **Synthesis Optimization:** Performs further optimizations at gate-level to improve timing on critical paths and recover area in less critical ones, serving as the final netlist to be used in the Back-End.

The following image generically represents the ordered steps of the Front-End part of the flow:

Figure 2.1: Front-End steps



Source: The Author

- **Back-End:**

Once the Front-End is concluded, the Back-End part starts and receives as input an initially optimized gate-level netlist, with its standard cells mapped to a given technology. In this part, not only the standard cells need to be properly placed and routed in the design, but the clock tree synthesis has to be performed as well (WESTE, 2010). Along with a final optimization and check step known as sign-off. This is achieved using what is usually known as EDA Place and Route tool.

From this perspective, the Back-End part of the implementation flow is devised in the following steps, as generically described:

- **Placement:** In this step the standard cells are placed in the design with the help of EDA placement algorithms according to the tool used. Placement along with routing influences in optimal PPA, making this step of critical importance for best results. The standard cells can be placed in multiple locations of the die, while some of them might be reserved for the clock tree routing and for the power grid. In addition, EDA Place and Route tools usually enable the user to guide the placement process using as input a predefined floorplan, defining blocks organization and location in the die.

- **Clock Tree Synthesis:** The Clock Tree Synthesis (CTS) step can be described as the clock distribution to all its sequential elements in the design. As the clock signals in practice cannot arrive at the same time to every component, causing a phenomenon known as "clock skew", the main objective of this step is to generate the clock tree trying to minimize the skew as much as possible, as strategical buffer insertion can be used to correct the difference in arrival clock times. During the CTS step the clock tree is as well routed, before the routing of the rest of the design take place.

- **Post Clock Tree Synthesis:** Complimentary step for extra timing optimizations after the clock tree synthesis, specially hold fixing.

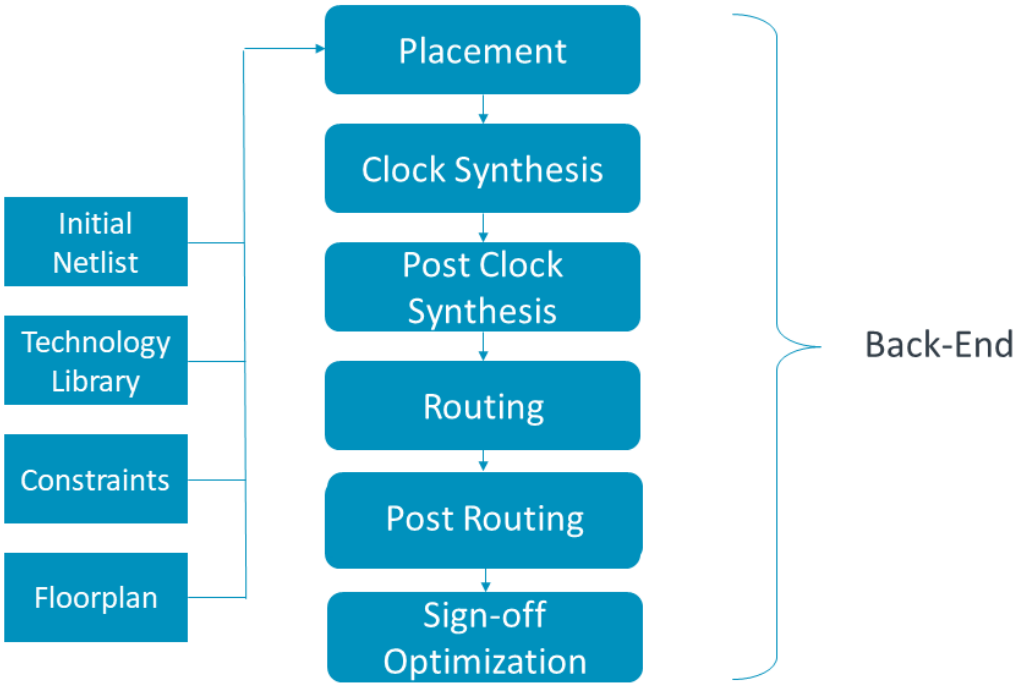
- **Routing:** After placement and clock tree synthesis steps had been finished, routing takes place to standard signals nets, when the tool applies its routing algorithms seeking for optimal routing, in terms of PPA preservation.

- **Post Routing:** Complimentary step for extra setup and hold timing optimizations after routing step. Usually the terms "Place & Route" or "PnR" are used in the industry to describe all the steps from Placement step to Post Routing step, including the Clock tree synthesis steps.

- **Signoff Optimization and Final Check:** Final optimizations for PPA improvements can be done in this step, being the main process performed what is known as STA (Static Timing Analysis) as setup time fixing and specific power recovery for dynamic and static power consumption, if available in the tool. In addition, checks like DRC (Design Rules Check), LVS (Layout Versus Schematic) and parasitic elements extraction (capacitance and resistance) can be performed as well.

The following image generically represents the ordered steps of the Back-End part of the flow:

Figure 2.2: Back-End steps



Source: The Author

2.2 TECHNOLOGY AND POWER CONSUMPTION

As previously stated, the semiconductor manufacturing process technology considered in this study for the given Arm® CPU architecture to be physically implemented is a 7 nanometer process technology node, that is based in the use of FinFET (Fin Field-Effect Transistor) model. After the process technology nodes had reached the nanometer scale, the continuous decrease in the channel length of the classical CMOS (Complementary Metal Oxide Semiconductor) planar transistor model commonly used in the industry, implied in the appearance of undesirable physical effects. Effects such as a considerable increase in the leakage current (not considerable before in the bigger technology nodes) and control loss or predictability in device switching behavior, lead to longer practical use of CMOS from a certain scale.

Using FinFET, in the other hand, enabled to avoid some of those undesirable effects due to its different 3D geometry and physical characteristics, generally offering as benefits in comparison to CMOS, less leakage current, a higher current drive for the same footprint (or per unit area) and the ability to operate at lower supply voltages. FinFETs present some disadvantages as well such as higher manufacturing cost and less versatility manipulating current drive. As equivalent to modifying channel width in CMOS to do so, current drive in FinFETs is manipulated with a quantized number of "fins".

When talking about PPA and specifically referring to power consumption in the digital circuit design industry, is important to be precise about the reason or origin of this consumption. Understanding the notion of dynamic and static power is crucial to more precisely evaluate the power consumption of a digital device and be able to make changes in the correct context when trying to achieve better power efficiency. The following concepts are already probably known by the reader, but its presentation is essential regarding the subject in study:

- **Dynamic Power Consumption:** Dynamic power consumption is referred to the situation when a transistor is in its switching stage, going from "on" to "off" or vice versa. Composed by a briefly short-circuit current formed between the supply voltage and ground while the gate is switching along with the charge and discharge of capacitive loads during this period (RABAEY, 1996).

