University of Arkansas, Fayetteville
ScholarWorks@UARK

Theses and Dissertations

12-2017

Asynchronous 3D (Async3D): Design Methodology and Analysis of 3D Asynchronous Circuits

Francis Corpuz Sabado
University of Arkansas, Fayetteville

Follow this and additional works at: http://scholarworks.uark.edu/etd

Part of the Computer Sciences Commons, Electrical and Electronics Commons, and the VLSI and Circuits, Embedded and Hardware Systems Commons

Recommended Citation
Sabado, Francis Corpuz, "Asynchronous 3D (Async3D): Design Methodology and Analysis of 3D Asynchronous Circuits" (2017). *Theses and Dissertations*. 2584.
http://scholarworks.uark.edu/etd/2584

This Dissertation is brought to you for free and open access by ScholarWorks@UARK. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of ScholarWorks@UARK. For more information, please contact scholar@uark.edu, ccmiddle@uark.edu.

Asynchronous 3D (Async3D):
Design Methodology and Analysis of 3D Asynchronous Circuits

A dissertation submitted in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy in Engineering

by

Francis Sabado II
University of Arkansas
Bachelor of Science in Computer Engineering, 2012
University of Arkansas
Master of Science in Computer Engineering, 2015

December 2017
University of Arkansas

This dissertation is approved for recommendation to the Graduate Council.

Dr. Jia Di
Dissertation Director

Dr. Dale Thompson
Committee Member

Dr. Jingxian Wu
Committee Member

Dr. J. Patrick Parkerson
Committee Member

ABSTRACT

This dissertation focuses on the application of 3D integrated circuit (IC) technology on asynchronous logic paradigms, mainly NULL Convention Logic (NCL) and Multi-Threshold NCL (MTNCL). It presents the Async3D tool flow and library for NCL and MTNCL 3D ICs. It also analyzes NCL and MTNCL circuits in 3D IC. Several FIR filter designs were implement in NCL, MTNCL, and synchronous architecture to compare synchronous and asynchronous circuits in 2D and 3D ICs. The designs were normalized based on performance and several metrics were measured for comparison. Area, interconnect length, power consumption, and power density were compared among NCL, MTNCL, and synchronous designs. The NCL and MTNCL designs showed improvements in all metrics when moving from 2D to 3D. The 3D NCL and MTNCL designs also showed a balanced power distribution in post-layout analysis. This could alleviate the hotspot problem prevalently found in most 3D ICs. NCL and MTNCL have the potential to synergize well with 3D IC technology.

ACKNOWLEDGEMENTS

First, I would like to express my sincere gratitude to my advisor Dr. Jia Di for the continuous support of my Ph.D. study and related research, for his patience, motivation, and immense knowledge. His guidance helped me in all the time of research and writing of this dissertation.

I am also grateful to my committee members: Dr. Dale Thompson, Dr. J. Patrick Parkerson, and Dr. Jingxian Wu for their guidance and support for my research work.

I would like to thank my colleagues at the TruLogic Circuit Design lab and Cato Springs Research Center for their support and collaboration on several projects during my graduate studies. Our deep discussions on various research topics have guided my research work these past years.

Last but not least, I would like to give a special thanks to my family who has been there during the hard times and the fun times. Thank you to my mom and dad, and my brother for their love and support. Your support and encouragements have given me the drive to succeed in all my endeavors.

DEDICATION

To my family whom are always loving and supportive.

TABLE OF CONTENTS

1 Introduction **1**

2 Problem Statement **4**

2.1 Dissertation Contributions **4**

2.2 Dissertation Organization **5**

3 Background **6**

3.1 NULL Convention Logic **6**

3.1.1 NULL Convention Logic Pipeline 9

3.2 Multi-threshold NULL Convention Logic (MTNCL) **11**

3.2.1 Multi-Threshold NCL Pipeline 14

3.3 Three Dimensional Integrated Circuits **16**

3.3.1 Motivation 17

3.4 TSV Process Flow **22**

3.4.1 Manufacturing technologies for 3D IC 23

3.5 Challenges **24**

3.6 Discussion **26**

4 Asynchronous 3D Tool Flow **27**

4.1 Proposed Flow **27**

4.2 Circuit Design **28**

4.2.1 Unified NCL Environment (UNCLE) 28

4.2.2 3D IC Partitioning 29

4.2.3 Automatic Synthesis of Transistor Networks (ASTRAN) 30

4.3 Physical Design **31**

4.4 Standard Library Flow **32**

4.4.1 Tezzaron 3D Design Kit 32

4.4.2 Cell Schematics 40

4.4.3 Cell Verification 40

4.5 Async3D Library Cell Area **41**

5 Case Study: Finite Impulse Response Filter **44**

6 Analysis of 3D Asynchronous Circuits **47**

6.1 Design Verification **47**

6.2 Layout Floorplan **48**

6.3 Circuit Performance **50**

6.4 Area/Wire Length **50**

6.4.1 2D IC Design Circuit Density 51

6.4.2 3D Design Circuit Density 54

6.5 Interconnect **58**

6.5.1 Wire Length Analysis - 2D IC Design 58

6.5.2 Wire length Analysis - 3D IC Design 59

6.5.3 Wire Length Analysis - 2D vs 3D IC Design 61

6.5.4 TSV Count 65

6.6 Power Analysis **65**

6.6.1 Vector-based Power Measurement 67

6.6.2 Power Data 68

7 Conclusion **73**

7.1 Future Work **73**

8 References 75

LIST OF TABLES

Table 1: NULL Convention Logic dual-rail encoding. 6

Table 2: NULL Convention Logic fundamental gates. 7

Table 3: Gate Transistor Sizing 39

Table 4: Async3D library timing specifications. 40

Table 5: Async3D NCL library cell area. 42

Table 6: Async3D MTNCL library cell area. 43

Table 7: Async3D MTNCL special cells area. 44

Table 8: 2D IC area comparison 53

Table 9: TSV/Micro-bumps count. 65

Table 10: 2D Designs power data. 69

Table 11: 3D Designs power data. 69

Table 12: 2D versus 3D power data. 70

LIST OF FIGURES

Figure 1: Transistor count adapted from [8]. 2

Figure 2: THmn NCL gates: (a) TH34, (b) TH34w2.................... 8

Figure 3: NCL threshold gate implementation using CMOS technology.................... 9

Figure 4: NCL micro-pipeline architecture [19]. 10

Figure 5: NCL single-bit dual-rail register [20].................... 10

Figure 6: Multi-threshold CMOS power gating structure.................... 12

Figure 7 Fined Grained MTNCL power gating structure [33]. 13

Figure 8: Multi-threshold NCL pipeline architecture. 14

Figure 9 Early completion detection block in MTNCL pipeline. 15

Figure 10: Trend leading towards 3D IC technology [35].................... 18

Figure 11: 2D versus 3D interconnect [36].................... 19

Figure 12: Global interconnect trends [36]. 20

Figure 13: 3D IC bandwidth potential [37].................... 21

Figure 14: Small form factor consisting of different layers [38]. 22

Figure 15: 3D cross-section of a TSV [39]. 22

Figure 16: Manufacturing 3D ICs [35]. 24

Figure 17: 3D IC thermal map [41]. 25

Figure 18: Synchronous vs NCL thermal distribution [40]. 25

Figure 19: Async3D tool flow. 28

Figure 20: Uncle tool flow overview [43] 29

Figure 21: ASTRAN layout style [45].................... 31

Figure 22: Async3D standard library tool flow. 32

Figure 23: Tezzaron 3D stack layer [46]. 33
Figure 24: Two-wafer logic stack [46]. 34
Figure 25: Async3D layout template. 37
Figure 26: Power rails. 38
Figure 27: Bulk body contacts. 38
Figure 28: Annotated TH44 schematic implementation. 40
Figure 29: Cell verification flow 41
Figure 30: FIR filter structure. 45
Figure 31: MTNCL FIR filter architecture. 46
Figure 32: MTNCL shift registers 47
Figure 33: FIR partitioning. 47
Figure 34: 2D 2.5×5mm floorplan. 49
Figure 35:3D 2.5×5mm floorplan. 49
Figure 36: 2D IC Circuit density 52
Figure 37: 2D IC standard cell area. 53
Figure 38: 3D IC Bottom layer density 55
Figure 39: 3D IC Top layer density 56
Figure 40: 3D Bottom layer standard cell area. 57
Figure 41: 3D Top layer standard cell area. 58
Figure 42: 2D IC total wire length. 59
Figure 43: 3D Bottom layer total wire length. 60
Figure 44: 3D Top layer total wire length 61
Figure 45: 2D vs 3D IC total wire length 62

Figure 46: NCL congestion map. 63

Figure 47: MTNCL congestion map. 64

Figure 48: State-dependent leakage data. 67

Figure 49: NCL total power density map: (a) Tier0, (b) Tier1. 71

Figure 50: MTNCL total power density map: (a) Tier0, (b) Tier1. 72

1 Introduction

Digital circuits has gained prominence since the invention of the metal-oxide-semiconductor (MOS) transistor in 1970 [1]. Digital circuits have been applied to many applications including mainframes, smartphones, tablets, wearable devices, and medical devices. The early digital integrated circuits were truly handcrafted full-custom designs involving manually placing, routing, and optimizing transistors [1]. As predicted by Gordon Moore in 1965, the advances in silicon technology have allowed the number of transistors to skyrocket from thousands to billions of transistor on a single chip as shown on Figure 1 [2], [3]. With the advent of very large-scale integration (VLSI), full-custom design became increasingly complex, leading digital designers to automation at higher abstraction. As a result, hardware description languages (HDL) such as VHDL and Verilog [4] and semi-custom design approaches like standard-cell methodology became more appealing and quickly gained prominence among VLSI designers [1], [4]. Using HDL, digital circuit designers could specify the behavior of digital circuits in a precise and formal manner in a textual description enabling technology independent flow that is suitable for code reuse and robust verification [4]. In addition, the standard-cell methodology provided designers with fully verified basic circuits called cells that can be used to develop more complicated ICs. The combination of standard-cell methodology and HDL is widely used in contemporary VLSI design and has been attributed as a key factor in the success of VLSI [5]–[7].

Figure 1: Transistor count adapted from [8].

Asynchronous circuits are sequential digital logic circuits that operate without a global clock signal. An asynchronous digital circuit uses local handshaking between components to synchronize, communicate, and operate [9], [10]. According to [9]–[13], asynchronous circuits provide the following benefits: (1) lower dynamic power consumption, (2) potentially faster operating throughput, (3) improved modularity, (4) lower electromagnetic emissions, and (5) higher robustness. In contrast to synchronous circuits, an asynchronous circuit is quiescent when idle and only computes a task whenever a request is received, resulting in lower energy

consumption. Asynchronous design has applications in low power systems since power consumption has become an important consideration in today's electronics industry. The impact of power dissipation has exponentially increased as the size of transistors decrease into nanoscale. This has caused a major paradigm shift where power dissipation has become as important as performance and area. In addition, due to the operating speed depending on the local latencies instead of a global signal, an asynchronous circuit operates in average-case delays rather than worst-case delays. Improved modularity is also inherent because of the reliance on local handshaking rather than a global signal. Since the registers switches at random points of time based on local request, an asynchronous circuit shows lower electromagnetic emissions. Finally, due to the lack of assumptions on inter-cell communication delays, asynchronous circuit has improved tolerance from process, voltage, and temperature (PVT) variations. An asynchronous circuit can also operate correctly in a large variation in power supply.

Despite these advantages, there are several drawbacks of asynchronous circuits including increased area overhead, and circuit and power consumption penalties in some cases [9], [10], [14].

Three-dimensional integrated circuit is a promising technology that could propel the advances of the semiconductor industry and solve the problem of scaling in deep-submicron technologies. It could be the technology that would scale beyond Moore’s law. Three-dimensional integrated circuits (3D ICs) consist of multiple layers of logic devices stacked together and connected via vertical interconnects. 3D ICs provide several benefits including shorter interconnect lengths, smaller area, improved interconnect bandwidth, and heterogeneous integration. While 3D IC provides these advantages, it suffers from a critical issue of thermal hotspots[15]. Asynchronous paradigms such as NULL Convention Logic (NCL) and Multi-

threshold NCL could be a potential solution to this thermal hotspot problem [16]. Since NCL and MTNCL lack the global clock signal of synchronous circuits, the power density is more evenly distributed, alleviating the hotspot problem. In addition, the shorter vertical interconnect length could improve the interconnect overhead of NCL and MTNCL designs. Asynchronous circuit designs such as NCL and MTNCL could synergize well with 3D IC technology, creating a more compact low power circuit.

2 Problem Statement

2.1 Dissertation Contributions

- **Develop an Asynchronous 3D Tool Flow:** The lack of EDA tool support for asynchronous circuits and 3D ICs has made the development of asynchronous 3D ICs especially difficult. This dissertation presents a semi-automated tool flow designed for asynchronous 3D ICs. It incorporates several prominent industry EDA tools as well as in-house scripts and open source programs. It utilizes a modified ASTRAN tool adapted for NULL Convention Logic and Multi-Threshold NCL. ASTRAN is an automated layout generation tool for quick creation of standard cells. It also includes tools for asynchronous 3D IC partitioning using Design Compiler. Physical design and verification of cells utilize industry standard tools such as Cadence Innovus, Voltus, and Mentor Graphics Calibre.
- **Develop Asynchronous 3D Standard Libraries (Async3D Library):** This dissertation includes the development of a low-power NCL and MTNCL standard library called Async3D. As of this writing, Async3D standard-cell library contains over 300 NCL and MTNCL cells. Several size variations for each cell were developed based on cell drive strength for fined-grained optimizations and low power applications. The library is fully verified with industry standard EDA tools including Mentor Calibre DRC/LVS. In addition, the cells have been

designed to be fully compatible with foundry provided Chartered 130nm synchronous standard cell library called CORELIB_LP, providing digital circuit designers the possibility of employing over a 1000 different types of cells in hybrid architectures. This is especially useful for synchronous and asynchronous 3D heterogeneous applications.

- **Analysis of Asynchronous 3D ICs:** This dissertation includes an analysis of asynchronous circuits designed in NCL and MTNCL architecture. The NCL and MTNCL circuits are implemented using the Async3D library while the synchronous counterparts were implemented using the CORELIB_LP library for comparison. Different circuits were implemented ranging in complexity using the Async3D tool flow. Chip area, wire-length, power consumption, and power density were analyzed for the asynchronous and synchronous designs to study design trade-offs. Circuit performance between designs was normalized to set up a fair comparison.

2.2 Dissertation Organization

The chapters are structured to provide the fundamental background information and presents technical analysis towards the end of the dissertation. Finally, the data are analyzed and summarized. Chapter 3 provides the background information on NULL Convention Logic, Multi-Threshold NCL and 3D ICs. Chapter 4 presents the Async3D tool flow and the Async3D library used to implement the NCL and MTNCL designs. Chapter 5 describes the FIR test circuit used to compare the different architectures. Chapter 6 presents the analysis of the NCL, MTNCL, and synchronous designs in terms of area, wire length, power consumption, and power density. Finally, Chapter 7 summarizes the findings and concepts presented on this dissertation and presents future work that could expand the application of asynchronous circuits with 3D IC.

3 Background

3.1 NULL Convention Logic

NULL Convention Logic (NCL) is a delay-insensitive (DI) asynchronous paradigm that lacks delay requirements and operates correctly as long as the transistors switch properly. NCL circuits utilize a local hand shaking protocol and multi-rail encoding to achieve delay-insensitivity. The most prevalent multi-rail encoding scheme is the dual-rail encoding, which uses two signals to represent four states. As shown on Table 1, the two signals *rail0* and *rail1* determine the current data value of a dual-rail encoding. Data0 is represented when *rail0* is logic 1 and *rail1* as logic 0, and data1 is denoted when *rail0* is logic 0 and *rail1* is logic 1. NULL and INVALID states are represented when both rails are either logic 0 or logic 1, respectively.

Table 1: NULL Convention Logic dual-rail encoding.

	DATA0	**DATA1**	**NULL**	**INVALID**
Rail0	1	0	0	1
Rail1	0	1	0	1

An NCL circuit is designed using 27 fundamental gates called threshold gates. Table 2 shows all threshold gates and their equivalent logic 1 set functions. These threshold gates were originally proposed by Theseus Logic, Inc. [17]. NCL threshold gates have the following properties: (1) consist of *n* inputs, (2) only becomes asserted when at least *m* of the *n* input become asserted, (3) special function gates are denoted by *D* or *N* refer to resettable gates to either zero or one, and (4) hysteresis property where the output de-asserts only when all the inputs de-assert.

Table 2: NULL Convention Logic fundamental gates.

NCL Gate	Set Function
TH12	A+B
TH22	AB
TH13	A+B+C
TH23	AB + AC + BC
TH33	ABC
TH23w2	A + BC
TH33w2	AB + AC
TH14	A+B+C+D
TH24	AB + AC + AD + BC + BD + CD
TH34	ABC + ABD + ACD + BCD
TH44	ABCD
TH24w2	A + BC + BD + CD
TH34w2	AB + AC + AD + BCD
TH44w2	ABC + ABD + ACD
TH34w3	A + BCD
TH44w3	AB + AC + AD
TH24w22	A + B + CD
TH34w22	AB + AC + AD + BC + BD
TH44w22	AB + ACD + BCD
TH54w22	ABC + ABD
TH34w32	A + BC + BD
TH54w32	AB + ACD
TH44w322	AB + AC + AD + BC
TH54w322	AB + AC + BCD
THxor0	AB + CD
THand0	AB + BC + AD
TH24comp	AC + BC + AD + BD

By utilizing *m*-of-*n* threshold gates, the design complexity of NCL circuits is significantly improved. NCL gates use the following naming convention of TH*mn* gates, where *m* represents the threshold requirement and *n* represents the number of inputs pins of the gate. Each gate transitions to a logic 1 value only if a certain *threshold* of input pins transitions to a logic 1. The output of the gate will be logic 1 when any *m* inputs pins have switched to logic 1 and can be set low only when all inputs are logic 0. Using this convention, a C-element can be interpreted as *n-of-n* threshold gate with hysteresis. Hysteresis is a special property of NCL gates.

Once the output of an NCL gate is logic 1, it remains high until all the inputs are logic 0. An example gate is a TH34 gate shown on Figure 2a and a TH34w2 gate shown on Figure 2b.

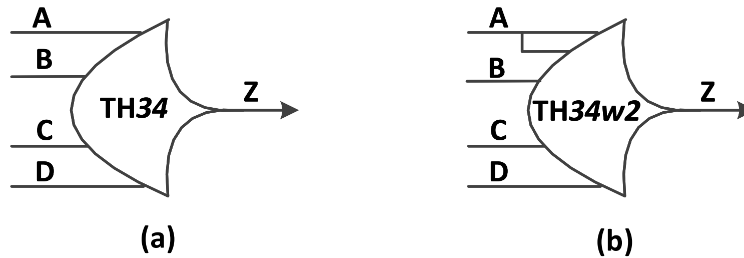


Figure 2: THmn NCL gates: (a) TH34, (b) TH34w2.

In addition to the threshold and hysteresis characteristics, NCL gates also utilize input weights. Weighted threshold gates are a variation of the basic threshold gates and are denoted by TH*mnW*$w_1w_2 \ldots w_R$ where $1 < w_R \leq m$. The values of $w_1, w_2, \ldots w_R$ indicate the weights of the input pins in order starting from input *A*. The weight of input A is indicated by w1, w2 denotes the weight of input B, etc. For example, a weighted threshold gate is shown on Figure 2b and has the following naming convention of TH34w2. A TH34w2 gate has 4 inputs and has a value threshold of 3 with a weighted input A having a weight of 2. Since input A has a weight of 2, TH34w2 could be set to logic 1 with input A as logic 1 and another input pin set to logic 1. The weights of input pins B, C, and D are not indicated on the naming convention because they have a weight of 1. Another variation are gates that have resettable functionally [18]. Resettable gates are often included to initialize the state of the gate. A reset input pin in added into the gate and can set the output of the gate to either logic 1 or logic 0. Resettable gates have an added notation of *N* or *D*, where having the *N* notation indicates a resettable gate to logic 0, and having a *D* notation indicates a resettable gate to logic 0. Resettable gates are used in designing shift registers in an NCL circuit.

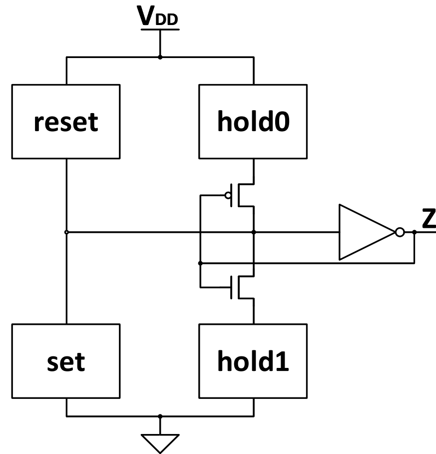


Figure 3: NCL threshold gate implementation using CMOS technology.

NCL threshold gates could be implemented using CMOS technology. A generic threshold gate consists of 5 components: set, reset, hold0, hold1, and an output inverter as shown on Figure 3. The set component controls how the gate will be asserted to logic 0 and corresponds to the set functions described on Table 2, while the reset component controls how the gate will be de-asserted. The hold1 component is a complement of set component, while the hold1 component is a complement of the reset component.

3.1.1 NULL Convention Logic Pipeline

The micro-pipeline framework for NCL is shown on Figure 4. The NCL pipeline follows DATA-NULL cycles in which DATA inputs are always followed by a NULL spacer. In a DATA cycle, the values in the gates are set by sending data inputs to the gates. The DATA cycle must then be followed by a NULL cycle where the inputs are all set to logic 0 to reset the gates to a NULL state. The NULL cycle acts as a boundary to prevent the current DATA cycle from being overwritten by the next DATA cycle. An NCL pipeline consist of several NCL combinational logic blocks separated by NCL registers. An NCL pipeline is similar to a synchronous pipeline architecture in that they both use registers between combinational blocks. However, special

registers are used for the NCL architecture. An NCL single-bit dual-rail register is shown on Figure 5.

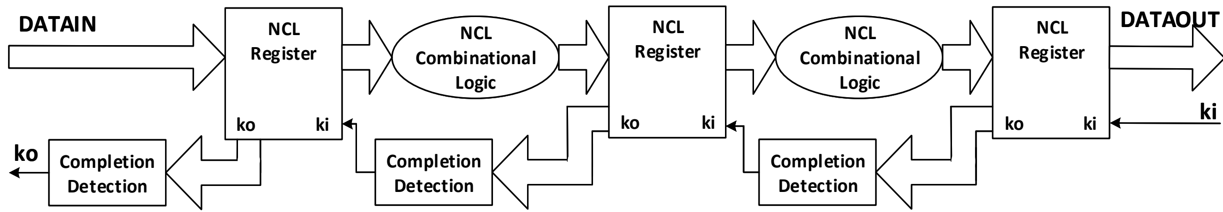


Figure 4: NCL micro-pipeline architecture [19].

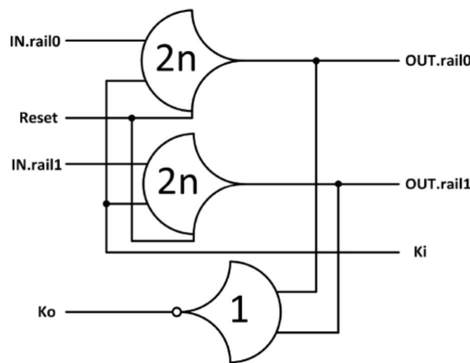


Figure 5: NCL single-bit dual-rail register [20].

These special registers are used to maintain delay-insensitivity and performs handshaking during sequential operations. An NCL register consist of resettable and inverting NCL gates. In the NCL pipeline architecture, one pipeline stage consists of an NCL combination block between two NCL registers. The registers between pipeline stages interact via *ki* and *ko* signals. NCL registers sends *ki* signal as input and *ko* signal as output. The *ki* represents the request signal and the *ko* represents the acknowledge signal. When a register's *ki* signal is logic 0, only DATA is allowed to pass. On the other hand, when the register's ki signal is logic 1, only NULL is allowed to pass. The acknowledge signal *ko* can represents a Request for NULL *(rfn)* via logic 0 and a Request for DATA *(rfd)* via logic l. In addition to registers, the NCL architecture also utilizes a special component block called completion detection. The completion detection

consists of a tree of C-elements and is used for data validation. A completion detection is required when multiple single registers are combined to form an *n*-bit register.

