A COST AND POWER EFFICIENT DDR4/GDDR5X/GDDR5 TRANSMITTER WITH 3-TAP EQUALIZER

By

NG HOONG CHIN

A Dissertation submitted for partial fulfilment of the requirement for the degree of Master of Science (Microelectronic Engineering)

August 2016

Acknowledgement

I would like to thank my company, Intel PSG, and my manager for providing the environment and infrastructure that are conducive for the research. The unlimited access to IEEE Xplore Digital Library and EDA tools such as Cadence Virtuoso and Synopsys HSPICE are critical in conducting the research within a short timeframe. A special appreciation to my supervisor, Dr. Nor Muzlifah Mahyuddin for providing clear guidance and critical feedbacks through the research. The aids are irreplaceable, especially when defining the correct expectation and scope for the research. The completion of the research would not be possible without her supervision. Finally, I would like to thank my friends and colleagues for their constant supports and information sharing.

Table of Contents

List of Tables

List of Figures

List of Abbreviations

Abbreviation	Meaning
3D	3 Dimensions
AC	Alternating Current
ADC	Analog-to-Digital Converter
ALU	Arithmetic Logic Unit
ASIC	Application-Specific Integrated Circuit
BER	Bit Error Ratio
CTT	Center Tap Terminated
DAC	Digital-to-Analog Converter
DC	Direct Current
DCD	Duty Cycle Distortion
DDR	Double Data Rate
EOL	End-Of-Life
EQ	Equalization
ESD	Electro-Static Discharge
FF	Fast NMOS, Fast PMOS
FIR	Finite Impulse Response
FS	Fast NMOS, Slow PMOS
GDDR	Graphics Double Data Rate
HBM	High Bandwidth Memory
HMC	Hybrid Memory Cube
I/O	Input/Output
IC	Integrated Circuit
ISI	Inter-Symbol Interference
JEDEC	Joint Electron Device Engineering Council
LSB	Least Significant Bit
MOM	Metal-Oxide-Metal
MSB	Most Significant Bit
PCB	Printed Circuit Board
POD	Pseudo Open Drain
PRBS	Pseudo Random Binary Sequence
PSIJ	Power Supply Induced Jitter
PVT	Process, Voltage and Temperature
RTL	Register-Transfer Level
SF	Slow NMOS, Fast PMOS
SS	Slow NMOS, Slow PMOS
SST	Source-Series Terminated
TT	Typical NMOS, Typical PMOS
UHD	Ultra High Definition
UI	Unit-Interval

Abstract

The demand on memory bandwidth has been increasing due to the rapid development in graphics intensive applications and big data analytics. Limited of the I/O and thermal densities, widening of data bus is no longer a feasible option for increasing memory bandwidth. Therefore, high-speed DDR transmitter is the current solution to the increasing demand on memory bandwidth until the through-silicon via technology is matured for 3D integration of memory modules. At high data rate, the channel losses at interconnect have necessitated the equalization on transmitting signal that will incur extra power and area to the transmitter. Achieving low cost and low power has become the greatest challenge in the design of transmitter with equalizer. For this objective, novel driver architecture in hybrid coding scheme was introduced to reduce pre-driver and routing resources. In concert with dual-function pre-driver cells, 3-tap equalization was achieved by manipulating the driver impedance code in a low power ALU. The proposed transmitter has achieved 6.4 Gb/s transmit rate in DDR4 standard with 40 % frequency margin, 10.0 Gb/s transmit rate in DDR5X standard with 5 % frequency margin and 8.0 Gb/s transmit rate in DDR5 standard with 15 % frequency margin. It has a size of 1175 um^2 and an energy efficiency of 1.1 pJ/bit at 10 Gb/s GDDR5X interface, which is 50 % lower in power and 35 % lower in area cost when comparing against the power and area cost of the related works. The hybrid-coded driver with ALU-controlled equalizer demonstrates a viable solution to a cost and power efficiency transmitter for DDR4, GDDR5X and GDDR5 standards.

Abstrak

Kemajuan dalam aplikasi-aplikasi yang grafik intensif dan analisis data yang besar telah meningkatkan permintaan terhadap bandwidth memori. Terhad oleh kepadatan I/O and haba, perleluasan bas data bukan lagi satu pilihan yang munasabah bagi meningkatkan bandwidth memori. Oleh itu, pemancar DDR kelajuan tinggi adalah penyelesaian semasa untuk bandwidth memori yang semakin meningkat selagi teknologi untuk mengintegrasi modul-modul memori secara 3D tidak matang. Pada kadar data yang tinggi, kebocoran saluran di antara sambungan telah mewajibkan penggunaan penguat dalam penghantaran isyarat di mana ia akan menambahkan pengunaan kuasa dan kawasan litar oleh pemancar. Oleh itu, kos dan kuasa yang rendah telah menjadi cabaran yang paling besar untuk diperoleni dalam reka bentuk pemancar dengan penguat isyarat. Bagi tujuan ini, seni bina pemancar baru yang mengguna skim pengekodan hibrid telah diperkenalkan untuk mengurangkan bilangan pra-pemancar dan jalur sambungan. Bersama dengan sel-sel pra-pemancar dwi-fungsi, penguat tiga tap telah dicapai dengan memanipulasi kod pemancar dalam ALU kuasa rendah. Pemancar yang dicadangkan telah mecapai kadar data 6.4 Gb/s dengan margin sebanyak 40 % dalam aplikasi DDR4, kadar data 10.0 Gb/s dengan margin sebanyak 5 % dalam aplikasi GDDR5X dan kadar data 8.0 Gb/s dengan margin sebanyak 15 % dalam aplikasi GDDR5. Ia mempunyai saiz 1175 um^2 dan kecekapan tenaga setinggi 1.1 pJ/bit dengan GDDR5X pada kadar 10 Gb/s, iaitu 50 % lebih rendah dari segi kuasa dan 35 % lebih rendah dari segi kos fabrikasi apabila berbanding dengan kajian-kajian yang berkaitan. Pemancar berkod hibrid dengan penguat berkawalan ALU telah menbentangkan satu penyelesaian untuk pemancar kos dan kuasa rendah dalam aplikasi DDR4, GDDR5X dan GDDR5.

CHAPTER 1

INTRODUCTION

1.1 Background

The demand on a higher memory bandwidth has been consistently increasing with the development of graphics intensive applications such as autonomous driving, virtual or augmented reality and cloud services for ultra-high definition (UHD) contents [1]. The recent boom in the big data analytics for cloud services and machine learning is also contributing to the demand on memory bandwidth [2]. In meeting the demand, the memory data bus could be widened and the data rate per I/O pin could be increased [3]. However, increasing the data bus width is not power and cost efficient as there will be more signal traces that occupy a larger board area and contribute more parasitic capacitances. Moreover, today high performance systems have already reached the limits of their I/O density and thermal heat dissipation [1]. As a result, inserting additional I/O pins to the devices is no longer a feasible solution to memory bandwidth. In order to overcome the limit of I/O density, there has been a drive toward 3D integration of memory modules such as Hybrid Memory Cube (HMC) and High Bandwidth Memory (HBM) [4]. However, until the through-silicon via technology is matured for 3D integration [5], higher bandwidth has to be enabled by increasing data rate per pin for better cost and energy efficiencies. This implies that, high-speed double data rate fourth-generation (DDR4), graphic double data rate fifth-generation extension (GDDR5X) and graphic double data rate fifth-generation (GDDR5) interfaces are expected to meet the needs of server, client, graphics and mobile platforms in the near term [6].