- **Static Power Consumption:** The term static power consumption is often used to describe the situation when all the inputs of logic gates are held in some logic valid level and the circuit is not changing states. Classical CMOS logic gates are known for having low static power consumption for the reason of one the devices of the complementary pair be always in "off" state, due to the technology scaling there is the appearance of what is known as leakage current. Leakage current became a big concern as the scaling down of transistor size arrived around certain range in nanometers, being considered as of great negative impact in power efficiency and must be controlled (RABAEY, 1996).

2.3 POWER OPTIMIZATION TECHNIQUES

During the development of this research work there were conducted several testcases applying different modifications in a standard physical implementation flow in the trial to investigate techniques that could improve PPA, exploring the various capabilities offered by

the EDA tools used, evaluating its benefits and drawbacks. Different physical implementation flow configurations were executed.

In those lines, the following techniques or characteristics were applied on the base standard physical implementation flow referred for the given research and will be presented as following:

- **Multi bit flip flops**

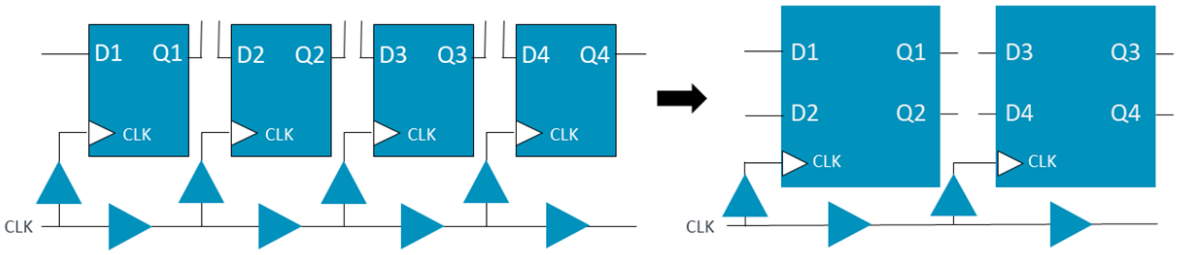
Clock trees are among the largest contributors to dynamic power consumption in digital design (SANTOS et al., 2012; HO et al., 2009), making the clock tree synthesis one of the crucial steps to keep meeting the aggressive power targets for advanced process nodes. The main goal in the clock tree synthesis is to improve latency and minimize skew, i.e. reduce as much as possible the difference in arrival time of the clock signal at the various clock pins in the design. Trying to correct skew usually implies in multiple buffer insertions, hence a bigger clock tree that increases area and power consumption.

There are several occasions in digital circuit design when multiple flip flops, also known as DFF (D-type Flip Flop), are connected in a chain sharing the same clock signal side by side. This kind of configurations can be improved with the help of a special feature present in advanced Place & Route EDA tools, which enables the merging of multiple enchained single bit flip flops in multibit flip flops.

The main benefit related to the application of this feature, is the reduction of buffer insertions to minimize skew, resulting in a smaller clock tree. Consequentially, as less buffers are inserted, this feature can reduce power consumption and possibly some area (SANTOS et al., 2012; YAN et al., 2010).

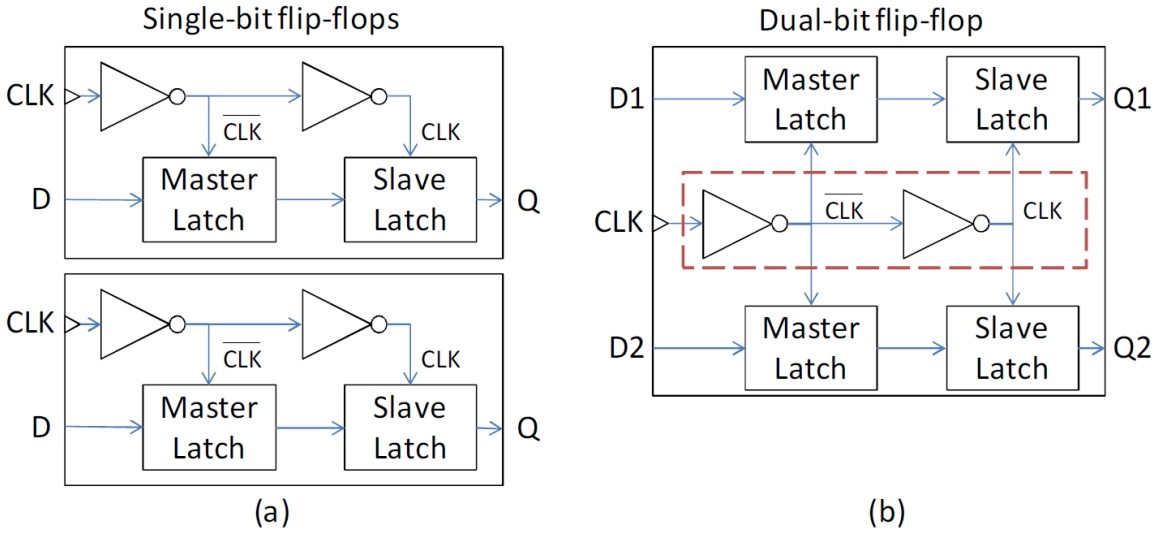
The following figures represents the multi bit flip flops when the merging of single bit flip flops can occur:

Figure 2.3: Clear Representation of Multi Bit Flip Flops Merging



Source: The Author

Figure 2.4: Internal difference between single-bit (a) and multi-bit (b) flip flops



Source: The Author

- **Multi VT Technology**

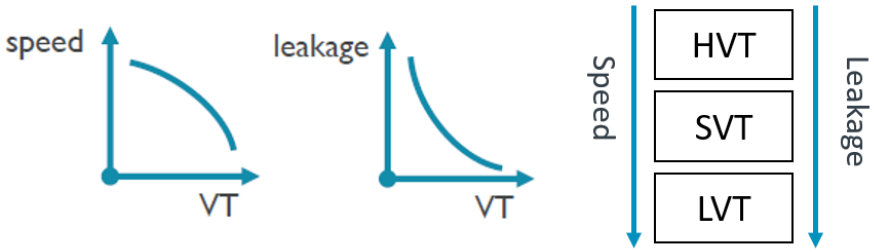
It is not new in the digital design industry that manipulating the threshold voltage, also known as VT, i.e. the minimum gate-to-source voltage that is needed to make a transistor operate in a conduction state, has a considerable impact in power consumption, specifically in terms of leakage current reduction. With the constant scaling down of process node technologies, leakage current (as also known to leakage power) has considerably increased to become a major concern to achieve optimal power efficiency.

Concerning the physical effects caused by this scaling down, the subthreshold drain current, among other factors, is considered one of the greatest contributors to generate this leakage current (FALLAH et al., 2005; MUTOH et al., 1995). As gate and supply voltages were being reduced to decrease dynamic power consumption, the threshold voltage had to be reduced as well, which has as effect the increase in subthreshold drain current (FALLAH et al., 2005; MUTOH et al., 1995). However, decreasing threshold voltage from another point of view, can reduce switching time, increasing performance, for a faster working transistor.