There are two important requirements when designing an NCL requirement: input-completeness [21] and observability [22]. Input-completeness requires that all outputs of the combinational blocks do not transition until all the inputs arrive. Observability requires that no orphans may propagate through a gate [23]. An orphan is defined as a wire that transitions during the current DATA cycle but is not used to determine the output. The observability condition ensures that every gate that transitions is necessary to transition at least one of the outputs.

Compared to a synchronous circuit design, an asynchronous circuit paradigm such as NCL provides several advantages. Past scientific research has demonstrated that NCL circuit has the potential to have a wide-range of applications and benefits as described by the following:

- NCL circuits show resilience when considering performance-energy tradeoffs [24].
- Energy and performance advantages have been demonstrated in normal and near-threshold voltage regions [25].
- NCL circuits have shown promising application in hardware security [26], [27].
- NCL circuits have exhibited lower electromagnetic interference [11], and immunity to meta-stable behavior [28].
- The resilience of NCL circuits have been validated in a wide range of process, voltage, and temperature variations [29], [30] .

3.2 Multi-threshold NULL Convention Logic (MTNCL)

The advantages of NCL circuits could be improved by utilizing power gating into the NCL gates, creating a new logic paradigm called Multi-Threshold NCL (MTNCL). MTNCL is an optimized low power asynchronous paradigm that achieves better performance-energy tradeoff

when compared to NCL. This is achieved by reducing static power during quiescent NULL states while maintaining high performance during DATA propagation. The static power consumption is reduced by implementing sleep-able MTNCL gates during NULL states. Multi-threshold technology is commonly used in synchronous designs for power-gating mechanism by utilizing several transistors with varying threshold voltages (V_{th}). Figure 6 shows a typical power-gating structure for CMOS technology. The threshold voltage of a field-effect transistor (FET) is the minimum gate-to-source voltage potential that is required to create a conducting path between the source and drain terminals. The threshold voltage depends on the choice of the oxide and on oxide thickness. Low-V_{th} transistors are faster but have higher subthreshold leakage current running through the device, while high-V_{th} transistors are slower but have less leakage current [31].

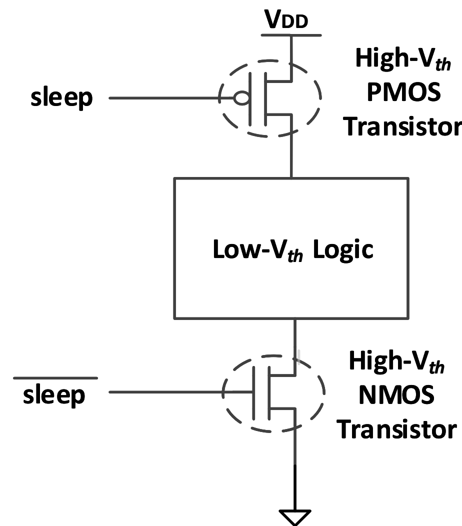


Figure 6: Multi-threshold CMOS power gating structure.

MTNCL gates have reduced leakage current during idle states while also maintaining performance while processing data [32]. High-V_{th} transistors are used to reduce subthreshold leakage during idle states, while low-V_{th} transistors are used to maintain performance while processing data. The high-V_{th} transistors are controlled by the *sleep* signal. When the *sleep* signal

is logic 0, the MTNCL gate is in active mode and the gate functions normally. When the sleep signal is logic 1, the MTNCL gate is in sleep mode and the output low-V_{th} transistor is turned on to set the output of the MTNCL gate to logic 0, while the high-V_{th} transistors are turned off to reduce subthreshold leakage. A more fined grained MTNCL power gating structure shown on Figure 7 have been proposed to eliminate the output wake-up glitch and provide several improvements.

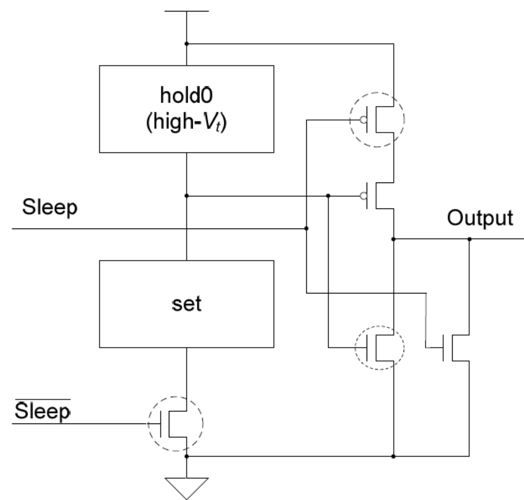


Figure 7 Fined Grained MTNCL power gating structure [33].

The sleep mode and active mode of an MTNCL gate is based on the NCL DATA-NULL cycles. The sleep mode of an MTNCL circuit is equivalent to the NULL cycle while the active mode is equivalent to the DATA cycle. As shown on Figure 7 of the improved MTNCL gate power gating structure, the reset block and its corresponding complement hold1 block has been removed from the NCL gate structure because the sleep mode can generate a NULL cycle. With these changes, MTCNL gates have fewer transistors than NCL gates. In addition, the input-completeness and observability requirements from NCL circuit design can be eliminated. The hysteresis property of NCL gates is also only required by a small subset of MTNCL gates [26].

3.2.1 Multi-Threshold NCL Pipeline

The MTNCL pipelining framework is very similar to the NCL counterpart. A pipeline stage consists of a combination logic block with two sets of MTNCL registers. Figure 8 shows the MTNCL pipeline architecture.

Figure 8: Multi-threshold NCL pipeline architecture.

The MTNCL registers are now sleep-able and are used to generate a NULL cycle. When the MTNCL combinational logic are set to sleep mode, the outputs of the gates are set to logic 0. This is equivalent to a NULL state. The completion detection has also been modified and improved by utilizing early completion [34]. Early completion detection, shown on Figure 9, is used to improve the throughput of the pipeline while maintaining delay-insensitivity. It is also used to ensure that all the outputs of a pipeline stage are NULL before allowing the DATA cycle to pass.

The corresponding request and acknowledge signals *ki* and *ko* are still required for the handshaking protocol, but they are also used to control the sleep signal. The output of the completion logic *ko* signal is used to sleep the MTNCL registers, completion logic, and the combination block of the next stage in the pipeline.

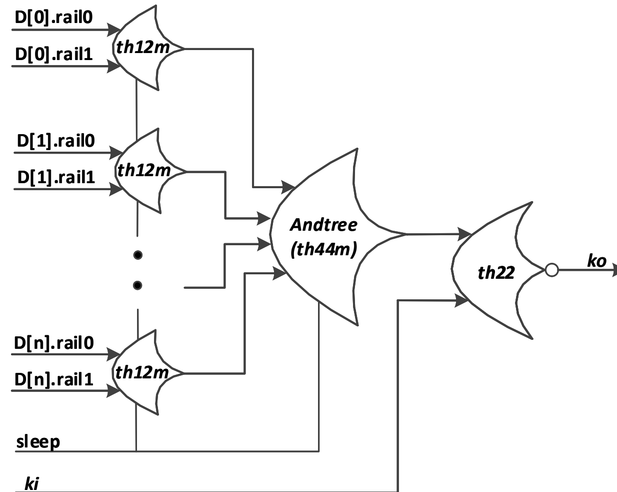


Figure 9 Early completion detection block in MTNCL pipeline.

After the first DATA cycle, the output *ko* signal of the completion logic will be logic 0 which represents a Request for NULL. This *ko* signal activates the active mode for the next stage in the pipeline and allows the combination logic to propagate the input data. This *ko* signal remains de-asserted until a NULL cycle is detected on the input ports and the completion logic is forced to sleep. When the output of the completion logic is logic 1, representing a Request for DATA, it also activates sleep mode for the next stage of the pipeline, propagating the NULL cycle. The DATA-NULL cycle propagation continues until a valid data is present on the output port.

There are several major differences between NCL and MTNCL and they can be summarized by the following: (1) sleep signal controls the local handshaking between pipeline stages in MTNCL pipeline, (2) MTNCL have lower leakage power consumption due to power gating (3) MTNCL gates forego the NCL hysteresis property.

3.3 Three Dimensional Integrated Circuits

Three-dimensional integrated circuit (3D IC) is a circuit manufactured by 3D stacking multiple silicon wafers/die into one device. The main enabler of such technology is the through-silicon via (TSV) that enables vertical interconnects between the different layers. 3D ICs with TSVs offer new levels of efficiency, power, performance, and form-factor advantages to the semiconductor industry. As an emerging technology, 3D IC promises to leap forward the Semiconductor Industry beyond the challenges of Moore's law. As demand for increasing density, higher bandwidths, and lower power intensifies, 3D IC integration provides a potential solution to these problems by packing in a great deal of functionality in a small form factor. 3D IC also provides designers with the freedom to create multiple heterogeneous die on the same circuit. Circuits as logic, memory, analog, RF, and micro-electrical mechanical systems (MEMS) could be integrated into one 3D IC with different process nodes suitable for each component. For example, a 3D IC could harness the cutting-edge advantages of a 28nm process node for high-speed logic while using legacy 130nm process node for analog components. Heterogeneous integration could alleviate cost associated with node migration, and allow designers to take advantage of existing legacy nodes.

The impact of 3D IC is very broad. Applications of 3D IC are prominent in areas such as networking, graphics, mobile communications, and computing. Considering of the current trends where mobile devices are pervasive, 3D IC could cause a significant paradigm shift to the way ICs are manufactured. One area where 3D IC could be a potential alternative is System-on-Chip (SoC). SoC in the modern world has allowed the advent of "smart" devices where different components are integrated as one unit to save floor space, power, and increase bandwidth. 3D IC allows the same integration in an even smaller area, lower power, and faster bandwidth. The

most exciting and more useful advantage of 3D IC over conventional 2D process is the reduction in the interconnect length between components. Such reduction will be most advantages in circuits such as microprocessors, memory, and Field Programmable Gate Arrays (FPGA).

3.3.1 Motivation

There are several advantages 3D ICs have over traditional 2D circuits. The advantages could be summarized by the following:

- Shorter interconnect length
- Increased interconnect bandwidth
- Heterogeneous Integration
- Smaller form factor

Due to the closer integration of components, the bandwidth could be significantly improved. 3D ICs are ideal for compact mobile devices due to the space saving on the board. In addition, there is potential for power reduction on 3D ICs due to the reduction in resistance-inductance-capacitance (RLC) on the interconnect between devices. The interconnects between components are reduced and can improve the overall performance of chip. 3D IC provides the ability to create modularize components in different process nodes through heterogeneous integration.

When compared to other technologies such as SoC/System-in-Package (SiP), 3D IC offers improved level of integration on a smaller form factor. Figure 10 shows the industry trend of integration from traditional systems to SoC, and finally to 3D IC in the future.

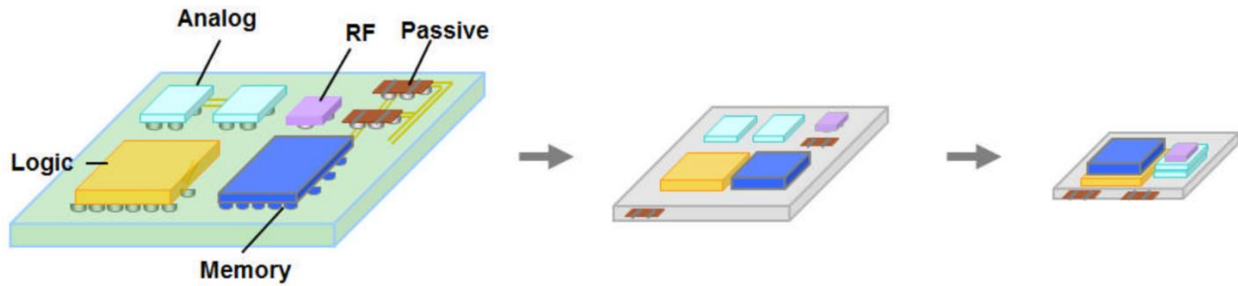


Figure 10: Trend leading towards 3D IC technology [35].

3D IC provides significant improvements over the traditional model, but there are additional challenges that must be solved. These challenges include thermal issues, timing, and power management concerns.

3.3.1.1 Shorter interconnect

The most valuable advantage of 3D IC is the reduction of interconnects. This is because as technology scaled down, wire has not scaled along with it. Two factors contribute to the delay of the overall IC, i.e., the switching speed and propagation delay. Unfortunately, narrower wire only results in increased resistance, while smaller pitches increase capacitance. Due to this fact, as technology advances to the sub-micron level, the RC delay has become the dominant factor over the switching delay. The active power (switching) consumption in microprocessor interconnect contribute greater than 50% of the overall power [36]. In addition, over 90% of the interconnect power is consumed by only 10% of the wires. In memory applications, an 8GB four-stack 3D DDR3 DRAM using TSVs can reduce standby power by 50% and active power by 25%. Another example is in microprocessors with a 3D implementation of the Pentium 4 gaining 15% performance improvement while simultaneously lowering power consumption by 15% at the cost of only 14°C rise in temperature. Figure 11 is a diagram showing the advantage of the TSV vertical interconnects over the traditional 2D wire interconnect.

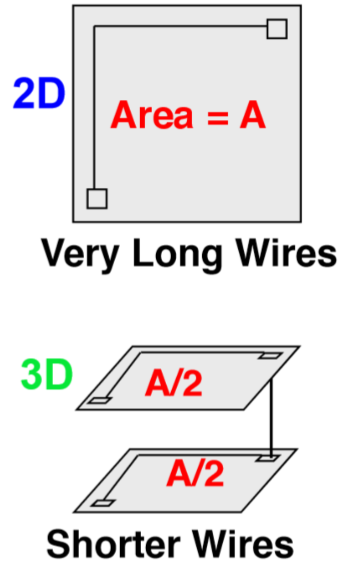


Figure 11: 2D versus 3D interconnect [36].

3.3.1.2 Wiring Delay

Shorter interconnects come with several improvements over the overall circuit. The speed of devices is improved and capacitance on the wire is reduced. Furthermore, this reduction in wire length reduces power consumption because less power is required to drive the wire. Figure 12 shows the comparison of the relative delays between interconnects and gate delays. The graph shows that as the feature size becomes smaller from 250nm to 32nm, the global interconnect dominates the delay, especially if repeaters are not used. The relative difference between the global interconnect versus the gate delay is in the order of several magnitudes.

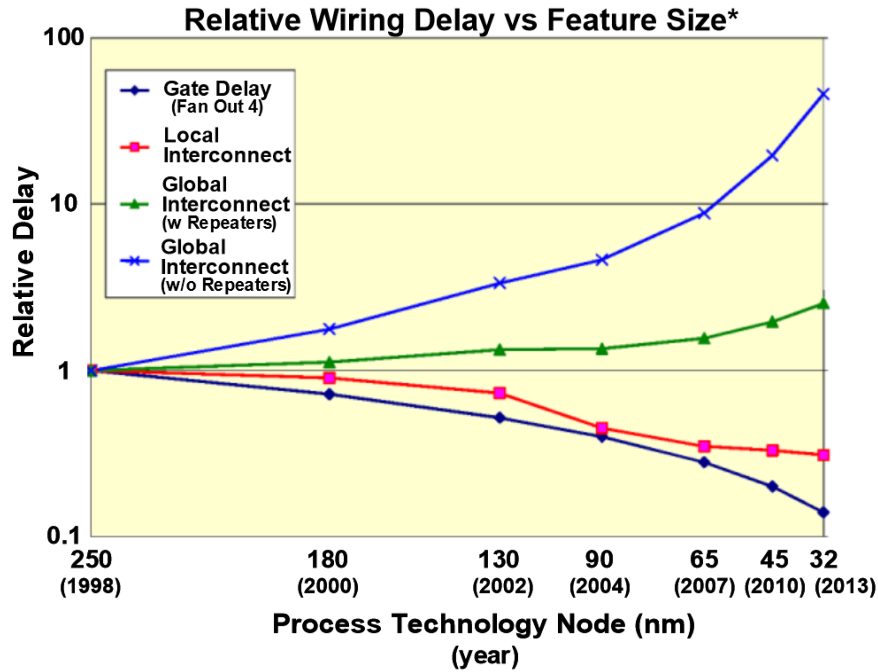


Figure 12: Global interconnect trends [36].

3.3.1.3 Memory Bandwidth

The memory bandwidth demands on computing systems are rapidly growing. The trend in mobile devices is the same, requiring 50 Gbps and higher. Graphics and networking can easily benefit from 1Tbps, as well as data-driven workloads requiring cross-system bandwidth. 3D IC can provide high memory bandwidth at better power efficiencies than conventional packaging.

3.3.1.4 Bandwidth density

3D IC integration includes vertical vias to connect the different layers, allowing the construction of wide-bandwidth buses between different functional blocks. The typical application of a stack layer is the integration of a processors and a memory on a 3D stack. This arrangement could potentially solve the memory wall problem. The memory wall or the bandwidth gap is a known issue in industry. Due to the dramatic difference between the processing speed of the processor and the memory, the memory has been the bottleneck in terms

of performance. Figure 13 shows the potential bandwidth between different technologies with 3D IC showing the highest potential, reaching 1 Tbps/mm^2.

Technology	Approximate limit
High density laminate	~ 50 Gbps/mm
Silicon interposer	~ 150 Gbps/mm
3DIC - microbumps	> 10 Tbps/mm ^{2}
3DIC - TSV	> 1 Tbps/mm ^{2}

Figure 13: 3D IC bandwidth potential [37].

3.3.1.5 Heterogeneous Integration

Heterogeneous integration allows a single silicon die to include an incredible amount of functionality by including processors, digital logic, memory, and analog components. Furthermore, 3D IC integration has an advantage over traditional SoCs by utilizing different process nodes. Traditionally, analog and RF designs in advanced process nodes are challenging and could potentially require significant amount of time to develop and test. This is further exasperated by different process variabilities. Compared to the traditional implementation, 3D IC could integrate different functional layers based on optimized process nodes. For example, a digital logic in an advanced node (e.g., 22 nm) could be placed on top of analog circuits in a legacy node (e.g., 180 nm).

3.3.1.6 Small Form Factor

Due to the 3D stacking, 3D ICs are compact and can be designated to have multiple layers. An example of a 3D stacked device consisting of different types of components is shown in Figure 14. The circuit consist RF sensors/harvester, low-power ASIC, memory, and analog devices. TSVs provide the vertical interconnect between these layers.

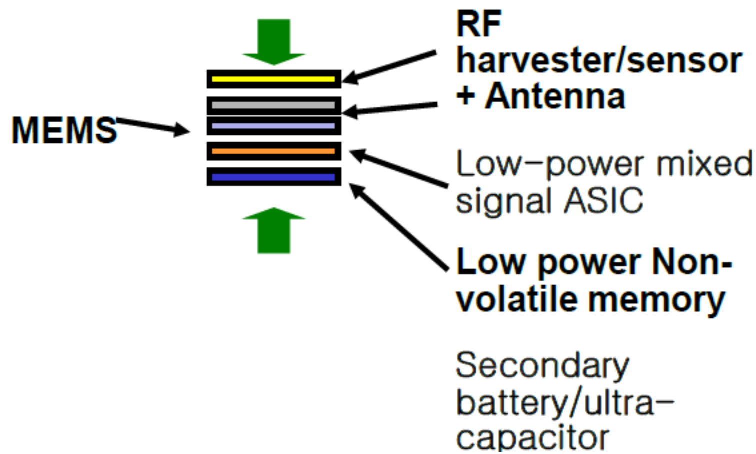


Figure 14: Small form factor consisting of different layers [38].

3.4 TSV Process Flow

TSV has been the main enabler of 3D IC integration. TSVs are electrically conducting electrical connections that allow multiple layers on a chip to communicate. TSVs are copper vias with diameters that may range from 1 to 30 microns. Figure 15 shows the cross-section of a 3D die with TSV.

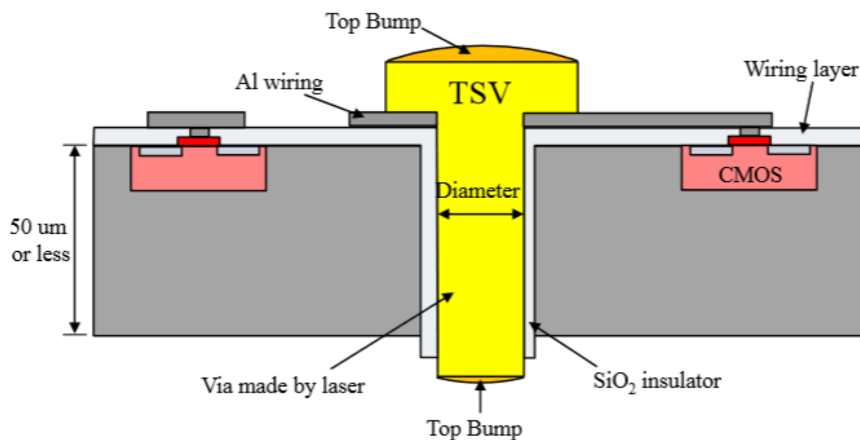


Figure 15: 3D cross-section of a TSV [39].

First step in the TSV creation is the etching process to create an etched hole where a TSV will be placed. The etching process is done through deep reactive-ion etching. In addition, sidewall passivation is performed where an insulation layer is created around the etched region to separate

the TSV from the bulk. TSV filling is accomplished by a vapor deposition process. The conductive fill is typically copper or tungsten. Afterwards, wafer thinning is done by grinding and chemical mechanical planarization (CMP). Finally, alignment is one of the last critical steps before permanent bonding occurs.

After the creation of each layer, the 3D stacked using three possible configurations: face-to-face, face-to-back, and back-to-back. Different configurations are possible in single devices, especially when more than two dies are stacked.

3.4.1 Manufacturing technologies for 3D IC

There are three possible integration methods for 3D ICs: wafer-to-wafer (W2W), die-to-wafer (D2W), or die-to-die (D2D). W2W involves directly bonding the wafer stacks together and then dicing them into individual die stacks. This bonding process is associated with the highest throughput, but requires the different layers to have the same form factor. Due to this requirement, W2W is more suitable for homogeneous integration, e.g., 3D stacked memory. In addition, W2W could suffer from significant yield loss due to the stacking of good and bad die. In comparison, D2W and D2D hope to resolve these issues by dicing the bare die from the wafer and then stacking them on die/wafers. These bonding strategies could achieve higher stack yield by only bonding known good die. Figure 16 shows the three types of bonding process.

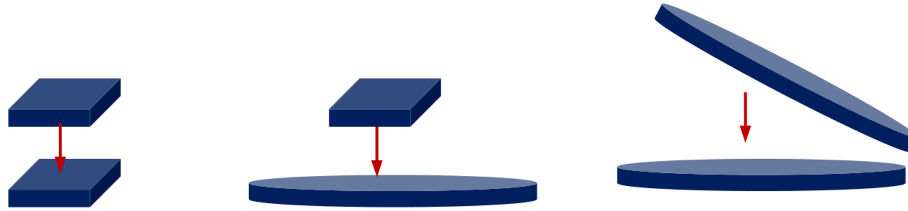


Figure 16: Manufacturing 3D ICs [35].

3.5 Challenges

When compared to the logic gates and other circuit features, the TSV is very large and requires careful consideration on placement and usage. Excessive use of TSVs could be detrimental and result in increased wire length. Due to coupling, TSVs also require keep-out zones, reducing mechanical stress placed on nearby devices. This in turn causes thermal hot spots; therefore, careful floor planning is necessary to find optimal TSV placements on the die. Thermal issues and mechanical stress could affect the performance of the device. Figure 17 shows the thermal hotspots of a 3D IC denoted in bright red. In addition, die stack order is important as the middle stack is more susceptible to thermal problems due to the reduce access to cooling. The bottom stack is generally closest to the cooling system and so the most thermally sensitive die is best placed at the bottom. The densely packed stack is also prone to generate a lot of heat. There are several solutions to the thermal issues. Air-cooling is possible but it requires active power management. Another potential solution is liquid cooling. Another viable solution is to evenly distribute the power density throughout the chip to prevent the hotspot problem. Previous research has explored the application of NCL circuits with 3D IC as a potential solution to the thermal hotspot issue [40]. Figure 18 shows the comparison between synchronous and NCL circuits in terms of thermal map distribution. The NCL circuit showed a more evenly distributed thermal map than the synchronous counterpart.