The development of DDR transmitter in this research thesis will be conducted using the 20 nm process node where the minimal channel length of thin oxide transistors is 20 nm. However, the transmitter is unable to directly benefit from the shrinking of process node as the supply voltage of DDR interface is higher than the operating voltage of thin oxide transistors in this deep sub-micron process node [7]. With the I/O supply voltages rated at 1.2 V for DDR4, 1.35 V for GDDR5X and 1.5 V for GDDR5, thick-oxide transistors are required in the traditional driver design, that leads to a decrease in performance and an increase in power and area costs. In order to regain the advantages from the shrinking of process node, circuit technique is employed to enable the transmitter to be built using only thin oxide transistors. With this, producing a transmitter with high data rate, low power and low area costs is now possible.

1.2 Problem Statement

The quality of high-speed interconnects are crucial for memory buses. However, with the ever-increasing demand for high data rate in the memory interfaces, board materials and connectors used in these interconnects have experienced comparatively little improvement [8]. As the data rate of the interface increases, inter-symbol interference (ISI) becomes one of the key deterrents to signal integrity. As a signal transmits through a printed circuit board (PCB) channel, it suffers from high frequency losses such as skin effect and dielectric loss as illustrated in Figure 1.1. As a result, the slew rate of the transmitting signal may degrade until its transition extends beyond one unit-interval (UI). This phenomenon, known as ISI, can be addressed by using multi-tap equalization method that imitates the characteristic of finite impulse response (FIR) filter. However, the implementation of a conventional 3-tap equalizer that can compensate both pre-cursor and post-cursor ISI causes significant overhead in power and area. In term of power, there are dynamic power dissipation from the addition of data paths and logics for the equalization purpose and static power dissipation from the constant impedance de-emphasis scheme. In term of area, there is area overhead

from the dedicated driver and pre-driver. Therefore, this research thesis will explore a low cost and low power design of a 3-tap equalizer for DDR4/GDDR5X/GDDR5 transmitter.

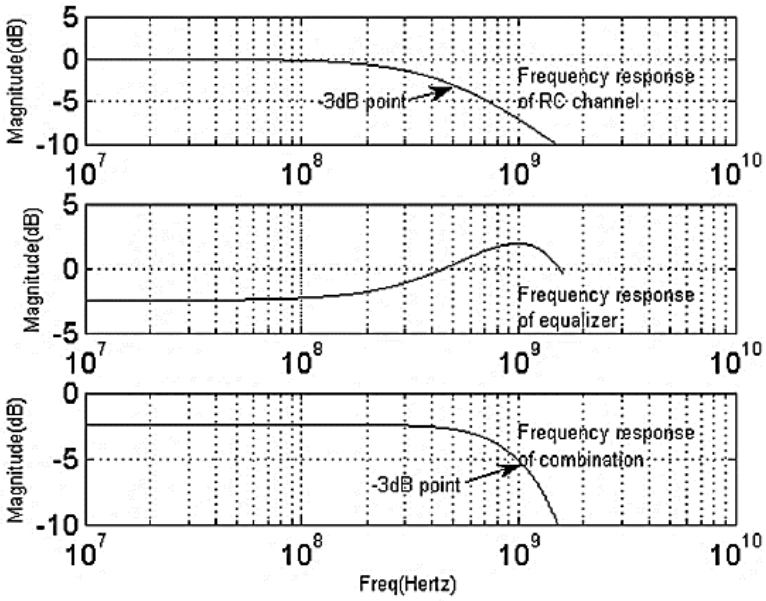

Figure 1.1: Frequency responses of a distributed RC interconnect channel, equalizer and their combination [9]

1.3 Objectives

The objectives of this research thesis are as the following

1. To achieve high transmit rate using the proposed transmitter with 3-tap equalizer
    a. To achieve 6.4 Gb/s transmit rate in DDR4 standard
    b. To achieve 10.0 Gb/s transmit rate in GDDR5X standard
    c. To achieve 8.0 Gb/s transmit rate in GDDR5 standard
2. To achieve high energy efficiency and small area for the proposed transmitter with 3-tap equalizer

1.4 Project Scope

This research will cover the design of a high-speed DDR4/GDDR5X/GDDR5 transmitter that has a 3-tap equalization capability. The data registers to the transmitter and its clock source will not be covered. The proposed transmitter will be designed using the 20 nm process node. It will be characterized in HSPICE simulations using pre-layout netlist and validated against the specifications defined by JEDEC. Complete interconnect channel simulation will be conducted for eye diagram analysis at the target toggle rate for each I/O standards. Then, the performance, power consumption and area cost of the proposed transmitter will be analyzed and compared with previous design approaches.

1.5 Thesis Outline

This thesis is written and structured into five chapters. The first chapter is this introductory chapter that gives an insight into the background, problems, objectives and scope of the thesis. The outline of subsequent chapters is as below.

In Chapter 2, a literature review on driver architecture and equalizer architecture will be presented. The scope of the review will be bounded to the prior designs that are applicable to single-ended transmitter in double data rate (DDR) memory interface. In the context of driver architecture, binary-weighted, linear-weighted and hybrid driver structures will be covered. In the context of equalizer architecture, dedicated and shared equalizer schemes will be discussed. Moreover, the advantages and disadvantages of each related work will be investigated and compared against each other.

In Chapter 3, the methodology of designing a high-speed DDR transmitter will be detailed. Initially, the operating conditions of the transmitter and the overall project workflow will be established to form the guiding framework. Then, the functionality of all three modules in transmitter – driver, ALU and pre-driver will be elaborated. Furthermore, the design considerations made during the design process for achieving the objectives of the thesis will be highlighted.

In Chapter 4, various performance parameters of the transmitter will be examined through HSPICE simulations to ensure that all the goals of the research are met. As the foundation to the verification, the assumption and setup used in HSPICE simulation will be defined upfront. Then, the simulations results on duty cycle distortion and output slew rate will be presented and verified against specifications. The performance of the transmitter in term of maximum toggle rate will be characterized using the statistical eye diagram approach. Finally, the power and area of the transmitter will be tabulated and compared against related works.

In Chapter 5, the conclusion and future works of the research will be presented. A conclusion will be drawn from all the design activities, considerations and simulation results documented in preceding chapters. The remaining works for the research and future research area will be proposed to improve the performance and robustness of the proposed transmitter.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

In this chapter, prior researches on high-speed transmitter and equalization technique will be scrutinized. Driver architecture and equalizer architecture will be the two focus areas as they are the foundation of a transmitter that will determine its performance, power and area. The pros and cons of each related works will be highlighted and pitted against each other. In Section 2.2, various driver structures of the single-ended source-series terminated (SST) driver, which was used in DDR memory interface, will be investigated. In Section 2.3, various equalization schemes for reducing ISI in high-speed application will be considered.