With that being said, semiconductor foundries (e.g TSMC and Samsung) had developed what is known as multi VT technologies, where standard cells with different VT options are available for the physical implementation. These standard cells can be usually named as "Low VT" (LVT), "Standard VT" (SVT) and "High VT" (HVT). Multi VT technology can be used on the design on different ways, as to give priority to delay reduction or diminish leakage power in different levels, as different parts of the design might be more time critical than others. EDA Synthesis and Place & Route tools can then choose from different libraries which standard cells are used in different occasions, seeking for optimal PPA. One way that foundries use to modulate VT is by using several implant pass or different oxide thickness.

The following figure show in general terms the influence of VT in leakage power and transistor delay, along with a multi VT standard cells diagram:

Figure 2.5: Multi VT Technology Influence



Source: The Author

Considering the context of this research and work, the 7 nanometers technology process node used has available three possible options of VT specification for the standard cells:

- **Low VT**: Offers the smallest delay between the other options, but as expense of the highest amount of leakage power.

- **Standard VT:** An intermediary VT option, offering a balance between leakage power and delay.

- **High VT:** Offers the least amount of leakage in comparison to the others but has the biggest delay among the other VT options.

The used names are generic and are not related to the specific technology.

- **Multi Channel Technology**

Transistor channel length along with voltage threshold is an important parameter to control performance and power in digital design. As classically known changing the width of a MOSFET (Metal-Oxide Semiconductor Field-Effect Transistor) channel impacts in performance, as increasing the channel width more current can flow through it in the same amount of time, but at the expense of increasing power.

From a different angle, increasing the transistor channel length increases transistor tend to increase the gate capacitance and delay, while being capable of reducing leakage power with some increase in dynamic power (FALLAH et al., 2005; SIRISANTANA et al., 2000). With this in mind, as with multi VT technologies, is it possible to give the option to the EDA tools to use faster or slower standard cells in different paths, depending when timing is critical or not.

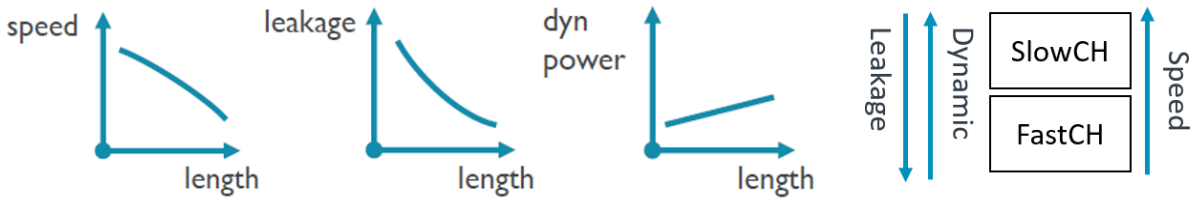
Concerning the 7 nanometers process technology node used in this work, the available options given by the foundry were:

- **Slow Channel (SlowCH):** Slower cell, offering less leakage power than the faster cell but having slightly higher dynamic power.

- **Fast Channel (FastCH):** Faster cell, with higher leakage power and less dynamic power.

The following figure show in general terms the influence of channel length in leakage power, dynamic power and transistor delay, along with a multi-channel standard cells diagram:

Figure 2.6: Multi Channel Technology Influence



Source: The Author

3 METHODOLOGY

The methodology used in the development of this work was based from the starting point of a fixed specific advanced modern Arm® CPU architecture using a standard physical implementation flow to implement it, in a 7-nanometer process technology node. From its physical implementation with the execution of the referred flow, metrics relative to power consumption, operating frequency and area, were collected to create the base data that contains the figures to be compared to each test case of implementation flow configuration, presenting improvements or not. The following test cases were created and englobe the power optimization techniques discussed theoretically previously:

- **Test Case 01:** Multi bit flip flop merging feature.
- **Test Case 02:** The use or enabling of multiple options of VT and channel length at different steps of the flow.
- **Test Case 03:** Dynamic power optimization feature after routing.

3.1 Collected and Evaluated Metrics

Considering the multiple testcases elaborated from the standard physical implementation flow referred trying to improve PPA for the given architecture, certain metrics were selected to evaluate the benefits and drawbacks from one configuration to another. As result of that, the following main metrics were considered:

- **Frequency:** Considered as the internal operating frequency of the CPU.

- **Power consumption:** As form of Dynamic Power and Static Power (also known as Leakage Power).

- **Standard cell area:** Indicates the total standard cell area of the design.

- **Runtime per step of the implementation flow:** As the physical implementation flow is composed of several steps, indicates the run time of a given step.

- **Total runtime of the implementation flow:** Indicates the total runtime of the physical implementation flow, considering all steps.

The retrieval of the listed metrics was made possible with help of the reports generated by the EDA tools used, such as timing reports, area reports, and power extraction reports for a specific benchmark. Obviously, as the focus of this project is centered in PPA improvements, power consumption, operating frequency and area are the metrics of main concern most of the time. However, in some cases analyzing the runtime of the flow execution can be interesting, as some added or removed feature might affect runtime significantly, being necessary to evaluate its real benefits, at expenses of increase or decrease in runtime.

It is important to remember that all metrics listed here will not be presented as raw figures, but always in a form of ratio or gain/loss comparison to the metrics presented in the standard physical implementation flow referred, as they are confidential information.

3.2 Power Extraction and Switching Activity Generation

To properly collect the figures used to evaluate the power consumption of the given design, the methods used differ from the standard vectorless power reports generated by the Synthesis and Place & Route EDA tools, since they do not consider precisely and realistically switching activity compared to what we would see in real-world applications. Taking that into account, for better QoR (Quality of Results) when analyzing power, the quality of switching activity applied is crucial. The technology library files in the flow contain the power information about each cell such as static power and dynamic power, but not about the switching activity, which can be generally described as a form of measurement of changes in signal values.

The way to generate a proper switching activity is to realize what is known as Gate Level Simulation (GLS). GLSs, which can be performed after synthesis, are used for verification in many contexts in the design cycle, such as to estimate power, to spot ‘X’(“Don’t Care”) propagation and help reveal glitches on edge sensitive signals due to combinational logic. They are less optimistic and more precise compared to RTL simulations as they consider real timing information as opposed to “zero delay”, but at an expense of much larger turnaround time.

One of the other uses of performing Gate-Level Simulation is to generate switching activity for each gate in the design, that can be used as an input for power analysis in other

tools. This method was used in project and considering the given architecture, which contains around 2 million instances, the amount of information stored in a switching activity file in those terms is considerably big. To generate a switching activity file, the tool that performs the GLS usually takes as main inputs a gate-level netlist, a delay annotation file and a test bench file.

3.3 Tools and Benchmark for Power Extraction

After having generated a switching activity file performing GLS, it is possible to use it as input in an EDA power extraction tool. Power extraction tools can apply different kind of benchmarks when realizing power analysis. The selected kind of benchmark used to evaluate the power consumption in the given architecture is known as Dhrystone.

