University of Udine

DEPARTMENT OF ELECTRICAL MANAGEMENT AND MECHANICAL ENGINEERING
Ph.D. Programme in Industrial and Information Engineering, XXVI Cycle

Ph.D. Dissertation

Analysis and Design of High Speed Serial Interfaces for Automotive Applications

Candidate
Andrea Cristofoli

Advisors
Prof. Pierpaolo Palestri
Dr. Nicola Da Dalt
Prof. Luca Selmi

External Reviewers
Dr. Franz Dielacher
Prof. Andrea Mazzanti

Udine, 4 April 2014

Abstract

The demand for an enriched end-user experience and increased performance in next generation electronic applications is never ending, and it is a common trend for a wide spectrum of applications owing to different markets, like computing, mobile communication and automotive. For this reason High Speed Serial Interface have become widespread components for nowadays electronics with a constant demand for power reduction and data rate increase.

In the frame of gigabit serial systems, the work discussed in this thesis develops in two directions: on one hand, the aim is to support the continuous data rate increase with the development of novel link modeling approaches that will be employed for system level evaluation and as support in the design and characterization phases. On the other hand, the design considerations and challenges in the implementation of the transmitter, one of the most delicate blocks for the signal integrity performance of the link, are central.

The first part of the activity regarding link performance predictions lead to the development of an enhanced statistical simulation approach, capable to account for the transmitter waveform shape in the ISI analysis, a characteristic that is missed by the available state-of-the-art simulation approaches. The proposed approach has been extensively tested by comparison with traditional simulation approaches (Spice-like simulators) and validated against experimental characterization of a test system, with satisfactory results.

The second part of the activity consists in the design of a high speed transmitter in a deeply scaled CMOS technology, spanning from the concept of the circuit, its implementation and characterization. Targets of the design are to achieve a data rate of 5 Gb/s with a minimum voltage swing of 800 mV, thus doubling the data rate of the current transmitter implementation, and reduce the power dissipation adopting a voltage mode architecture. The experimental characterization of the fabricated lot draws a twofold picture, with some of the performance figures showing a very good qualitative and quantitative agreement with pre-silicon simulations, and others revealing a poor performance level, especially for the eye diagram. Investigation of the root causes by the analysis of the physical silicon design, of the bonding scheme of the prototypes and of the pre-silicon simulations is reported. Guidelines for the redesign of the circuit are also given.

Sommario

Nel panorama delle applicazioni elettroniche il miglioramento delle performance di un prodotto da una generazione alla successiva ha lo scopo di offrire all'utilizzatore finale nuove funzioni e migliorare quelle esistenti. Negli ultimi anni grazie al costante avanzamento della tecnologia integrata, si è assistito ad un enorme sviluppo della capacità computazionale dei dispositivi in tutti i segmenti di mercato, quali ad esempio l'information technology, la comunicazione mobile e l'automotive. La conseguente necessità di mettere in comunicazione dispostivi diversi all'interno della stessa applicazione e di traferire grosse quantità di dati ha provocato una capillare diffusione delle interfacce seriali ad alta velocità, o High Speed Serial Interfaces (HSSIs). La necessità di ridurre il consumo di potenza e aumentare il bit rate per questo tipo di applicazioni è diventata dunque un ambito di ricerca di estremo interesse.

Il lavoro discusso in questa tesi si colloca nell'ambito della trasmissione di dati seriali a bit rate superiori ad 1Gb/s e si sviluppa in due direzioni: da un lato, a sostegno del continuo aumento del bit rate nelle nuove generazioni di interfacce, è stato affrontato lo sviluppo di nuovi approcci di modellazione del sistema, che possano essere impiegati nella valutazione delle prestazioni dell'interfaccia e a supporto delle fasi di progettazione e di caratterizzazione. Dall'altro lato, si è focalizzata l'attenzione sulle sfide e sulle problematiche inerenti il progetto di uno dei blocchi più delicati per le prestazioni del sistema, il trasmettitore.

La prima parte della tesi ha come oggetto lo sviluppo di un approccio di simulazione statistico innovativo, in grado di includere nell'analisi degli effetti dell'interferenza di intersimbolo anche la forma d'onda prodotta all'uscita del trasmettitore, una caratteristica che non è presente in altri approcci di simulazione proposti in letteratura. La tecnica proposta è ampiamente testata mediante il confronto con approcci di simulazione tradizionali (di tipo Spice) e mediante il confronto con la caratterizzazione sperimentale di un sistema di test, con risultati pienamente soddisfacenti.

La seconda parte dell'attività riguarda il progetto di un trasmettitore integrato high speed in tecnologia CMOS a 40 nm e si estende dallo studio di fattibilità del circuito fino alla sua realizzazione e caratterizzazione. Gli obiettivi riguardano il raggiungimento di un bit rate pari a 5 Gb/s, raddoppiando così il bit rate dell'attuale implementazione, e di una tensione differenziale di uscita minima di 800 mV (picco-picco) riducendo allo stesso tempo la potenza dissipata mediante l'adozione di una architettura Voltage Mode. I risultati sperimentali ottenuti dal primo lotto fabbricato non delineano un quadro univoco: alcune performance mostrano un ottimo accordo qualitativo e quantitativo con le simulazioni pre-fabbricazione, mentre prestazioni non soddisfacenti sono state ottenute in particolare per il diagramma ad occhio. Grazie all'analisi del layout del prototipo, del bonding tra silicio e package e delle simulazioni pre-fabbricazione è stato possibile risalire ai fattori responsabili del degrado delle prestazioni rispetto alla previsioni pre-fabbricazione, permettendo inoltre di delineare le linee guida da seguire nella futura progettazione di un nuovo prototipo.

Contents

Chapter 1

Introduction

1.1 High Speed Serial Interfaces

Technology advancement of the semiconductor industry over the last decades has repeatedly shown to follow Moore's Law in that the number of transistors in an integrated circuit doubles roughly every two years. This continuous shrinkage of the feature size enabled higher operation speed, logic density, integration, and lower power consumption per logic function. As a direct consequence of this scaling process, the number of functionalities crammed into processing units has became enormous and has generated a corresponding increase in the amount of data exchanged between chips to guarantee the increase in the overall system performance.

The two conventional methods to increase chip communication bandwidth consists in increasing the number of I/O channels or in raising the data rate per channel. However, cost containment imposes an upper limit to the I/O pin count and to the number of board traces and board material, preventing a significant increase in the I/O channel number. As a consequence, the increasing bandwidth demands must be satisfied pushing the data rate towards higher limits. But this presents a remarkable challenge, considering that, while usually high-performance I/O circuitry can leverage the technology improvements that enable increased core performance, unfortunately the bandwidth of the electrical channels used for the communication does not scale in the same manner. To further raise the bar, pushing to higher power consumption to overcome channel limited bandwidth is not an option, as containing the power budget is becoming an urgent need throughout the whole electronics panorama.

If in addition to the elements thus mentioned, we also consider how in our everyday life we are experiencing the pervasive presence of communication and multimedia equipment, it appears clear why high-speed serial connectivity has risen to the rank of critical enabling technology in so many markets. High Speed Serial Interfaces (HSSIs), in fact, find their place in a large number of electronic applications, used for [1]:

- Chip-to-chip, board-to-board, and system-to-system links;

- Data communication and telecommunication networks, e.g. Gigabit Ethernet;

- Component interfaces internal to optical network equipment, e.g. Optical Internetworking Forum (OIF) standards like Interlaken [2];

- Computing I/Os, e.g. Peripheral Component Interconnect Express (PCIe), Hypertransport [3,4];

- Storage area networking, e.g. Serial Advanced Technology Attachment (SATA), Fibre Channel;

- Wireless networking, linking the radio equipment control and the radio equipment in wireless base stations, e.g. Common Public Radio Interface (CPRI) [5];

- High-performance embedded processing, e.g.Serial Rapid IO (SRIO) [6].

To give an idea of the typical data rates we can distinguish today's high-speed link standards minding the maturity of the market they address [7, 8]. Therefore, considering relatively established markets, typical data rates range from 5 to 6 Gb/s, as in the case of PCIe Gen2 (5 Gb/s), HyperTransport and SATA III/SAS II (6 Gb/s). The leading-edge and next-generation standards push data rates to the $8 - 11$Gb/s range, as in the case of IEEE 10G Ethernet, IEEE 40G/100G Ethernet (802.3ba, 4×/10× 10.3125 Gb/s), PCIe Gen3 (8 Gb/s), and Fibre Channel 8× (8.5 Gb/s). Finally, new standards for emerging market segments go even beyond these figures, as in the case of 16G Fibre Channel (14.1 Gb/s) and SATA IV/SAS III (12 Gb/s).

1.2 Building Blocks of a Serial Link

Figure 1.1 shows the main components of a typical HSSI [9, 10]. One of the main purposes of the HSSIs is to limit the number of high-speed I/O pins in chip packages and relax the Printed Circuit Board (PCB) wiring constraints in terms of copper traces, connector pin count, and so on. For this reason, the first building block we encounter at the transmitter side is a *serializer* which, as the name suggests, serializes an input bus of parallel data into a unique stream. The transmitter task is to generate an accurate voltage swing on the channel while also maintaining proper output impedance in order to attenuate any channel-induced reflections. Either current- or voltage-mode drivers are employed as output stages [11, 12]. The generation of the timing reference for the serializer and transmitter circuits is performed by means of a frequency synthesis Phase Locked Loop (PLL): it generates the high-frequency clock, usually at a frequency equal to the desired data rate, by multiplying the low frequency reference clock of the elaboration digital core.

At the receiver side, the incoming signal is sampled and compared to a threshold to properly discriminate bits '0' and '1', regenerated to CMOS values, and finally deserialized. At the receiver, the clock providing the sampling instants is aligned to the incoming data stream by a timing recovery system which usually incorporates a PLL and some additional circuits needed to synchronize the receiver with the incoming data stream.

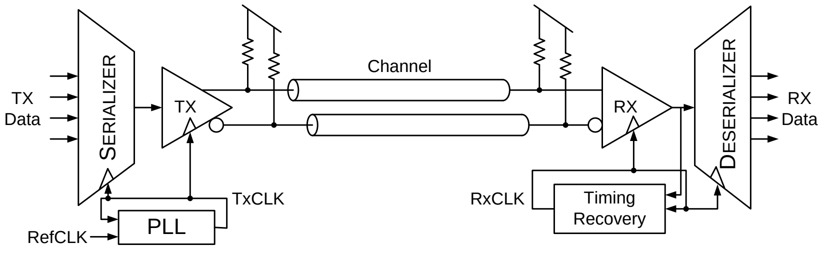


Figure 1.1: Block diagram of a typical high speed serial link [9].

The timing blocks of the link, at both the transmitter and receiver side, are critical for high-speed operation of the link since they provide accurate spacing of transmitted data symbols and sampling of the signal waveforms at the receiver.

1.3 Interference Sources in Serial Links

The typical structure of a serial link is represented in Figure 1.2, which shows the cross section of a backplane link. In such a system, a number of components constitute the transmission medium: as can be seen in the figure, in addition to the backplane channel, vias, connectors, and board traces are present. As each element has non-ideal characteristics, it is not transparent to the signal propagation thus interferes with the transmitted data.

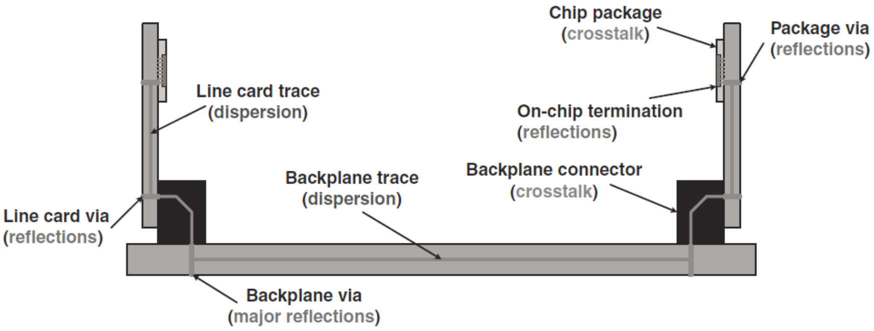


Figure 1.2: The cross section of a backplane link as an example of electrical channel [9].

A first rough classification of the interference sources in a link can be made distinguishing among interferences due to the transmission medium and noise.