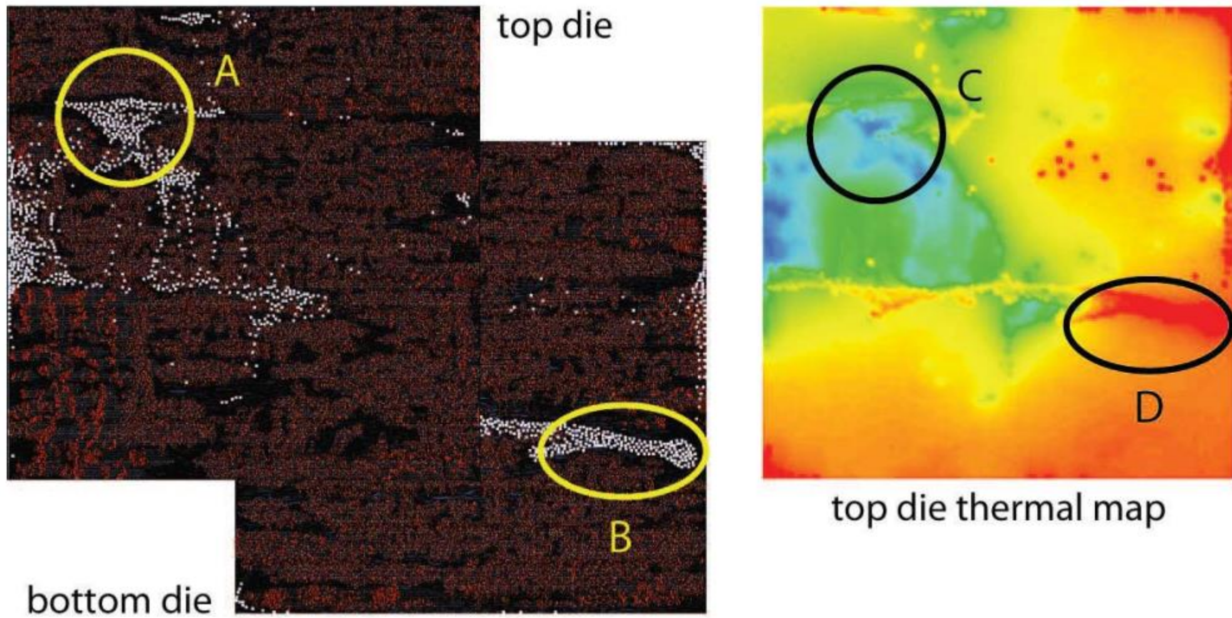


Figure 17: 3D IC thermal map [41].

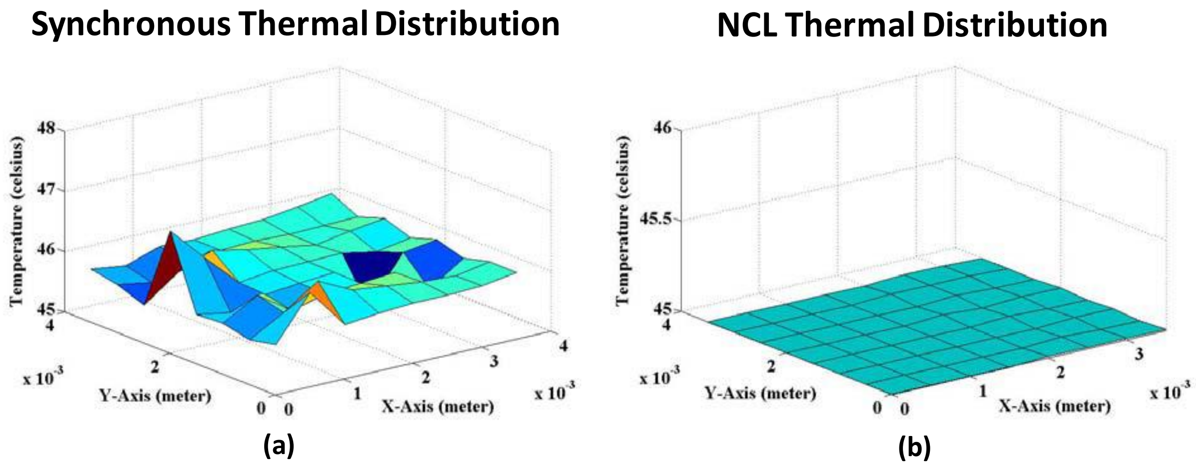


Figure 18: Synchronous vs NCL thermal distribution [40].

3D IC also comes with increased design complexity. Especially with heterogeneous integration, the 3D stack could combine digital and analog/RF circuitry, requiring tools with strong analog/mixed-signal capabilities. In addition, the manufacturing process demands the expertise in several domains from 3D manufacturing to 3D IC packing, requiring close collaboration and co-design between groups in the whole design chain.

Testing 3D ICs is a challenge due to the tight integration between active layers in the 3D stack. A significant amount of interconnect is associated between each layer for communication. Conventional testing methods do not account for the overhead associated with TSVs as well as the different modules on the stack. To achieve high yield rates and reduce cost, each die must be tested independently. One of the challenges in testing 3D IC is gaining access to the die inside the 3D stack. 3D IC testing included two levels, wafer test and package test. Wafer test involves testing each silicon die while package test involves testing after assembling the die into a package. Compared to the traditional testing method, 3D IC includes many more intermediate steps due to the die stacking and TSV bonding process. Wafer test is critical in cost reduction as it reduces the chance of a bad die being used in a package. A failed check could result in a failed package-level test.

3.6 Discussion

3D IC has shown significant improvements over the traditional paradigm of 2D counterpart. 3D IC integration offers improvements in power consumption, form factor, and performance. It also allows heterogeneous integration by incorporating different process nodes. While the advent of 3D IC integration has yet to be realized, it has shown potential in revolutionizing the way circuits will be design in the future. The application of 3D IC includes networking, graphics, mobile communications, and computing. However, the shift to 3D IC integration comes with several challenges. First, a 3D IC must be designed from the initial planning process to take full advantage of the 3D process. In addition, the associated cost in manufacturing 3D ICs must be reduced to encourage widespread use. Furthermore, thermal issue remains a concern and must be carefully evaluated. Testing 3D IC is another hot research topic.

4 Asynchronous 3D Tool Flow

4.1 Proposed Flow

This chapter presents the Async3D tool flow that was developed and used for this dissertation work. The proposed flow is shown on Figure 19. It consists of three main components: circuit design, physical design, and standard cell library development. The bottom and top layers of the 3D stack are called Tier0 and Tier1 in the tool flow.

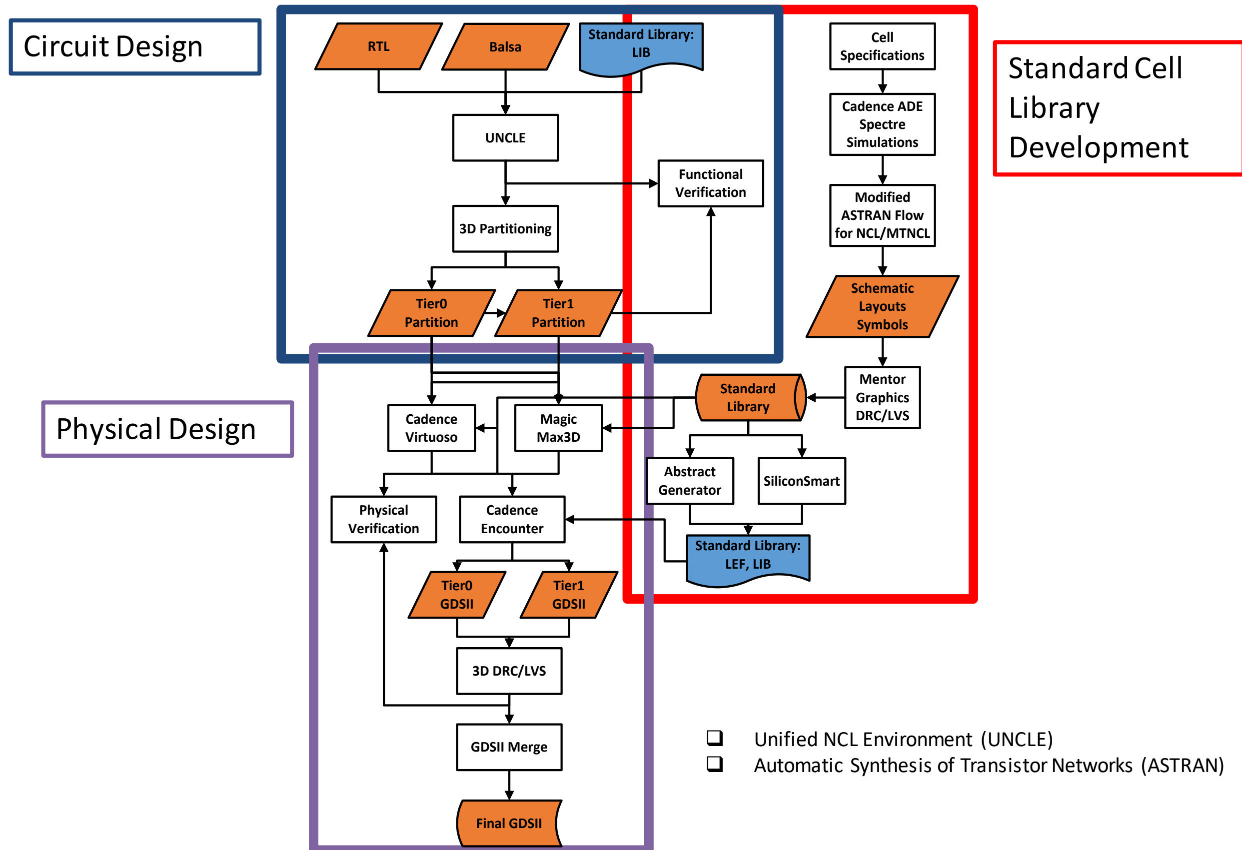


Figure 19: Async3D tool flow.

4.2 Circuit Design

4.2.1 Unified NCL Environment (UNCLE)

The Unified NCL Environment (UNCLE) is a toolset developed for the synthesis of dual-rail NCL designs from a Verilog RTL specification [42]. The tool supports two NCL based system implementations: data-driven approach and control driven approach. The data-driven approach utilizes NCL gates for registers while the control driven approach uses Balsa style registers and control. UNCLE features several optimizations including net buffering, latch balancing, relaxation, and cell merging. In addition, UNCLE allows logic synthesis via commercial toolset such as Synopsys Design Compiler, and can automatically generate the dual-rail expansion and acknowledge pipelines. Figure 20 shows the full UNCLE tool flow.

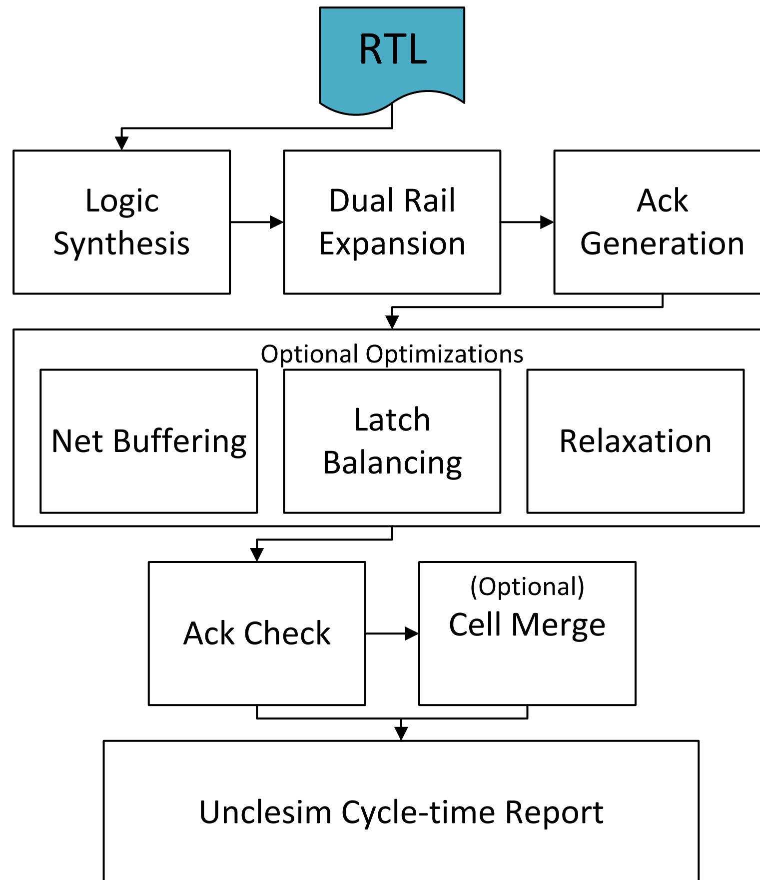


Figure 20: Uncle tool flow overview [43] .

UNCLE provides full support for NCL designs. As of this writing, the latest version 0.2.6 can generate a netlist from an NCL design with some limited support for MTNCL designs.

4.2.2 3D IC Partitioning

After the synthesis, the netlist must be partitioned to create the top and bottom netlist. Synopsys Design Compiler was used for the 3D IC partitioning phase of the tool flow. Previous work has proposed a set of scripts for 3D IC partitioning for NCL and MTNCL circuits [44]. Their approach used python scripts to parse through the netlist and create TSV components between the partitioned netlist. While this automated the 3D partitioning, the partitioning scripts became hard to maintain and produced netlist that were very difficult to debug and verify. One of

the issue with the 3D IC partitioning phase is that the post-synthesis netlist is often flattened, removing all hierarchies. Partitioning this manually would be error prone and laborious. The Async3D flow leverages existing industry tools to partition the netlist. It takes advantage of the naming convention used in the flattened netlist. By using regular expressions in conjunction with the group and ungroup feature of Design Compiler, the design structure could be restored and the 3D netlist for the top and bottom tiers can be created in a more convenient manner. In addition, this method has the benefit of easier debugging.

4.2.3 Automatic Synthesis of Transistor Networks (ASTRAN)

The Async3D tool flow utilizes an automated layout generation tool to generate the cell libraries. Automatic Synthesis of Transistor Networks (ASTRAN) is an open-source tool used for physical synthesis [45]. ASTRAN can generate the layouts of CMOS cells using a transistor-level netlist specification. The cell netlist must follow the SPICE format. The area overhead of ASTRAN versus an optimized hand-made layout is around 3.7% on average. ASTRAN provides several features including transistor sizing, floorplanning, cell placement, and routing. The layout tool also includes several optimizations such as transistor folding, intracell routing, and 2-D layout compaction. The layouts can be exported in CIF, GDSII, and LEF formats. These formats can then be imported to Cadence Virtuoso for viewing or further optimizations. The version used in this dissertation has been modified to support the cell dimensions and specifications described on the Async3D library section. In addition, the code has been improved to support multi-threshold transistors used for MTNCL. Figure 21 shows an example of an ASTRAN style layout. The project is active with plans to support 45nm process and beyond.

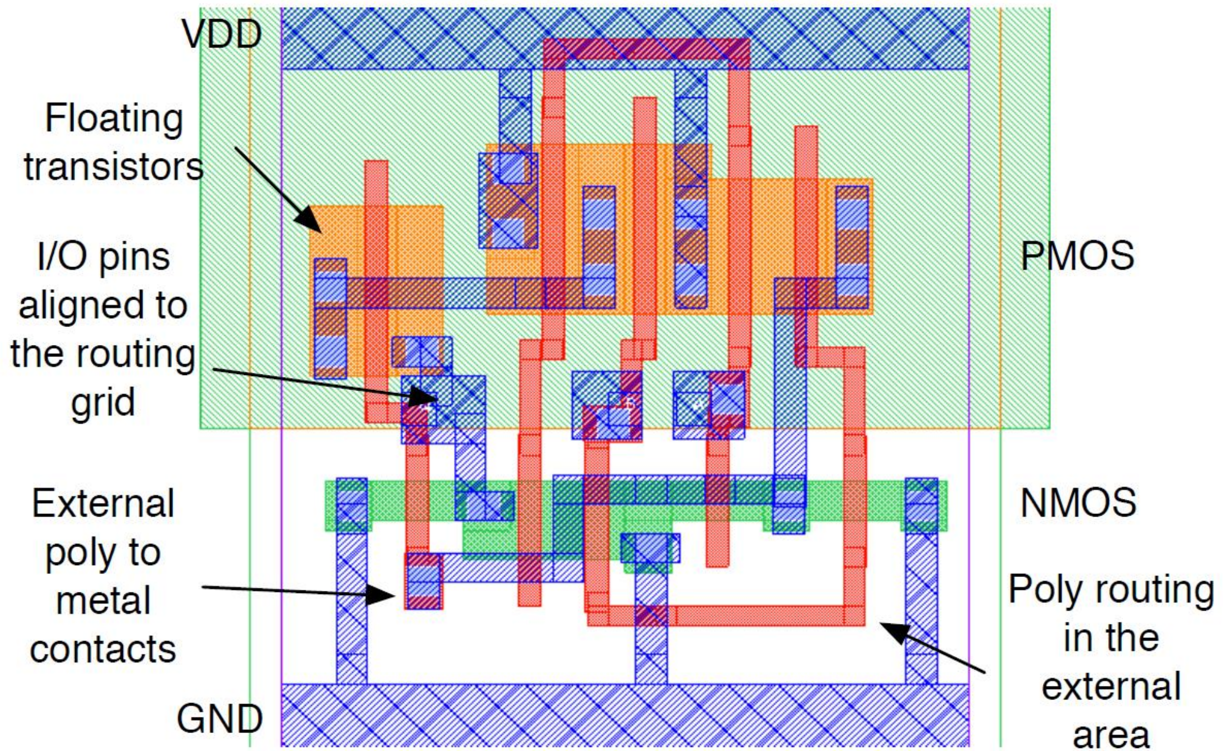


Figure 21: ASTRAN layout style [45].

4.3 Physical Design

The main tool used for physical design phase was Cadence Innovus. The latest version of Innovus has included some features for 3D IC design. After Tier0 and Tier1 are placed and routed, the GDSII are extracted. This post-layout file can then be imported back to Cadence Virtuoso for verification or merged to create final GDSII. For the Async3D flow, a 3D DRC and LVS are included the foundry kit for verification. The 3D DRC and LVS utilizes Mentor Calibre for post-layout verification. After post-layout verification, the two files are merged using a tool provided in the foundry kit to create the final GDSII combining Tier0 and Tier1.

4.4 Standard Library Flow

This section presents the Async3D standard library. The standard library could be divided into three main parts: cell library specification, cell creation, and cell verification. The standard library tool flow is shown on Figure 22.

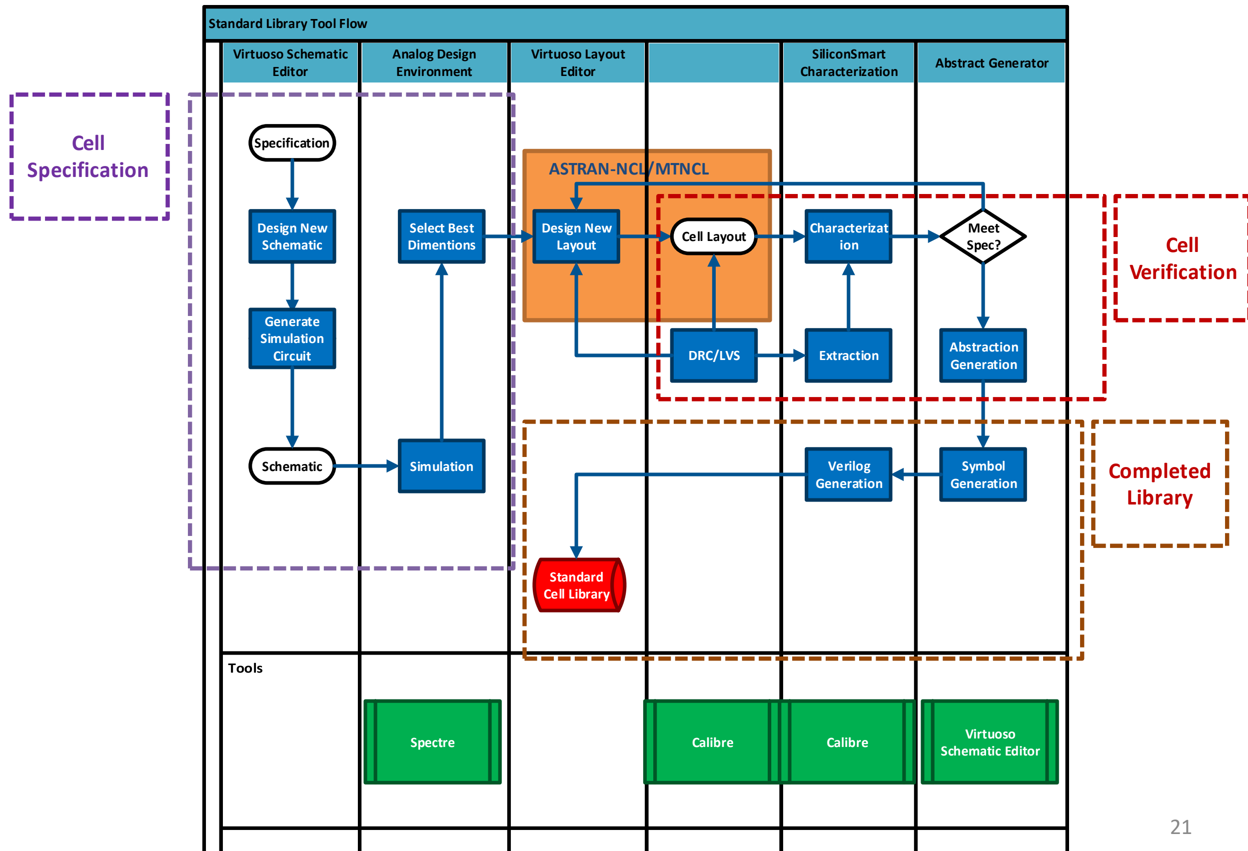


Figure 22: Async3D standard library tool flow.

4.4.1 Tezzaron 3D Design Kit

The 3D process design kit used for this dissertation is the Tezzaron 3D design kit. It supports a 2-layer stacked device based on the Chartered 130nm Low Power process. The 2-layer devices are stacked top of each other to create a 3D stack. There is an option to create a 2-layer stack as well as an additional 3rd layer for the use of a memory module. This dissertation work uses the 2-layer stack configuration.

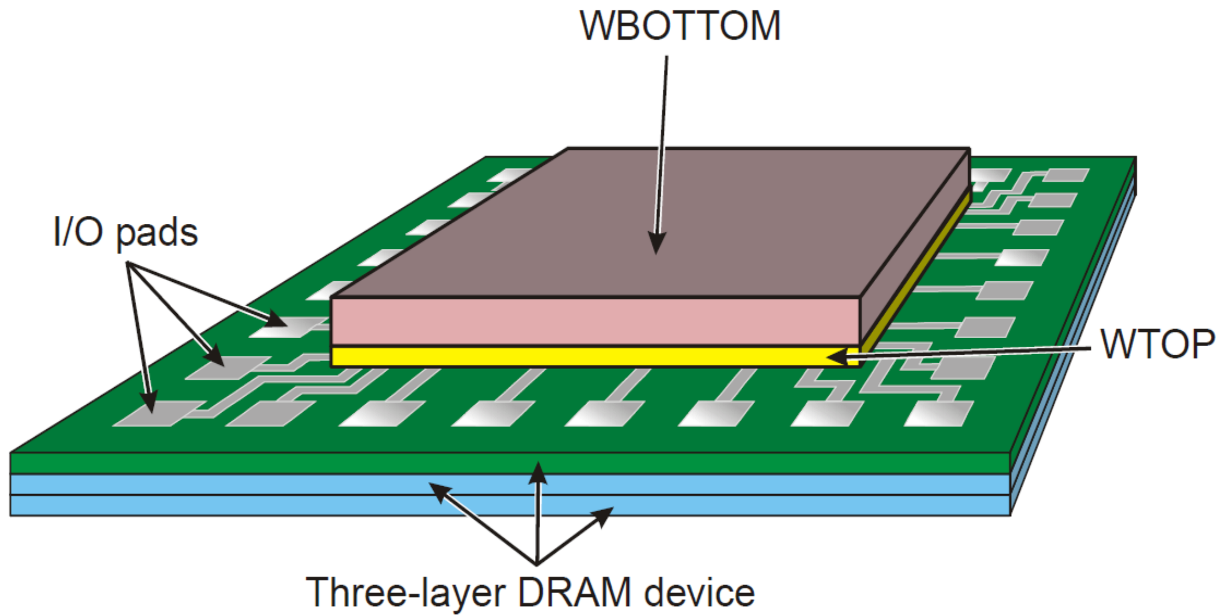


Figure 23: Tezzaron 3D stack layer [46].