Dhrystone is an integer-based type of benchmark, in another words, it does not perform floating-point operations, and it is considered one the main representatives of general-purpose “integer” CPU performance. It is interesting to add that a Dhrystone score can be more meaningfully than a MIPS (MIPS Microprocessor without Interlocked Pipelined Stages) score, simply because instruction count cannot be directly compared between different instruction sets, such as, RISC and CISC (Complex Instruction Set Computer). Floating-Point based benchmarks (e.g. Saxpy) could be planned to be executed as well, but the benchmark Dhrystone alone is already a very well-known way to evaluate PPA in the industry.

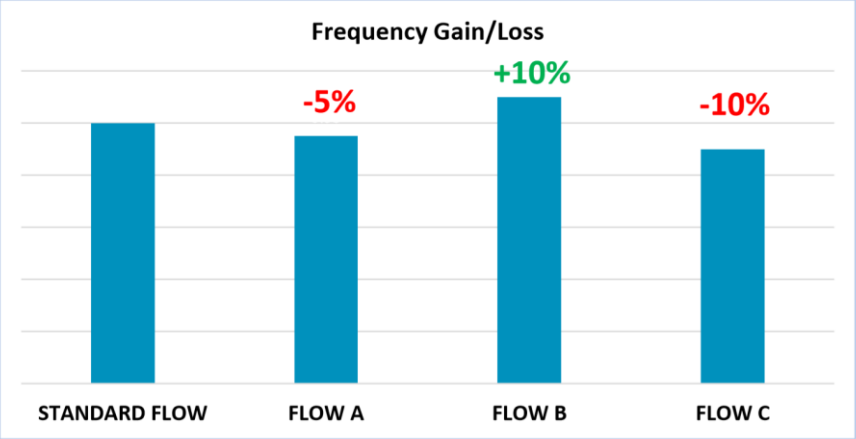
4 RESULTS

This chapter is dedicated to the results presentation of the power optimization techniques discussed on chapter two. Before the results of each test case are presented, the method of results presentation and comparison between each implementation flow configuration is explained.

4.1 Results Presentation

Respecting the confidentially disclaimer presented in the beginning of this work, all the figures shown will be relative (not raw) to the figures of the standard flow configuration test case, as shown in the following figure that serve as example:

Figure 4.1: Example - relative figures comparison between different implementation flows



Source: The Author

The Figure 4.1 represents the gain or loss in operating frequency for a given architecture, in different physical implementation flows, compared to one base standard configuration. During the development of the practical research there were developed several test cases that performed some kind of change or modification in the way the physical implementation flow was executed for the given architecture from the original flow. Not every test case will be commented or reported in this work, except the ones listed previously, as many others considered the use or change in characteristics of proprietary ownership, such as certain

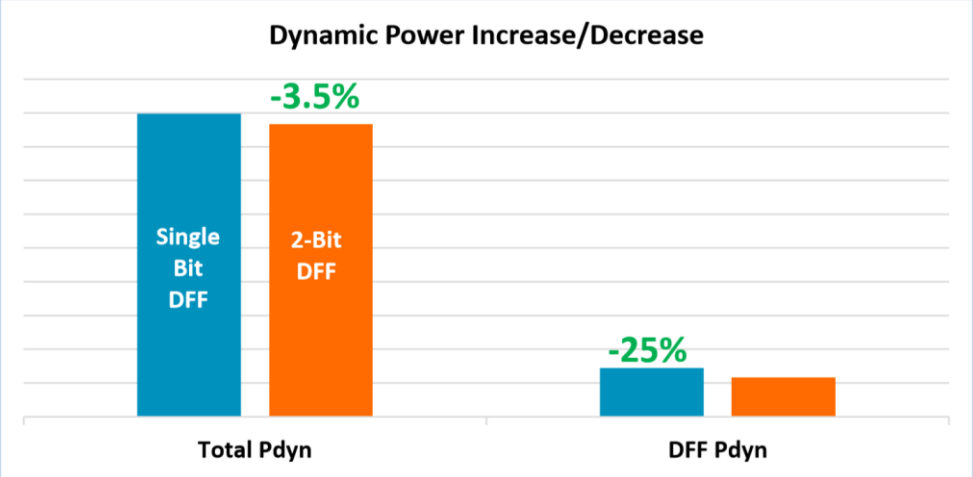
optimization algorithms or parameters that cannot be disclosed, neither explained in generic terms.

4.2 Test Case 01: Multi Bit Flip Flops

The multi bit flip flop merging feature in the used tool chain has as available options 2-bit and 4-bit multibit flip flops versions. For the trials in this project, it was compared the effect of enabling and not enabling the merging of multi single bit flip flops into 2-bit flip flops.

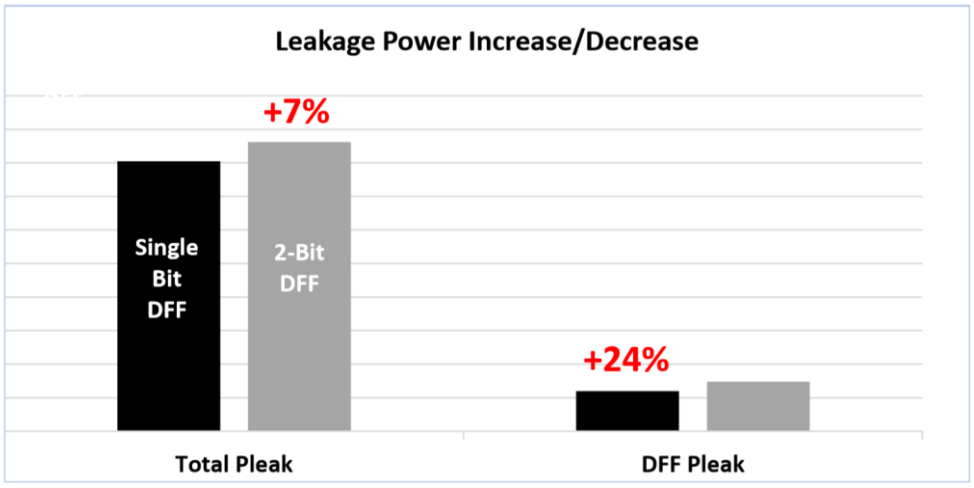
The collected results concerning power were initially evaluated in terms of dynamic and leakage power improvements or degradation, to finally an overall comparison and flip-flop merging rate in the design. The results are shown in the figures below:

Figure 4.2: Multi bit flip flops dynamic power



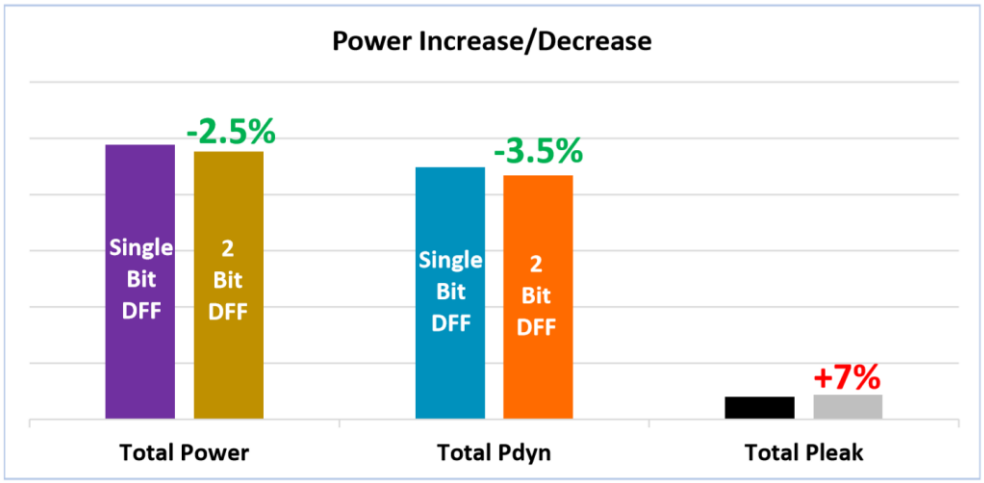
Source: The Author

Figure 4.3: Multi bit flip flops leakage power



Source: The Author

Figure 4.4: Multi bit flip flops total power



Source: The Author

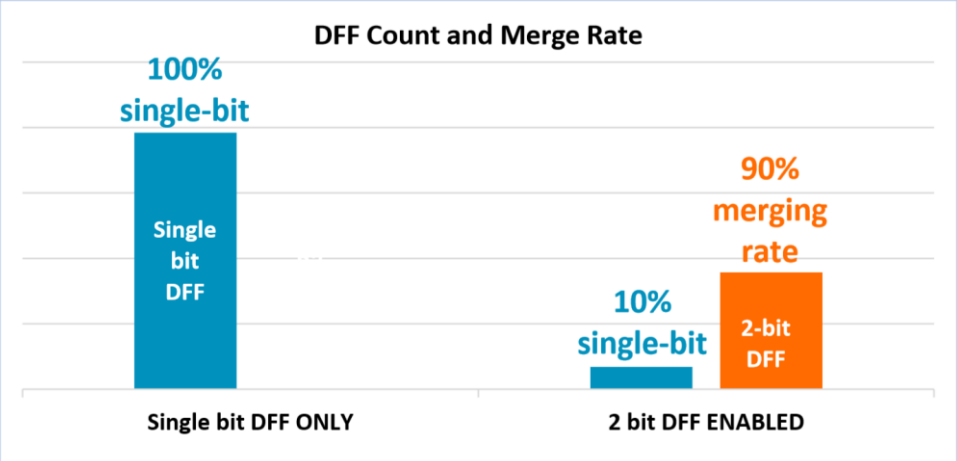
According to the reports generated, the collected power shows that when merging multi bit flip flops feature is enabled, it implies in a very clear reduction of 25% relative specifically to DFF dynamic power, giving a 3.5% improvement in total dynamic power. For the leakage power, the data collected showed a degradation of 24% relative specifically to DFF leakage power, equating to a 7% of degradation in total leakage power, but leakage remains in proportion much lesser to the importance of dynamic power as shown in the diagram above.

Looking at the results from an overall power consumption point of view, there is a 2.5% reduction in the total power (i.e. dynamic power plus leakage power). It is true that 2.5% in total overall power consumption improvement might not look a lot alone in this context, but any 1% of power improvement that can be achieved and combined with other improvements or techniques can possibly translate in the future in a significant power efficiency impact for a given architecture.

It is enough to say that the standard cell area of the design was not affected considerably, having a 0.6% improvement, which is not very representative in terms of area improvement for this context and process technology node. Operating frequency was not affected and stayed in the same range as well.

It is interesting to add as information the merging rate of the DFFs, i.e. the percentage of single bit flip flops that were considered while merging into 2-bit multi bit flip flops in the design. The representative data of this characteristic is showed below:

Figure 4.5: Multi bit flip flops merging rate



Source: The Author

The last figure shows that around 90% of single-bit DFFs were considered when the merging occurs for this given architecture, while each 2-bit DFF is equivalent to two single-bit DFFs, which implies that when counting DFFs without differentiating its number of bits occurs a big reduction in the number of cells used. However, the equivalent number of single-bit DFFs remains the same as compared to when this feature is disabled. It is also interesting to add that the addition of the multibit DFF feature did not impact in extra considerable runtime in the steps of the physical implementation, while preserving area and frequency.

4.3 Test Case 02: Different Technology Selection at Different Steps of the Flow

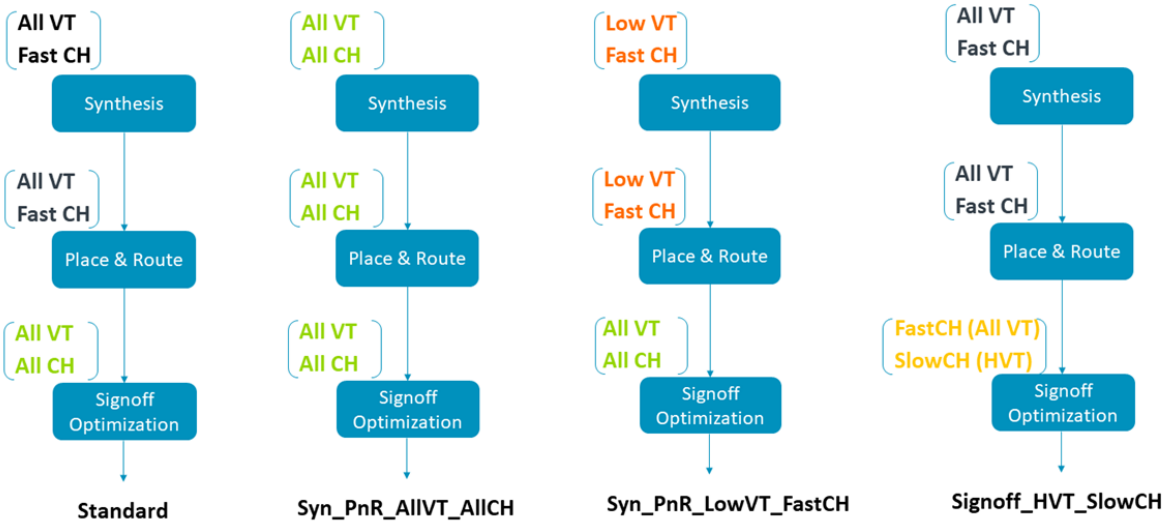
In the base standard physical implementation flow referred different types of standard cell are made available in different steps to try to help the EDA Synthesis and Place & Route tools make the best decisions along the design cycle. As previously presented Multi-VT and Multi-Channel length technologies are a very interesting way to seek for PPA improvements whenever possible, while different parts of the design might have different needs, as some of them can be less timing critical than others.