1.3.1 Interferences due to transmission medium

This subset of interferences is determined by the non-ideal electrical properties of the medium. They are:

Channel insertion loss. When traveling through a transmission medium, electrical signals experience an attenuation that increases at high-frequency. Main causes are the *skin effect* and the *dielectric loss*. In the first case, the effective cross section of the wire, or trace, decreases at high frequencies because the majority of the current tends to flow near the conductor surface. This results in a resistive loss term that is proportional to the square root of the frequency [9, 13]. In the second case, at high frequencies, energy is absorbed from the signal trace and transferred into heat due to the rotation of the dielectric atoms in an alternating electric field.This results in the dielectric loss term increasing proportional to the signal frequency [9, 13].

Return Loss. Return loss is a measure of signal energy loss due to reflection. When impedance discontinuity exists, part of the signal is reflected back at the impedance discontinuity point, thus reducing the signal energy being transmitted.

Other Loss Factors. Other causes of losses are *mode conversion* and *radiation*. The first effect takes place when in a differential signaling system the balance between the two conductors is not perfect and part of the differential signal is converted to common mode, resulting in an energy loss of the desired differential signal [14]. The second effect is due to electromagnetic waves radiating energy into the air, e.g. in the case of the formation of standing waves in the line [13].

Regardless of the mechanism, loss in the channel is typically measured in terms of decibel at the Nyquist frequency of the data stream, i.e. $F_s/2$, where F_s is the nominal data rate.

1.3.2 Noise Disturbances

They consist in *crosstalk* from other channels and noise due to random signal fluctuations.

Crosstalk occurs due to both capacitive and mutual inductive coupling [15] between neighboring signal lines. It takes place especially at connectors and chip packages, where spacing between signal lines is smaller, and shielding is less effective. Crosstalk is classified either as Near-End Crosstalk (NEXT), where aggressor and victim are on the same chip, or Far-End Crosstalk (FEXT), where the aggressor energy couples and propagates along the channel to a victim on another chip. NEXT is commonly the most detrimental crosstalk, as energy from a strong transmitter can couple onto a weak received signal on the same chip, which has been attenuated by the band-limited channel.

Random signal fluctuations are due to the inherent thermal and shot noise of the passive and active system components. Random fluctuation are particularly important when dealing with timing of signals, as they can cause deviation of the signal characteristics, usually the edge crossings, with respect their ideal value. We speak in this case of *jitter*, that will be extensively treated in Section 2.2.

1.4 Equalization

Frequency dependent channel loss can reach magnitudes sufficient to make simple Non-Return to Zero (NRZ) binary signaling undetectable. Thus, in order to continue scaling electrical link data rates, systems that compensate for frequency-dependent loss, i.e. that *equalize* the channel response have been developed [16].

As reported by the simplified plot in Figure 1.3(top), the idea behind equalization is to insert into the serial link an "equalizer" such that its transfer function compensates the dispersion of the channel. By multiplying the low-pass transfer function of the transmission medium with that of the equalizer, as the whole system is linear, the goal is to have a resultant transfer function that is relatively "flat" up to the required frequency.

Figure 1.3(bottom) shows how equalization acts on the signal: the signal launched by the transmitter (left) is attenuated by the transmission medium (center). When passing through the equalizer, the original signal is restored (right).

Since linearity is assumed, it is important to note that signal conditioning can be applied before or after the interconnect. In this example, the equalizer is placed at the far end, at the receiver. Similarly, operating at the transmitter side the signal is pre-distorted so that after it goes through the interconnect, the resulting signal is much easier to be recover at the receiver.

Equalization can be implemented with digital techniques, e.g. with Finite Impulse Response (FIR) or Infinite Impulse Response (IIR) filters, or analog techniques, such as Continuous Time Linear Equalization (CTLE). Furthermore, equalization techniques could be either linear or adaptive (non-linear), such as Decision Feedback Equalization (DFE). In the

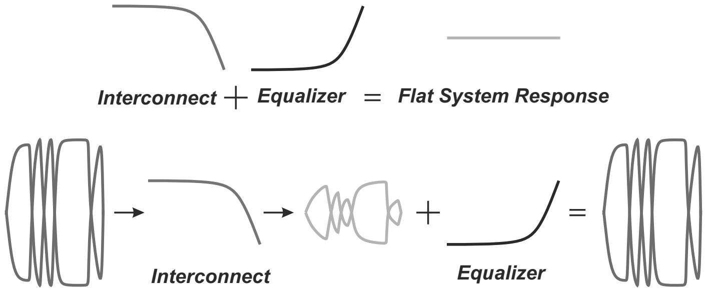


Figure 1.3: Simplified scheme showing the idea behind equalization (top), and effects of equalization at signal level (bottom) [1,7].

following a brief summary of nowadays most diffused transmitter and receiver equalization techniques will be given.

1.4.1 Transmit Equalization

Transmit equalization is implemented by preconditioning the signal before being applied to the channel: this type of signal conditioning is called *emphasis*. It can operate in two ways: the signal can be distorted such that its high frequency contents are amplified or, at the opposite, the signal low-frequency contents are reduced [1,7,17]. In the former case we denote it as *pre-emphasis*, in the latter as *de-emphasis*.

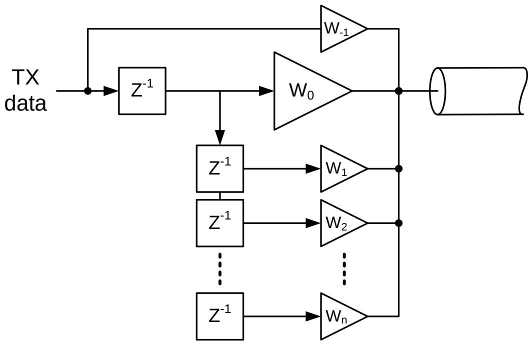


Figure 1.4: Conceptual scheme of transmitter equalization implementation based on registers holding prior and upcoming bits [9].

Emphasis is relatively simple to implement and does not require large additional power: for this reason it is the most common technique used in HSSI [18]. The common implementation is based on the FIR filter approach: as all serial data is available at the transmitter side, 1-bit spaced versions of the transmitted data can be easily created by means of registers that hold prior and upcoming serial data bits (see Figure 1.4), and then summed with proper weights to generate the appropriate voltage level of the current bit.

Figure 1.5 shows an example of de-emphasized waveform [3], obtained with a simple FIR filter with only one bit delay (so called one *tap*). Each logic value output by the transmitter can be represented with two different voltage levels: all transition bits corresponding to a change in the logical output state (from ′0′ to ′1′ or ′1′ to ′0′) are driven to the full swing amplitude ($V_{TX-DIFF-PP}$ in Figure 1.5). On the contrary, when multiple bits of the same logic value (′1′ or ′0′) are output in succession, they are driven to the de-emphasized amplitude ($V_{TX-DE-EMPH-PP}$ in Figure 1.5). The de-emphasis ratio ($DE - RATIO$) is thus defined

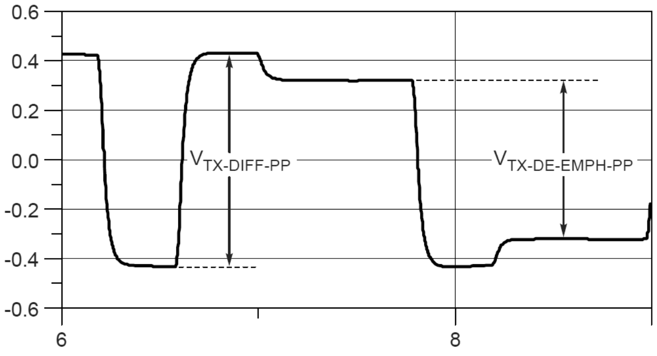


Figure 1.5: Transmitter waveform in presence of de-emphasis.

as [3]:

$$DE - RATIO = -20\log_{10}\left(\frac{V_{TX-DIFF-PP}}{V_{TX-DE-EMPH-PP}}\right) \tag{1.1}$$

In serial link designs, equalization is most often located at the transmitter side: the main reason is that here the input of the equalizer circuit is a binary data pattern instead of an analog voltage. Therefore the equalizer is simple to implement using simple digital and analog techniques, e.g. as Direct FIR [19] or Segmented DAC [20] transmitters.

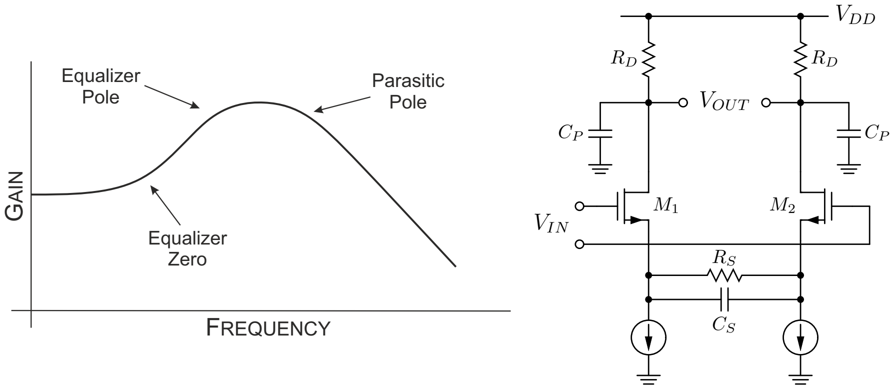
1.4.2 Receiver Equalization

Common equalization techniques at the receiver side fall under three categories: CTLE, RX FIR and DFE [17].

CTLE

As the name implies, Continuous Time Linear Equalization is a linear technique and operates continuously in time. Therefore can be considered an analog technique in nature [21]. Similar to transmit pre-emphasis, CTLE addresses pre-cursor and post-cursor ISI, but in continuous time instead of being limited to a pre-defined number of transmitter taps. Figure 1.6(a) shows an example of a first-order CTLE transfer function: this technique aims to compensate the poles in the low-pass channel transfer function by inserting a zero value in correspondence of the frequencies close to the link data rate.

An implementation example of CTLE with a continuous-time amplifier is shown in Figure 1.6(b) [22,23]. The RC-degeneration in the source-couple pair creates a high-pass filter transfer function if the zero frequency is designed to be much lower of the dominant pole [16]. While this implementation is a simple and low-area solution, one issue is that the amplifier has to supply gain at frequencies close to the full signal data rate. This gain-bandwidth requirement potentially limits the maximum data rate. Multiple equalizer stages implementations can be devised to increase the order of the equalizer and thus increase the maximum boost achieved in a given frequency interval. However, tuning the parasitic poles and their locations of such multiple stage design across PVT variations can be hard [21]. For this reason CTLE compensation is usually limited to the 1st order.



(a) (b)

Figure 1.6: (a) CTLE high-pass transfer function [8]. (b) Continuous-time amplifier as an active implementation of CTLE [23].

RX FIR

Analogously to transmit pre-emphasis, the RX FIR equalization scheme employs a FIR filter to compensate for channel losses. RX FIR equalization can be implemented as either discrete or analog [16, 17].

The discrete RX FIR conceptual block diagram is shown in Figure 1.7(a). It adopts a linear digital filter similar to the one used for transmit pre-emphasis. Anyway, as the input of the filter is the analog output of the channel, in this case a Sample and Hold Amplifier (SHA) and an Analog to Digital Converter (ADC) are required to interface the channel output to the filter. The analog to digital conversion is particularly critical, as achieving a high resolution of the taps (e.g. in the order of 10 bits [21]) and at the same time operate at the full data rate of the interface requires large power and area overhead [16]. The high-speed ADC implementation thus poses serious limitations to the adoption of discrete RX FIR.

An analog RX FIR equalizer obviates the need for a high-speed ADC. It is therefore attractive for high-speed operation with potentially lower power consumption as just the relatively simple sample and hold circuit is required. A conceptual block diagram of an analog FIR equalizer is shown in Figure 1.7(b). As opposed to the digital delays used in the discrete FIR, an analog delay chain is required. As the overall structure is analog, non-idealities limiting the overall performance of the circuit come from errors in the sampled voltage due to sampling jitter and charge leakage, non-linearity of the equalizer taps and summing circuits, and offset currents due by device mismatch. If not handled properly, all these issues can negate the benefits versus the digital implementation [21].

A common problem faced by linear receiver-side equalization, thus both CTLE and RX FIR, is that high-frequency noise content and crosstalk are amplified along with the incoming signal. Nonetheless, one of the major advantage of receiver-side equalization is that the filter tap coefficients can be adaptively tuned to the specific channel, which is not possible with transmit-side equalization unless a back-channel is implemented [9].