The 2-layer logic stacks are connected in a face-to-face (wafer-to-wafer) configuration via Cu-to-Cu thermocompression process. The two logic die on Figure 23 are aptly named WBOTTOM and WTOP. A vertical cross-section of the 3D stack is shown on Figure 24. WTOP will be thinned to expose the TSV that will be connected to the memory. The number of metals used is 6 metal layers, with Metal 6 being used as a layer for the micro-bumps.

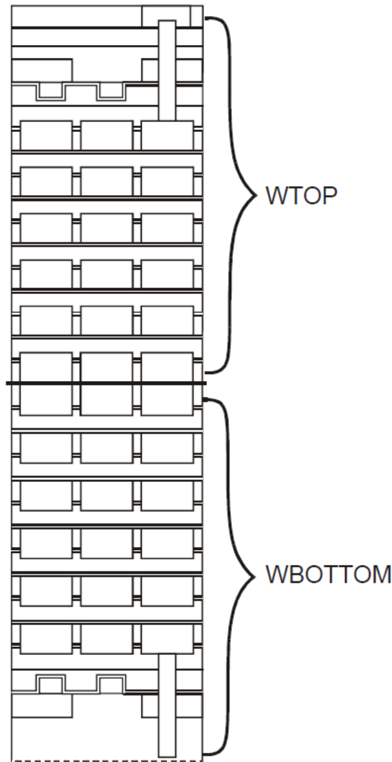


Figure 24: Two-wafer logic stack [46].

The maximum TSV spacing is 1000 microns and the minimum spacing is 100 microns. Higher TSV density is recommended for 3D stacks that have more layers to allow more surface for bonding. The diameter of the micro-bumps is 1.2 microns with a depth of 6 microns. The parasitic capacitance between the face-to-face interconnects is negligible, and is essentially the same as the top metal layer.

The Chartered 130nm process consists of 1.5V and 3.3V transistors. This dissertation work uses the 1.5V transistors. In addition, low-V_{th} and normal-V_{th} transistors are used to implement the multi-threshold gate configurations.

4.4.1.1 NCL vs MTNCL Naming Convention

The naming convention follows THmnWw1w2…wR where:

- TH - stands for threshold

- *m* - denotes that at least *m* of the *n* inputs must be asserted before the output will become asserted
- *n* - number of inputs
- *W* - means the inputs are weighted
- *w*1 - weight of input1
- *w*2 - weight of input2
- *w*R - weight of inputR

For this dissertation, lowercase naming was used to resolve an issue with case-sensitivity in the VHDL and Verilog netlist. MTNCL cell names are derived from NCL cell names. Adding "_*m*" suffix denotes that the gate is an MTNCL gate. For example, TH12 becomes TH12*m*.

4.4.1.2 Cell Layout Dimensions

The cell layout dimensions were determined based on the foundry provided synchronous library. This was done such that the synchronous and asynchronous libraries are fully compatible. In addition, all cells can be abutted side by side, reducing total chip area. The cell layout requirements are shown on Figure 25. The cell layout dimensions are specified by the following rules:

- Height multiple of horizontal pitch: Height = 0.41×
- Width multiple of vertical pitch: Width = 0.46×

Pitch is defined as the distance from center to center of a layer. This dimension is important because it is used to determine the routing grid. On-grid pin placements reduce computation resource requirements during place and route. The cell height dimension is 3.69µm and is a multiple of the horizontal pitch. The cell width can vary as long as it is a multiple of the vertical pitch. The cell sizes are calculated from VDD and GND metal pitch. The power rails have a

fixed height of 500nm. The VDD and GND are placed on top and bottom of the cells. Figure 26 shows an example of the VDD and GND rails. In addition, Figure 27 shows a layout example for the bulk body contacts.

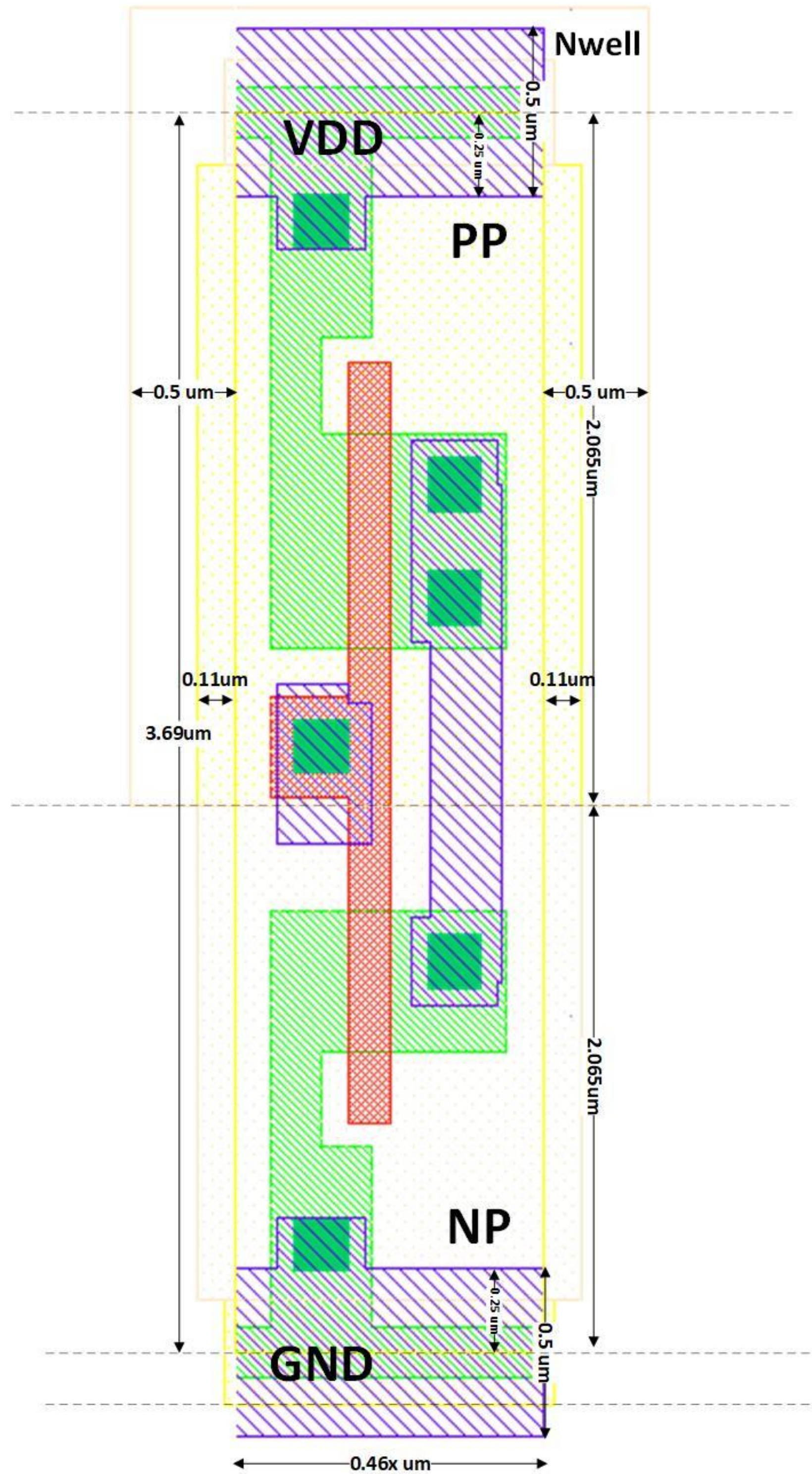


Figure 25: Async3D layout template.

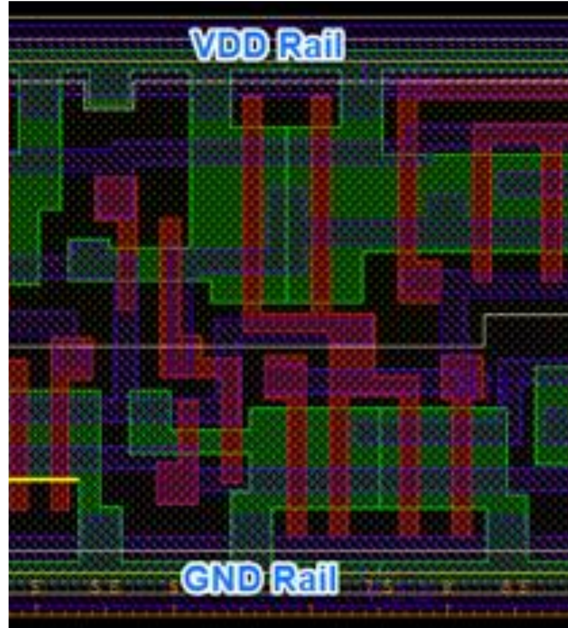


Figure 26: Power rails.

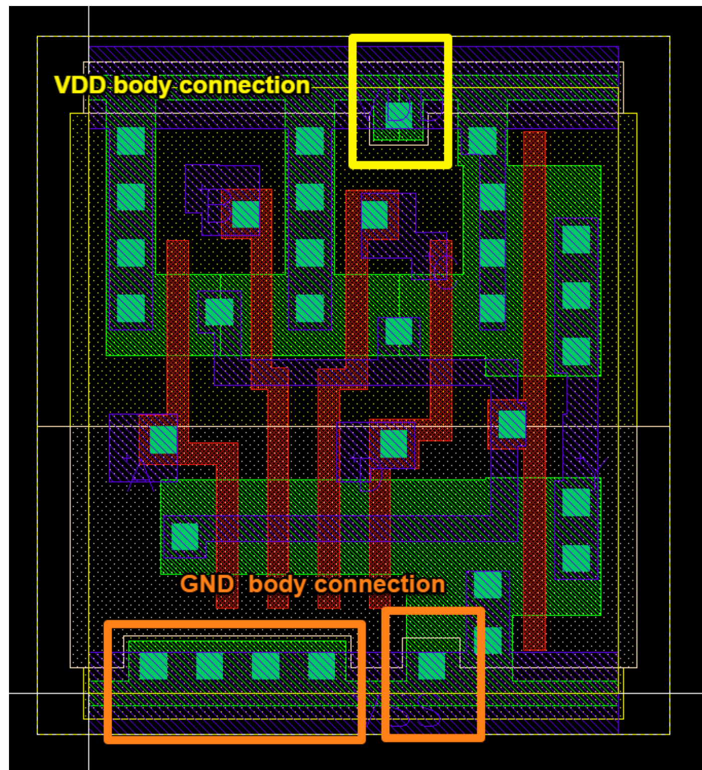


Figure 27: Bulk body contacts.

4.4.1.3 Cell Specification

Cadence Spectre was used to simulate the cells. The transistor gate sizes are shown on Table 3 and the library timing parameters are shown on Table 4. A load capacitance of 500 fF was used to characterize the cells for cell sizing. The variables represent the following: value of *riseTimeIZ* and *riseTimeZ* are the time required for *IZ* and *Z* nets to rise from 10% to 90% of VDD, while the value of *fallTimeIZ* and fallTimeZ is time required for *IZ* and *Z* net to fall from 90% of VDD. The net *IZ* represents the internal net driving the output inverter while net *Z* is the output of the inverter. These values are used to calculate the linearized load. The variables *riseKLoad* and *fallKload* represent the linearized load factor using rise and fall times. It is one of the metrics used to determine the drive strength of the cells. This value is based on the load capacitance, *riseTimeZ,* and *fallTimeZ*. This is derived from the CORELIB_LVT specification document. It is calculated using Equation 1. These metrics are matched to create well-balanced and power efficient cells.

Equation 1: Linearized load factor.

$$(a) \quad \text{riseKload} = \frac{riseTime}{2LoadCapacitance}$$

$$(b) \quad \text{fallKload} = \frac{fallTime}{2LoadCapacitance}$$

Table 3: Gate Transistor Sizing

Transistor Size	PMOS Size (um)	NMOS Size (um)
XL	0.3	0.16
X1	0.64	0.25
X2	1.28	0.5
X4	2.29	1
X6	3.84	1.5
X8	4.68	2

Table 4: Async3D library timing specifications.

Gate Size	riseTimeIZ (ns)	fallTimeIZ (ns)	riseTimeZ (ns)	fallTimeZ (ns)	riseKLoad (ns/pf)	fallKLoad (ns/pf)
XL	Less than 0.100 ns	Less than 0.100 ns	10 – 10.3	10 – 10.3	10 – 10.3	10 – 10.3
X1	Less than 0.100 ns	Less than 0.100 ns	4.8 – 5.2	4.8 – 5.2	4.8 – 5.2	4.8 – 5.2
X2	Less than 0.100 ns	Less than 0.100 ns	2.3 – 2.5	2.3 – 2.5	2.3 – 2.5	2.3 – 2.5
X4	Less than 0.100 ns	Less than 0.100 ns	1.2 – 1.4	1.2 – 1.4	1.2 – 1.4	1.2 – 1.4
X6	Less than 0.100 ns	Less than 0.100 ns	0.76 – 0.79	0.76 – 0.79	0.76 – 0.79	0.76 – 0.79
X8	Less than 0.100 ns	Less than 0.100 ns	0.61 – 0.63	0.61 – 0.63	0.61 – 0.63	0.61 – 0.63

4.4.2 Cell Schematics

Figure 28 shows an annotated schematic from a TH44 gate. The schematic transistors are broken down to set, reset, hold1, hold0, and output inverter. This schematic follows the NCL threshold gate specification discussed in Chapter 3.

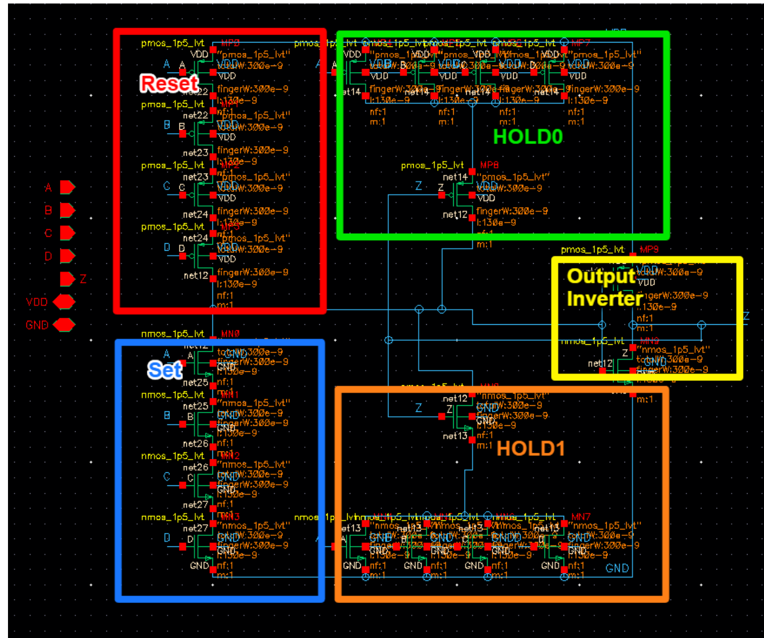


Figure 28: Annotated TH44 schematic implementation.

4.4.3 Cell Verification

Mentor Graphics Calibre was used for cell verification. The cell verification flow is shown on Figure 29. The verification phase includes design rule check (DRC), layout versus schematics (LVS), and parasitic extraction (PEX). DRC is the process that determines whether

the generated layouts meet the recommended physical design rules. This is a major step of the cell verification phase. Any problems in this step must be resolved before moving on to LVS. After a successful DRC, LVS must be passed. LVS is the phase where the verification tool determines whether the cell layout matches the schematic diagram. The final step is parasitic extraction where the parasitic effects are calculated from the devices and wire interconnects. This creates an accurate model of the circuit with signal delays, and it is used to generate the liberty characterization file.

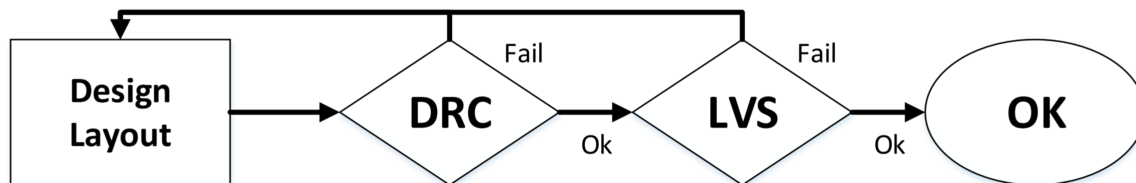


Figure 29: Cell verification flow.

4.5 Async3D Library Cell Area

Table 5-7 shows a full list of the completed Async3D NCL and MTNCL libraries. As of this writing, the library contains 6 different sizes ranging from the low power cells such as TH*mn*_XL and large drive strength cells such as TH*mn*_X8. The tables also include the post-layout cell area metrics. The special cells on Table 7 were developed to support the fined-grained MTNCL optimizations developed in [47].

Table 5: Async3D NCL library cell area.

Cell Name	Cell Height (um)	Cell Width (um)	Cell Area (um^2)
th12_X1	3.69	2.3	8.487
th12_X2	3.69	2.3	8.487
th12_X4	3.69	2.76	10.1844
th12_X6	3.69	4.14	15.2766
th12_X8	3.69	4.6	16.974
th12_XL	3.69	2.3	8.487
th13_X1	3.69	3.22	11.8818
th13_X2	3.69	2.76	10.1844
th13_X4	3.69	4.14	15.2766
th13_X6	3.69	4.6	16.974
th13_X8	3.69	5.98	22.0662
th13_XL	3.69	3.22	11.8818
th14_X1	3.69	3.68	13.5792
th14_X2	3.69	3.68	13.5792
th14_X4	3.69	6.9	25.461
th14_X6	3.69	5.52	20.3688
th14_X8	3.69	6.44	23.7636
th14_XL	3.69	3.68	13.5792
th22_X1	3.69	3.68	13.5792
th22_X2	3.69	3.68	13.5792
th22_X4	3.69	4.14	15.2766
th22_X6	3.69	5.98	22.0662
th22_X8	3.69	6.44	23.7636
th22_XL	3.69	3.68	13.5792
th23w2_X1	3.69	5.52	20.3688
th23w2_X2	3.69	5.52	20.3688
th23w2_X4	3.69	5.06	18.6714
th23w2_X6	3.69	5.98	22.0662
th23w2_X8	3.69	6.9	25.461
th23w2_XL	3.69	5.52	20.3688
th23_X1	3.69	5.98	22.0662
th23_X2	3.69	5.52	20.3688
th23_X4	3.69	7.36	27.1584
th23_X6	3.69	7.82	28.8558
th23_X8	3.69	10.12	37.3428
th23_XL	3.69	5.52	20.3688
th24comp_X1	3.69	6.9	25.461
th24comp_X2	3.69	6.44	23.7636
th24comp_X4	3.69	6.9	25.461
th24comp_X6	3.69	6.9	25.461
th24comp_X8	3.69	7.36	27.1584
th24comp_XL	3.69	5.98	22.0662
th24w22_X1	3.69	6.9	25.461
th24w22_X2	3.69	7.36	27.1584
th24w22_X4	3.69	8.74	32.2506
th24w22_X6	3.69	9.66	35.6454
th24w22_X8	3.69	10.12	37.3428
th24w22_XL	3.69	6.9	25.461
th24w2_X1	3.69	6.44	23.7636
th24w2_X2	3.69	6.44	23.7636
th24w2_X4	3.69	7.82	28.8558
th24w2_X6	3.69	8.28	30.5532
th24w2_X8	3.69	8.28	30.5532
th24w2_XL	3.69	6.9	25.461
th24_X1	3.69	9.2	33.948
th24_X2	3.69	8.74	32.2506
th24_X4	3.69	8.74	32.2506
th24_X6	3.69	9.2	33.948
th24_X8	3.69	9.66	35.6454
th24_XL	3.69	8.74	32.2506
th33w2_X1	3.69	5.98	22.0662
th33w2_X2	3.69	5.06	18.6714
th33w2_X4	3.69	6.44	23.7636
th33w2_X6	3.69	6.9	25.461
th33w2_X8	3.69	8.28	30.5532
th33w2_XL	3.69	5.06	18.6714
th33_X1	3.69	4.6	16.974
th33_X2	3.69	4.6	16.974
th33_X4	3.69	5.06	18.6714
th33_X6	3.69	7.36	27.1584
th33_X8	3.69	8.28	30.5532
th33_XL	3.69	4.6	16.974
th34w22d_X1	3.69	8.28	30.5532
th34w22d_X2	3.69	9.2	33.948
th34w22d_X4	3.69	9.2	33.948
th34w22d_X6	3.69	9.2	33.948
th34w22d_XL	3.69	8.74	32.2506

(a)

Cell Name	Cell Height (um)	Cell Width (um)	Cell Area (um^2)
th34w2_X1	3.69	7.36	27.1584
th34w2_X2	3.69	7.36	27.1584
th34w2_X4	3.69	10.12	37.3428
th34w2_X6	3.69	10.58	39.0402
th34w2_X8	3.69	12.88	47.5272
th34w2_XL	3.69	6.9	25.461
th34w32_X1	3.69	6.44	23.7636
th34w32_X2	3.69	5.52	20.3688
th34w32_X4	3.69	5.98	22.0662
th34w32_X6	3.69	6.44	23.7636
th34w32_X8	3.69	6.44	23.7636
th34w32_XL	3.69	5.98	22.0662
th34w3_X1	3.69	6.44	23.7636
th34w3_X2	3.69	6.44	23.7636
th34w3_X4	3.69	6.9	25.461
th34w3_X6	3.69	6.9	25.461
th34w3_X8	3.69	7.82	28.8558
th34w3_XL	3.69	5.98	22.0662
th34_X1	3.69	6.9	25.461
th34_XL	3.69	6.9	25.461
th44w22_X1	3.69	7.82	28.8558
th44w22_X2	3.69	7.82	28.8558
th44w22_X4	3.69	10.12	37.3428
th44w22_X6	3.69	10.58	39.0402
th44w22_X8	3.69	12.42	45.8298
th44w22_XL	3.69	7.82	28.8558
th44w2_X1	3.69	7.36	27.1584
th44w2_X4	3.69	13.8	50.922
th44w2_XL	3.69	7.36	27.1584
th44w322_X1	3.69	6.9	25.461
th44w322_X2	3.69	7.82	28.8558
th44w322_X4	3.69	11.04	40.7376
th44w322_X8	3.69	17.02	62.8038
th44w322_XL	3.69	6.9	25.461
th44w3_X1	3.69	6.44	23.7636
th44w3_X2	3.69	5.06	18.6714
th44w3_X4	3.69	6.9	25.461
th44w3_X6	3.69	8.28	30.5532
th44w3_X8	3.69	10.12	37.3428
th44w3_XL	3.69	5.06	18.6714
th44_X1	3.69	5.52	20.3688
th44_X2	3.69	5.98	22.0662
th44_X4	3.69	8.28	30.5532
th44_X8	3.69	9.66	35.6454
th44_XL	3.69	5.52	20.3688
th54w22_X1	3.69	6.44	23.7636
th54w22_X4	3.69	8.28	30.5532
th54w22_X6	3.69	8.74	32.2506
th54w22_X8	3.69	11.04	40.7376
th54w22_XL	3.69	7.36	27.1584
th54w322_X1	3.69	7.82	28.8558
th54w322_X2	3.69	7.36	27.1584
th54w322_X4	3.69	7.82	28.8558
th54w322_X8	3.69	9.2	33.948
th54w322_XL	3.69	7.36	27.1584
th54w32_X1	3.69	6.44	23.7636
th54w32_X2	3.69	5.98	22.0662
th54w32_X4	3.69	9.2	33.948
th54w32_X6	3.69	9.66	35.6454
th54w32_X8	3.69	11.04	40.7376
th54w32_XL	3.69	6.9	25.461
thand0_X1	3.69	6.9	25.461
thand0_X2	3.69	9.66	35.6454
thand0_X4	3.69	10.12	37.3428
thand0_X6	3.69	16.1	59.409
thand0_X8	3.69	19.78	72.9882
thand0_XL	3.69	6.9	25.461
thxor0_X1	3.69	6.9	25.461
thxor0_X2	3.69	6.44	23.7636
thxor0_X4	3.69	8.74	32.2506
thxor0_X6	3.69	10.12	37.3428
thxor0_X8	3.69	11.5	42.435
thxor0_XL	3.69	7.36	27.1584
bufSleep_X1	3.69	2.3	8.487
th12b_X1	3.69	3.68	13.5792
th22d_X1	3.69	5.06	18.6714
th22n_X1	3.69	6.44	23.7636

(b)

Table 6: Async3D MTNCL library cell area.