Taking this into account, while in physical implementation process it is important to explore the different possibilities given by the tools and its results offered, as they are not evident initially and can vary from design to design and from technology to technology. EDA Synthesis and Place & Route tools are very complex and take its decisions based in several algorithms interconnected that can work sometimes in a non-deterministic way.

With that in mind, for the continuous seek for PPA improvements and run the physical implementation flow in an optimal way, one of the trials performed during this research was the exploration of enabling different technologies options to the EDA Synthesis and Place & Route tools during its different steps, concerning Multi-VT and Multi-Channel technology.

There were performed three trials along with the standard flow in this context, as described in the following figure, which indicates (in parenthesis) the types of VT and Channel technology selection in the different steps:

Figure 4.6: Technology selection at different steps

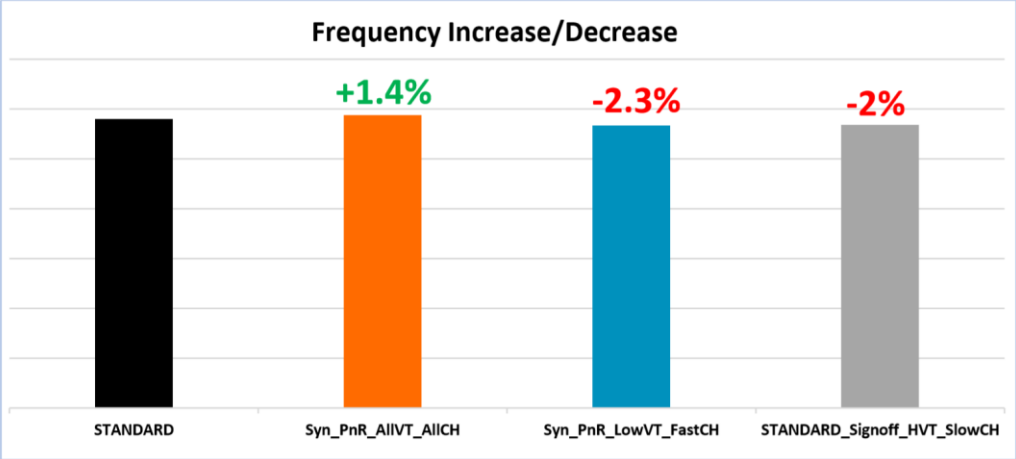


Source: The Author

Starting from the base standard physical implementation flow, which from the Synthesis to Place & Route steps uses only the fastest cells (Fast Channel and Low VT) and enables use of All VT and Channel at Sign-off, different trials were performed as in the above diagram.

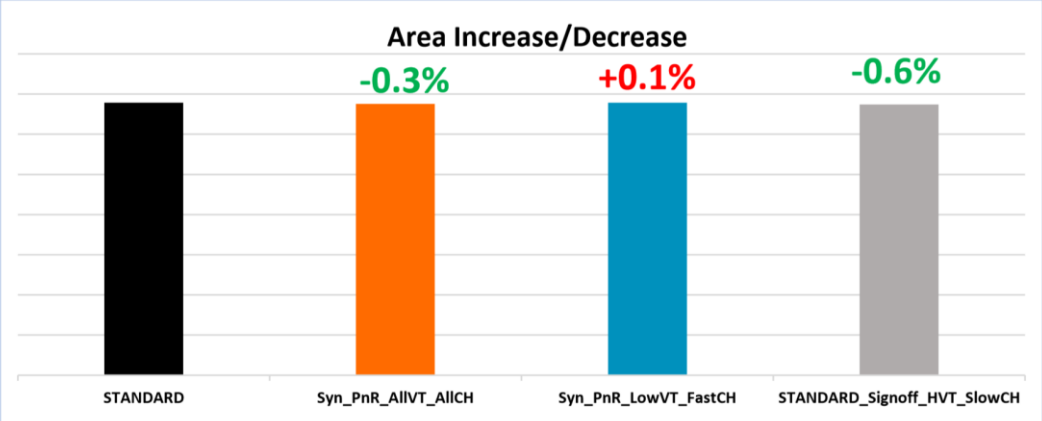
After the physical implementation flow execution and power extraction were performed for all trials, the metrics of interest were collected as shown in the following figures:

Figure 4.7: Technology selection at different steps - Frequency



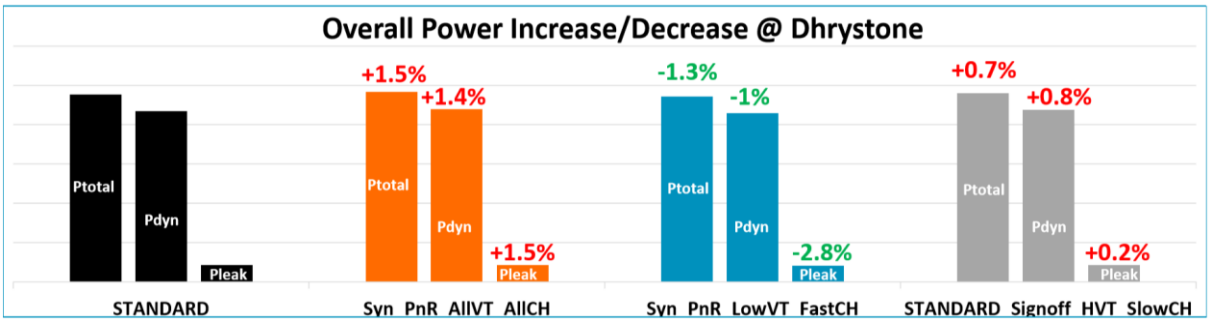
Source: The Author

Figure 4.8: Technology selection at different steps – Area



Source: The Author

Figure 4.9: Technology selection at different steps – Total Power



Source: The Author

Analyzing all three parameters of PPA there is no evident configuration that points directly to one trial that works the best. Area variation is so minimal that is hard to be considered, along with the characteristics changed that should not highly affect area. In frequency, minimal improvements when opening the possibility for the tool to use every kind of VT and Channel technology available, and a small decrease for the other trials that could be linked to just a non-deterministic small variation between runs. It is important to point that at the run "STANDARD_Signoff_HVT_SlowCH" in the Signoff Optimization step is different from all the other runs, when the opportunity to use Low VT and SVT cells are removed for Slow Channel cells.

For the power consumption the results are quite interesting and unexpected as the configuration that uses initially the fastest and most power demanding cells (FastCH and LowVT) in Synthesis and Place & Route step, at the end offered the best results. But at the same time, the improvement in power is minimal (1%), making it difficult to not discharge the reason of maybe just a small variability between runs.

An interesting point that can be used to validate the real necessity or not of opening the possibility to the tool of using more channel or more VT options earlier in the flow, i.e. at Synthesis and Place & Route step, is to analyze the impact of flow runtime while doing that, correlating to the real benefits provided at the end.