2.2 Driver Architecture

The driver of a transmitter is the final stage on an integrated circuit (IC) device before the data signal is sent off chip through a PCB channel to one or more destination devices. Due to the large parasitic capacitance from the channel trace and the receiving device, the dimensions of the driver are scaled up to attain the required current strength, making it harder to be driven by a single pre-driver. Furthermore, channel reflection has imposed a requirement on impedance matching between the driver and the channel to avoid signal integrity issue. Hence, the output driver is segmented into binary-weighted driver slices, linear-weighted driver slices or the hybrid of the two former types of driver slices to enable the control on its impedance through multiple pre-drivers.

2.2.1 Binary-weighted Driver Slice

Voltage-mode driver with binary-weighted driver slices is the conventional driver architecture that utilizes the concept of parallel resistive digital-to-analog converter (DAC) [1]. The impedance of a binary-weighted driver slice is designed to have half of the impedance of the driver slice of its next less significant bit. For this purpose, the width of transistor is twice of the width of previous transistor, as moving from least significant bit (LSB) to most significant bit (MSB) as depicted in Figure 2.1.

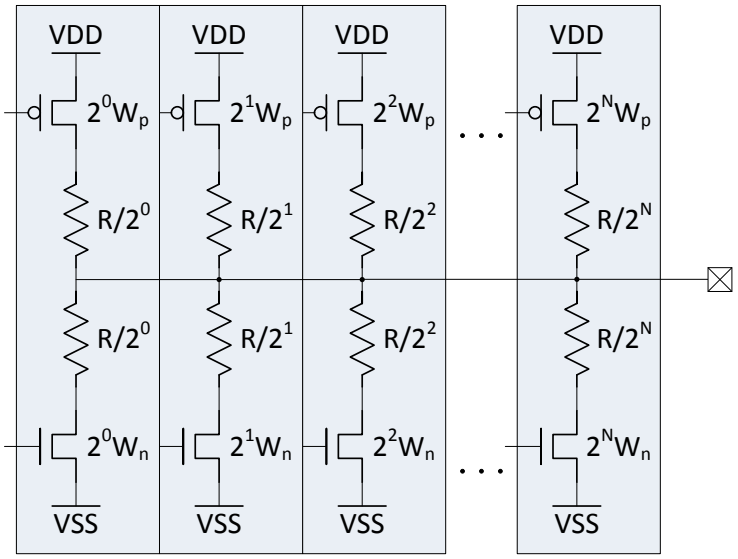

Figure 2.1: Driver using binary-weighted driver slices

The key advantage of binary-weighted driver architecture is the simplicity in the implementation of driver and pre-driver. In [10], [11] and [12], the driver was segmented in a single binary-weighted pattern and each of the controls to the driver was meant to carry the information on data signaling and impedance control. Furthermore, a high impedance resolution was achieved with minimal number of binary-weighted driver slices as an inherited property of binary-weighted DAC, leading to the saving in dynamic power and die area [13].

However, the binary-weighted driver architecture is not conducive for incorporating an equalizer as explained in [1] and [14]. The ratio of equalization level to output impedance is uncontrollable and the dynamic range of equalization is limited without further slicing of the

binary-weighted driver slices. As an attempt to obtain an effective equalization in a 5-bit binary-weighted driver, each binary-weighted driver slice was further segmented into 4-bit binary-weighted sub-drivers for controlling the equalization level in [15]. Nevertheless, the numbers of control signals and pre-drivers were increased by four times in the proposed design. Essentially, the attempt has defeated the principal benefit of binary-weighted driver architecture.

2.2.2 Linear-weighted Driver Slice

In order to overcome the complication in implementing equalizer using the binary-weighted driver slice, linear-weighted driver slice was introduced. The linear-weighted driver slice architecture is commonly constructed using multiple parallel driver slices with equal impedance. As shown in Figure 2.2, each of the driver slices was designed to produce an impedance of 480 Ω after calibration as presented in [7]. 480 Ω was selected to provide extra resolution on top of supporting the industrial output impedance standards that ranges in the factor of 240 Ω. The desired ratio of equalization level to output impedance can easily be controlled by disabling the same ratio of equal impedance driver slices that are enabled by a thermometer code.

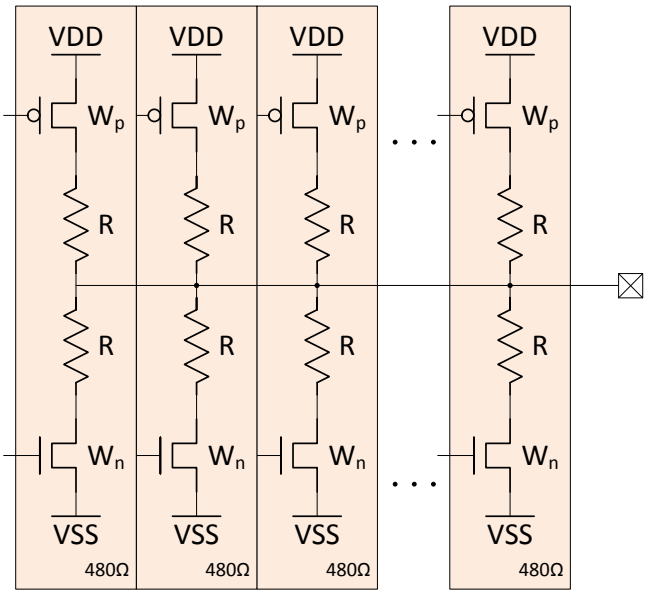

Figure 2.2: Driver using 480 Ω linear-weighted driver slices

In [7], the driver slice was built with stacked transistors in series with resistors on both its pull-up and pull-down networks. Within the stacked transistors, the transistors that are closer to the I/O pad were used to propagate data to I/O pad whereas the transistors that were connected to power or ground node were used to control the impedance of driver slice via 8-bit calibration code respectively. The stacked transistors setup is critical in reducing the number of pre-drivers and connections between the pre-driver and driver slice. All the data transistors on the pull-up or pull-down networks within a driver slice were correspondingly driven by a common pre-driver. Furthermore, the 8-bit calibration codes, separately for the pull-up and pull-down networks, were shared among all driver slices. However, the stacked transistors architecture has a disadvantage on the capacitance of I/O pin and the area of driver. Due to the decrease of effective drive strength when transistors were stacked, the width of each transistor in the driver slice has to be increased to regain the effective drive strength. An increase in the width of transistors within the driver slice will translate to an increase of the capacitance of I/O pin which is dominated by the junction capacitance of transistors.