DFE

The third equalization topology commonly implemented at receiver side is Decision Feedback Equalization (DFE). The block diagram of the DFE is shown in Figure 1.8. The idea behind

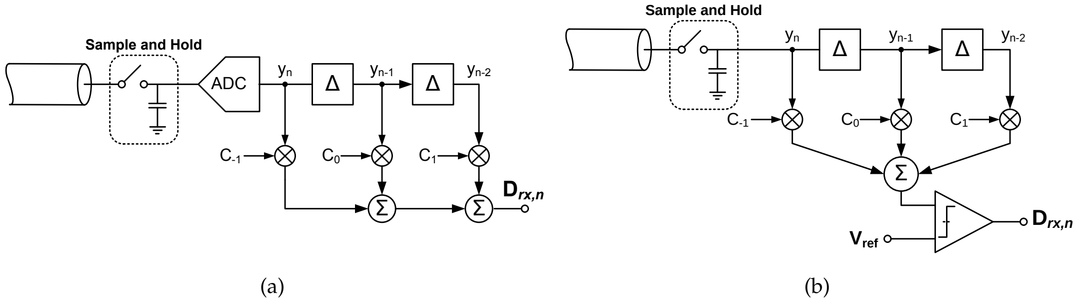


Figure 1.7: RX FIR implementations: digital (a) and analog (b) [16].

it is to cancel Intersymbol Interference (ISI) directly from the incoming signal using the last resolved data to control the polarity of the equalization taps: thus it is a non-linear technique. Working on quantized input values, this technique does not operates a high frequency boost on the analog signal and therefore has the advantage of not amplifying noise and crosstalk, on the contrary of what happens with linear equalizers. Due to the feedback structure, DFE addresses only post-cursor ISI, i.e. ISI caused by the previous bits, and leaves the pre-cursor ISI uncompensated. As a consequence, a separated feed-forward equalizer, e.g. CTLE as in [24], is still required to accommodate the pre-cursor ISI [1,16].

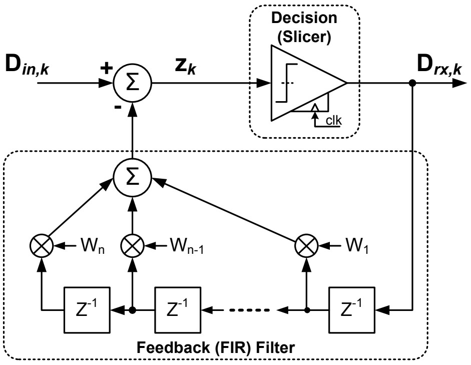


Figure 1.8: Receiver equalization with DFE [9].

In addition to this, the issue of error propagation arises. The error propagation phenomenon happens if the noise is large enough to determine a wrong decision of the current data. At this point, the bit is fed back through the ISI cancellation filter and determines an erroneous coefficient computation for the present data sample. Therefore, the error on a single bit capture affects few consecutive bits until it propagates out of the filter and thus correct samples are obtained again [16].

Another major challenge in DFE implementation arises from the need for the first tap feedback to be ready before the next bit comes. In other words, the computation of the first tap coefficient must be done in one bit period. Therefore this critical timing path needs to be highly optimized [9] or different filter architectures are needed, i.e. decision look-ahead schemes [16], when the first tap loop delay can not be reduced below 1 Unit Interval (UI) due to the very high data rate.

1.5 High Speed Links in the Automotive Environment

In the last decade, the massive introduction of electronic devices and products in every aspect of our life has been driven by advancements in the integrated electronics. Three principles can be cited to understand this impressive development: transparency, i.e. the user can be helped unobtrusively with electronics, pervasiveness, i.e. any common object can host electronics thanks to integration, and intelligent environments, i.e. environments that are sensitive and responsive to the presence of people thanks to embedded sensors and systems.

Vehicles have not been immune to this trend: in fact, automotive innovation today is almost entirely driven by semiconductor actuation, control, or monitoring, and it involves every different domain inside a vehicle (Powertrain, Driver Comfort, Safety, Infotainment, Chassis, Driver Assistance). The list of electronically assisted functions and systems that may be found on a today's upscale automobile is very long. Figure 1.9 tries to mention some of them and gives an idea of the deep penetration of electronics into the automotive environment.

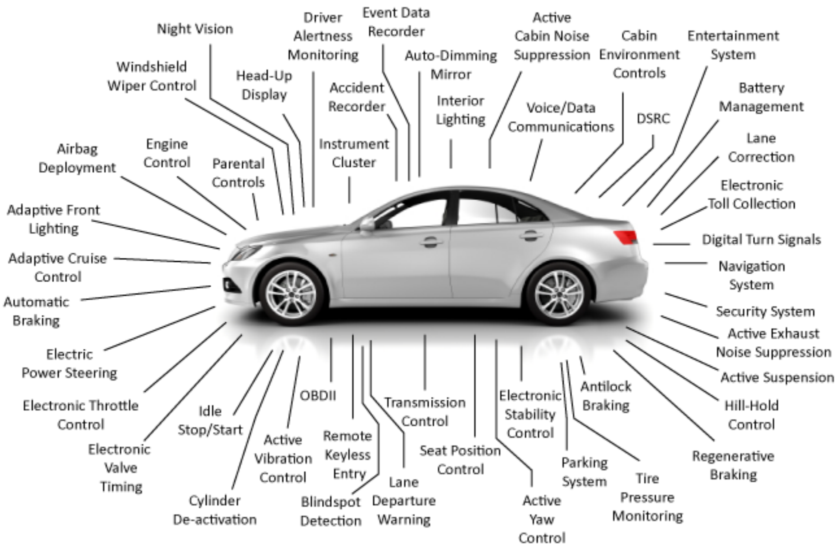
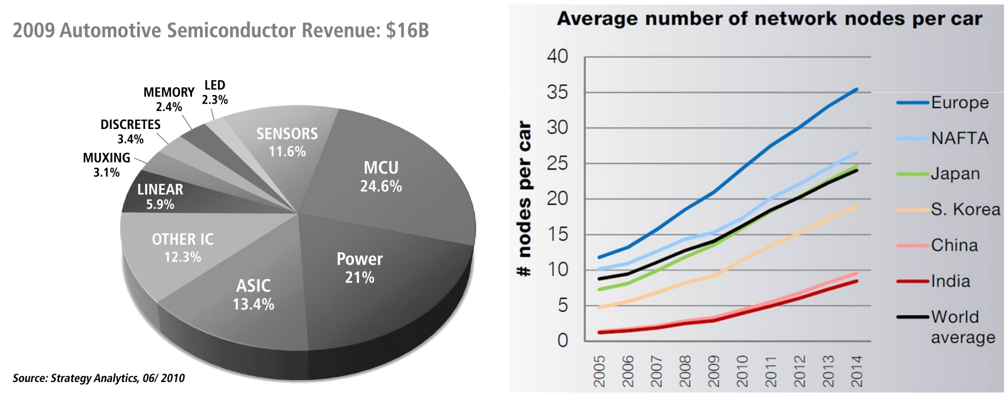


Figure 1.9: Some of the functions and systems assisted by electronics available in nowadays vehicles.

This picture is not likely to change in the future [25–27]. In fact, rising fuel prices and environmental concerns aiming at the reduction of carbon dioxide emission are pushing the adoption of more and more sophisticated engine and powertrain control schemes. The same applies to electric vehicles too, as power and battery management is a key aspect to make them suitable for modern human mobility needs. Secondly, improving the passive and active safety is a constant target as the available technologies make advancements possible (e.g. the so-called Advanced Driver Assistance Systems (ADAS)). Thirdly, automobile producers will try to enhance the on-board user experience increasing the connectivity solutions (GPS, Mobile Internet Connectivity) and body comfort systems. The need to handle all these various functions demands solid computations capabilities in a number of different locations in the vehicle, thus asking for an increase of the number of Microcontroller Units (MCUs) used on-board. In fact it is estimated that today's well-equipped upscale automobile generally relies on more than 80 electronic control units [26], and this number is constantly increasing. This is also confirmed by the analysis of the yearly semiconductor revenue referred to the automotive segment [28]. Distinguishing the revenue depending on the type of device, as shown in Figure 1.10(a), it is possible to see that approximately 1/4 of the automotive devices sold worldwide are MCUs. The same growth prediction is devised in [29], where the number of nodes constituting the on-board network in new generation of vehicle is analyzed over the last 10 years. As can be seen in Figure 1.10(b), this number is steadily increasing world wide, testifying from one side that the need to include advance computation capability in order to handle the various functions in the vehicle is a strong trend, and on the other side that the connection of all these computational nodes is becoming a challenging problem.



(a)

(b)

Figure 1.10: (a) In 2009, MCUs were the market-share leaders among the various semiconductor device types used in vehicles [28]. (b) Each MCU is at the same time a node of the vehicle network: over the last 10 years the increase in the number of network nodes, thus in the number of MCUs on board, has been a common trend worldwide [29].

The vehicle on-board networking structure is realized nowadays based on bus standards like Controller Area Network (CAN), Local Interconnect Network (LIN) and FlexRay [30], with data rates spanning from a few tens of kb/s (in the case of LIN) to the 10 Mb/s per lane of FlexRay (see Figure 1.11). The implementation of electronic applications involving the transmission of audio or video streams, either in the field of the active and passive safety (e.g. the already mentioned ADAS) or in that of the entertainment of the passengers, will demand for the transfer of a high quantity of data. It will thus require the use of connectivity solutions capable of reaching higher transfer rates. In fact, in the near future the adoption of Ethernet as a standard for networking in automotive applications [27] goes exactly in the direction of allowing the transfer of high volumes of data. Nevertheless the trend revealed by Figure 1.11 is quite likely not to stop at Ethernet. We see in fact that in terms of data rates, serial application in the automotive are trailing the path opened by the telecommunication and consumer markets, with a delay time frame of approximately 10-15 years [31]. We can therefore say that HSSI implementations in the automotive allowing for data rates of few Gb/s, are state of the art nowadays. We can also expect that vehicle electronic applications hosting HSSIs will become more and more common in the next years.

Nevertheless, a direct technology transfer from the telecommunication and consumer market is not possible, due to the serious challenges that the automotive environment poses to the implementation of electronic applications in general, and of HSSIs in particular. The requirements of automotive electronics are much more stringent and demanding with respect to the consumer segment. In Table 1.1 the principal differences are highlighted. The more im-

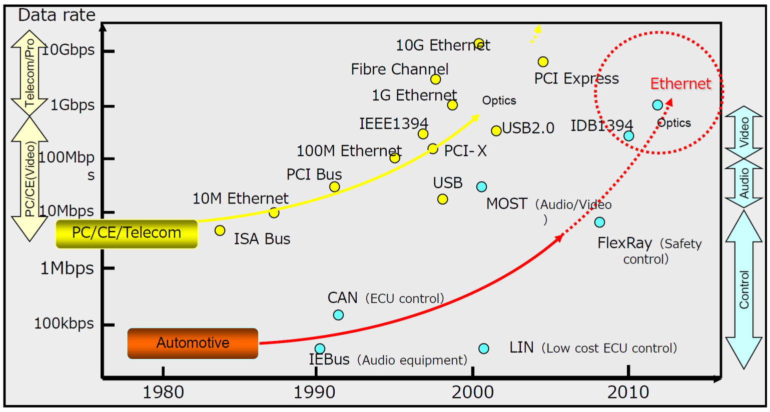


Figure 1.11: The data rate improvement of connection solutions for the automotive market is following the same trend as in the telecommunication and consumer markets, but 10-15 year later [31]. Serial interfaces allowing data transmission in the order of Gb/s are nowadays the state of the art in the field.

pressive numbers are in the broader temperature range, spanning from −40 °C up to 175 °C, in the expected operation time, which can be as high as 25 years and the ESD tolerance, which can be even double than the value required in consumer applications. It is clear then that implementation of HSSIs to be employed inside vehicles presents additional challenges to the design of such systems.