Cell Name	Cell Height (um)	Cell Width (um)	Cell Area (um^2)
th12m_X1	3.69	4.6	16.974
th12m_X2	3.69	6.44	23.7636
th12m_X4	3.69	8.28	30.5532
th12m_X6	3.69	10.12	37.3428
th12m_X8	3.69	12.42	45.8298
th12m_XL	3.69	3.68	13.5792
th13m_X1	3.69	5.06	18.6714
th13m_X2	3.69	7.36	27.1584
th13m_X4	3.69	10.12	37.3428
th13m_X6	3.69	11.04	40.7376
th13m_X8	3.69	14.26	52.6194
th13m_XL	3.69	4.14	15.2766
th14m_X1	3.69	7.36	27.1584
th14m_X2	3.69	11.04	40.7376
th14m_X4	3.69	13.8	50.922
th14m_XL	3.69	4.6	16.974
th22m_X1	3.69	5.06	18.6714
th22m_X2	3.69	6.9	25.461
th22m_X4	3.69	7.82	28.8558
th22m_X6	3.69	9.2	33.948
th22m_X8	3.69	10.58	39.0402
th22m_XL	3.69	4.14	15.2766
th23m_X1	3.69	6.44	23.7636
th23m_X2	3.69	11.5	42.435
th23m_X4	3.69	11.96	44.1324
th23m_XL	3.69	5.98	22.0662
th23w2m_X1	3.69	7.36	27.1584
th23w2m_X2	3.69	8.28	30.5532
th23w2m_X4	3.69	9.2	33.948
th23w2m_XL	3.69	5.98	22.0662
th24compm_X1	3.69	6.44	23.7636
th24compm_X2	3.69	7.36	27.1584
th24compm_X4	3.69	9.2	33.948
th24compm_XL	3.69	6.44	23.7636
th24m_X1	3.69	9.66	35.6454
th24m_X2	3.69	11.5	42.435
th24m_X4	3.69	11.96	44.1324
th24m_XL	3.69	9.2	33.948
th24w22m_X1	3.69	6.44	23.7636
th24w22m_X2	3.69	9.66	35.6454
th24w22m_X4	3.69	14.26	52.6194
th24w22m_XL	3.69	5.52	20.3688
th24w2m_X1	3.69	8.28	30.5532
th24w2m_X2	3.69	9.2	33.948
th24w2m_X4	3.69	10.58	39.0402
th24w2m_XL	3.69	7.82	28.8558
th33m_X1	3.69	5.98	22.0662
th33m_X2	3.69	6.44	23.7636
th33m_X4	3.69	9.2	33.948
th33m_XL	3.69	3.68	13.5792
th33w2m_X1	3.69	6.9	25.461
th33w2m_X2	3.69	6.9	25.461
th33w2m_X4	3.69	8.28	30.5532
th33w2m_XL	3.69	5.52	20.3688
th34m_X1	3.69	8.74	32.2506
th34m_X2	3.69	9.66	35.6454

(a)

Cell Name	Cell Height (um)	Cell Width (um)	Cell Area (um^2)
th34w22dm_X1	3.69	8.74	32.2506
th34w22dm_X2	3.69	10.12	37.3428
th34w22dm_XL	3.69	8.28	30.5532
th34w2m_X1	3.69	8.28	30.5532
th34w2m_X2	3.69	9.2	33.948
th34w2m_X4	3.69	14.26	52.6194
th34w2m_XL	3.69	7.82	28.8558
th34w32m_X1	3.69	6.44	23.7636
th34w32m_X2	3.69	7.36	27.1584
th34w32m_X4	3.69	8.74	32.2506
th34w32m_XL	3.69	6.44	23.7636
th34w3m_X1	3.69	6.44	23.7636
th34w3m_X2	3.69	7.82	28.8558
th34w3m_X4	3.69	9.2	33.948
th34w3m_XL	3.69	5.52	20.3688
th44m_X1	3.69	6.44	23.7636
th44m_X2	3.69	8.28	30.5532
th44m_X4	3.69	10.12	37.3428
th44m_XL	3.69	5.98	22.0662
th44w22m_X1	3.69	7.36	27.1584
th44w22m_X2	3.69	8.74	32.2506
th44w22m_X4	3.69	12.88	47.5272
th44w22m_XL	3.69	5.98	22.0662
th44w2m_X2	3.69	8.28	30.5532
th44w2m_X4	3.69	14.26	52.6194
th44w2m_XL	3.69	6.44	23.7636
th44w322m_X1	3.69	8.28	30.5532
th44w322m_X2	3.69	9.66	35.6454
th44w322m_X4	3.69	9.66	35.6454
th44w322m_XL	3.69	7.36	27.1584
th44w3m_X1	3.69	6.44	23.7636
th44w3m_X2	3.69	7.82	28.8558
th44w3m_X4	3.69	8.74	32.2506
th44w3m_XL	3.69	5.06	18.6714
th54w22m_X1	3.69	7.82	28.8558
th54w22m_X2	3.69	7.36	27.1584
th54w22m_X4	3.69	10.58	39.0402
th54w22m_XL	3.69	6.44	23.7636
th54w322m_X1	3.69	7.82	28.8558
th54w322m_X2	3.69	8.74	32.2506
th54w322m_X4	3.69	10.12	37.3428
th54w322m_XL	3.69	6.44	23.7636
th54w32m_X1	3.69	6.44	23.7636
th54w32m_X2	3.69	7.36	27.1584
th54w32m_X4	3.69	9.2	33.948
th54w32m_XL	3.69	4.6	16.974
thand0m_X1	3.69	7.36	27.1584
thand0m_X2	3.69	11.04	40.7376
thand0m_X4	3.69	14.26	52.6194
thand0m_XL	3.69	6.44	23.7636
thxor0m_X1	3.69	6.9	25.461
thxor0m_X2	3.69	7.36	27.1584
thxor0m_X4	3.69	11.5	42.435
thxor0m_XL	3.69	5.52	20.3688
th34m_X4	3.69	11.96	44.1324
th34m_XL	3.69	7.82	28.8558

(b)

Table 7: Async3D MTNCL special cells area.

Cell Name	Cell Height (um)	Cell Width (um)	Cell Area (um^2)
ANDc_X1	3.69	7.36	27.1584
NANDc_X1	3.69	7.36	27.1584
th12dm_X1	3.69	4.6	16.974
th12m_const_1_X1	3.69	3.68	13.5792
th12nm_X1	3.69	4.14	15.2766
th22m_const_0_X1	3.69	4.6	16.974
th23m_const_0_X1	3.69	5.98	22.0662
th23m_const_1_X1	3.69	5.06	18.6714
th23_noSleep_X1	3.69	4.14	15.2766
th24compm_const_0_X1	3.69	5.98	22.0662
th24compm_const_1_X1	3.69	5.06	18.6714
th34w2m_const_0_X1	3.69	7.36	27.1584
th34w2m_const_1_X1	3.69	5.98	22.0662

5 Case Study: Finite Impulse Response Filter

In digital signal processing, a finite impulse response (FIR) filter is a filter which responds to a finite length input in finite duration. This means that the impulse response eventually settles down to zero. An FIR filter structure is shown on Figure 30. The $x[n-i]$ terms are commonly known as taps. The taps provide the delayed inputs to the multiplication operations. The b_i values of the FIR filter are the coefficients of the FIR filter. The FIR filter implements the convolution shown on Equation 2.

Equation 2: FIR filter discrete convolution.

$$y(n) = b_0 x[n] + b_1 x[n-1] + \cdots + b_N x[n-N]$$

$$= \sum_{i=0}^{N} b_i \cdot x[n-i]$$

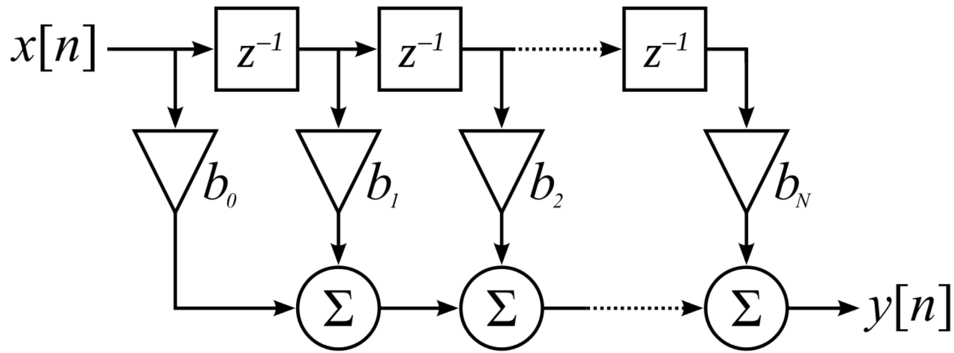


Figure 30: FIR filter structure.

This dissertation employs an FIR filter design to study and analyze 2D and 3D designs for NCL and MTNCL. In addition, a synchronous FIR filter design was also designed to allow comparison between synchronous and asynchronous architectures. The NCL and MTNCL FIR filters were based on the FIR design in [19] and [47]. The FIR filters consist of three main components: shift registers, Dadda multipliers, and carry-select adders. The shift registers are used to generate the taps. The multiplication of the coefficients and taps are done through the Dadda multipliers where the outputs are then summed using carry-select adders. All the outputs of the summation are kept until the final stage where it is truncated to 12-bits to maintain precision. Figure 31 shows a generalized architecture of the MTNCL design. The last register, multiplier, and adder are annotated with n variable to denote the last stages. The value n specifies the number of taps used in the design. For this dissertation, the taps instances are created for 16, 32, 64, and 96 taps. Developing multiple taps for measurement and study could show trends and develop better comparison between the different architectures.

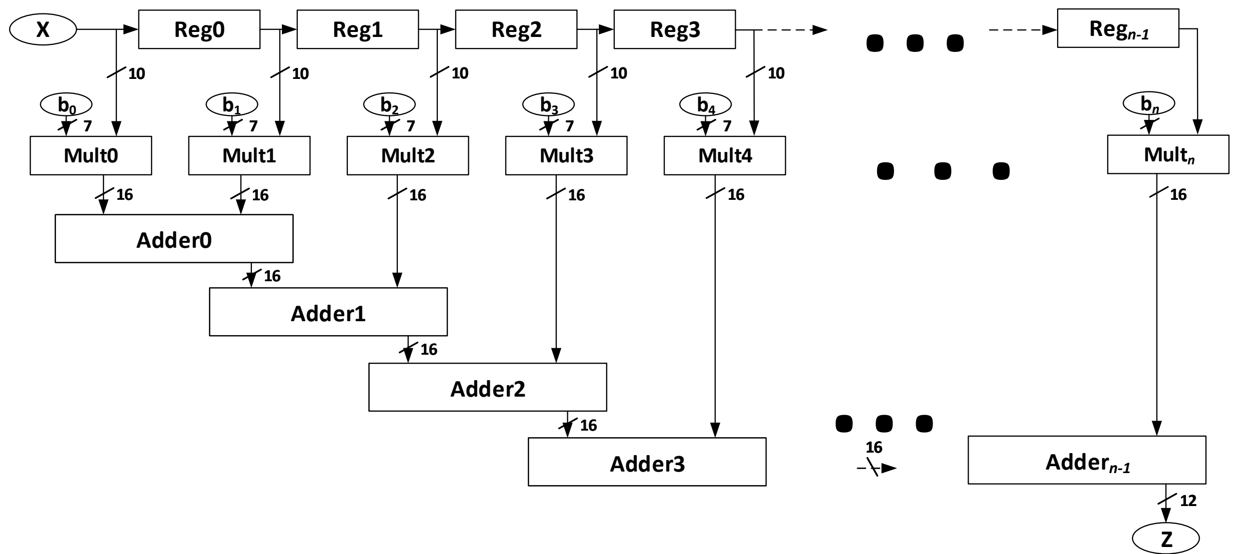


Figure 31: MTNCL FIR filter architecture.

While the synchronous design uses shift registers for the delay units, the MTNCL design uses special registers called *Regdm* and *Regnm* to implement the shifts registers [48]. These are shown on Figure 32. The naming notation *d* means that the gate is resettable to DATA1 and notation *n* means that the gate is resettable to DATA0. These special registers are created using resettable TH12m gates. The *Regdm* register is capable of being reset to a DATA1 with logic 1 reset, while *Regdn* register is capable of being reset to a DATA0 with logic 1 reset. In addition, the completion logic has been redesigned, replacing the last TH22 gate of the completion logic with a TH22*d* and TH22*n* to create *compd* and *compn,* respectively. The modified completion logic *compn* and *compd* are used to maintain data flow in the shift registers. All designs are pipelined and are written as generic VHDL code. The generic designs were used such that the taps could be modified easily without large structural changes. It also allowed consolidation of the designs into one design that can support any tap parameters. In addition, the FIR designs were partitioned using the Async3D tool flow and the partitioning is shown on Figure 33.

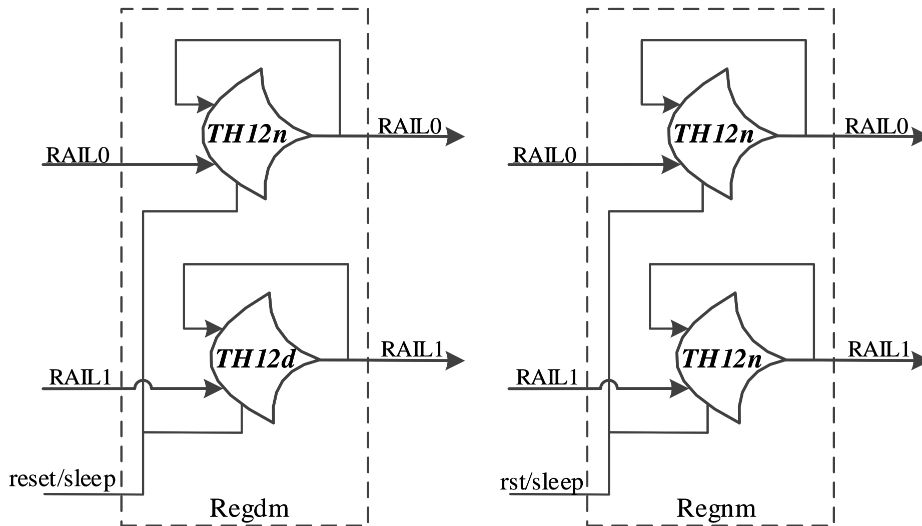


Figure 32: MTNCL shift registers

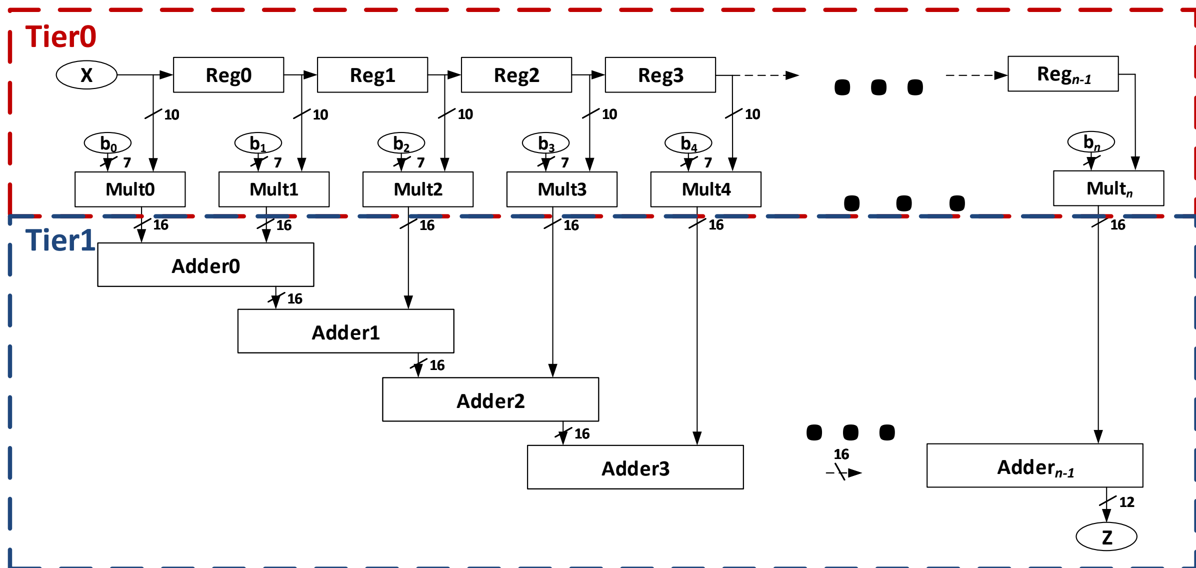


Figure 33: FIR partitioning.

6 Analysis of 3D Asynchronous Circuits

6.1 Design Verification

Design verification included full simulation using randomly generated data vectors. The tool used for simulation is Mentor Graphics ModelSim. The main format used for the design was a combination of VHDL and Verilog netlist format. Verilog files were required for several

physical design tools after synthesis. The design verification phase used standard delay format (SDF) and value change dump (VCD) files for accurate modeling of the designs. Standard delay format file contains the information about the interconnect and cell delays of a design. SDF files were used in ModelSim to correctly reflect delays. The SDF files were exported from Synopsys Design Vision for post-synthesis verification, and from Cadence Innovus for post-layout simulations. A VCD file logs all the signal transitions that occur during a simulation run. A VCD file is required for power analysis with Cadence Innovus. The VCD file could be used to extract the average activity of each circuit node.

6.2 Layout Floorplan

The floorplan used for circuit analysis are based on the Tezzaron 3D design guide. There are two options for the chip area, 2.5×5mm and 5×5mm chip. After previous place and route of preliminary designs, it was decided that 2.5×5mm chip floorplan was more appropriate as the other option was too large for the test circuits. The 2D IC layout floorplan is shown on Figure 34, while the 3D IC layout floorplan is shown on Figure 35. The 3D IC floorplan includes the micro-bump grids. The grid has a minimum pitch of 5 microns. The pitch used for the final designs were 20 microns.

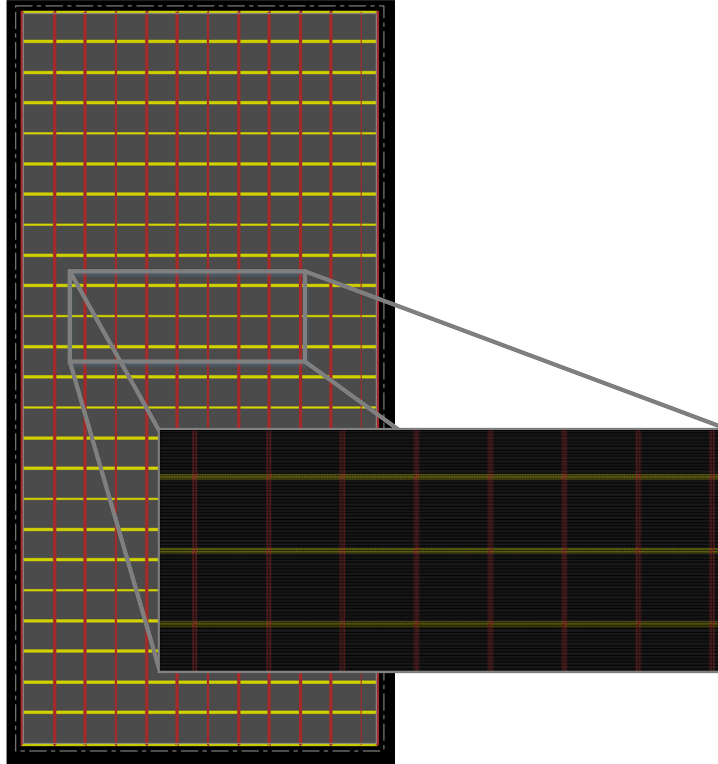


Figure 34: 2D 2.5×5mm floorplan.

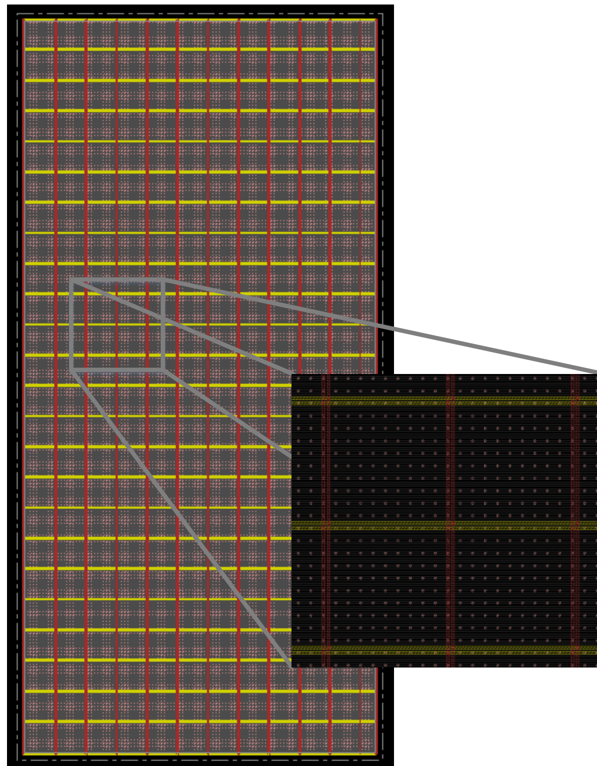


Figure 35:3D 2.5×5mm floorplan.

6.3 Circuit Performance

The FIR filter designs were normalized based on circuit performance. For the NCL and MTNCL circuits, DATA-to-DATA cycle time, denoted as T_{dd}, is the time its take the circuit to complete one cycle of operation, which is analogous to the clock speed of synchronous circuits. The T_{dd} is data-dependent and varies between cycles. Asynchronous circuits perform at average-case performance while a synchronous circuit must use the critical path to determine the clock speed. The T_{dd} was used to normalize the performance to compare multiple architectures. The NCL designs were simulated to calculate the average T_{dd} for all the random test vectors used in this dissertation. For NCL designs, the T_{dd} could be calculated using Equation 3. The equation accounts for variability of all the NCL pipeline stages. $Dcomb_i$ and $Dcomp_i$ are the combinational and completion logic delays of $stage_i$.

Equation 3: NCL T_{dd} estimation [49].

```
T_DDmax = 2 × (Dcomb_1 + Dcomp_1)
for (i = 2 to N) loop
     T_DDtemp = 2 × (Dcomb_i + Dcomp_i)
     T_DDmax = MAX(T_DDtemp, T_DDmax)
end loop
```

The T_{dd} used for normalization was 50ns (20 MHz) based on the analysis of the NCL designs. The MTNCL and synchronous designs were modified to meet this requirement.

6.4 Area/Wire Length

The post-layout results were gathered for the NCL, MTNCL, and synchronous designs. Multiples test circuits of varying complexity were designed to provide an accurate measurement of trends and scaling. The designs were implemented using 16, 32, 64, and 96 taps. By

implementing several test circuits, the data could show the design trade-offs of 3D ICs as complexity grows.

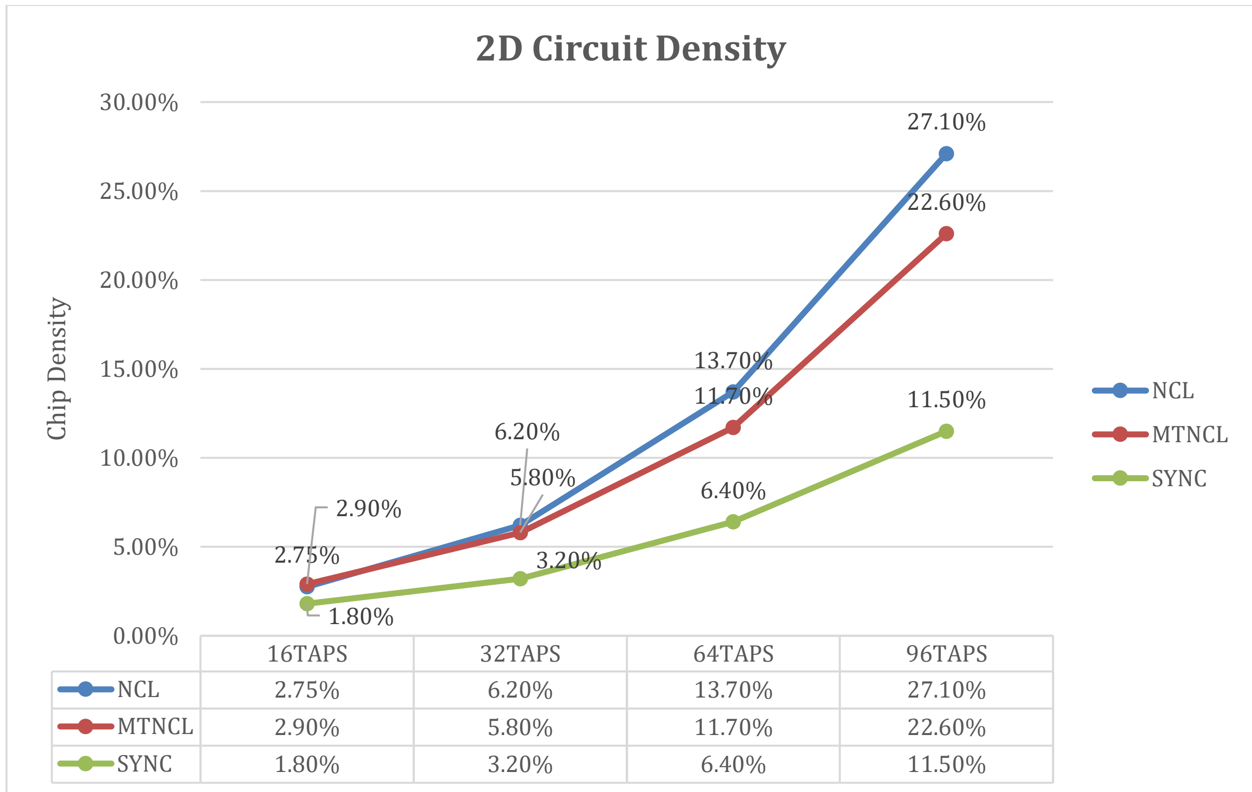
6.4.1 2D IC Design Circuit Density

The circuit density is the area required to place and route the chip divided by the total chip core area allotted. The circuit density is described on Equation 2.