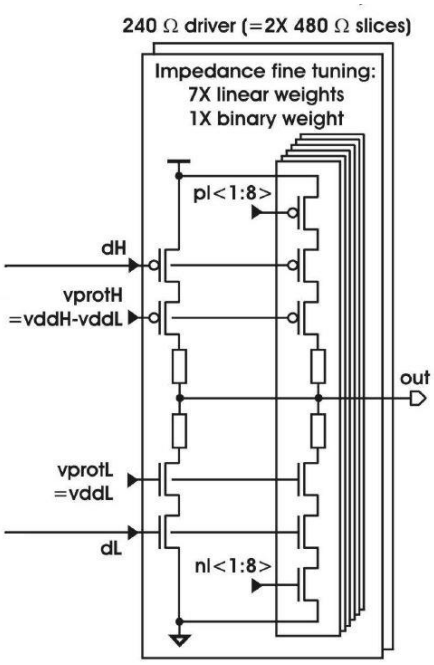

Figure 2.3: Linear-weighted driver slice using stacked transistors [7]

An alternative driver design that consists of multiple parallel driver slices using non-stacked transistors was proposed in [16] and [17] with 750 Ω and 240 Ω per slice respectively. In the implementation delineated in [17], the 240 Ω driver slice was formed by arraying 26 identical units of series combination of a resistor and a transistor for both pull-up and pull-down networks. Each of these units was driven by a unique signal, translating to a total of 52 control signals for each driver slice. With six instances of driver slices in the entire driver, routing resource is at risk with 312 signals between pre-drivers and driver slices. In addition, the dynamic power consumption is high at 13.7 pJ/bit as compared to 2.7 pJ/bit in [18] that was designed in the same 180 nm node due to the parasitic capacitance from large amount of interconnects.

In order to reduce the large number of pre-driver instances and its connections to multiple driver slices in [17], a driver slice with scalable impedance was introduced in [19]. Within the proposed driver slice, there are 8-bit binary-weighted legs that were constructed in the common series combination of a resistor and a transistor for both pull-up and pull-down networks to enhance the linearity of output impedance. Among the eight legs, only six legs are controlled by the impedance calibration code while the other two legs are used for impedance scaling through code shifting. The six LSB legs are driven by the calibration code when the driver slice is calibrating against the external 240 Ω reference resistor. Then, the 120 Ω and 60 Ω impedances of the drive slice are achieved by left shifting the calibration code by one and two bits respectively as shown in Figure 2.4.

With the scalability to 60 Ω per driver slice in [19], only two units of this driver slice are needed to obtain 30 Ω output impedance as opposed to seven units of the conventional 240 Ω driver slice are needed to obtain 34 Ω output impedance. As a result of the reduction in the number of drive slices, the associated pre-drivers and connections between pre-driver and driver slice were significantly scaled down accordingly. However, the scaling from 240 Ω to 60 Ω within a driver slice will impact the accuracy of impedance calibration. When performing upscaling of the calibration code to obtain 60 Ω, the original calibration code is

padded with two zeros at the LSB end. Consequently, the two LSB legs are disabled and the LSB is effectively shifted to bit 2 which corresponds to four times reduction in resolution.

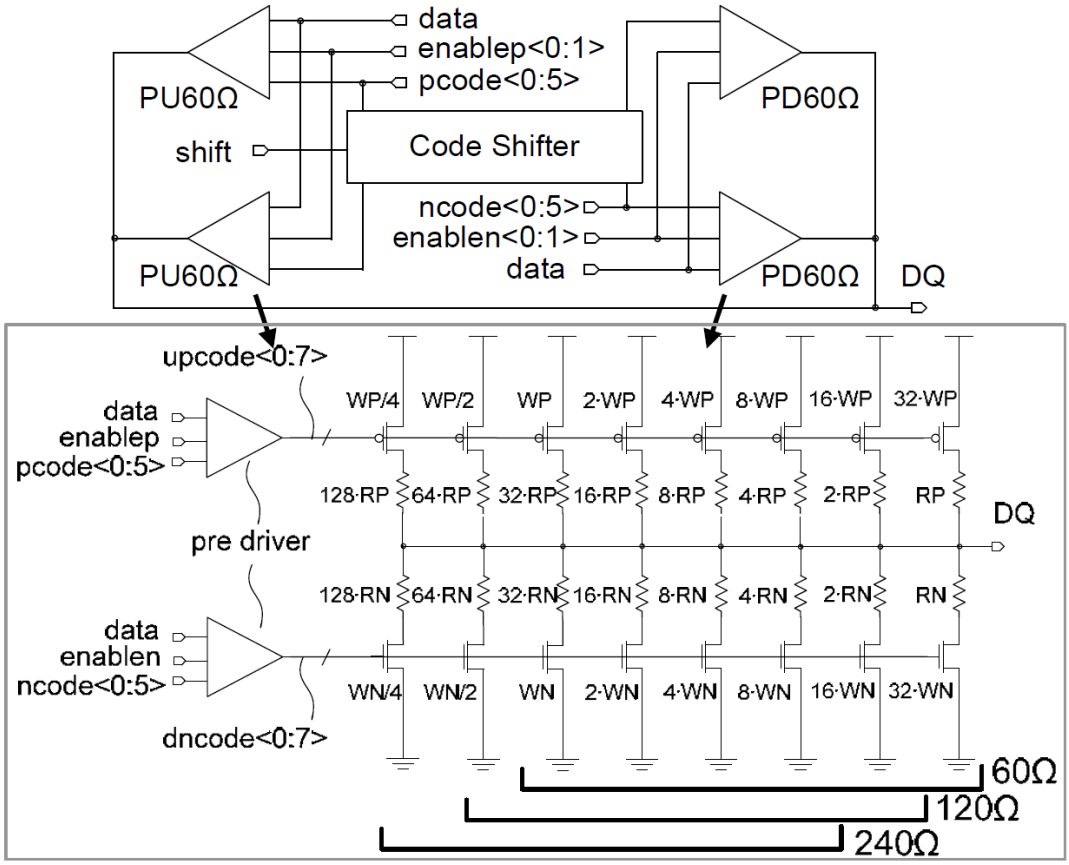

Figure 2.4: Driver using multiple identical drivers with scalable impedance [19]

2.2.3 Hybrid Driver Slice

As illustrated in Figure 2.5, a voltage-mode driver that consists of anchor legs and dynamic legs was proposed in [20]. It is a hybrid driver architecture where the anchor legs are thermometer-coded and the dynamic legs are binary-coded. Under this setup, the anchor legs were controlled by user setting to determine the equalization level whereas the dynamic legs were controlled by calibration code. This separation of legs by functionality is able to reduce the complexity on driver slices and pre-drivers, and at the same time, enlarge the dynamic range on the equalizer. The benefit in the purview of power and area from the binary-weighted architecture and the benefit in the scalability of equalizer from the linear-weighted architecture were realized in the hybrid driver architecture.