The following table shows the runtime increase or decrease percentage compared to the standard flow for the different trials in different steps:

Table 4.1: Runtime per step for testcase 02

	Increase or decrease in runtime comparison at step		
Flow	Synthesis	Place & Route	Sign-off Optimization
Syn_PnR_AllVT_AllCH	**+25%**	**+12%**	+3%
Syn_PnR_LowVT_FastCH	**-6%**	**-7.2%**	**-4.5%**
Signoff_HVT_SlowCH	+4.7%	+6.1%	-8.8%

Source: The Author

The above table, along with the PPA results showed previously, points that there`s no big evidence when enabling the use of all types of cells earlier in the flow, regarding channel length and VT. The trial 'Syn_PnR_LowVT_FastCH' confirms that when it assumes the use of the fastest cells (FastCH and LowVT) from the beginning till the Sign-Off Optimization step, while not affecting PPA and in a considerable shorter runtime. These results can point as well that the Sign-Off Optimization is performed very efficiently in way that no matter what the technology choice in the beginning is, it can recover from it if needed, when all the types of cells are enabled.

Aside from the runtime aspect, concluding the analysis of this trials is very hard to point or link one specific reason to decide which technology selection during the flow works the best in terms of PPA, as the variations were very small and spread along the trials. This variation maybe linked simply to somehow the "non-deterministic" way that EDA Synthesis and Place & Route tools can work, presenting small variations between runs, even if executed in the same exact conditions, depending the decisions of certain algorithms can take and statistical aspect of some of the methods applied. In any case, from the results, seems more reasonable to opt for the configuration that considers the faster cells from the beginning of the flow.

4.4 Test Case 03: Dynamic Power Optimization at Sign-off Optimization Step

At Sign-off Optimization step, the tasks performed to apply the final design optimizations generally can be described to what is known as STA (Static Timing Analysis), which try to fix timing violations, while preserving or improving PPA. The EDA Sign-Off optimization tool used enables several possibilities of optimizations features regarding timing, power and area at this step. Multiples features were still not tested before or not incorporated to evaluate its results for this process technology node (7nm).

Multiple features that involve proprietary algorithms were tested during this research couldn't be presented in this work as they would break confidentially policies and rules. However, it will be presented one feature that can be shown generically without infringing those terms. With that being said, at Sign-off Optimization step there was conducted one trial that activated a feature called "Dynamic Optimization", which aims for dynamic power recovery at this final step of the design. This dynamic power optimization feature has four configurable parameters listed and briefly described as the following:

- Swap Instances: Enables VT Swapping (the choice from swapping between LVT, SVT or HVT)
- Resize Instances: Enable cell current drive resize
- Add Instances: Enable buffer or delay insertion
- Delete Instances: Enables buffer deletion

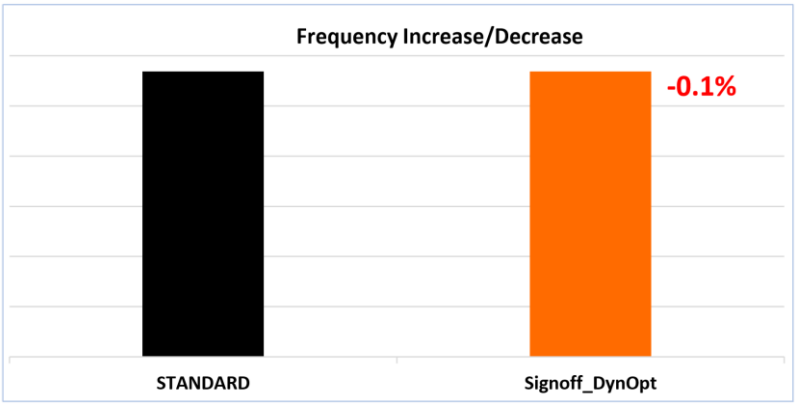
During the research developed there were planned the three different following trials, concerning the disabling or activation of different parameters, according to the labeling below:

Labeling: Enable ✔ Enable ❌

- Trial 01: Swap ✔ Resize ❌ Add ❌ Delete ❌
- Trial 02: Swap ✔ Resize ✔ Add ❌ Delete ❌
- Trial 03: Swap ✔ Resize ✔ Add ✔ Delete ✔

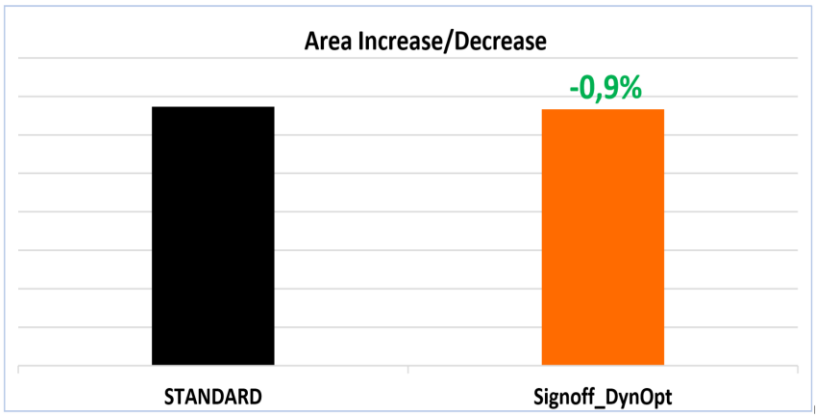
The results presented for this test case will only englobe the 'Trial 01', due to the fact that 'Trial 02' and 'Trial 03' were still running, while my internship contract period had been finished and the access to the data of interest was forbidden. The collected metrics of interest are shown below:

Figure 4.10: Dynamic optimization trial at Sign-off Optimization step – Frequency



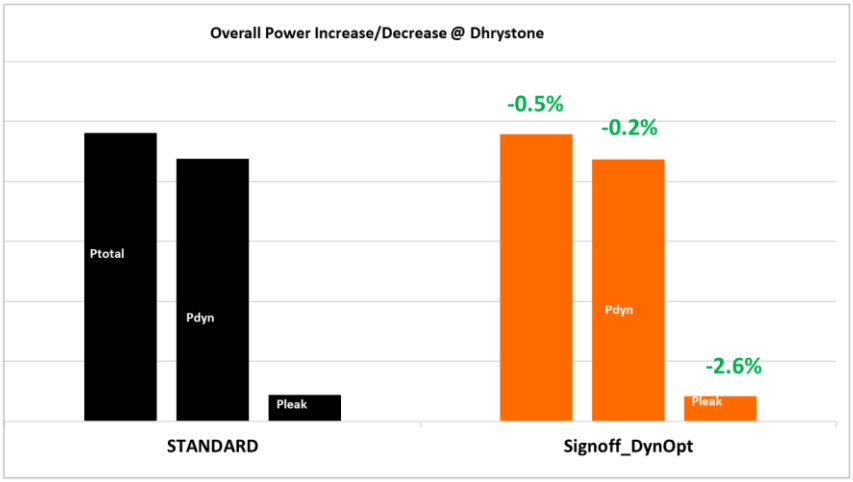
Source: The Author

Figure 4.11: Dynamic optimization trial at sign-off optimization step – Area



Source: The Author

Figure 4.12: Dynamic optimization trial at sign-off optimization step – Total Power



Source: The Author

Analyzing the results directly in the power figures, as the feature activated is related to optimization in dynamic power, it is shown a very small improvement in total power. Looking specifically in dynamic, there's a reduction of only 0.7%. The interesting result is that the improvement in leakage power is almost four times higher (in percentage, but differs in proportion to dynamic power importance), as the power recovery feature enabled is focused in dynamic power recovery. Area and frequency weren't considerably affected, staying in the same range.