Table 1.1: Requirements on automotive electronics [32]

Parameter	Consumer	Automotive
Temperature	0 °C → 40 °C	−40 °C → 85/175°C
Voltage	3.3 V	> 80 V
Operation Time	1-3 years	up to 25 years
Humidity	Low	0% up to 100%
Tolerated Field Failure Rate	<1000 ppm	Target: zero failure
ESD[a]	4 − 8kV	8 − 15kV

[a] Machine Model (MM)

1.6 Motivation of the work and thesis organization

From this brief introduction, it clearly appears that efficiently addressing the requirements in system bandwidth of nowadays applications achieving higher data rates and greater integration is becoming a challenge. This challenge includes targeting lower bit error rates and ensuring signal and power integrity while maintaining power efficiency and data rates, and optimizing design productivity. In this framework, the work discussed in this thesis develops in two directions. On one hand, with the aim to support the continuous data rate increase, the activity with major impact (as demonstrated by a publication in a peer-reviewed journal, see Candidate's Bio and Publications at page 115) has been the development of novel link modeling approaches to be employed for system level evaluation, design, and characterization. The major technical impact of this thesis stems from this first part, . On the other side, the design implementation of a high speed transmitter, one of the most delicate blocks for the signal integrity performance of the link, are carried out, with the aim of improving the current implementation in terms of power dissipation and supported data rate.

After this brief introduction, the thesis is organized in chapters, each presenting one of the activities carried out by the candidate.

- Chapter 2 deepens in the modeling of ISI and jitter in serial gigabit links. The approaches proposed in available literature to evaluate the impact of both effects on the link performance are described. In the case of ISI, the analysis highlights the advantages of the statistical approach over the worst-case one in terms of the insight offered to the designer and computation time. In the case of jitter, the picture is less clear, as the two most recent techniques available address modeling requirements of different link architectures and are quite complex to implement into a statistical approach.

- Chapter 3 describes the statistical approach proposed in this thesis to model ISI and jitter in HSSI and its implementation into a MATLAB program. This approach allows for an accurate modeling of the transmitter pulse shape, a feature that is missing in other statistical techniques due to the non-trivial problem of dealing with transmitter non-linearity. Validation by means of comparison with other simulation approaches follows. Its advantage in dramatically reducing the simulation time over traditional Spice-like techniques is pointed out, whereas a critical discussion of the limitations of the proposed simulation procedure in handling impedance discontinuities in the link is provided.

- Chapter 4 focuses on the verification of the proposed technique by comparison with experimental data from a high-speed test system (1.25 and 2.5 Gb/s). The agreement in terms of the eye diagrams and bathtub opening for various channel lengths is discussed, with satisfactory results.

- Chapter 5, by means of the review of recent works on high speed transmitters, discusses the clear improvement in the power efficiency of transmitters when adopting the Voltage-Mode topologies in place of the traditional CML implementations, thanks to their potential for lower power consumption and high swing capability. The details of the Source Series Terminated architecture are described, as it appears to be an attractive topology due to its potential in combining the advantage in power reduction of a Voltage Mode (VM) driver to a design completely based on digital switching techniques that cope well with nowadays deca-nanometer technologies.

- Chapter 6 provides the implementation details of a high speed transmitter. Design choices to face challenging targets in terms of power dissipation reduction, achievable data rate and voltage swing are discussed. Experimental verification on fabricated prototypes is also reported. Measurements draw a two-fold picture. For some of the relevant transmitter design figures the performance observed on the prototypes match very well simulation expectations. However, poor performance are observed in terms of eye diagram aperture. This aspect is object of in-depth investigation with detailed analysis of the physical layout and additional circuit simulations to target the root cause. Explanation of poor eye performance will be finally given in the closing part of the Chapter, with also directions for future improvement of the design.

Chapter 2

Modeling of ISI and Jitter in High Speed Links

Modeling of HSSI has become an active field of research in the last fifteen years, in parallel with the capability of modern integrated circuit technology to sustain the demand for continuously increasing data rates. As the performance of the links was ramping up at each technology node and the design margins were becoming more and more difficult to maintain, new modeling techniques and tools to accurately predict link performance have become necessary for successful first silicon. Traditionally the link signaling performance simulation involved time-domain simulations using random data sequences as inputs. Unfortunately, this approach does not assure that the worst case transmission is covered, given the fact that only small subsets of data pattern can be tested within a reasonable simulation time. This picture worsens when increasing the data rate and, as a consequence, also the channel interference: the channel response settling time becomes very long, imposing to simulate longer data patterns. In this way the combinations of bits to be tested increases exponentially, thus making the time domain approach rapidly unfeasible.

In this chapter the HSSI modeling approaches available in literature to model the effects of ISI will be reviewed. The available solutions to the problem of including jitter in the simulations will be reviewed too. This chapter is useful for the reader to fully understand the approach proposed in this thesis to develop a novel model for serial links, that will be object of the next Chapter 3.

2.1 Intersymbol Interference Modeling

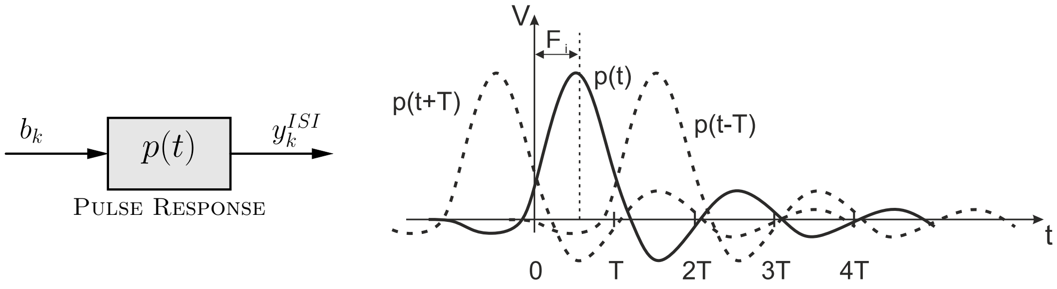
Many approaches have been proposed for the modeling of ISI. The most relevant are described in the following pages.

2.1.1 Peak Distortion Analysis

In a serial communication system (Figure 2.1(a)), we can define the pulse response $p(t)$ as the response of the channel to a pulse at its input of duration equal to the bit time T. Assuming that the physical channel is Linear and Time Invariant (LTI), the signal $y^{ISI}(t)$ at its output can be represented by the following expression (see Figure 2.1):

$$y^{ISI}\left((k+\Phi_i)T\right) = \sum_{j=-\infty}^{\infty} b_{k-j}\, p\left((j+\Phi_i)T\right) \tag{2.1}$$

where k and j are integers, Φ_i a number between 0 and 1 that can be considered as a phase inside T and b_k are the symbols voltage at the output of the transmitter[1]. The presence of ISI means that $p(t)$ extends outside the symbol period, causing portion of the signal of a bit to disturb the nearby bits.



(a) (b)

Figure 2.1: (a) Block diagram of the serial link model based on the pulse response $p(t)$ of the channel. (b) ISI is due to $p(t)$ extending outside T and disturbing neighboring bits.

The first approach proposed to efficiently model the effects of ISI was the Peak Distortion Analysis [33, 34]: it just observes, based on the principles of communication theory [35], that a worst-case eye diagram for a serial link could be extracted as a sum of all the interference sources. When the only source of interference is ISI, the worst-case for, e.g., a ′0′ is when the pattern of neighboring bits is such that the respective portion of responses adds over the ′0′ bit voltage level that one would have in absence of ISI. In this way we can identify a worst-case ′0′ ($s_0(\Phi_i T)$) and a worst-case ′1′ ($s_1(\Phi_i T)$) and define the area comprised between them as the worst-case eye $e(\Phi_i T)$:

$$s_0(\Phi_i T) < e(\Phi_i T) < s_1(\Phi_i T) \quad \text{for} \quad 0 \leq \Phi_i \leq 1. \tag{2.2}$$

In other words, $s_0(\Phi_i T)$ is the voltage level observed at the receiver side of the channel given that the transmitted bit is a ′0′ and the data pattern surrounding it is such that each unit portion of the tail of the channel response sums over the ′0′ bit voltage that one would have in absence of ISI. This concept can be mathematically expressed using the following:

$$\begin{aligned} s_1(\Phi_i T) &= p(\Phi_i T) - \sum_{\substack{k=-\infty \\ k\neq 0}}^{\infty} \left| p\left((\Phi_i - k)T\right) \right| \\ s_0(\Phi_i T) &= -p(\Phi_i T) + \sum_{\substack{k=-\infty \\ k\neq 0}}^{\infty} \left| p\left((\Phi_i - k)T\right) \right| \end{aligned} \tag{2.3}$$

Here we use the absolute value because if $p\left((\Phi_i - k)T\right) > 0$ the worst-case ISI contribution is given by a ′1′ bit, whereas if $y\left((\Phi_i - k)T\right) < 0$ it is given by a ′0′ bit (thus a -1 symbol). Note

[1] In this Chapter and in the following ones we will always refer to NRZ signaling systems with $b_i = +1$ for bit ′1′ and $b_i = -1$ for bit ′0′, i.e. the channel response to a ′1′ bit is $p(t)$ and to a ′0′ bit is $-p(t)$.

that using this principle it is also possible to find out the data pattern which is responsible of the worst-case eye, as shown as an example in Figure 2.2. Once the worst-case diagram has been calculated, an error-free transmission is possible if the data is sampled choosing a threshold voltage V_{th} and a sampling phase ϕ contained inside $e(\Phi_i T)$. This approach is quite simple, of straightforward application and it requires very limited computation time. On the other hand the insight given by the worst-case eye diagram is quite limited and can also lead to an over-design of the link. In fact, the probability to observe the worst-case pattern during data transmission is exponentially decreasing with the increase of the channel length and of the data rate, thus designing for very low probabilities may be highly inefficient in terms of silicon area and power.

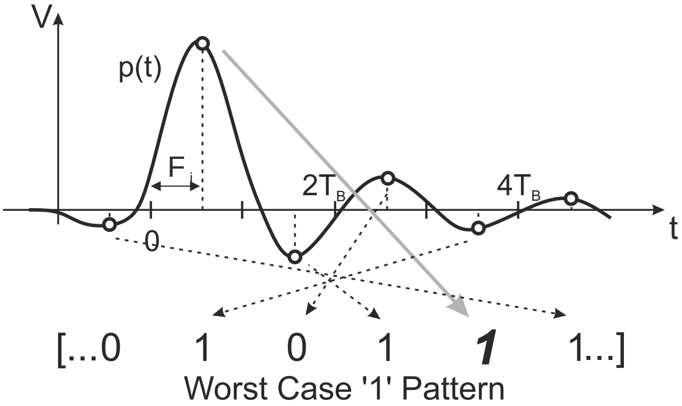


Figure 2.2: Example of how to extract the worst-case '1' pattern for a given Φ_i from the channel pulse response [34]. The sign of each sample $p((\Phi_i - k)T)$ determines the value the k^{th}-bit must assume for the worst-case. As an example, the sample $p((\Phi_i - 2)T)$ is positive, thus the worst-case condition occurs when 2 bits before the one of interest is transmitted a '0' bit: in this way the voltage $p((\Phi_i - 2)T)$ subtracts to $p(\Phi_i)$, causing a reduction of the sampling margin of the '1' bit.

2.1.2 Statistical Analysis based on the Single Bit Response

The statistical analysis based on the Single Bit Response (SBR) of the channel has been proposed to go beyond the worst-case approach by adding statistical information to the ISI eye diagram [33, 36, 37]. Final target is to calculate all possible eye contours that we can observe in an eye diagram depending on the data pattern, not just one contour representing the worst-case data transmission condition, and assign to each contour its respective probability to appear. To do this, we have to find a way to represent voltage quantities in the statistical domain. In particular, two questions must be answered:

1. How can we represent voltage values in the probability domain?

2. What are the operators in the probability domain associated to algebraic operations (sum, subtraction) of voltage values?

To answer the first question we note that a voltage v_1 can be represented by means of a Dirac's delta centered at v_1: in this way a Probability Distribution Function (PDF), function of v, is constructed such that all voltage values except v_1 have probability equal to zero. This representation allows to easily represent situations in which multiple voltage values are admissible at the same time, e.g. in case we want to represent the voltage values produced by different data patterns. In this case, the PDF consists of multiple Dirac's delta centered at each voltage value, with amplitudes equal to the probability to observe each value.