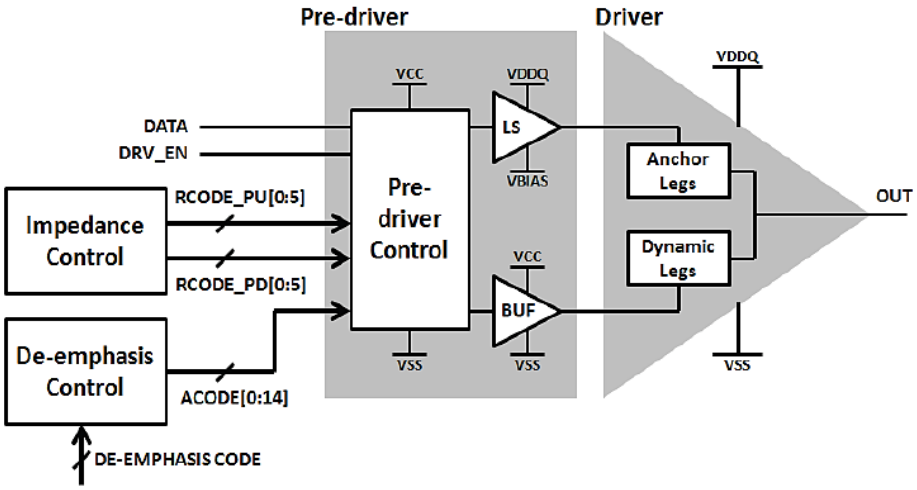

Figure 2.5: Voltage-mode driver with anchor legs and dynamic legs [20]

However, the driver architecture using anchor legs and dynamic legs is unable to maintain a fixed ratio of equalization level to output impedance across process, voltage and temperature (PVT) corners. As the anchor legs and dynamic legs operate independently, the PVT information that is available in the dynamic legs via calibration code is absent from the anchor legs. The usability of the equalizer is greatly degraded due to the requirement on manual tuning of equalization level in each device.

2.3 Equalizer Architecture

Equalization of transmitting signal is extensively used in high data rate link to overcome the frequency-dependent attenuations arising from dielectric loss and skin effect of transmission line. It is achieved by strengthening the high frequency components during the transitions of symbols (pre-emphasis) or weakening the low frequency components during the continuities of symbol (de-emphasis). For this purpose, an equalizer in the form of a dedicated driver or a shared driver with the normal driver is required.

2.3.1 Dedicated Equalizer Driver

Dedicated equalizer driver architecture is commonly used in the pre-emphasis equalization technique. Strengthening of transition edges is straightforward with direct

injection of current into a channel using an extra driver that operates in parallel with a normal driver. However, the dedicated equalizer is often not calibrated with the normal driver [3], [6], [9]-[10], [21]. Consequently, this architecture cannot provide a fixed ratio of equalization level to output impedance across PVT corners.

In [9], a 2-tap pre-emphasis setup of pre-cursor and main-cursor was constructed to function with a dedicated equalizer driver. This standalone driver was sized ten times larger than the normal output driver. In addition, the pre-emphasis driver was designed to produce a full rail-to-rail swing unlike the attenuated swing on the normal driver. With a strong, wide-swing pre-emphasis driver, peaking at the transitions of output signal is created to compensate for the channel losses. This peaking duration is determined by the separation between pre-cursor and main-cursor that is spaced by an inverter-based delay cell in [9]. As the delay cell is not PVT compensated, slow device will have longer pre-emphasis and this is aligned to the need for higher equalization in the slow device. Despite the simplicity of this equalizer circuit, the large pre-emphasis driver has led to large design area and I/O pin capacitance.

An attempt to reduce the size of equalizer driver was proposed in [3] where pre-emphasis has been applied to the output nodes of pre-drivers in addition to the final driver stage. As shown in Figure 2.6, the pre-emphasis on the internal nodes were engineered onto the pre-driver for normal driver and the pre-driver for pre-emphasis driver. As a result, a portion of equalization effect was contributed by the normal driver. Coupling with faster transitions at the input gate of equalizer driver, the size of pre-emphasis driver can be substantially scaled down while maintaining the similar equalization level. Although the impact on I/O pin capacitance was mitigated, this scheme does not improve on the design area and dynamic power consumption with its additional pre-emphasis circuits for the pre-drivers.

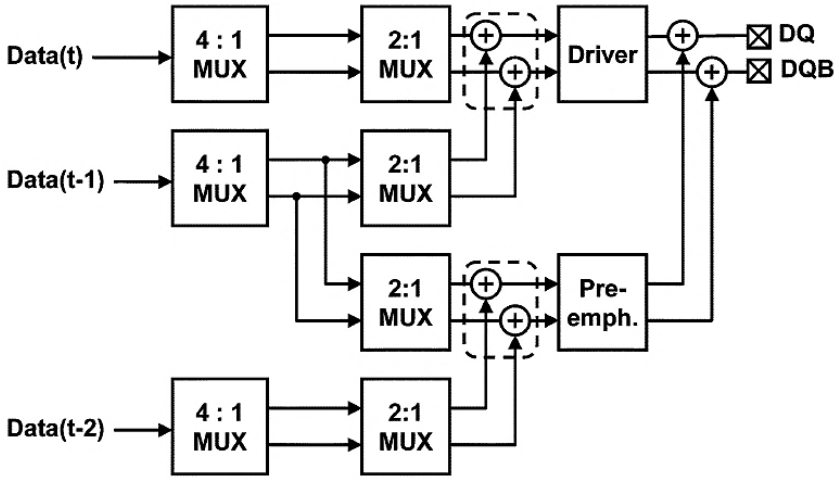

Figure 2.6: Equalization on the outputs of drivers and pre-drivers [3]

Another attempt on tackling the I/O pin capacitance due to extra equalizer driver was presented in [10]. In this effort, a passive capacitor and a transmission gate were connected in series between the output of an equalizer driver and the I/O pad as illustrated in Figure 2.7. The capacitor is used as a high pass filter that will limit the injection of current from the pre-emphasis driver onto I/O pad to the transitions of symbols, eliminating the need for a pre-cursor or post-cursor. On the other hand, the transmission gate is used to isolate the capacitor and the pre-emphasis driver from the I/O pad when in receiver mode. Although the parasitic capacitance from the equalizer driver was detached from the I/O pin capacitance in this configuration, new parasitic capacitance from the transmission gate was added. Moreover, the sizing of the transmission gate is problematic as a small sizing will cause high attenuation at the high pass filter while a large sizing will sacrifice the I/O pin capacitance. Then, the efficiency of equalizer driver is further reduced by the AC coupling from source to gate and from drain to gate at the large transmission gate.