Equation 4: Circuit density.

$$Circuit\ Density = \frac{ChipArea}{TotalCoreArea}$$

Figure 36 shows that the density is much higher for NCL and MTNCL designs when compared to the synchronous designs. When comparing NCL and MTNCL, MTNCL has lower density than NCL designs. As complexity grows, the trend shows that NCL and MTNCL is growing much faster than the synchronous design in terms of circuit density. The difference in circuit density is much smaller for lower taps designs. When comparing the 96-tap designs, the circuit densities of the NCL and MTNCL are approximately 2× that of the synchronous design.



	16TAPS	32TAPS	64TAPS	96TAPS
NCL	2.75%	6.20%	13.70%	27.10%
MTNCL	2.90%	5.80%	11.70%	22.60%
SYNC	1.80%	3.20%	6.40%	11.50%

Figure 36: 2D IC Circuit density

The total area of NCL and MTNCL designs were also much larger than that of their synchronous counterparts. On average, the NCL designs were 2.8× larger and the MTNCL 2.6× larger than the synchronous designs. This large area overhead is most likely attributed to the dual-rail encoding of the signal requiring two logic data paths for *rail0* and *rail1*, respectively. This also causes more interconnects, increasing the area of the NCL and MTNCL designs.

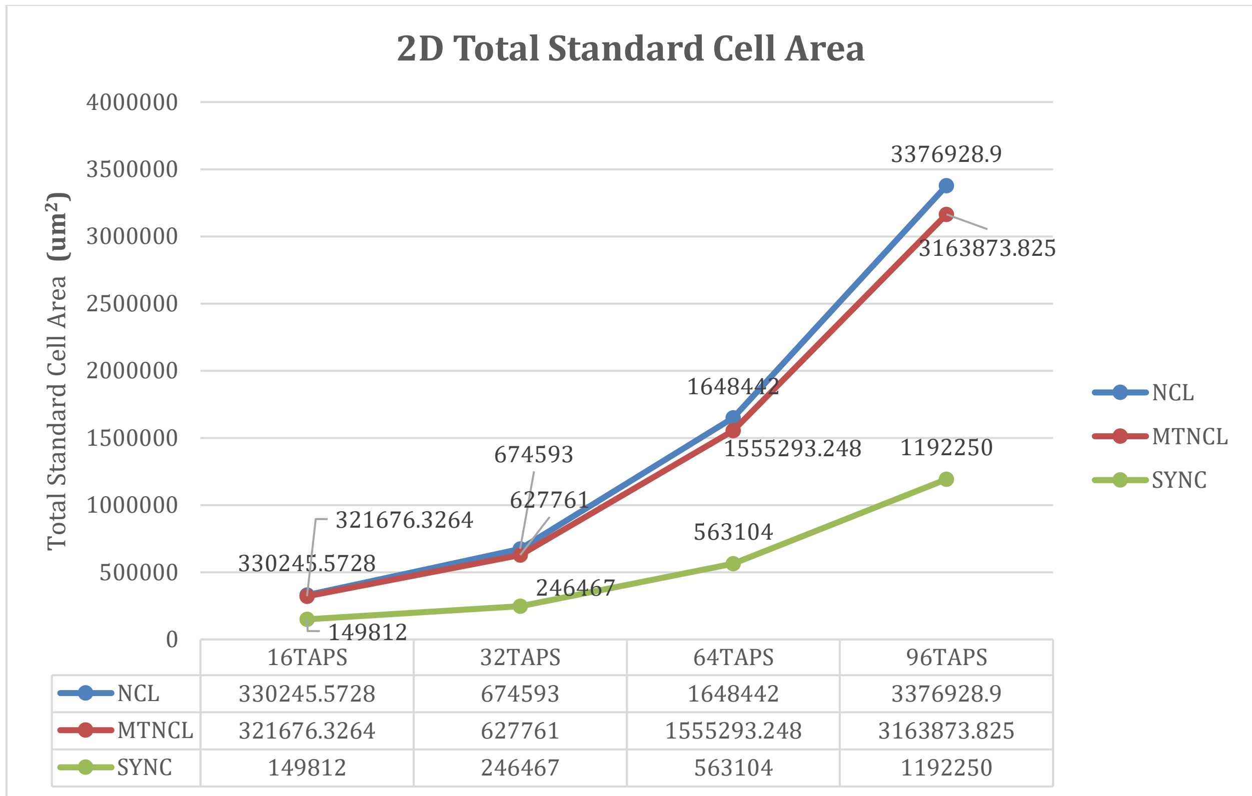


Figure 37: 2D IC standard cell area.

Table 8 shows a better comparison in terms of the ratio to the synchronous design. Based on the data, NCL and MTNCL has around 2.17 times more total standard cell area for 16 taps, 2.64 times for 32 taps, 2.84 times for 64 taps, and 2.74 times for 96 taps.

Table 8: 2D IC area comparison

Comparison	NCL_16TAPS	MTNCL_16TAPS	SYNC_16TAPS
Density %	152.78%	161.11%	100.00%
Total Standard Cell Length (mm)	220.44%	214.72%	100.00%
Total Standard Cell Area (um^2)	220.44%	214.72%	100.00%

(a)

Comparison	NCL_32TAPS	MTNCL_32TAPS	SYNC_32TAPS
Density %	193.75%	181.25%	100.00%
Total Standard Cell Length (mm)	273.71%	254.70%	100.00%
Total Standard Cell Area (um^2)	273.71%	254.70%	100.00%

(b)

Comparison	NCL_64TAPS	MTNCL_64TAPS	SYNC_64TAPS
Density %	214.06%	182.81%	100.00%
Total Standard Cell Length (mm)	292.74%	276.20%	100.00%
Total Standard Cell Area (um^2)	292.74%	276.20%	100.00%

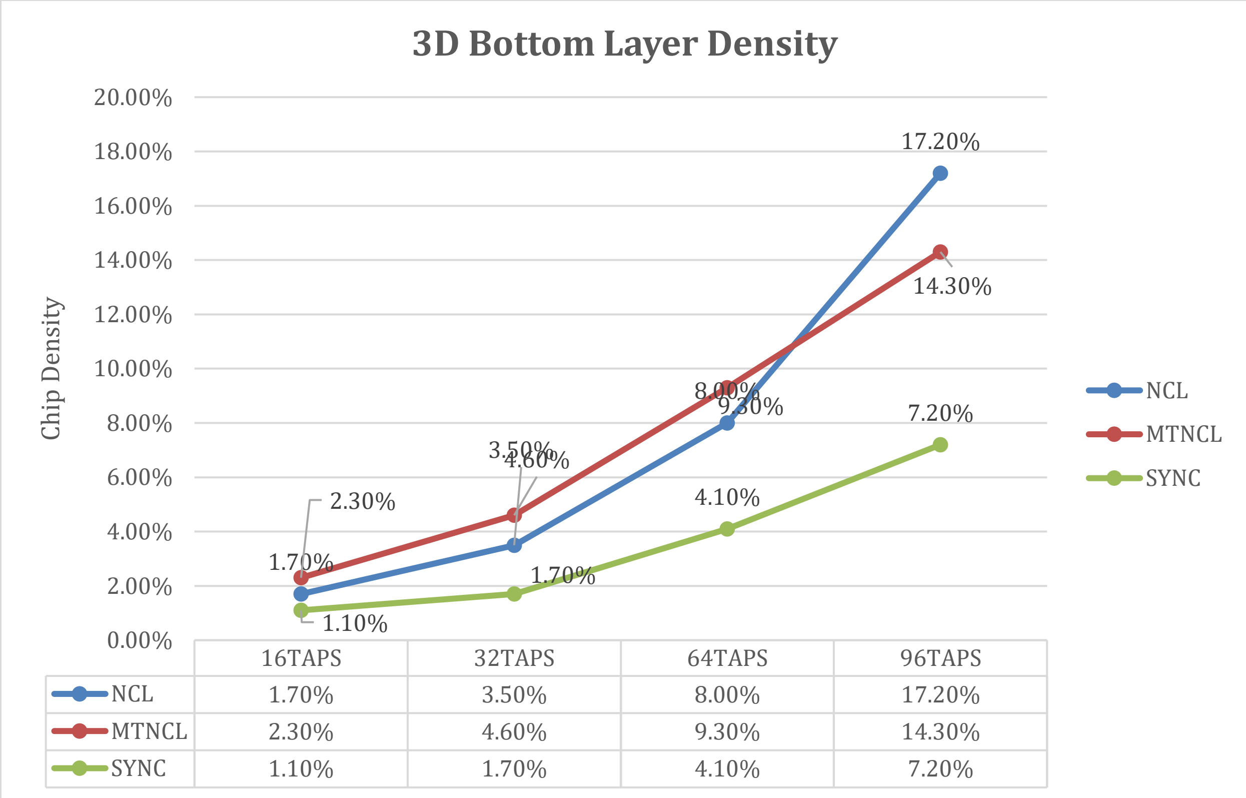
(c)

Comparison	NCL_96TAPS	MTNCL_96TAPS	SYNC_96TAPS
Density %	235.65%	196.52%	100.00%
Total Standard Cell Length (mm)	283.24%	265.37%	100.00%
Total Standard Cell Area (um^2)	283.24%	265.37%	100.00%

(d)

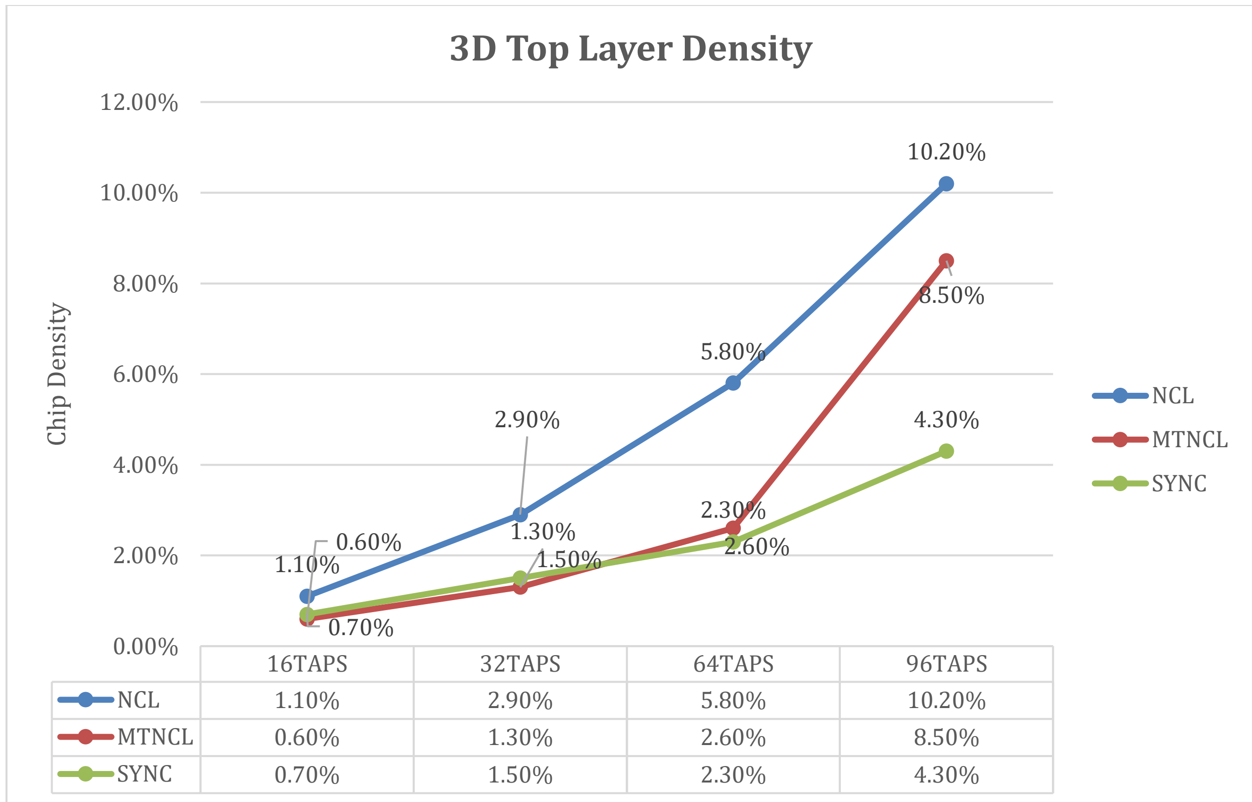
6.4.2 3D Design Circuit Density

This section presents the circuit density in term of Tier0 and Tier1 layers for all designs. Figure 38 and Figure 39 shows the bottom and top layer density of the designs. The graph suggests that NCL designs could be better for 96 taps than the MTNCL design. However, after a closer inspection of the netlist and the gate counts, the discrepancy between the density of NCL and MTNCL when compared to the 2D IC design is mostly attributed to the imprecise partitioning of the netlist. For example, certain designs might have more cells placed on the bottom layer, while others might have more cell placed on the top layer. As of this moment of writing, the 3D IC partitioning step could be improved. Nevertheless, the area overhead for NCL and MTNCL in terms of gate area cannot be reduced in a 3D design. The synchronous design maintains around 1 to 2.4 ratio versus NCL and MTNCL in terms of cell area.



	16TAPS	32TAPS	64TAPS	96TAPS
NCL	1.70%	3.50%	8.00%	17.20%
MTNCL	2.30%	4.60%	9.30%	14.30%
SYNC	1.10%	1.70%	4.10%	7.20%

Figure 38: 3D IC Bottom layer density



	16TAPS	32TAPS	64TAPS	96TAPS
NCL	1.10%	2.90%	5.80%	10.20%
MTNCL	0.60%	1.30%	2.60%	8.50%
SYNC	0.70%	1.50%	2.30%	4.30%

Figure 39: 3D IC Top layer density

Figure 40 and Figure 41 shows the total standard cell area for the bottom and top layers. The result is similar to the 2D total area analysis. The breakdown between the bottom layer and the top layer still follows 60%-40% ratio in terms of total standard cell area. Another possible cause for the large area overhead for NCL and MTNCL could be in the library design. The synchronous library is highly optimized because of its wide use in industry. This dissertation proposes an automated cell layout approach, but there are trade-offs between generating fast automated layouts and manually creating them. Manual creation of the cell layouts is laborious and error prone. However, it provides more fined-grained optimization and allows more compact and better performing cells. On the other hand, future improvements to the layout generation tool could reduce this area overhead.

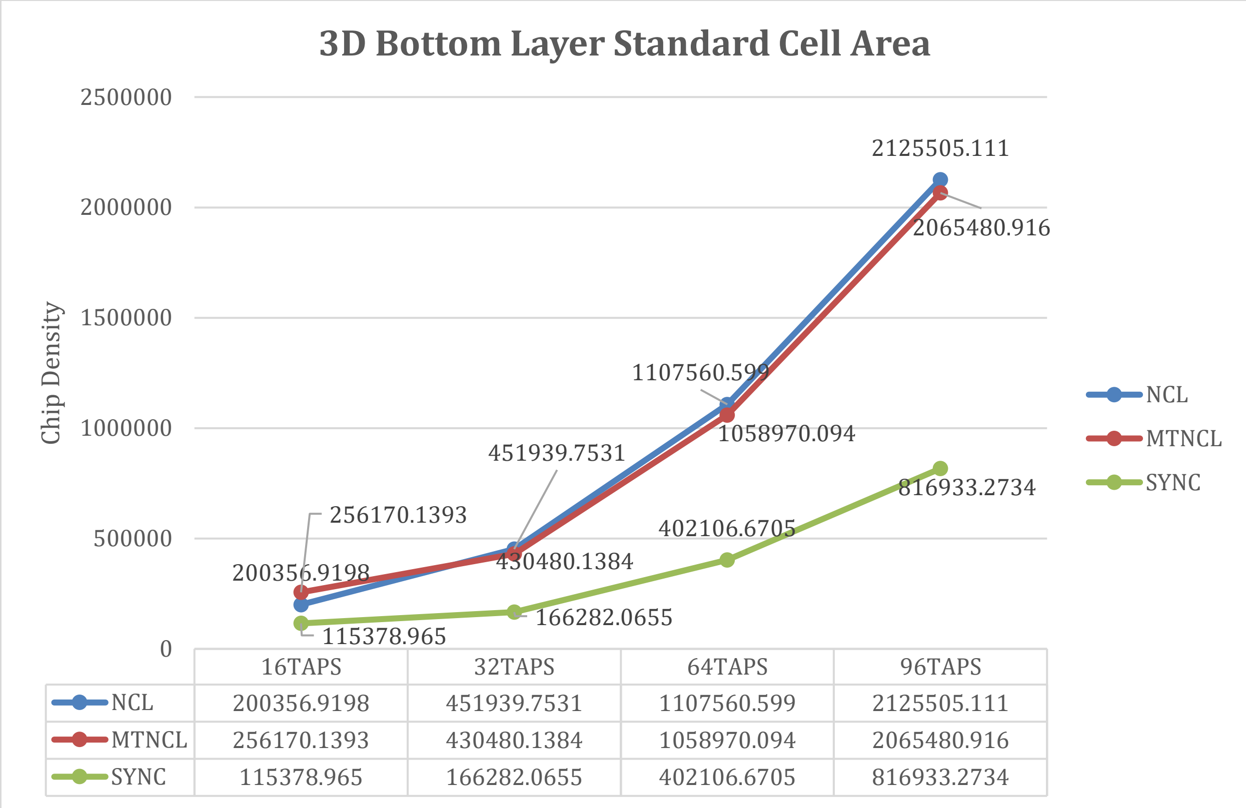
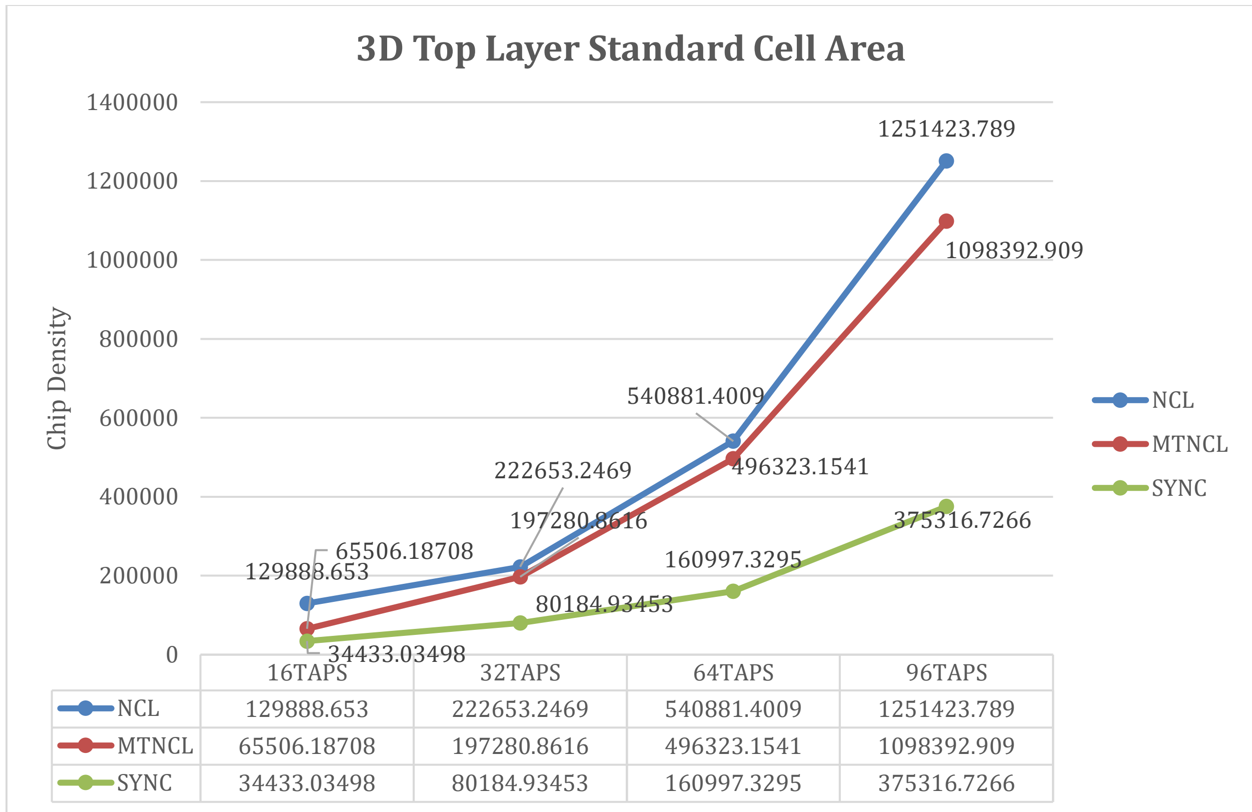


Figure 40: 3D Bottom layer standard cell area.



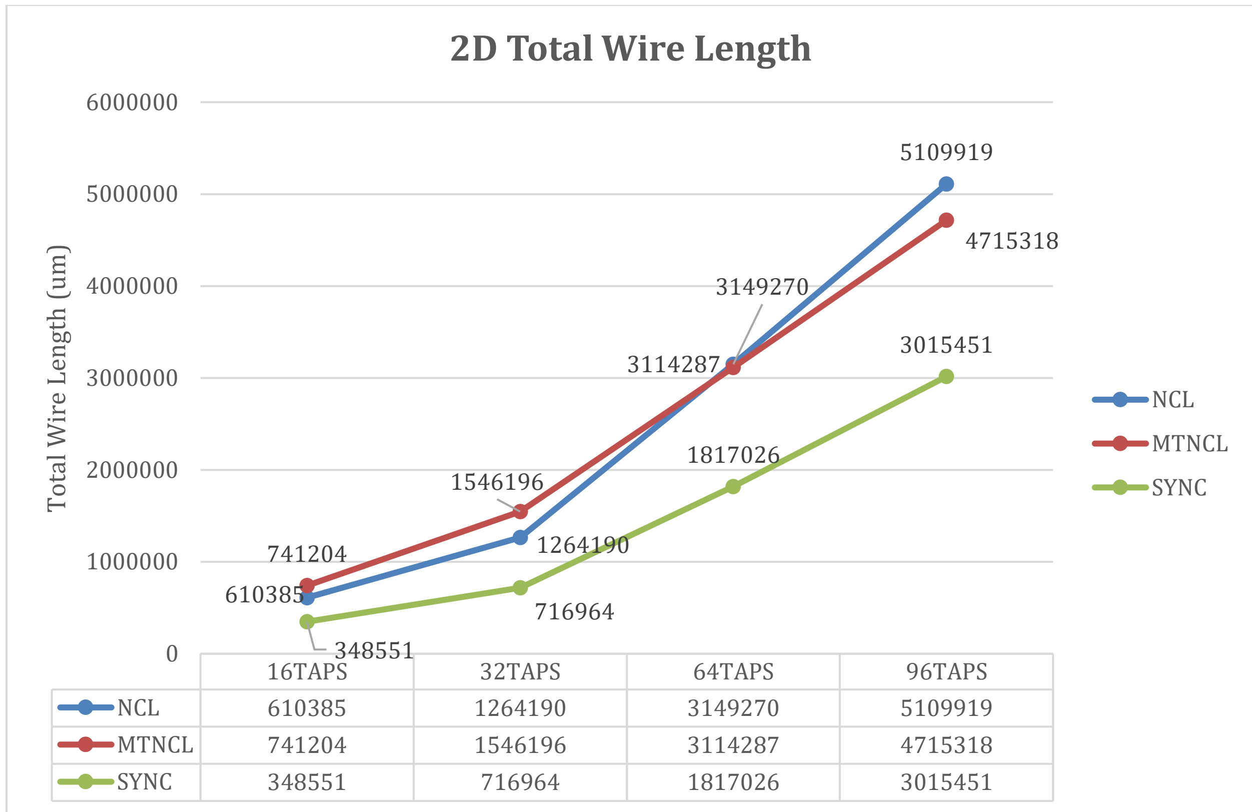
	16TAPS	32TAPS	64TAPS	96TAPS
NCL	129888.653	222653.2469	540881.4009	1251423.789
MTNCL	65506.18708	197280.8616	496323.1541	1098392.909
SYNC	34433.03498	80184.93453	160997.3295	375316.7266

Figure 41: 3D Top layer standard cell area.