Once assessed a proper representation of voltages through PDFs, the operator that allows to reproduce algebraic operations on voltages is the *convolution*. In fact, if we apply the convolution between two Dirac's delta centered respectively at v_1 and v_2:

$$\begin{aligned} pdf_{v_1}(v) * pdf_{v_2}(v) &= \int_{-\infty}^{\infty} pdf_{v_1}(v-\mu) \cdot pdf_{v_2}(\mu) d\mu \\ &= \int_{-\infty}^{\infty} \delta(v - v_1 - \mu) \cdot \delta(\mu - v_2) d\mu \\ &= \delta(v - (v_1 + v_2)) = pdf_{v_1+v_2}(v). \end{aligned} \tag{2.4}$$

So, we obtain a Dirac's delta centered at $v_1 + v_2$, as also shown in the top graph of Figure 2.3. Thus the convolution operation is able to represent in the probability domain sum and subtractions between voltages.

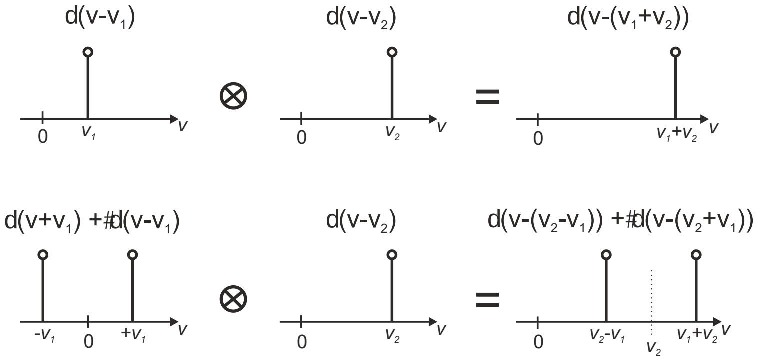


Figure 2.3: Convolution operation between PDFs representing voltage samples: the convolution of two Dirac's delta centered at v_1 and v_2 gives a Dirac's delta centered at $v_1 + v_2$ (top). The convolution of a PDF made of two Dirac's delta ad $\pm v_1$ and a single Dirac's delta at v_2 gives two Dirac's delta at $v_2 - v_1$ and $v_2 + v_1$ (bottom).

Let's now assume that we want to consider the data transmission of two bits, and we want to represent the ISI effect of the first transmitted bit, that could be either a '0' or a '1', on the following bit, that we assume to be a '1'. Under the assumption that each bit could be a '0' or a '1' with equal probability, and that the bit value is independent from the other bits in the data stream (i.e. random sequence with no coding), the voltage v_2 we would have for the '1' bit in absence of ISI will be disturbed by the tail of the first transmitted bit, as stated by eq. 2.1 and as shown in Figure 2.1. Assuming v_1 is the voltage given by $p(t)$ at T, if the bit is a 0, we have to subtract v_1 to v_2, while if the bit is a '1', we have to add v_1 to v_2. The two resulting voltages, $v_2 + v_1$ and $v_2 - v_1$, have a $1/2$ probability to be observed. We can represent this accumulation of ISI in the following way (see bottom graph of Figure 2.3):

1. construct a PDF for the '1' bit as a Dirac's delta $\delta(v - v_2)$;

2. construct a PDF that accounts for all possible ISI voltages due to the first transmitted bit as:
$$pdf = \frac{1}{2}\left[\delta(v - v_1) + \delta(v + v_1)\right] \tag{2.5}$$

3. convolve the two PDFs. The result of the convolution represents the possible voltages assumed by the '1' bit depending on the sign of the first transmitted bit.

This result is very important, because it provides us with all we need to compute ISI in the statistical domain.

The SBR approach [33, 36, 37] exploits the above results and determines the Bit Error Rate (BER) as a function of the data rate through the construction of a distribution plot that relates the BER to the sampling point, intended as the combination of threshold voltage and sampling time. The first step towards determining the BER is the calculation of a PDF of ISI for each time instant t of the eye diagram.

The PDF of ISI is calculated by convolving the individual ISI samples, determined from the channel pulse response $p(t)$ as in the worst-case analysis, using the following:

$$pdf_{k+1}(\mu,\Phi_i) = \begin{cases} \frac{\delta(\mu - p((\Phi_i - k)T)) + \delta(\mu + p((\Phi_i - k)T))}{2} \otimes pdf_k(\mu,\Phi_i) & \text{if } k \neq 0 \\ pdf_k(\mu,\Phi_i) & \text{if } k = 0 \end{cases} \tag{2.6}$$

where the calculation must be done from $k = -\infty$ to $k = \infty$. In eq. (2.6) we are:

1. associating to each ISI voltage sample $p((\Phi_i - k)T)$ its respective representation in the probability domain in the form of two Dirac delta functions centered at $\mu = \pm p((\Phi_i - k)T)$. The initial condition for the calculation is $pdf_{-\infty}(\mu,\Phi_i) = \delta(\mu)$, which follows from $p(-\infty) = 0$.

2. accumulating all ISI samples into a unique PDF by means of sequential convolution operations, identified by the symbol $\otimes$ in eq. (2.6).

Once determined $pdf_{k=+\infty}$, we have to couple it to the voltage levels of the '1' and '0' bit:

$$\begin{aligned} pdf(0)(\mu,\Phi_i) &= pdf_{k=+\infty}(\mu + p(t),\Phi_i) \\ pdf(1)(\mu,\Phi_i) &= pdf_{k=+\infty}(\mu - p(t),\Phi_i). \end{aligned} \tag{2.7}$$

which is just a shift of $pdf_{k=+\infty}$ at $p(t)$ (in the case of a '1' bit) or $-p(t)$ (in the case of a '0' bit).

We then combine $pdf(0)$ and $pdf(1)$ as:

$$pdf_{ISI}(\mu,\Phi_i) = \frac{1}{2}\left[pdf(0)(\mu,\Phi_i) + pdf(1)(\mu,\Phi_i)\right] \tag{2.8}$$

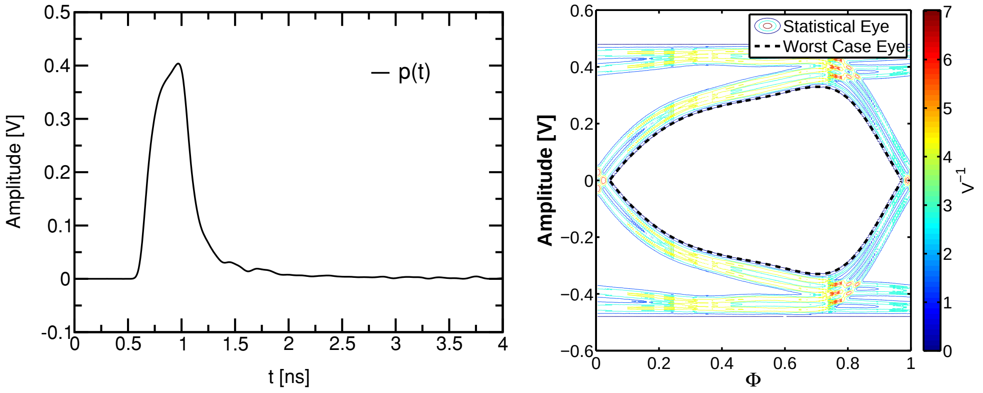
where we have assumed a 50% probability for both bit '0' and '1', and we contour plot it on the (μ, t) plane, as shown in Figure 2.4(b), obtaining a statistical eye diagram analogous to the one produced by an oscilloscope.

We can now calculate the BER, defined as the total probability of observing at the end of the channel a bit different from the transmitted one, i.e. in the case of a transmitted '1' we have $\mu < v$, where v is the sampling voltage. Using $pdf(0)(\mu,\Phi_i)$ and $pdf(1)(\mu,\Phi_i)$ calculated above, the BER is obtained using:

$$BER(v,t) = \int_{-\infty}^{v} \frac{pdf(1)(\mu,\Phi_i)}{2} d\mu + \int_{v}^{\infty} \frac{pdf(0)(\mu,\Phi_i)}{2} d\mu. \tag{2.9}$$

The first term of the equation determines the error probability of a '1' bit as the Cumulative Distribution Function (CDF) (thus a sum of probabilities) from $-\infty$ to v of $pdf(1)(\mu,\Phi_i)$. Similarly the second term of the equation determines the error probability of a '0' bit as the CDF from v to ∞ of $pdf(0)(\mu,\Phi_i)$.

The calculated $BER(v,t)$ can be contour plotted in the (v,t) plane: in this way the points on the (v,t) plane associated to a given BER level, e.g. 10^{-9} or 10^{-12}, can be visualized in



(a)

(b)

Figure 2.4: (a) Simple example of pulse response at 2.5 Gb/s of a channel with $S_{DD12} = -4.2$ dB at the Nyquist frequency. (b) Contour plot of the $pdf(v, \Phi)$ extracted with the SBR statistical ISI algorithm compared to the worst-case eye (black).

the same way of an eye obtained with the worst-case approach. This approach gives the designer the flexibility of accounting for reasonable design margins by choosing the target BER required by the application, thus saving in silicon area and power when some errors can be tolerated in the application, as most of the HSSI standards do nowadays.

2.1.3 Transitions-based Analysis

This approach uses as basis for ISI modeling the *transmit* segments, as opposed to transmit pulses employed in the SBR approach described in Section 2.1.2. As shown in Figure 2.5, transmit segments are defined as the transition from the left-side half-UI level to the right-side half-UI level. The transmitted data stream is divided into segments of length equal to one UI centered at the nominal data transition instant. The idea behind this modeling scheme is to compute the ISI contribution of the individual segments and then appropriately combine them to get the total effects of ISI at the channel output, represented by means of a statistical distribution [38,39].

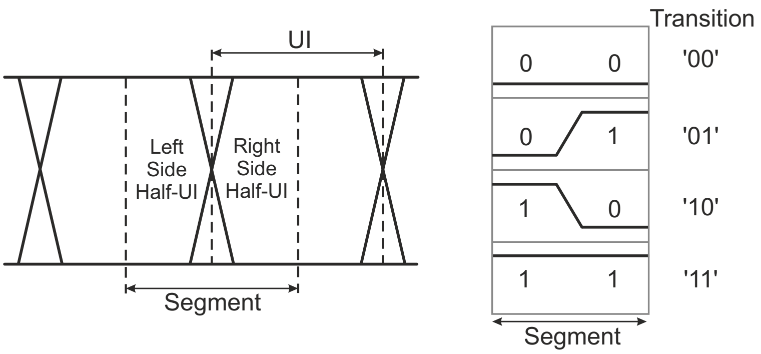


Figure 2.5: Segment definition with respect to the UI (left). The four transmit segment in the case of binary NRZ signaling [38] (right).

The first step to implement the transition-based approach is to tabulate all possible segments as combinations of the initial and final transition voltage values. For instance, in the very simple case of a binary NRZ signaling scheme the possible voltage values are only two, thus the possible combinations are four, as shown in Figure 2.5. In case of a larger number of voltage levels in the TX data stream, i.e. when Pulse Amplitude Modulation (PAM) or transmit equalization are employed, the number of segments to handle is larger. For each segment in this table, a waveform is constructed. The voltage value defined over the UI across the nominal transition instant is equal to the segment itself, while outside this time interval the waveform must not contribute, thus the voltage is equal to zero, as shown in Figure 2.6.

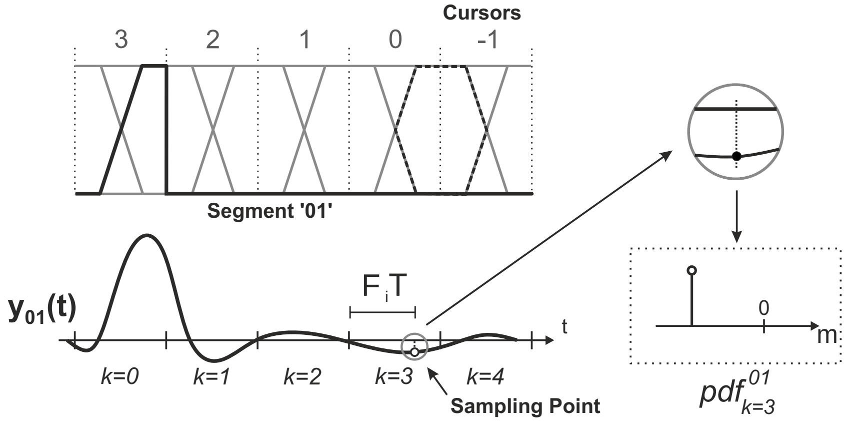


Figure 2.6: Construction of the transition PDF for the segment '01' at cursor $k = 3$. The first step is the construction of the segment waveform (top). After computing the channel response $y_{01}(t)$, the sampling process at multiples of UI identifies the voltage samples to be considered to build the transition PDFs for each cursor [39] (bottom).