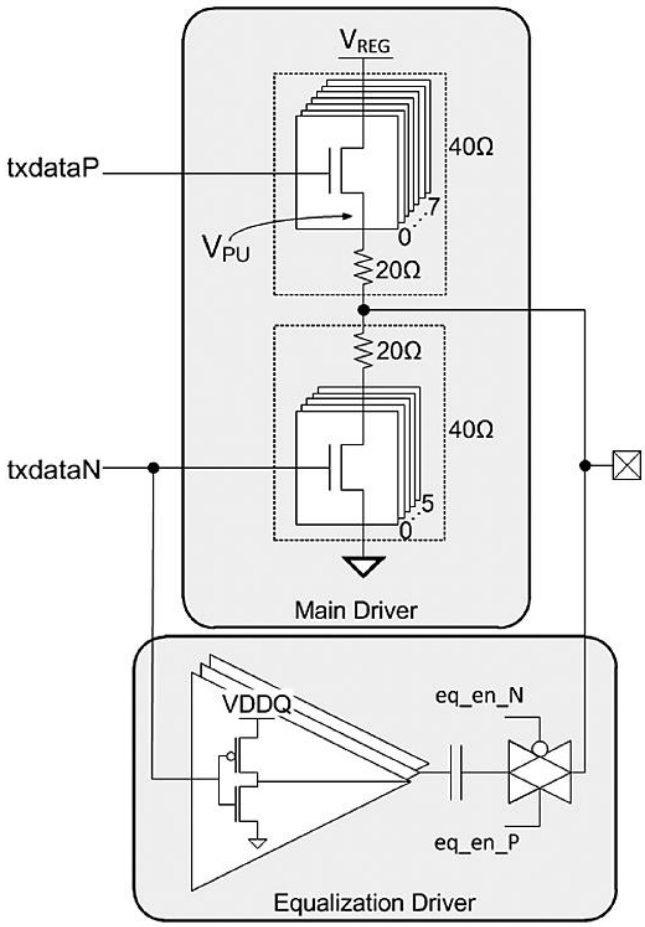

Figure 2.7: Equalization using AC-coupled driver [10]

2.3.2 Shared Equalizer Driver

In contrast to the dedicated equalizer driver architecture, shared equalizer driver architecture is commonly used in the de-emphasis equalization technique. There are two approaches to weaken the low frequency components of a transmitting signal in the shared equalizer driver architecture. First is the non-constant impedance approach where a portion of the driver slices are switched off when no change in symbol [20], [22]. Another is the constant impedance approach where a portion of the driver slices are driven into opposite polarity [7], [14]-[15], [18]. In the shared equalizer driver architecture, a configurable ratio of the entire active driver is used for de-emphasis except the hybrid driver architecture descripted in Section 2.2.3. Hence, it can generate a fixed ratio of equalization level to output impedance across PVT corners. However, the reduction in the voltage level of a transmitting

signal in the de-emphasis technique will deteriorate its voltage margin, limiting the maximum equalization level of this architecture [15], [23].

In the non-constant impedance de-emphasis scheme presented in [20], a 2-tap de-emphasis setup of main-cursor and post-cursor was used to control all 15 anchor legs. Programmable through user setting, a number of these anchor legs can be switched off when no signal transition to produce a smaller output voltage swing. With this configuration, a substantial power is conversed during long symbols but the output impedance is varied when the de-emphasis occurs, causing reflections in the channel. As the impedance mismatch increases with the level of de-emphasis, the usability of a high de-emphasis level is restricted, making the non-constant impedance approach unsuitable for high-speed links above 2.4 Gb/s [20].

On the other hand, the constant impedance de-emphasis scheme was chosen by [7], [14], [15] and [18] that have operating data rate at 3.6 Gb/s and above. In this approach, the de-emphasis is achieved by driving an inverted data to the input gate of a number of selected driver slices. The original transmitting signal will be countered by these drive slices, causing a drop in the output voltage swing. In contrast to the non-constant impedance approach, there is still a path to power or ground through the driver slices that are used for de-emphasis. Due to the fact that both power and ground are perceived as AC ground, the effective output impedance does not change even if the driver slices are driven to the opposite polarity. Furthermore, only 20 % of its overall driver slices were used for de-emphasis in [7] to avoid over-designing to an impractical de-emphasis level as in [20]. This design consideration is able to converse the area and the power used by the pre-drivers. However, the power that is consumed during long symbols is significantly higher in the constant impedance de-emphasis approach. This is because of a crowbar current path that flows across normal driver slices and the inverted de-emphasis driver slices.

2.4 Summary of Related Works

Table 2.1 summarizes the architecture, key performance parameters, power and area of three related works that were designed in the process node around 20 nm. In the transmitter proposed in [6] and [10], binary-weighted driver architecture was employed to attain area and power savings at the pre-driver. However, these savings were consumed by the dedicated equalizer driver in these works as a shared equalizer driver is not possible under the binary-weighted driver architecture. The impedance of the driver is also limited to a single 40 Ω configuration in both works in order to further reduce the area impact due to the dedicated equalizer driver.

Table 2.1: Summary of related works

Reference	[10]	[6]	[7]
Technology	28 nm	22 nm	22 nm
Driver Architecture	binary-weighted	binary-weighted	linear-weighted
Equalizer Architecture	dedicated equalizer	dedicated equalizer	shared equalizer
Equalization Scheme	AC-coupled pre-emphasis	non-constant impedance de-emphasis	constant impedance de-emphasis
Equalizer Tap Configuration	2-tap with post-cursor	2-tap with post-cursor	2-tap with post-cursor
I/O Standard	custom 1.0 V I/O	DDR4, GDDR5	DDR3, DDR4
Impedance [Ω]	40	40	24 – 40
Data Rate [Gb/s]	6.4	6.4	5.0
Efficiency [pJ/bit]	2.68	2.5	2.2
Area [um^2]	-	5400	10956

On the other hand, linear-weighted driver architecture was selected in the transmitter introduced in [7] to enable a shared equalizer driver. However, a large area was consumed by the linear-weighted driver because its driver slices were designed with tripled-stacked transistors as shown in Figure 2.3. There is further area and power penalties due to complex pre-drivers and routings for controlling these linear-weighted driver slices. This research thesis will address the issues of binary-weighted and linear-weighted driver architecture by proposing a new architecture that avoids the complexity of driver and pre-driver in the linear-weighted driver architecture while enabling a shared equalizer driver.

CHAPTER 3

METHODOLOGY

3.1 Introduction

The transmitter in this work is consisted of three major modules – an arithmetic logic unit (ALU), a pre-driver and a driver – as illustrated in Figure 3.1. These modules were constructed using implementation strategies that are most appropriate by taking into account of their nature and requirement. Hence, the driver and pre-driver were designed at transistor level using full custom approach where performance is critical at every stage of the high-speed data paths. On the other hand, the ALU that handles all the static control signals was implemented using cell-based approach to reduce design time. Regardless of their implementation strategy, all these modules were implemented on the 20 nm process node.

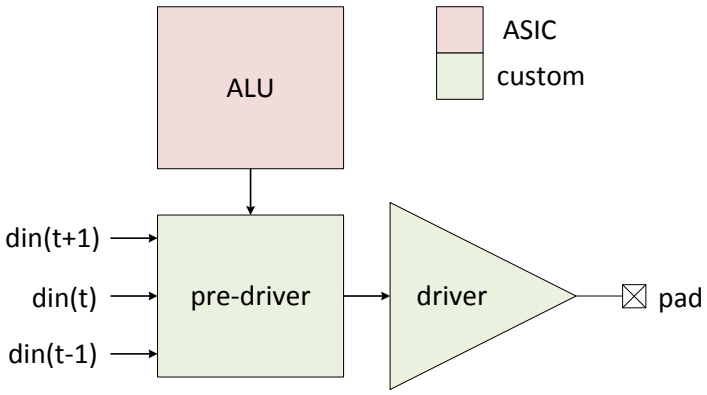

Figure 3.1: Block diagram of transmitter

In the subsequent sections within this chapter, the methodology of designing a high-speed transmitter will be presented. First, the operating conditions of the transmitter will be defined in Section 3.2 as the high-level design constraints. Then, the project methodology will be outlined in Section 3.3 and the design process of all three modules in transmitter –

driver, ALU and pre-driver – will be elucidated in Section 3.4, Section 3.5 and Section 3.6 respectively.