6.5 Interconnect

6.5.1 Wire Length Analysis - 2D IC Design

Figure 42 shows the comparison between the 2D IC designs in terms of total wire length. NCL has shorter total interconnect wire length than MTNCL for 16 taps and 32 taps, while the trend changes for 64 and 96 taps. Overall, total wire length of the NCL and MTNCL designs are very similar. The difference is around 8-12% on average. When comparing NCL, MTNCL, and synchronous designs, the graph shows that NCL and MTNCL have much longer total wire lengths than the synchronous counterpart. This is due to the dual-rail encoding where 2 signals are required to represent logic 1 and logic 0.

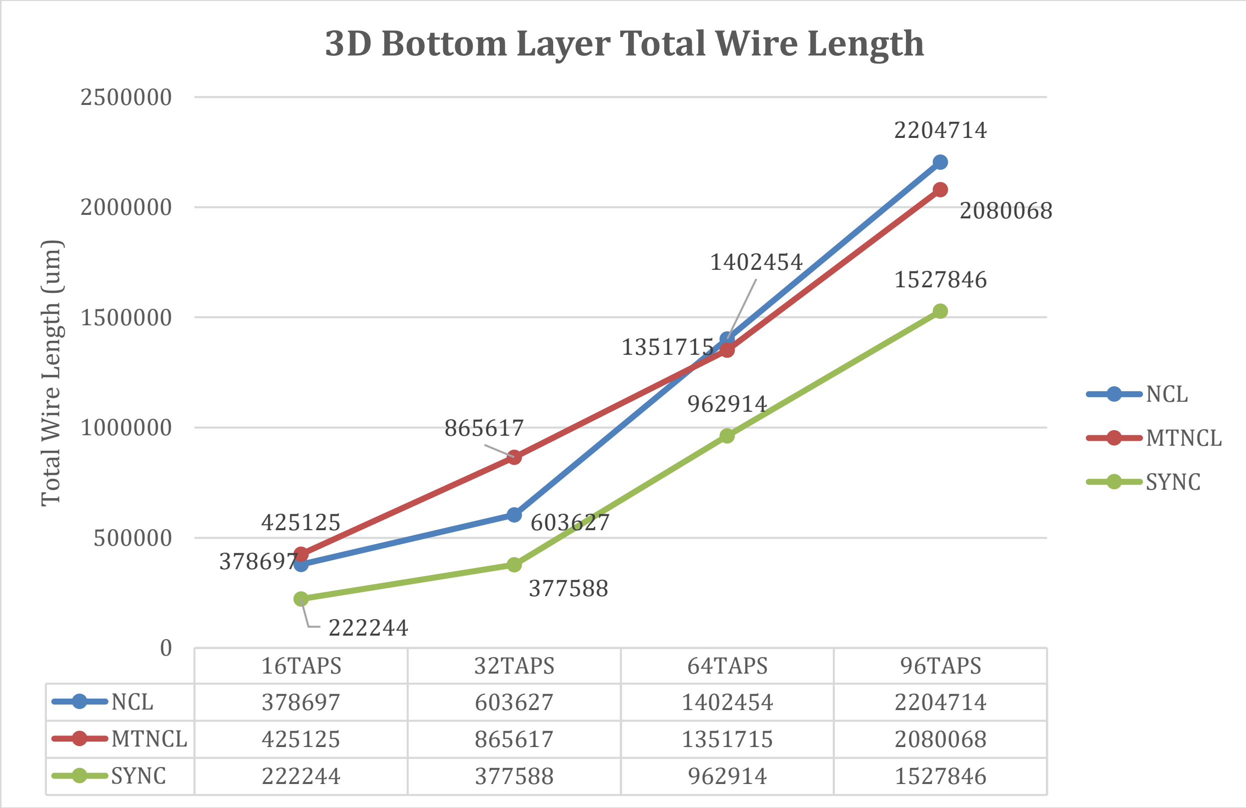


	16TAPS	32TAPS	64TAPS	96TAPS
NCL	610385	1264190	3149270	5109919
MTNCL	741204	1546196	3114287	4715318
SYNC	348551	716964	1817026	3015451

Figure 42: 2D IC total wire length.

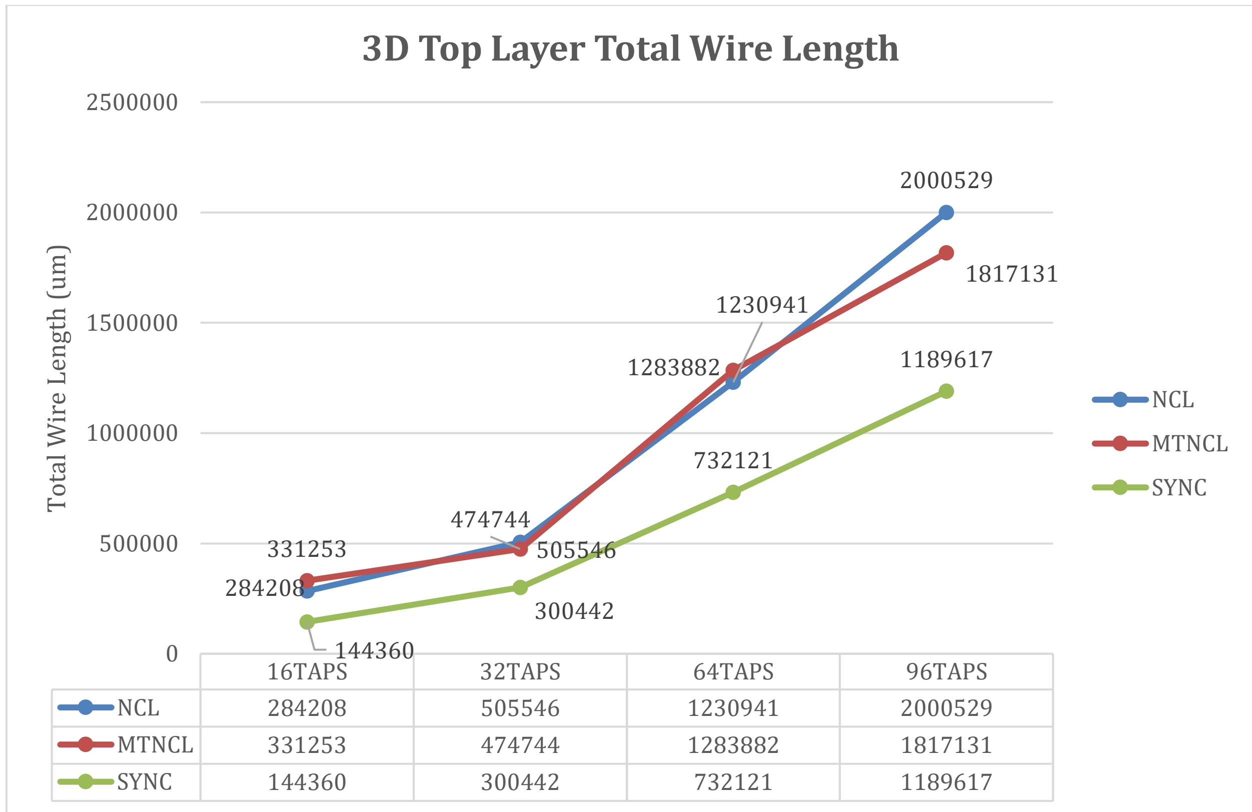
6.5.2 Wire length Analysis - 3D IC Design

Figure 43 and Figure 44 shows the bottom layer and top layer total wire length for the 3D designs. The total wire length for the NCL and MTNCL designs are very similar. On average, all three architectures have improved in terms of interconnect length. The interconnect is longer for Tier0 than Tier1. This is because the Dadda multipliers are much larger than the carry-select adders. As previously mentioned in Chapter 5, the Dadda multipliers are partitioned to Tier0 while the adders are partitioned to Tier1.



	16TAPS	32TAPS	64TAPS	96TAPS
NCL	378697	603627	1402454	2204714
MTNCL	425125	865617	1351715	2080068
SYNC	222244	377588	962914	1527846

Figure 43: 3D Bottom layer total wire length.



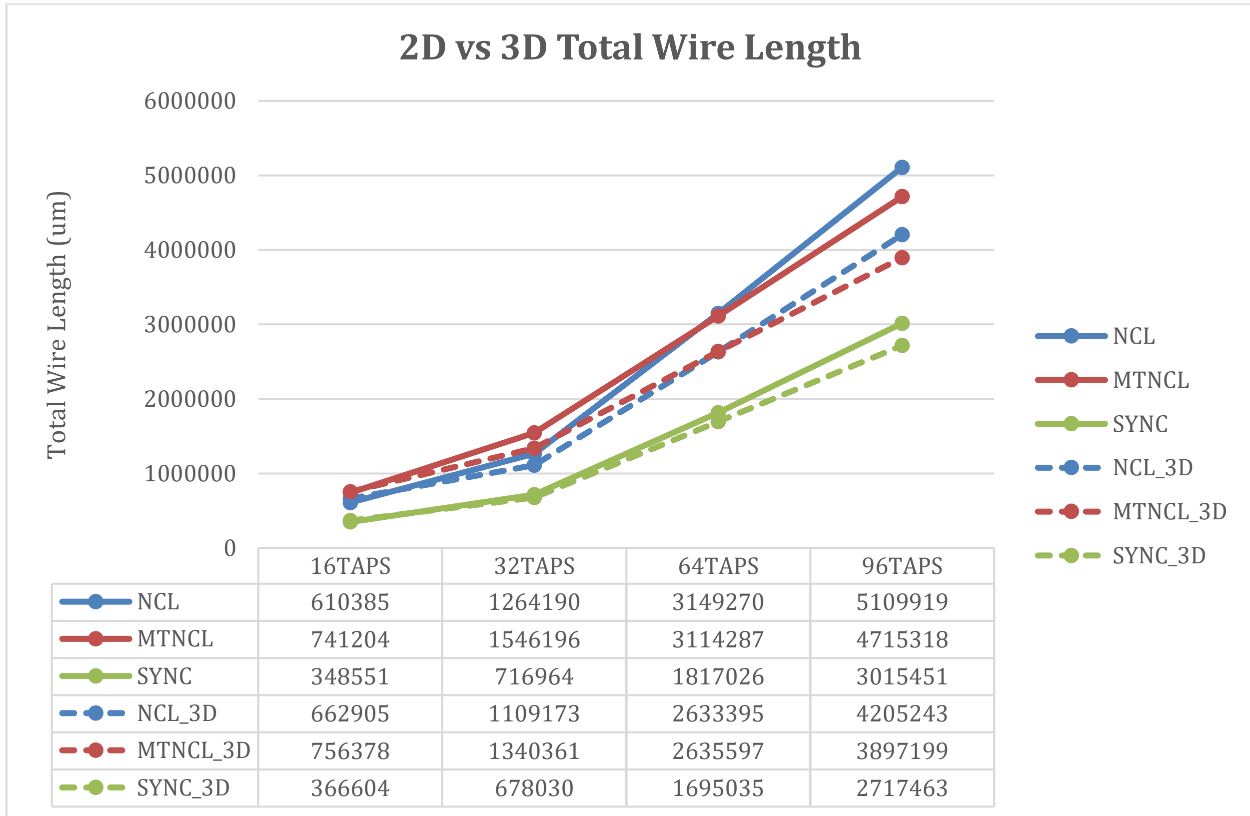
	16TAPS	32TAPS	64TAPS	96TAPS
NCL	284208	505546	1230941	2000529
MTNCL	331253	474744	1283882	1817131
SYNC	144360	300442	732121	1189617

Figure 44: 3D Top layer total wire length

6.5.3 Wire Length Analysis - 2D vs 3D IC Design

Figure 45 shows the comparison of the total wire length of the 2D and 3D IC designs. The 3D designs are shown as dotted-lines for better visibility. Overall, all 3 designs have improved in terms of total wire length when moving to a 3D design. In addition, the improvement for NCL and MTNCL are much more significant than the improvement for the synchronous design. This trend could most likely be attributed to the large area overhead and therefore higher density of the NCL and MTNCL. Due to the much higher density of the NCL and MTNCL designs, the vertical interconnects have a much higher impact. The use of the TSVs has reduced the total wire length of the design. For smaller designs, the effect is much smaller. The total wire length of the 16-tap designs increased by around 5% on average when comparing 2D vs 3D. Given that the starting floorplan for the test designs is large, the smaller density

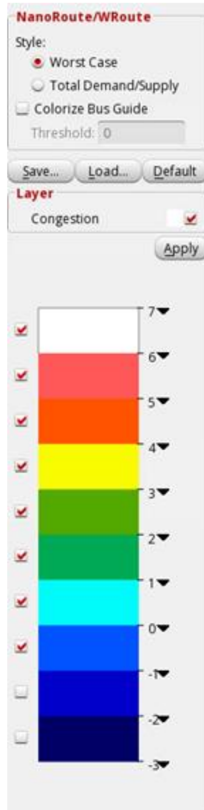
design did not see much improvement. It is important to note, however, that this trend might not be attributed to the 16-tap design itself, but is more due to the small density when compared to floorplan.

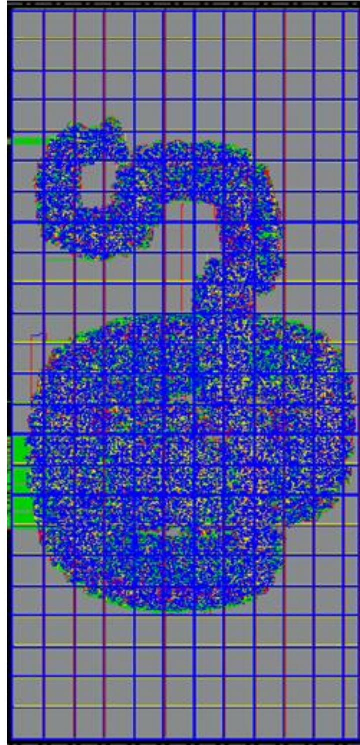


	16TAPS	32TAPS	64TAPS	96TAPS
NCL	610385	1264190	3149270	5109919
MTNCL	741204	1546196	3114287	4715318
SYNC	348551	716964	1817026	3015451
NCL_3D	662905	1109173	2633395	4205243
MTNCL_3D	756378	1340361	2635597	3897199
SYNC_3D	366604	678030	1695035	2717463

Figure 45: 2D vs 3D IC total wire length

Larger designs with higher density such as the 64 taps and the 96 taps design have an improvement of around 12.86% and 14.98%, respectively. The graph in Figure 45 shows a trend that larger designs could benefit more from 3D IC integration. Figure 46 and Figure 47 shows the congestion map for the NCL and MTNCL 96-tap designs. Based on the data, the congestion of the NCL and MTNCL 96-tap designs are very low with the highest reported congestion of 2.3% horizontal congestion and 3.1% vertical congestion. These values are based on the horizontal and vertical routing grids.





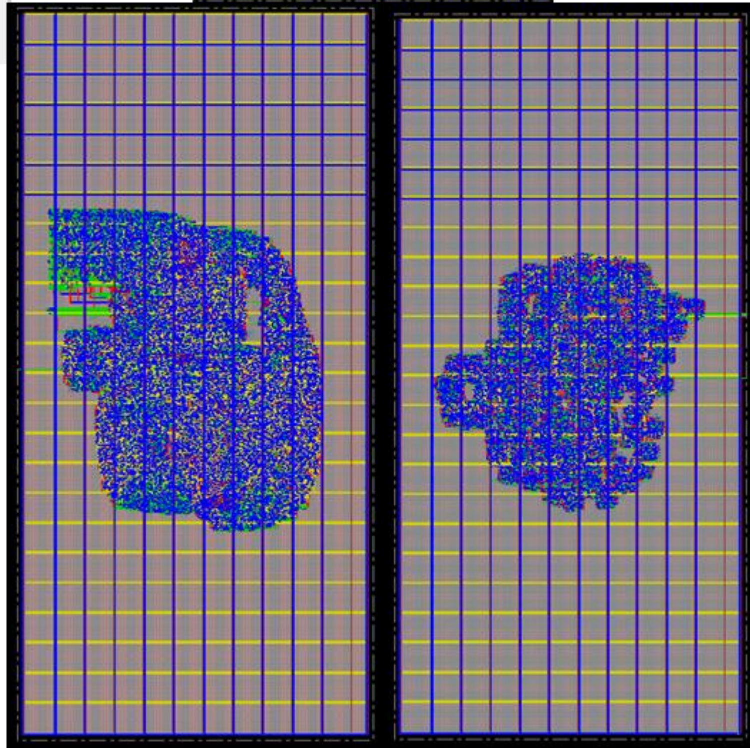


Figure 46: NCL congestion map.

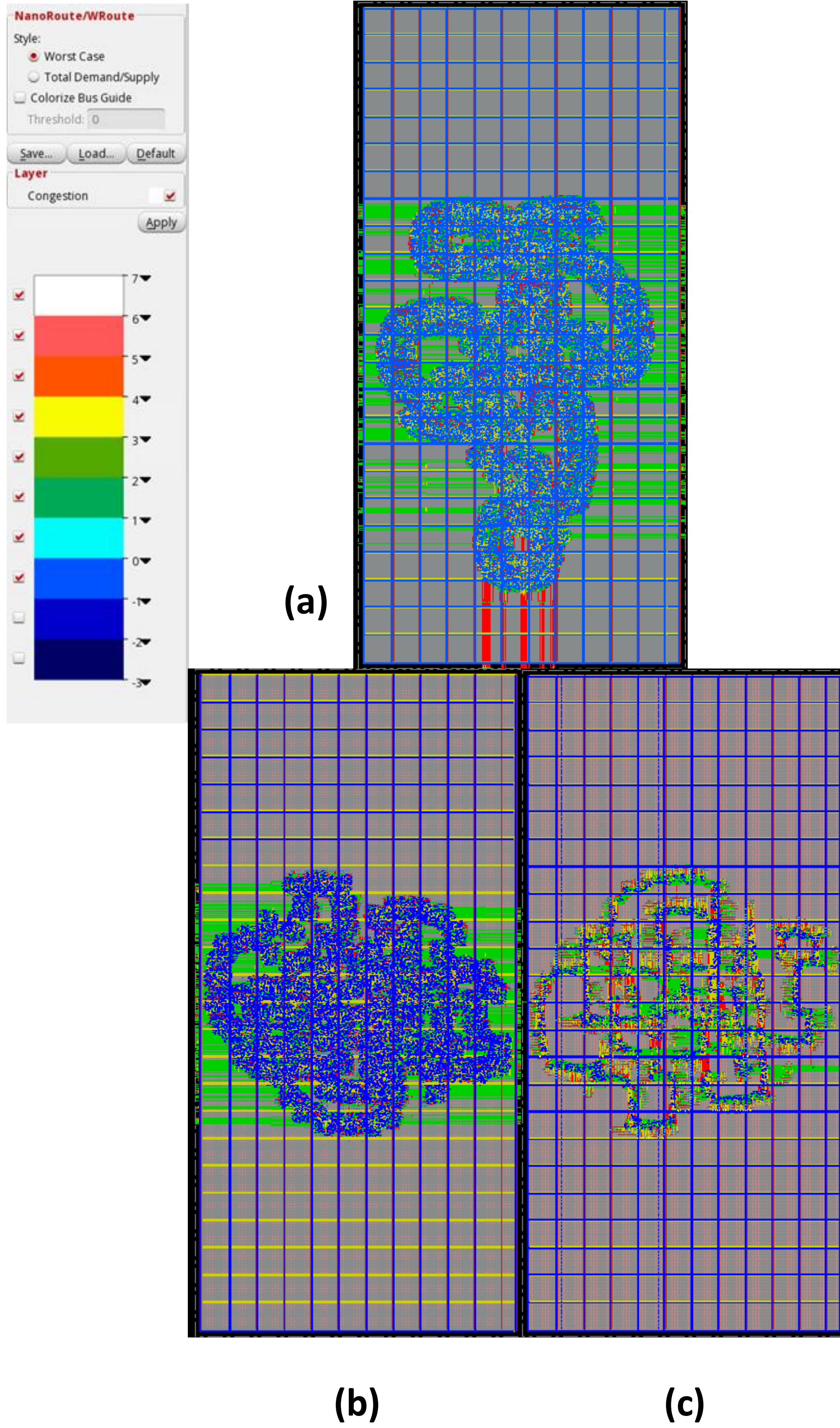


Figure 47: MTNCL congestion map.

6.5.4 TSV Count

The TSV count was calculated based on the number of TSVs used on the grid. Table 9 shows the TSV and micro-bumps count of the 3D designs. The TSV count is highly dependent on the partitioning method. The wires that connect the logic die become the vertical interconnects in the designs.

Table 9: TSV/Micro-bumps count.

3D	Number of Super Contacts	Ratio
NCL_16TAPS	992	1.907692308
MTNCL_16TAPS	1107	2.128846154
SYNC_16TAPS	520	1

(a)

3D	Number of Super Contacts	Ratio
NCL_32TAPS	2055	2.046812749
MTNCL_32TAPS	2122	2.113545817
SYNC_32TAPS	1004	1

(b)

3D	Number of Super Contacts	Ratio
NCL_64TAPS	4210	2.06372549
MTNCL_64TAPS	4652	2.280392157
SYNC_64TAPS	2040	1

(c)

3D	Number of Super Contacts	Ratio
NCL_96TAPS	6105	2.02689243
MTNCL_96TAPS	6445	2.139774236
SYNC_96TAPS	3012	1

(d)

6.6 Power Analysis

After placing and routing the designs, the power consumptions were calculated using Cadence Voltus that is integrated with Innovus [50]. Voltus is capable of measuring static and dynamic power consumptions. For better accuracy, the power simulations were gathered post-layout and included various layout parasitics that were added after placement and routing. The data reported are internal power, switching power, and leakage power.

The internal power is the power consumed by the charging and discharging of interconnect and device capacitances within the cell. The internal power is calculated using the power tables from the liberty characterization file.

Switching power is the power consumed in the charging and discharging of interconnect capacitances. This value will be large if there are large drivers driving high capacitance loads. Switching power is calculated using the following equations:

$$SwitchingPower = CAV^2F$$

$$C = Loading\ net\ capacitance$$

$$V = Voltage$$

$$A = Nodal\ activity$$

$$F = Operating\ frequency$$

The product of the nodal activity and the operating frequency ($A * F$) is the transition density (D) calculated during the activity propagation. The transition density includes both rising and falling transitions, and so the equation could be modified to calculate the power rail using transition density. The modified equation is given by the following:

$$SwitchingPower = \frac{1}{2}CV^2D$$

When the net is driven by multiple outputs, the capacitance is split and divided amongst the output drivers. An example of this is a clock mesh driven by parallel clock drivers.

The leakage power is the power consumed by devices when it is not switching. This includes the state-dependent leakage which is the leakage that depends on the internal state of the cell. This is also gathered using the liberty characterization file. Figure 48 shows an example state-dependent leakage data from liberty characterization.

```
cell_leakage_power : 14.335 ;
leakage_power() {
when : "!A1 !A2" ;
value : 9.120 ;
}
leakage_power() {
when : "!A1 A2" ;
value : 16.467 ;
}
leakage_power() {
when : "A1 !A2" ;
value : 12.364 ;
}
leakage_power() {
when : "A1 A2" ;
value : 19.390 ;
}
```

Figure 48: State-dependent leakage data.