The computation of the channel response to each segment is the following step in the procedure, and it is done by convolving the segment with the channel impulse response. The resulting waveform is then sampled at multiples of the UI, and a transition PDF is constructed for each voltage sample, in the form:

$$pdf_k^{seg} = \delta(\mu - y_{seg}(t - kT)) \tag{2.10}$$

where $y_{seg}(t)$ is the channel response to one of the segments. Once the PDFs of all precursors and postcursors[2] have been constructed, they are combined to compute the total ISI contribution. This operation is the most delicate of the whole procedure. In fact, at this step the samples coming from the responses to the different segments must be properly combined in a sequential fashion, taking care that in the cascade of segments the final value of one transition coincides with the initial value of the neighboring one. If this is not verified, we have a bit changing exactly in the middle of a UI, which is not admissible.

Figure 2.7 shows the sequential operations in the case of two precursors and two postcursors. At each step, starting from the last significant postcursor segment ($k = 2$ in the figure), the PDFs are averaged and then combined by means of convolution with appropriate transition PDFs of the neighboring segment. To understand this basic operation, let's consider the postcursor segment '2' (see Figure 2.7):

[2] In the framework of ISI modeling the word *cursor* is used to identify the voltage samples at k multiples of UI in a generic channel response. These are the samples that must be accumulated to account for the total ISI at t. *Precursors* are the samples with $k < 0$ and *postcursors* the samples with $k > 0$, respectively. The cursor corresponding to $k = 0$ represents the voltage at the output of the channel in absence of ISI.

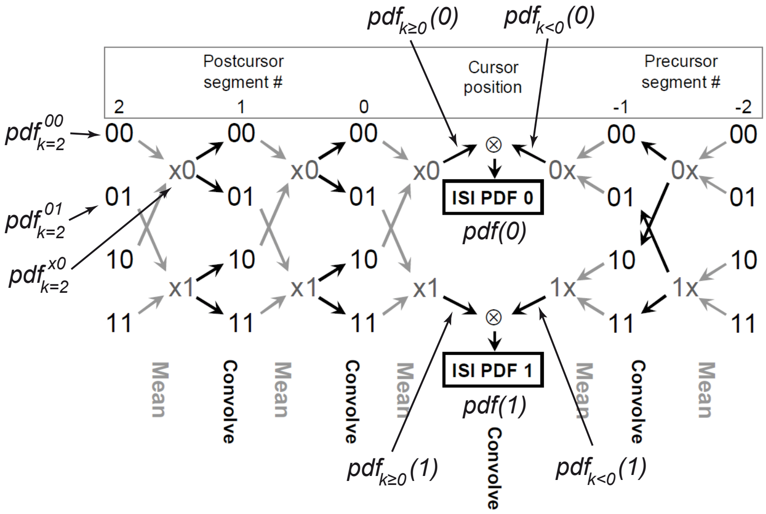


Figure 2.7: Diagram of the sequential accumulation of ISI in the simple case of two postcursors and two precursors only [38].

1. we average $pdf^{00}_{k=2}$ (segment "00") with $pdf^{10}_{k=2}$ (segment "10"), obtaining a temporary PDF $pdf^{x0}_{k=2}$;
2. we convolve the latter with $pdf^{01}_{k=1}$ of segment "01" at cursor '1'. That means we are computing the ISI contribution of all possible data patterns that have a 0-to-1 transition 1UI before cursor '0'.
3. we repeat steps 1. and 2. for all possible combinations of segments until coming back to cursor '0'.

By repeating this procedure we can calculate separately two PDFs for bit 0 ($pdf_{k\geq 0}(0)$) and for bit 1 ($pdf_{k\geq 0}(1)$) that take into account ISI effects due to all possible data patterns leading to a 0 bit or to a 1 bit at cursor '0'. To end the ISI calculation having considered all possible data patterns we apply the same procedure also for all the precursors. The last step is the convolution of $pdf_{k<0}(0)$ and $pdf_{k<0}(1)$, see again Figure 2.7, with respectively $pdf_{k\geq 0}(0)$ and $pdf_{k\geq 0}(1)$. Like in the SBR-based statistical algorithm the result of the ISI computation are two separate PDFs, one referred to the bit '0' and one referred to the bit '1'.

The BER calculation is done analogously to eq. (2.7), using:

$$BER(v,t) = \frac{1}{2}\left[\int_{-\infty}^{v} pdf(1)(\mu,t)d\mu + \int_{v}^{+\infty} pdf(0)(\mu,t)d\mu\right] \tag{2.11}$$

where $pdf(0)(\mu,t)$ and $pdf(1)(\mu,t)$ are the PDFs for bit 0 and bit 1, respectively, resulting from the sequential algorithm.

2.2 Jitter

In the literature, jitter is a concept shared among clock signals and data signals. While intuitively we are referring in both cases to the time deviation of a signal with respect to a reference, the jitter definitions used in the two applications are quite different. In the following of the thesis we will deal mainly with serial data, so we will use mostly the definitions belonging to this field, like Random Jitter (RJ), Deterministic Jitter (DJ), Duty Cycle Distortion (DCD), and so on. Nevertheless most of the jitter in a serial data stream stems directly from the clock signal driving the transmitter, thus we will encounter definitions as "period jitter" and "absolute jitter". For this reason, we will firstly introduce here the most used definitions of jitter in the field of clocking and, afterward, we will describe the jitter classification adopted when dealing with serial data.

2.2.1 Clock Jitter

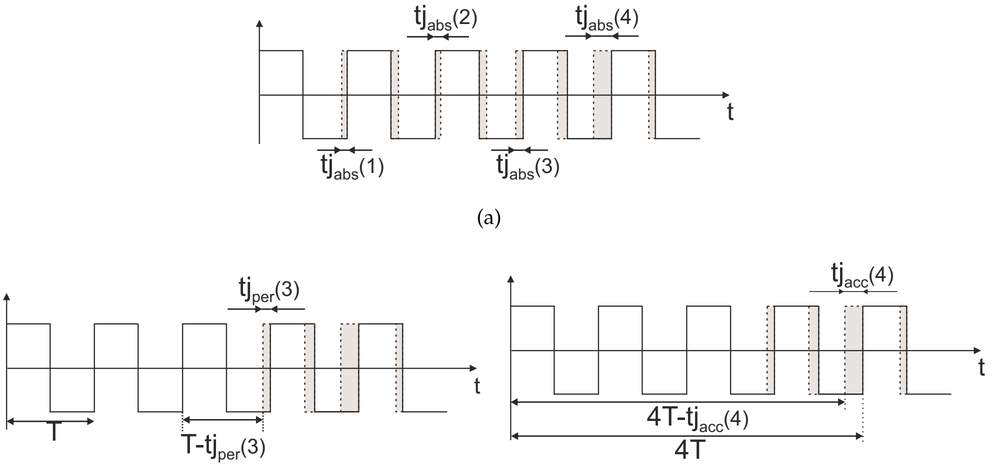
When dealing with clocking applications and their timing non-idealities, a number of definitions are given depending on the way jitter is measured [40, 41]:

Absolute Jitter: time displacement between the edge of an ideal clock and the real one. It is usually indicated as j_{abs} (Figure 2.8(a)).

Period Jitter: time variation of the clock period, measured as the time between one clock edge and the preceding one. It is indicated as j_{per} (Figure 2.8(b)).

Accumulated Jitter: time displacement of one clock edge relative to a starting edge of the same clock more than one clock cycle away. Accumulated jitter is a function of the number of cycles m and it is indicated as $j_{acc}(m)$. If $m = 1$ we have $j_{acc}(1) = j_{per}$ (Figure 2.8(c)).

For all these jitter types the value could be provided as rms, 1σ, 3σ, peak or peak-to-peak.



(a)

(b)

(c)

Figure 2.8: Clock jitter definitions: (a) absolute jitter j_{abs}, (b) period jitter j_{per} and (c) accumulated jitter j_{acc} [40]. The solid line represents the ideal unjittered waveform.

2.2.2 Jitter in Serial Data

In serial data communications, jitter is defined as the deviation of the timing properties of a signal with respect to a specified reference time and, historically, it is measured at the nominal switching threshold of the signal [42]. Jitter classification into categories is needed because jitter components accumulate differently in the link depending on their characteristics. Moreover breaking jitter into its various components allows to develop techniques to support budgeting of jitter in the design phase and leads to an efficient diagnosis of the causes of the jitter.

The first main distinction of jitter in serial communications is between *bounded* and *unbounded* jitter. Bounded jitter has the property that no population exists beyond specific limits regardless of the number of events observed while for unbounded jitter some finite population exists at all values of jitter (assuming an infinite sample size). By definition, all bounded jitter is *deterministic* jitter (DJ) and all unbounded is *random* jitter (RJ) [42].

Deterministic jitter could be further divided into different classes [42–44] (see also Figure 2.9):

Duty Cycle Distortion (DCD): is jitter due to different pulse widths for '1' bits compared to '0' bits. It is most easily observed in a clock-like data pattern and has a dual Dirac distribution. DCD does not depend on the data pattern.

Data Dependent Jitter (DDJ): is jitter that is correlated with the data pattern. Data Dependent Jitter (DDJ) is the effect in the time domain of the ISI phenomenon as we will see later in Section 2.3.

Periodic Jitter (PJ): is jitter that repeats in a cyclic fashion. Since any periodic waveform can be decomposed into a Fourier series of harmonically related sinusoids, this kind of jitter is sometimes called sinusoidal jitter. It is caused by external noise sources coupling into a system, such as switching power supply noise or a strong local RF carrier.

Bounded Uncorrelated Jitter (BUJ): it is usually caused by crosstalk coupling from adjacent interconnects. It is bounded in amplitude and uncorrelated to the data pattern and has a random distribution similar to RJ, but with limited spread (no tails).

Random jitter is Gaussian in nature [42]; thus, it can theoretically reach any magnitude (within physical limits). It is expressed as a single Gaussian distribution or a combination of multiple Gaussian distributions.

Deterministic jitter is measured as a peak to peak value for any distribution, random jitter is given as rms.

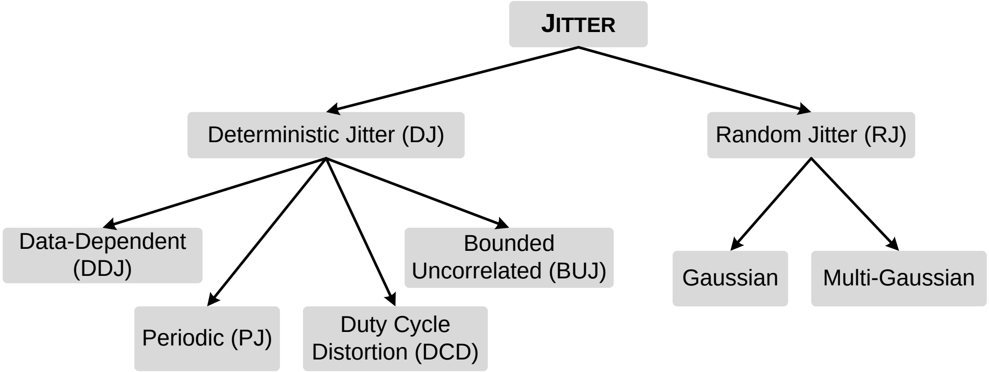


Figure 2.9: Jitter hierarchy [1].