3.2 Operating Conditions

The I/O interface standards for DDR4, GDDR5 and GDDR5X applications are supported by the transmitter of this work. These I/O standards are the JEDEC standards with their specification defined in the following documents – JESD209-4A (DDR4), JESD212C (GDDR5) and JESD232 (GDDR5X) [24]-[26]. Using the specifications in the JEDEC's documents as its foundation, Table 3.1 was constructed to set the operating conditions of the transmitter.

Table 3.1: Operating conditions for the transmitter

Symbol	Parameter		DDR4	GDDR5X	GDDR5	Unit
VDD	I/O supply voltage	Min	1.1400	1.3095	1.4550	V
		Typ	1.2000	1.3500	1.5000	
		Max	1.2600	1.3905	1.5450	
VCC	Core supply voltage	Min	0.87	0.87	0.87	V
		Typ	0.90	0.90	0.90	
		Max	0.93	0.93	0.93	
VSS	Common ground	Min	-	-	-	V
		Typ	0	0	0	
		Max	-	-	-	
T_{OPER}	Operating temperature	Min	-40	-40	-40	°C
		Typ	25	25	25	
		Max	125	125	125	
F_{MAX}	Maximum operating frequency	Min	2.666	4.000	3.000	GHz
		Typ	-	-	-	
		Max	3.200	5.000	4.000	
Rs	Driver impedance for pull-up and pull-down	Min	34	34	34	Ω
		Typ	-	-	-	
		Max	60	60	60	
C_{IO}	Pin capacitance	Min	-	-	-	pF
		Typ	-	-	-	
		Max	1.3	1.0	1.0	
I_L	Pin leakage	Min	-	-	-	µA
		Typ	-	-	-	
		Max	20	20	20	
RZQ	External reference resistor for ODT calibration	Min	-	-	-	Ω
		Typ	240	240	240	
		Max	-	-	-	

As a part of these operating conditions, the transmitter is required to tolerate a variable I/O supply voltage that ranges from 1.2 V for DDR4 to 1.5 V for GDDR5. Furthermore, it must be compliant to the operating temperature for automotive grade that ranges from -40 °C to 125 °C. The transmitter is also targeted to achieve the highest speed grade for DDR4 and GDDR5 standards, and to reach 5 GHz data frequency for GDDR5X. At the same time, the output driver of the transmitter must support programmable impedance that ranges from 34 Ω to 60 Ω is required. Ultimately, Table 3.1 will serve as the constraints during the design process of driver, ALU and pre-driver.

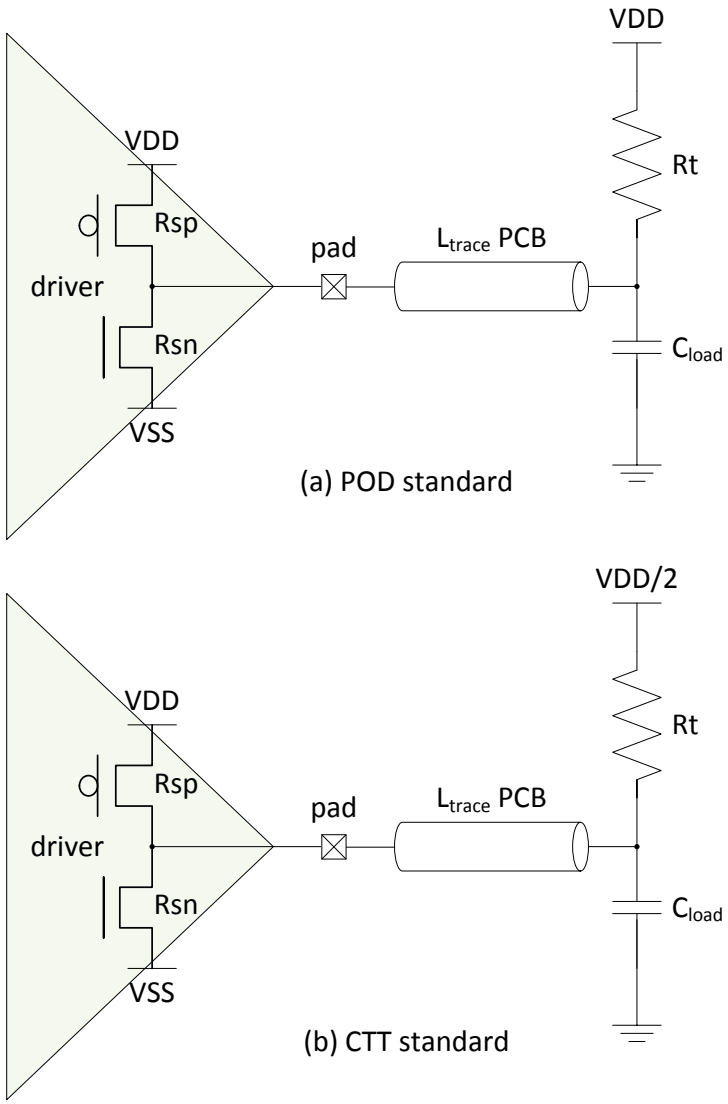

Figure 3.2: I/O standard types and their termination scheme – (a) POD standard (b) CTT standard

In the context of channel termination requirement, the DDR4, GDDR5 and GDDR5X I/O standards are belonged to the Pseudo Open Drain (POD) I/O standard family where its termination scheme is different from the Center Tap Terminated (CTT) scheme used by their precedent I/O standards like the DDR3 standard. In the POD standard, the channel is terminated to VDD voltage level; whereas, in the CTT standard, the channel is terminated to VDD/2 voltage level. As a result, the output logic high in the POD topology is at VDD voltage level and consumes no power. However, as the cost for a 50 % power saving over the CTT topology, a more stringent requirement on the linearity of output driver was imposed due to the asymmetrical output swing in the POD topology. Unlike the output signal in the CTT standard that is symmetrical about VDD/2, the voltage across the pull-down driver is higher than the pull-up driver in the POD standard. Consequently, the pull-down driver is exposed to higher non-linearity effect than the pull-up driver. In order to minimize this mismatch, the driver for supporting the POD standards is required to have high linearity.