As the leakage component of the power consumption increases, an accurate estimation becomes more and more critical. Finally, total power consumption is also reported by adding the internal power, switching power and leakage power as shown on Equation 5:

Equation 5: Total power calculation.

$$Power_{Total} = Power_{Internal} + Power_{Switching} + Power_{Leakage}$$

6.6.1 Vector-based Power Measurement

The power calculation used a vector-based power calculation. The vector-driven approach uses a VCD file output from a logic simulator. For this dissertation work, the VCD files were generated using Mentor Graphics ModelSim. The VCD files are used to obtain the number of transitions for each net. Mentor Graphics ModelSim was used to capture a complete simulation run and was used to determine the exact switching activity for all the nets. ModelSim is then used to generate VCD files from the post-layout netlist with all the node information. In addition, the VCD files included 256 randomly generated data vectors. Randomly generating the data vectors would provide the average-case power measurements. Power consumption is data

dependent. It is important to make proper choices for stimuli vectors to present a more accurate comparison. The gate-level simulations provided at least 99% coverage for each gate and net. In addition, the power engine also calculates the duty cycle for each net for state-dependent internal and leakage power calculation.

Dynamic power consumption directly depends on the switching activity. Using this method, the power consumption could be estimated on Cadence Innovus without the running costly simulations.

6.6.2 Power Data

The power measurement data are shown on Table 10-12. Using synchronous design as a baseline, Table 10 shows that the NCL design consumes around 2.20× more total power, while the MTNCL design consumes around 2.10× more total power. In terms of leakage, the low power optimizations for MTNCL have reduced the leakage power significantly. When comparing the designs in terms of leakage power, the MTNCL design has around 60% reduced leakage power than NCL and 57% reduced leakage than synchronous design. In terms of switching power and internal power, NCL and MTNCL consume about 2.13× more than the synchronous design. This was expected because of the area overhead and the dual-rail encoding. The increased power could also be attributed to increased wire interconnect of the NCL and MTNCL designs. Looking at the 3D data on Table 11, the power breakdown between the two layers is around 50-60% to 37-49% with Tier0 being higher than Tier1 in most cases. Table 12 shows the total power comparison between the designs. There are some improvements in total power for 32, 64, and 96 taps. The 96-tap NCL and MTNCL FIRs have improved by around 6.5% in terms of total power when moving to 3D IC. As the density increases, 3D IC provides more benefits to the NCL and MTNCL designs. Larger designs could show better improvements.

The reduced power could be the result of shorter interconnect length. Shorter interconnects result in lower load capacitance.

Table 10: 2D Designs power data.

Designs	$P_{internal}$ (mW)	$P_{switching}$ (mW)	$P_{leakage}$ (mW)	P_{total} (mW)
NCL16	20.555	5.235	0.002766	25.793
MTNCL16	19.999	5.553	0.001252	25.554
SYNC16	9.272	2.575	0.002111	11.849
NCL32	44.417	10.223	0.005444	54.646
MTNCL32	40.648	9.409	0.002507	50.059
SYNC	18.764	5.205	0.004274	23.973
NCL64	95.321	18.057	0.011834	113.390
MTNCL64	86.636	16.125	0.004914	102.767
SYNC64	36.923	10.447	0.008621	47.379
NCL96	113.858	23.294	0.018075	137.170
MTNCL96	104.448	20.172	0.007217	124.627
SYNC96	54.152	14.509	0.012908	68.674

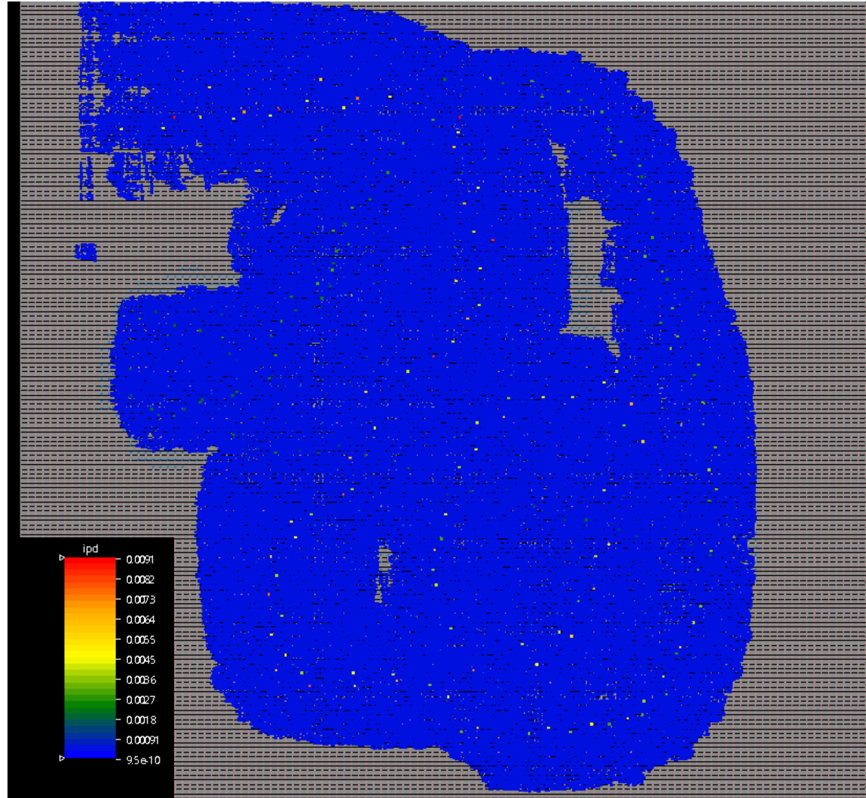
Table 11: 3D Designs power data.

Designs	$P_{internal}$ (mW)		$P_{switching}$ (mW)		$P_{leakage}$ (mW)		P_{total} (mW)	
	Tier0	Tier1	Tier0	Tier1	Tier0	Tier1	Tier0	Tier1
NCL16	13.561	9.351	2.442	2.280	0.000700	0.000278	13.561	11.631
MTNCL16	13.796	7.649	2.589	2.317	0.000598	0.000155	13.796	9.966
SYNC16	5.096	3.986	1.272	2.289	0.000678	0.001486	5.096	6.277
NCL32	25.822	17.023	5.796	5.594	0.001360	0.000741	25.822	22.618
MTNCL32	24.010	16.116	5.111	4.273	0.001199	0.000319	24.010	20.389
SYNC	7.955	7.249	4.713	4.423	0.002989	0.001464	7.955	11.674
NCL64	47.157	33.215	13.267	13.377	0.003153	0.003120	47.157	46.595
MTNCL64	42.983	30.332	12.288	12.083	0.003698	0.001166	42.983	42.416
SYNC64	18.073	11.622	8.639	7.466	0.005992	0.002844	18.073	19.091
NCL96	62.010	30.120	17.243	18.666	0.005896	0.004774	62.010	48.791
MTNCL96	53.403	32.929	15.591	14.553	0.005457	0.001638	53.403	47.483
SYNC96	26.579	17.523	11.671	11.096	0.009028	0.004293	26.579	28.623

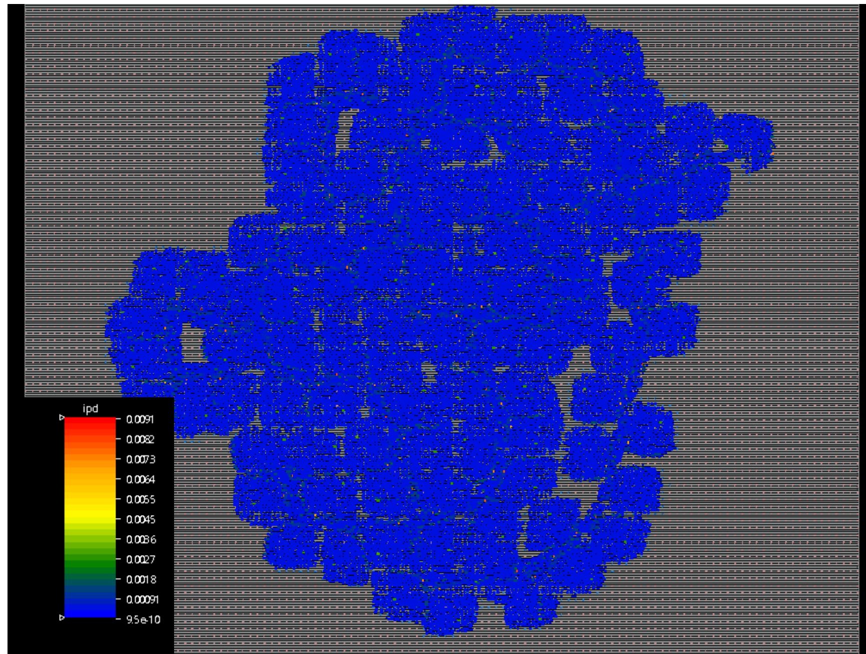
Table 12: 2D versus 3D power data.

Designs	2D P_{Total} (mW)	3D P_{Total} (mW)	Improv.%
NCL16	25.793	27.635	-7.14%
MTNCL16	25.554	26.351	-3.12%
SYNC16	11.849	12.646	-6.73%
NCL32	54.646	54.237	0.75%
MTNCL32	50.059	49.512	1.09%
SYNC	23.973	24.345	-1.55%
NCL64	113.390	107.022	5.62%
MTNCL64	102.767	97.691	4.94%
SYNC64	47.379	45.809	3.32%
NCL96	137.170	128.050	6.65%
MTNCL96	124.627	116.482	6.54%
SYNC96	68.674	66.882	2.61%

Using the power data gathered above, Innovus can create power map file that can be used to create an overlay over the original layout. The total power density was generated for the NCL and MTNCL design. Figure 49 and Figure 50 shows the total power density of the NCL and MTNCL 96-tap 3D designs. A blue color means low density while a red color means very high density. Based on the power map, the NCL and MTNCL designs are evenly distributed in terms of total power.

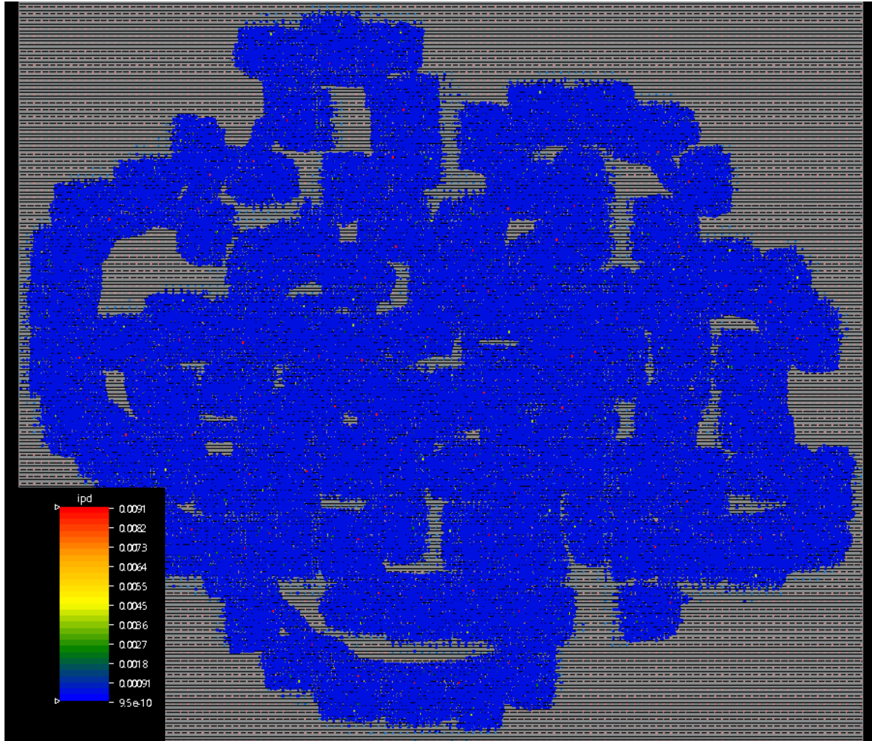


(a)



(b)

Figure 49: NCL total power density map: (a) Tier0, (b) Tier1.



(a)

(b)

Figure 50: MTNCL total power density map: (a) Tier0, (b) Tier1.

7 Conclusion

This dissertation work presents the developed Async3D tool flow and library for NCL and MTNCL 3D ICs. In addition, a series of NCL and MTNCL FIR filter designs were implemented in varying complexity using the flow to analyze NCL and MTNCL 3D ICs. A set of synchronous FIR filters was also implemented to show design trade-off between asynchronous and synchronous designs. This dissertation work shows that the application of NCL and MTNCL with 3D IC is very promising. The NCL and MTNCL 3D IC designs have showed improvement in all metrics. Simulation results show that MTNCL have much lower leakage than NCL and synchronous designs. Although NCL and MTNCL designs have around 2× area overhead over synchronous design, the overhead could be reduced with 3D IC technology. The results show that the NCL and MTNCL designs have reduced power consumption of around 2-6% when moving to 3D IC. In addition, the NCL and MTNCL 3D designs have improved in terms of total interconnect length by up to 15%. As density of the circuit increases, 3D IC provides more benefits to the NCL and MTNCL circuits. Finally, NCL and MTNCL asynchronous paradigms could be a viable option in solving the thermal hotspot problem of 3D ICs.

7.1 Future Work

There are several challenges in implementing asynchronous circuits in 3D IC. A major segment of the EDA industry was developed to support standard cell synchronous design. While the EDA support for synchronous design has matured, the support for asynchronous methodologies lagged behind [51]. For instance, while EDA vendors like Cadence and Synopsys have developed automated tool support for clock tree and clock gating generation, test mechanism insertion, and even optimizations such as retiming, time borrowing, and skew management [50], [52], they target synchronous design without any support for asynchronous

design. This dissertation hopes to bridge the gap slightly through the Async3D tool flow for asynchronous designs in 3D ICs.

One major area of improvement in the tool flow is 3D IC partitioning. Design-aware partitioning could provide significant improvements to the final NCL and MTNCL 3D IC designs.

8 References

[1] J. M. Rabaey, A. P. Chandrakasan, and B. (Assistant P. Nikolić, *Digital integrated circuits : a design perspective*. Pearson Education, 2003.

[2] G. E. Moore, "Cramming more components onto integrated circuits (Reprinted from Electronics, pg 114-117, April 19, 1965)," Sep. 1965.

[3] G. E. Moore, "No exponential is forever: But 'forever' can be delayed!," *Solid-State Circuits Conf. 2003. Dig. Tech. Pap. ISSCC. 2003 IEEE Int.*, vol. 1, pp. 20–23 vol.1, 2003.

[4] W. Neil and D. Harris, *CMOS VLSI Design: A Circuits and Systems Perspective (4th Edition)*, 4th ed. Pearson Education.

[5] H. Eriksson, P. Larsson-Edefors, T. Henriksson, and C. Svensson, "Full-custom vs. standard-cell design flow - An adder case study," *Proc. Asia South Pacific Des. Autom. Conf. ASP-DAC*, vol. 2003–Janua, pp. 507–510, 2003.

[6] M. Hashimoto, K. Fujimori, and H. Onodera, "Standard cell libraries with various driving strength cells for 0.13, 0.18 and 0.35 µm technologies," *Proc. Asia South Pacific Des. Autom. Conf. ASP-DAC*, vol. 2003–Janua, pp. 589–590, 2003.

[7] A. B. Jambek, A. R. Noorbeg, and M. R. Ahmad, "Standard cell library development," *Proc. Int. Conf. Microelectron. ICM*, vol. 2000–Janua, no. November, pp. 161–163, 1999.

[8] Wikipedia, "Moore's Law," 2016. [Online]. Available: https://en.wikipedia.org/wiki/Moore%27s_law. [Accessed: 10-Oct-2016].

[9] J. Sparsø and S. Furber, *Principles of asynchronous circuit design: a systems perspective*. Kluwer Academic Publishers, 2001.

[10] H. Jacobson, "ASYNCHRONOUS CIRCUIT DESIGN," no. May, p. 404, 1996.

[11] G. F. Bouesse, N. Ninon, G. Sicard, M. Renaudin, A. Boyer, and E. Sicard, "Asynchronous logic Vs Synchronous logic : Concrete Results on Electromagnetic Emissions and Conducted Susceptibility," no. November 2016, 2007.

[12] K.-S. Chong, B.-H. Gwee, and J. S. Chang, "Design of several asynchronous-logic macrocells for a low-voltage micropower cell library," *IET Circuits, Devices Syst.*, vol. 1, no. 2, p. 161, 2007.

[13] L. F. Cristófoli *et al.*, "On the comparison of synchronous versus asynchronous circuits under the scope of conducted power-supply noise," *2010 Asia-Pacific Symp. Electromagn. Compat. APEMC 2010*, pp. 1047–1050, Apr. 2010.

[14] K.-S. S. Chong, B.-H. H. Gwee, and J. S. Chang, "Energy-efficient synchronous-logic and asynchronous-logic FFT/IFFT processors," *IEEE J. Solid-State Circuits*, vol. 42, no. 9, pp. 2034–2045, Sep. 2007.

[15] S. Das, A. Chandrakasan, and R. Reif, "Timing, energy, and thermal performance of three-dimensional integrated circuits," in *Proceedins of the 14th ACM Great Lakes symposium on VLSI - GLSVLSI '04*, 2004, p. 338.

[16] L. Caley, C.-W. Lo, F. Sabado, and J. Di, "A comparative analysis of 3D-IC partitioning schemes for asynchronous circuits," in *ICICDT 2014 - IEEE International Conference on Integrated Circuit Design and Technology*, 2014, pp. 1–4.

[17] M. Ligthart, K. Fant, R. Smith, A. Taubin, and A. Kondratyev, "Asynchronous design using commercial HDL synthesis tools," *Proc. - Int. Symp. Asynchronous Circuits Syst.*, pp. 114–125, 2000.

[18] B. Sparkman and S. C. Smith, "Reducing Energy Usage of NULL Convention Logic Circuits using NULL Cycle Reduction Combined with Supply Voltage Scaling."

[19] L. Men, "Asynchronous Data Processing Platforms for Energy Efficiency, Performance, and Scalability," *Theses Diss.*, Aug. 2016.

[20] L. J. Caley, "High Temperature CMOS Silicon Carbide Asynchronous Circuit Design," 2015.

[21] S. C. Smith, "Completion-Completeness for NULL Convention Digital Circuits Utilizing the Bit-wise Completion Strategy."

[22] S. K. Bandapati and S. C. Smith, "Design and characterization of NULL convention arithmetic logic units," *Microelectron. Eng.*, vol. 84, no. 2, pp. 280–287, 2007.

[23] A. Kondratyev, L. Neukom, O. Roig, A. Taubin, and K. Fant, "Checking delay-insensitivity: 10/sup 4/ gates and beyond," in *Proceedings Eighth International Symposium on Asynchronous Circuits and Systems*, pp. 149–157.

[24] W. Cilio, M. Linder, C. Porter, J. Di, D. R. Thompson, and S. C. Smith, "Mitigating power- and timing-based side-channel attacks using dual-spacer dual-rail delay-insensitive asynchronous logic," *Microelectronics J.*, vol. 44, no. 3, pp. 258–269, 2013.

[25] P. Beerel and M. Roncken, "Low Power and Energy Efficient Asynchronous Design," *J. Low Power Electron.*, vol. 3, no. 213, pp. 1–60, Dec. 2007.

[26] M. Hinds, B. Sparkman, J. Di, and S. Smith, "An asynchronous advanced encryption standard core design for energy efficiency," *J. Low Power Electron.*, vol. 9, no. 2, pp. 175–188, Aug. 2013.

[27] J. P. T. Habimana, F. Sabado, and J. Di, “Multi-threshold dual-spacer dual-rail delay-insensitive logic: An improved IC design methodology for side channel attack mitigation,” in *Proceedings - IEEE International Symposium on Circuits and Systems*, 2016, vol. 2016–July, pp. 750–753.

[28] D. S. Bormann and P. Y. K. Cheung, “Asynchronous Wrapper for Heterogeneous Systems,” *IEEE Int. Conf. Comput. Des. VLSI Comput. Process.*, pp. 307–314, 1997.

[29] B. Hollosi *et al.*, “Delay-insensitive asynchronous ALU for cryogenic temperature environments,” *Midwest Symp. Circuits Syst.*, pp. 322–325, 2008.

[30] N. Karaki, T. Nanmoto, H. Ebihara, S. Utsunomiya, S. Inoue, and T. Shimoda, “A flexible 8b asynchronous microprocessor based on low-temperature poly-silicon TFT technology,” *ISSCC. 2005 IEEE Int. Dig. Tech. Pap. Solid-State Circuits Conf. 2005.*, pp. 272–274, 2005.

[31] M. Klein, “Static Power and the Importance of Realistic Junction Temperature Analysis,” 2005.

[32] L. Zhou, S. C. Smith, and J. Di, “Bit-Wise MTNCL : An Ultra-Low Power Bit-Wise Pipelined Asynchronous Circuit Design Methodology,” *2010 53rd IEEE Int. Midwest Symp. Circuits Syst.*, pp. 217–220, Aug. 2010.

[33] L. Zhou, R. Parameswaran, F. A. . Parsan, S. C. . Smith, and J. Di4, “Multi-Threshold NULL Convention Logic (MTNCL): An Ultra-Low Power Asynchronous Circuit Design Methodology.,” *Journal of Low Power Electronics & Applications*, vol. 5, no. 2. Multidisciplinary Digital Publishing Institute, pp. 81–100, 18-May-2015.

[34] P. Palangpour and S. C. Smith, “Sleep Convention Logic using partially slept function blocks,” in *2013 IEEE 56th International Midwest Symposium on Circuits and Systems (MWSCAS)*, 2013, pp. 17–20.

[35] M. Farooq, “3D IC Technology - Introduction,” 2013.

[36] C. Leitz, “Wafer to Wafer Technology.”

[37] P. Franzon, “System Design for 3D Integration,” 2013.

[38] J. Li, “Chapter 8: Introduction to 3D Integration Technology using TSV,” 2016. [Online]. Available: http://www.ee.ncu.edu.tw/~jfli/vlsia10/lecture/ch08.

[39] P. Marchal *et al.*, “3-D Technology Assessment: Path-Finding the Technology/Design Sweet-Spot,” *Proc. IEEE*, vol. 97, no. 1, pp. 96–107, Jan. 2009.

[40] B. Hollosi, T. Zhang, R. S. P. Nair, Y. Xie, J. Di, and S. Smith, “Investigation and comparison of thermal distribution in synchronous and asynchronous 3D ICs,” in *2009 IEEE International Conference on 3D System Integration, 3DIC 2009*, 2009, pp. 1–5.

[41] K. Athikulwongse, M. Ekpanyapong, and S. K. Lim, “Exploiting die-to-die thermal coupling in 3-D IC placement,” *IEEE Trans. Very Large Scale Integr. Syst.*, vol. 22, no. 10, pp. 2145–2155, 2014.

[42] R. B. Reese, S. C. Smith, and M. A. Thornton, “Uncle - An RTL approach to asynchronous design,” in *Proceedings - International Symposium on Asynchronous Circuits and Systems*, 2012, pp. 65–72.

[43] J. Di *et al.*, “Recent advances in low power asynchronous circuit design,” *J. Low Power Electron.*, vol. 13, no. 3, 2017.

[44] R. J. Thian, “Multi-threshold CMOS Circuit Design Methodology from 2D to 3D,” University of Arkansas, 2010.

[45] A. Ziesemer, R. Reis, M. T. Moreira, M. E. Arendt, and N. L. V Calazans, “Automatic layout synthesis with ASTRAN applied to asynchronous cells,” *2014 IEEE 5th Lat. Am. Symp. Circuits Syst. LASCAS 2014 - Conf. Proc.*, 2014.

[46] Tezzaron, “Tezzaron 3D Design Guide,” 2010.

[47] B. Bell, W. Bouillon, S. Li, E. Logal, and J. Di, “Application of Multi-Threshold NULL Convention Logic to Adaptive Beamforming Circuits for Ultra-Low Power,” pp. 77–80.

[48] L. Men and J. Di, “An Asynchronous Finite Impulse Response Filter Design for Digital Signal Processing Circuit,” pp. 25–28, 2014.

[49] S. C. Smith and J. Di, *Designing Asynchronous Circuits using NULL Convention Logic (NCL)*. 2009.

[50] “Cadence Design Systems.” [Online]. Available: http://www.cadence.com.

[51] P. Beerel, R. Ozdag, and M. Ferretti, *A Designer’s Guide to Asynchronous VLSI*. Cambridge University Press, 2010.

[52] “Synopsys.” [Online]. Available: http://www.synopsys.com.