Jitter budgeting of a link is specified through Total Jitter (TJ). Due to the presence of bounded terms side by side to unbounded terms, total jitter is specified as the time interval where all but a specified fraction of the population falls. Given that a jitter occurrence outside the TJ time interval means a bit error during data transmission, the fraction of population to specify is equal to the BER we can tolerate. Therefore, giving a TJ number without the respective BER value is meaningless. In modern link standards, e.g. PCIe [3], frequently the specified BER is 10^{-12}. The TJ distribution is assumed to be a dual Dirac:

$$pdf_{tj}(t) = \frac{1}{2\sqrt{2\pi}} \frac{1}{\sigma_{rj}} \left\{ e^{-\left[\frac{(t-dj/2)^2}{2\sigma_{rj}^2}\right]} + e^{-\left[\frac{(t+dj/2)^2}{2\sigma_{rj}^2}\right]} \right\}. \tag{2.12}$$

where dj is the peak-to-peak DJ value and σ_{rj} is the rms RJ value. To calculate TJ we thus have to calculate the CDF of the dual Dirac distribution. We can also use the approximated relation [3]:

$$tj = dj + 2Q_{BER} \cdot \sigma_{rj}. \tag{2.13}$$

Q_{BER} is a function of BER, and is calculated using the inverse error function [45]:

$$Q_{BER} = \sqrt{2} \cdot erf^{-1}\left[1 - \frac{1}{\rho_T} BER\right] \tag{2.14}$$

where ρ_T is the data transition density, which is equal to $1/2$ if we assume the same probability for bit '0' and '1'. Some significant Q_{BER} values are 5.99 for BER= 10^{-9}, 7.04 for BER= 10^{-12} and 7.94 for BER= 10^{-15}.

2.2.3 Jitter Modeling

In a high speed serial link design, margins at the receiver are equivalently affected by ISI, due to the band limited nature of the channel, and by timing uncertainty of the clock and data. Accounting for jitter is therefore as important as ISI when focusing on the prediction of the link performance. Numerous papers have been devoted to this topic, proposing different methodologies for handling all the various mechanisms responsible for deterministic and random jitter. In the following the main approaches reported in literature are reviewed.

Receiver Sampling Distribution Model

Historically, the receiver sampling distribution model is the first model developed to account for jitter in HSSI [36, 37], and it is also the simplest approach. It is based on the assumption that all jitter sources in a link could be treated as uncertainty in the receiver sampling distribution, no matter whether the jitter comes from the transmitter or from the receiver itself. This means that transmitter jitter and receiver jitter are considered as uncorrelated, thus possible jitter tracking effects of the Clock and Data Recovery (CDR) circuit are neglected [46] with possible overestimation of jitter in the link. Moreover, when considering the transmitter jitter the same way as receiver sampling uncertainty, possible effects of the channel limited bandwidth over jitter are ignored. Therefore, the so-called jitter amplification, reported in [47, 48], is not taken into account, causing possible under-estimation of jitter in the link model.

On the other hand, the receiver sampling distribution model has the great advantage of being of straightforward implementation in a ISI statistical simulation framework. In fact, it is possible to calculate a PDF that takes into account the combined effects of ISI and jitter [33, 37, 49], starting from the PDF of ISI $pdf_{ISI}(v,t)$ calculated with one of the statistical approaches described in Section 2.1 and the PDF of jitter ($pdf_{jitter}(t)$), using:

$$pdf(v,t) = \int_{-\infty}^{\infty} pdf_{ISI}(v, t-\tau) \cdot pdf_{jitter}(\tau) d\tau \tag{2.15}$$

which is a convolution of the two PDFs in the time domain. Note that when choosing $pdf_{jitter}(\tau)$ one must not account for DDJ effects of the channel, because they are already included in the ISI analysis, as we will demonstrate in Section 2.3.

Equivalent Voltage Noise Model

This alternative approach to jitter modeling in the statistical domain stems from the observation [34,36,50,51] that the transmitted pulse train $x(t)$ can be written as:

$$x(t) = \sum_{k=-\infty}^{\infty} (b_k - b_{k-1}) \cdot u(t - kT) \tag{2.16}$$

where $u(t)$ is the unit step function, defined as $u(t) = 1$ for $t > 0$ and $u(t) = 0$ otherwise. Noting that the output of the channel can be characterized by its impulse response $h(t)$, the channel output can be determined by convolving the input pulse train with the channel impulse response $h(t)$:

$$y^{ISI}(t) = x(t) \otimes h(t) = \sum_{k=-\infty}^{\infty} [(b_k - b_{k-1}) \cdot s(t - kT)] \tag{2.17}$$

where $s(t) = u(t) \otimes h(t)$ is the step response of the channel.

The first step to account for jitter is to observe that at transmitter side the time instants of the data edges are not ideal but affected by transmitter jitter ϵ_k^{TX}, and rewrite eq. (2.17) as:

$$y(t) = \sum_k (b_k - b_{k-1}) s(t - \epsilon_k^{TX} - kT). \tag{2.18}$$

At the receiver side the data output by the channel is sampled at jittered time instants $t_m = mT + \epsilon_m^{RX}$. The sampled signal y_m can be written as:

$$y_m = \sum_k (b_k - b_{k-1}) s(\epsilon_m^{RX} - \epsilon_k^{TX} + (m - k)T). \tag{2.19}$$

Note that to derive this expression no approximation has been introduced, thus completely describes data transmission over a band-limited channel in the presence of jitter. Note also that ϵ_m^{RX} is not a function of the index k because it does not alter the transmitted bit as it happens for ϵ_k^{TX}. If we now use the Taylor series expansion and we truncate it to the first order we obtain:

$$\begin{aligned} y_m &\cong \sum_k (b_k - b_{k-1}) s((m - k)T) + \\ &+ \sum_k (b_k - b_{k-1}) \epsilon_k^{TX} h_{m-k} + \epsilon_m^{RX} \sum_k (b_k - b_{k-1}) h_{m-k} \\ &= \sum_k b_k p_{m-k} + n^{TX} + n^{RX} \end{aligned} \tag{2.20}$$

where h_{m-k}, obtained by a time derivation of the step response $s\left((m-k)T\right)$, is the data-rate sampled impulse response of the channel, while n^{TX} and n^{RX} are defined as the Equivalent Voltage Noise (EVN) terms for transmitter and receiver jitter, i.e. the contribution to the received voltage due to transmitter and receiver jitter, respectively. The term $\sum_k b_k p_{m-k}$ represents the received signal in absence of jitter. Eq. (2.20) shows that the EVN terms could be determined independently from ISI calculation.

To better understand the model for jitter provided by eq. (2.20), we may refer to Figure 2.10. The top part shows how a data pulse affected by jitter at the transmitter side can be

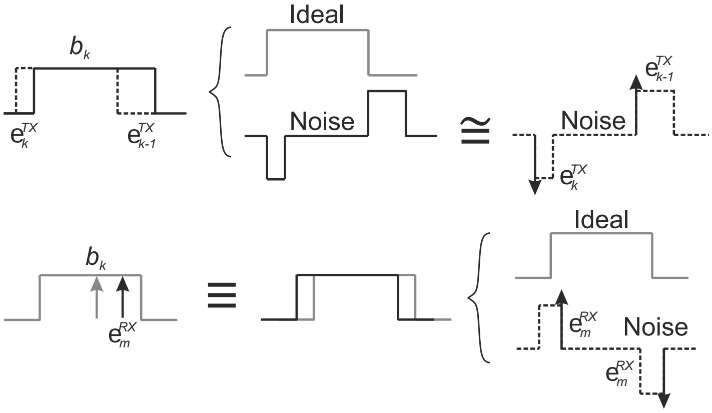


Figure 2.10: Models for transmitter (top) and receiver (bottom) jitter in the framework of the equivalent voltage noise model [50].

represented as the sum of an ideal pulse representing the data without jitter, and two pulses of time amplitude equal to the jitter magnitude placed at the ideal time crossings. If TX jitter affecting the data is small, these pulses can be approximated as impulses (Dirac's delta) whose amplitude is the jitter magnitude. The same approximation is used for the receiver jitter, as shown on the bottom plot of Figure 2.10. The sampling time uncertainty could also be viewed as a rigid time shift of the data pulse due to ϵ_m^{RX}, which translate in the sum of an ideal data pulse and two pulses of ϵ_m^{RX} width. Again if ϵ_m^{RX} is small, the two pulses could be represented as impulses (Dirac's delta). We understand then that the first-order approximation of the Taylor series expansion in the conversion step from timing jitter to voltage impulses is limiting the accuracy of EVN model to small jitter only.

Segment-based Model

The segment-based approach to jitter modeling is tightly bound to the transition-based ISI analysis described in Section 2.1.3. In fact, the two approaches have been developed together in [38].

We have already shown how ISI is modeled in the transition-based method by means of the construction of segments representing all the possible transitions between voltage levels at the transmit side (see Figure 2.5). Now we have to consider that each transition is affected by jitter; thus, we have to take it into account when constructing the segments. This is done by considering a group of closely spaced segments instead a single one for each transition to model, as shown in Figure 2.11. The number of segments in the group depends on the TX jitter distribution, e.g. in Figure 2.11 the transition '01' has 5 segment shapes because its jitter PDF is discretized with 5 possible values. The channel responses to the segment group are then determined and sampled at each cursor. The different voltage samples obtained at each cursor position produce a PDF made of a group of Dirac's delta (see bottom of Figure 2.11) which, after multiplication with the jitter PDF, give the cursor transition PDF that binds ISI and TX jitter. Once the cursor transition PDFs for all the possible transitions have been determined, they are combined using the same sequential algorithm depicted in Figure 2.7. The advantage of this combined ISI and jitter modeling approach is that it produces a PDF accounting for both ISI and jitter, and also their interactions. The latter in particular is the missing element in the receiver sampling distribution approach.

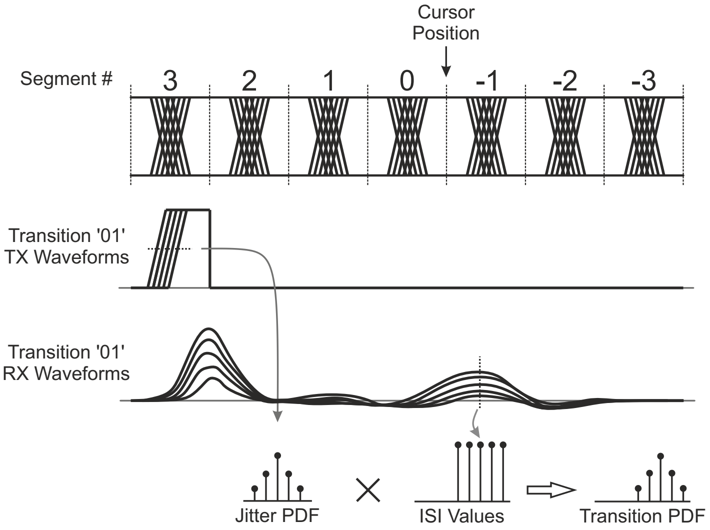


Figure 2.11: Transition PDF calculation example in the case of a ’01’ transition and a TX jitter PDF with 5 possible values [38].

The receiver contribution to the link jitter is modeled using the receiver sampling distribution approach.

2.3 Intersymbol Interference vs. Data-Dependent Jitter

As we have seen in Section 2.2.2, DDJ is defined as the threshold-crossing time deviations from a reference time due to the memory of previous data bits [52]. From this definition we can affirm that DDJ is the distortion on the threshold-crossing time resulting from ISI. In fact DDJ is determined by the high frequency dispersion of the medium, electromagnetic reflections or low frequency coupling and other mechanisms related to the frequency response of the link.

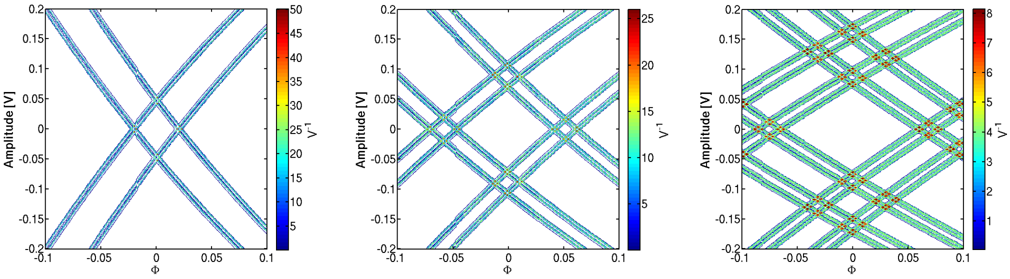
To understand DDJ, it is useful to consider as a link medium a first-order system, described by the transfer function:

$$H(s) = \frac{1}{1+\tau s}. \tag{2.21}$$

Here τ is the system time constant, thus the associated $-3\,\text{dB}$ bandwidth is $1/(2\pi\tau)$. When transmitting serial data over a band-limited link, the bandwidth of the system itself is not the only parameter necessary to fully describe the DDJ or ISI effects. In fact one must relate the bandwidth and the data rate, because finite-bandwidth limitation is much more severe when forcing high bit rates. It is possible to account for the bandwidth-data rate relationship by defining a parameter $\alpha \equiv e^{-T/\tau}$. This variable relates the system time constant to the bit time [52] thus giving a measure of the channel bandwidth to data rate ratio. On one hand, if α approaches zero the system has a large bandwidth compared to the bit rate, and we can expect a low impact of ISI on the transmission (thus low values of DDJ). On the other hand, small bandwidth compared to the bit rate means that the data transition between neighboring bits is slow, thus the observed DDJ will be quite large. In the case of the first-order system, it is possible to express the peak-to-peak DDJ value Δt_{pp} as a function of α as [52]:

$$\Delta t_{pp} = -\tau \cdot ln(1-\alpha). \tag{2.22}$$

From the above expression we can derive the upper limit for α. In fact, the worst case for DDJ is when the bit transition crosses the voltage threshold (assumed to be equal to half of the swing between high and low bit level) at T. Thus, posing $\Delta t_{pp} = T$ in eq. 2.22, we obtain $\alpha = 0.5$. This condition corresponds to a bandwidth of only 11% of the data rate.