3.3 Project Methodology

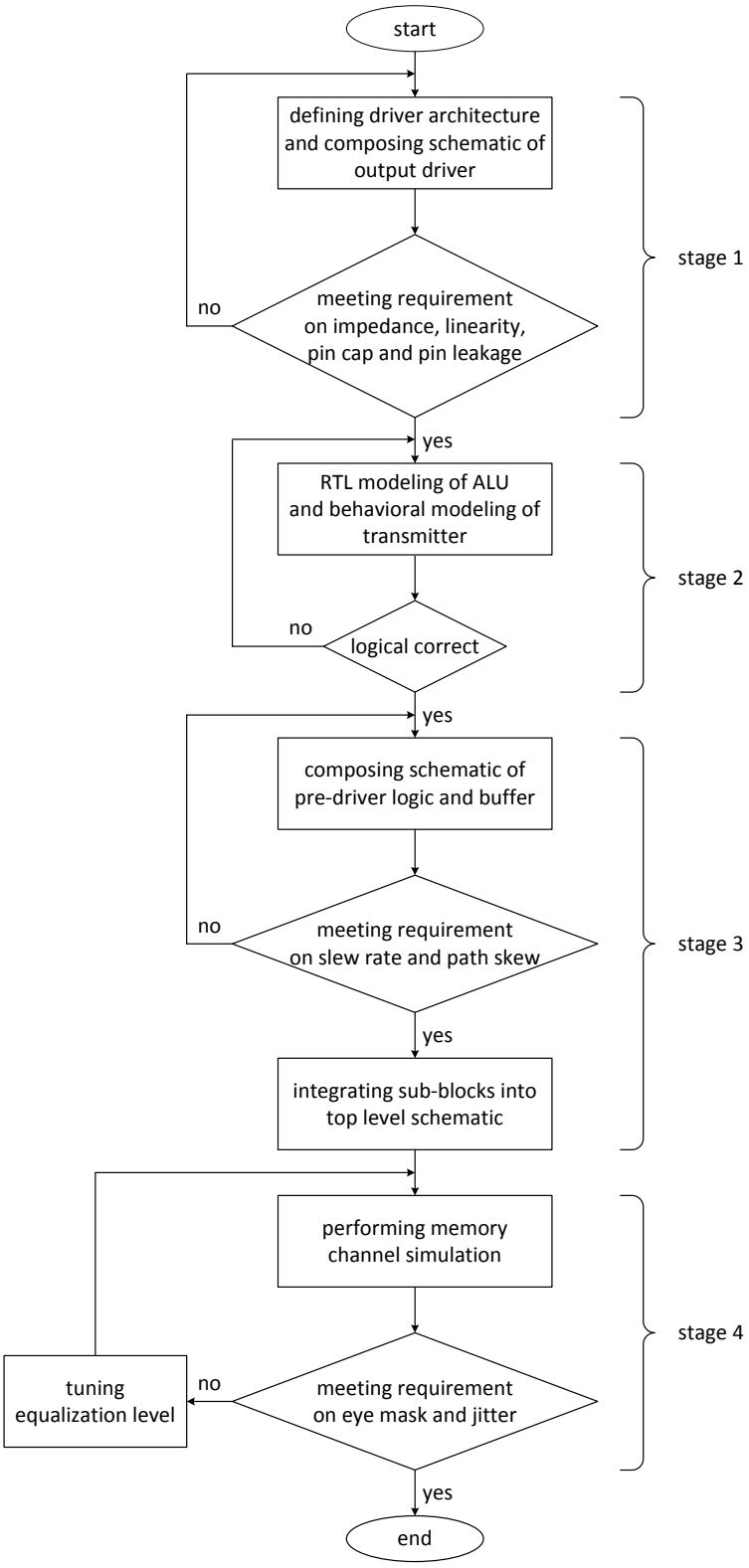

Figure 3.3: Design flow

The design flow used in the development of the transmitter is outlined in Figure 3.3. In the first stage (Section 3.4), the architecture of output driver was defined and its schematic was composed. This was followed by the second stage (Section 3.5) involving the RTL modelling of ALU that functions coherently with the driver architecture. Then, the focus was on the design of pre-driver in the third stage (Section 3.6), towards the completion of the transmitter. Finally, the performance of the transmitter was validated in channel simulation in the fourth stage that will be discussed in Section 4.6 of the next chapter.

3.4 Driver

3.4.1 Driver Architecture

The driver is the most important block of a transmitter, especially for this work that utilizes novel hybrid driver architecture. In this architecture, the driver is partitioned and controlled by a pair of 18-bit codes – one code, named as *drvp_code[17:0]*, for the pull-up driver and another code, named as *drvn_code[17:0]*, for the pull-down driver. These codes, generalized as *drv_code[17:0]*, are a combination of binary-weighted code at bits [3:0] and linear-weighted code at bits [17:3] as shown in Figure 3.4. The entire driver is comprised of 31 units of full-sized driver slices and one unit each of half-sized and quarter-sized driver slices. This built-up is equivalent to a 7-bit binary-weighted driver and this implies that the 18-bit *drv_code* is decoded from a 7-bit binary code (The decoding of *drv_code* is detailed in Section 3.5.3.).

The driver was designed to have the resolution of a 7-bit binary-weighted DAC in order to have less than 10% tolerance in calibrated driver impedance that was specified in the JEDEC's DDR4, GDDR5 and GDDR5X standards while providing sufficient drive strength to meet the minimum driver impedance of 34 Ω. Based on Eq. 3.1, the impedance of LSB slice must be greater than 1200 Ω at the fast PVT corner under the conditions that the impedance of RZQ is 240 Ω and the limit of resolution error is 10 %. Then, the impedance of

a full-sized driver must be greater than $\frac{1200}{4} \times \frac{1.3}{0.7} = 557\ \Omega$ assuming that the variation from both fast and slow PVT corners to typical PVT corner is 30 %. Using Eq. 3.2, at least 7-bit binary-weighted driver slices, 31.75 units of full-sized slices, are required to obtain less than 34 Ω in driver impedance. Concisely, the driver design with 7-bit resolution is optimum for the required design specifications and any further increase in resolution will result in an unnecessary increase in the pre-driver and routing resources.

$$impedance\ resolution\ error, \% = \frac{RZQ\ impedance}{2 \times min\ impedance\ of\ LSB} \times 100 \quad (3.1)$$

$$minimum\ driver\ impedance, \Omega = \frac{max\ impedance\ of\ one\ full\ driver\ slice}{total\ number\ of\ driver\ slices} \quad (3.2)$$

With the total number of driver slices defined, each full-sized slice was designed to be close to 800 Ω at typical corner as listed in Table 3.2. As a result, the driver in this work has resolution error of 4.75 % and minimum driver impedance of 31.75 Ω from the calculations in Eq. 3.3 and Eq. 3.4. In regard to the resolution error, the 5.25 % tolerance margin is reserved for other sources of error such as the variation of package impedance across different I/O pins. In regard to the driver impedance, the 6.6 % margin is reserved for end-of-life (EOL) degradation.

Table 3.2: Impedance of driver slices at different PVT corners

Parameter	Typical	Fast	Slow
Impedance of LSB slice, Ω (% from typical)	3168 (+0.0 %)	2524 (-20.3 %)	4032 (+27.3 %)
Impedance of full-sized slice, Ω (% from typical)	792 (+0.0 %)	631 (-20.3 %)	1008 (+27.3 %)

$$impedance\ resolution\ error, \% = \frac{240}{2 \times (4 \times 631)} \times 100 = 4.75 \quad (3.3)$$

$$minimum\ driver\ impedance, \Omega = \frac{1008}{31 + 0.5 + 0.25} = 31.75 \quad (3.4)$$