It is very important as well to evaluate if this added feature increased or not considerable extra runtime at the sign-off optimization step, with the intent to verify the benefits to drawbacks ratio of enabling it. The following runtime table shows this data:

Table 4.2: Runtime per step for testcase 03

Flow	Runtime increase @ total flow	Runtime increase @ sign-off
Signoff_dyn_opt	7.6 %	**30 %**

Source: The Author

For a final conclusion about this trial, from the results shown it was clear that for the given architecture the dynamic optimization feature applied did not offered great results. The benefits were minimal at the expense of considerable extra runtime, determining that this feature should not be active in this context.

5 CONCLUSIONS

Exploring the capabilities of PPA improvements, and more specifically to power efficiency at the physical implementation level, showed to be a very hard task. It is true that sometimes reducing 1% or 2% in total overall power consumption improvement might not look a lot alone in a certain context, but any recognizable power improvement that can be achieved and combined with other improvements or techniques can possibly translate in the future in a significant impact in power saving for a given architecture. The EDA tools used in the industry are very complex and offer numerous features and possibilities that should be tested in different situations and contexts to achieve certain results, and sometimes requiring lots of trial and error, as they can behavior in a very indeterministic way. Runtime showed to be an important factor as well to decide if some feature or optimization has value or not as the benefit to runtime expense ratio might not always be interesting.

Departing from the goal of optimizing power in a standard physical implementation flow, by the research and investigation of different techniques that could be applied, many trials were performed during the work of this project. Despite from the fact that some test were hidden to not infringe confidentiality aspects on this research, from the test cases that could be presented, it was already possible to show interesting results and characteristics that possibly can be merged to a standard physical implementation flow at the selected process node technology to see some improvement, even if minor.

REFERENCES

FALLAH, F.; PEDRAM, M. Standby and Active Leakage Current Control and Minimization in CMOS VLSI Circuits. **IEICE TRANSACTIONS on Electronics Vol.E88-C No.4**, p. 509-519, 2005.

HO, Pei-Hsin. Industrial Clock Design. **Proceedings of the ACM International Symposium on Physical Design (ISPD)**, San Diego, California, USA, p. 139-140, 2009.

MUTOH, S. et al. 1-V power supply high-speed digital circuit technology with multithreshold-voltage CMOS. **IEEE Journal of Solid-State Circuits (Volume: 30 , Issue: 8)**, p. 847 – 854, 1995

RABAEY, M. **Digital Integrated Circuit: A design perspective.** Prentice Hall: NJ, USA 1996.

SANTOS, C.; REIS, R. et al. Multi-bit flip-flop usage impact on physical synthesis. **25th Symposium on Integrated Circuits and Systems Design (SBCCI)**, Brasilia, Brazil, 2012.

SHYU, Y. et al. Effective and Efficient Approach for Power Reduction by Using Multi-Bit Flip-Flops. **IEEE Transactions on Very Large Scale Integration (VLSI) Systems (Volume: 21, Issue: 4)**, p. 624 - 635**,** 2013.

SIRISANTANA, N. et al. High-performance low-power CMOS circuits using multiple channel length and multiple oxide thickness. **Proceedings 2000 International Conference on Computer Design**, Austin, TX, USA, 2000.

WESTE, N.; HARRIS, D. **CMOS VLSI Design: A Circuits and Systems Perspective**. 4th edition. USA: Addison-Wesley Publishing Company, 2010.

YAN, J.; Chen, Z. Construction of constrained multi-bit flip-flops for clock power reduction. **International Conference on Green Circuits and Systems**, Shanghai, China, 2010.

APPENDIX - ARM®: ABOUT THE COMPANY

- **Special Acknowledgements**

The experience of doing a graduation internship at Arm® was an amazing journey, where the amount of daily learning from a technical and personal standpoint was immensurable. Working first-hand in the leading company of IP design for embedded CPUs at physical implementation level was a wonderful opportunity to learn from one of the best in the industry, acquiring as much knowledge and experience as possible. Being able to interact with very educated people on the field, seeing and experiencing closely how is a day of an engineer in a big company like Arm® company, added a lot in terms of personal experience and field perspective as well. Overall, it was a wonderful experience working as an intern for Arm® and highly recommending it to future students wouldn't be a surprise, but in fact a recurrence.

- **Brief History**

Arm® is a British company in the semiconductors and software design industry founded in 1990 that has its headquarters base in the city of Cambridge, United Kingdom. Its foundation, in a nutshell, can be described as starting from the company Acorn Computers Limited with the development of the "Acorn RISC Machines" also known as "Advanced RISC Machines", structuring a joint venture with Apple Inc. and VLSI (Very Large-Scale Integration) Technology at the time. Its main product and market rely in the design of what is known as RISC (Reduced Instruction Set Computer) architectures for computer processors, that can also be referred as embedded processors. This kind of architectures generally strive for low power consumption, small silicon area, but at the same time offering enough and respectable performance. Starting in 1990 as a very small company of around 12 engineers based in a "barn" like office in Cambridge, Arm® climbed to actually established itself as the leader in the design of embedded processors, as the majority of every portable device and embedded system, like smartphones and microcontrollers, are using designs licensed by Arm®.

- **Business Model**

The business model of the company generically can be explained as the selling of licenses of its Intellectual Property (IP), in this case its designs, to be integrated in the System

On-Chip (SoC) devices of a customer. It is a business model that can be viewed as kind of "unique" in this industry, as other companies known as "fabless semiconductor companies" like NVIDIA Corporation and AMD Inc., that designs its products to be manufactured at "for-hire" foundries (e.g. TSMC and Samsung), instead of licensing to anyone that wants to use it to incorporate in their products, like Arm® does. In addition to that, Arm® profits from royalties in the sale of each of its licenses, being for every chip that contains an Arm® IP, a royalty associated to it. As previously said, the primary business of the company is centered in the development of ARM processors (CPUs), however GPU (Graphic Processing Unit) IPs, SoC infrastructure, IoT (Internet of Things) SoC (System on Chip) solutions, Security IPs, are between other products developed by Arm®.

- **Offices and Structure**

Arm® company is well-spread around the globe having multiple offices in different locations, such as England (Cambridge), USA (Austin), France (Sophia Antipolis), Japan (Tokyo), China (Shanghai) and India (Bangalore). My internship took place in France, at the Sophia Antipolis Arm® offices. The Arm® office based in Sophia is mainly responsible for CPU (Central Processing Unit) design and physical IP design.

The CPU design team is divided as following:

- Design Team
- Implementation Team
- Verification Team
- CPU Modelling Team