(a)

(b)

(c)

Figure 2.12: Eye diagram magnified around the threshold crossing time for a first order system with (a) $\alpha = 0.1$, (b) $\alpha = 0.2$ and (c) $\alpha = 0.3$.

Figure 2.12 shows the eye diagram threshold-crossings for the first-order system and three increasing values of α, obtained with a SBR-based statistical ISI algorithm as the one described in Section 2.1.2. The increasing amount of DDJ coming with the increase of α is evident. Another interesting effect is visible: the number of signal paths crossing the threshold increases with α too. In fact, in Figure 2.12(a) the total paths crossing the threshold is two for $\alpha = 0.1$, while in Figure 2.12(b) for $\alpha = 0.2$ we have four crossing paths, and they become eight in Figure 2.12(c) for $\alpha = 0.3$. We can explain this behavior noting that when the bandwidth to bit rate ratio decreases, the number of prior bits whose residual response affects the current bit increases. The number of possible combinations of patterns to account for in determining DDJ increases as power of two of the number of prior bits involved. Each combination of patterns determines in turn a different deviation from the ideal crossing time, thus the splitting into multiple crossing paths [52]. Therefore, for $\alpha = 0.1$, only the first neighboring bit is affecting the current one, thus only 2^1 paths are possible. With $\alpha = 0.2$ the affecting bits are 2 thus giving 2^2 possible paths, and so on.

Thanks to the fact that DDJ and ISI are different representations of the same physical mechanism, we can affirm that DDJ is naturally accounted for in all the ISI statistical algorithms described in Section 2.1.

Figure 2.13 demonstrates that the analytical DDJ model and the same SBR-based statistical ISI algorithm considered in Figure 2.12 give exactly the same results. This is done by comparing Δt_{pp} from eq. (2.22) with the difference between T and the BER bathtub opening obtained for the same first-order system using the statistical ISI algorithm. Similar demonstration could be given also in the case of channels with a higher order than the first one assumed here. In this case extracting Δt_{pp} as a function of α in a closed form [52] would have been not trivial, or even impossible, thus making the reasoning less intuitive.

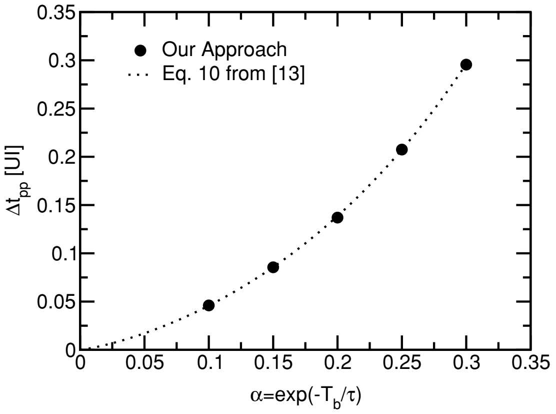


Figure 2.13: Comparison of the peak-to-peak DDJ Δt_{pp} in the case of a first-order channel obtained from a SBR-based statistical ISI algorithm (as the one presented in Section 2.1.2) and from Eq. (2.22) [52]. T is the bit period and τ is the system time constant. α approaches zero in the case of a system with large bandwidth compared to the input data rate.

Chapter 3

Improved ISI and Jitter Modeling

When approaching the modeling of HSSIs, and in general of any other electronic system, one finds himself in front of two possible choices: either a *top-down* or a *bottom-up* approach. The bottom-up approach, i.e. from a schematic view of the single blocks up to the very complete system, might be desirable because allows to analyze in the very detail each block. On the other side, a top-down approach makes easier to have a less accurate but more immediate estimate of the performance of the entire system. It is thus straightforward to determine the most important impact factors on the performance, without the need to deal with full details of each block implementation.

We have mentioned in Chapter 2 that HSSI performance simulation using a traditional time-domain approach is pretty unpractical in terms of computation complexity, and thus simulation time. In the case of HSSIs a bottom-up approach is therefore not desirable if the target is the analysis is the overall system performance. The top-down approach provided by statistical simulation techniques, on the contrary, is quite promising as it allow for an immediate overview of the ISI link performance thanks to the computation efficiency of such techniques.

This chapter describes a new statistical algorithm, developed in this Ph.D. thesis, to model the ISI and transmitter jitter in HSSI. This approach aims at improving the accuracy of the statistical ISI simulation technique with a more accurate modeling of the transmitter driver waveform, a feature that is not available in the approaches presented in literature. Then, it presents the approach followed to include in the analysis also the effects of transmitter jitter, and the detailed validation of the results by means of comparison with other simulation approaches.

3.1 Transmitter Waveform and Validity of the LTI Assumption

The main assumption behind the statistical approach is that the transmission system is Linear and Time Invariant (LTI) [33, 36, 37, 49]. Linearity is needed when calculating the overall response to a stream of bits as the sum of SBRs while time invariance is needed to assure that the SBR does not change in time. These two assumptions are valid when considering common passive physical channels (i.e. cable, printed circuit board, etc.) but become inaccurate when considering also the transmitter as a part of the system to be simulated. For this reason a concern when trying to apply a statistical algorithm to the simulation of a transmitter-channel system falls over the LTI property of the transmitter itself. In fact, the transmitter has a substantial non linear behavior, which could be easily understood observing Figure 3.1: here the waveforms at the differential output of a 2.5 Gb/s CMOS transmitter obtained from SPICE simulations are compared with the waveforms obtained as linear combination of the transmitter single bit waveforms. In the top graph of Figure 3.1 it is possible to see that the waveform produced by the transmitter in the case of two '1' bits separated by one '0' bit is equal to the waveform obtained by the linear combination of the differential single pulse and the same differential single pulse shifted by two bit periods. On the contrary, in the bottom graph of Figure 3.1 we see that the same procedure is not valid in the case of two consecutive '1' bits. In this case the waveform obtained by linear combination of the TX differential single pulse shows an artifact at the transition between the two '1' bits, demonstrating that the transmitter operates as a non linear element. Figure 3.2 shows how the non linear behavior of the transmitter is transferred to the statistical eye diagram resulting from a SBR-based statistical algorithm like the one described in paragraph 2.1.2. Here the contour plot of the PDF obtained with the statistical algorithm considering the transmitter alone (no channel) is compared with the eye diagram obtained from a SPICE simulation of the same transmitter. We see that the distortion related to two consecutive '1' bits visible in the bottom graph of Figure 3.1 (@$t = 0.7$ ns) affects also the eye diagram and, in particular, the trajectories associated to the transitions between two '0' bits and two '1' bits. This distortion is not acceptable because it hampers the prediction capability of the link model.

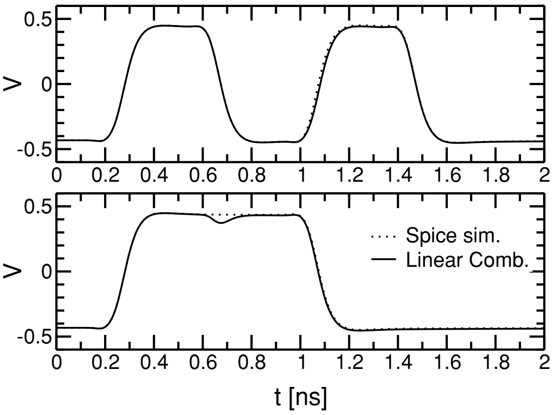


Figure 3.1: (Top) Differential waveform of two '1' bits separated by one '0' bit obtained with a Spice simulation of a transmitter driving two ideal 50 Ω resistors compared with the waveform obtained by summing the differential single pulse response with the same single pulse response shifted by two bit periods. The transmitter is working at 2.5 Gb/s. (Bottom) The same waveforms for the case of two consecutive '1' bits.

3.2 Edge Responses

A viable strategy to include in the link analysis also the real transmitter waveform is the adoption of a statistical algorithm based on the edge responses [38]. This approach maps into different pulses all the possible waveforms representing a transition between two consecutive bits as produced by the transmitter during data transmission. This technique, which models each edge as ideally vertical, was initially developed in order to decouple the rising and falling edges in the data stream at the channel input in order to use different jitter dis-

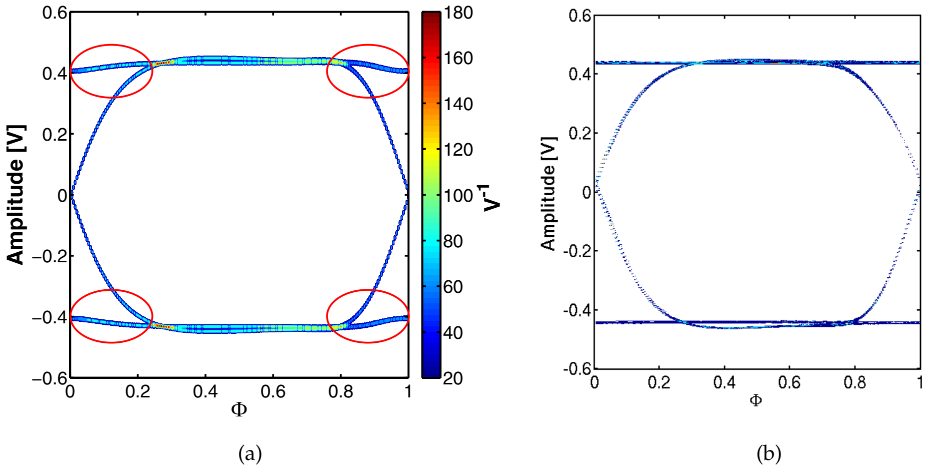


Figure 3.2: Contour plot of the PDF of both levels ′0′ and ′1′ obtained using the differential single bit response of the transmitter (left) and the eye obtained from a Spice simulation of a transmitter driving two ideal 50 Ω resistors (right). Circles in the left panel highlight discrepancies with Spice simulations.

tributions for each of the two edges [38]. Besides the increased flexibility in jitter modeling, this technique permits to model carefully the time interval across the transition between two consecutive bits, where the non linear behavior of the transmitter reveals itself, by considering the real rising and falling edge shape instead of ideally vertical edges. The number of transitions to account for depends on the signaling scheme of the system; for instance, in the case of NRZ signaling the total number of possible transitions is four.

The real transition waveforms can be extracted from a SPICE simulation of the transmitter driven with a pattern of ′0′s and a single ′1′ bit and considering ideal 50 Ω loads (see Figure 3.3). Figure 3.4 (top) shows the typical waveform obtained simulating a transmitter for 2.5 Gb/s HSSI. This single pulse waveform allows to extract the four transition waveforms necessary to completely model the transmitter behavior in the following way (see Figure 3.4):

- the ′00′ case (center-left) is a horizontal line equal to the low value of the pulse;
- the ′01′ and ′10′ cases are constructed by splitting the pulse waveform exactly in the middle;
- the ′11′ case (bottom-right) is a trapezoidal pulse obtained by keeping constant over the whole bit interval the value of the pulse at the middle of the UI and adding sharp transitions with the same slope as for the low-high and high-low transitions, respectively.

Once computed the transmitter transition waveforms, the next step is the determination of the time domain response of the channel to each transition. These responses, referred as *transition responses*, will then be input to a statistical algorithm to compute the ISI Probability Distribution Function (PDF).

3.3 Channel Response

The most common representation of a physical channel is through its S parameters, which represent how the channel reflects or transmits the incident power waves over frequency at

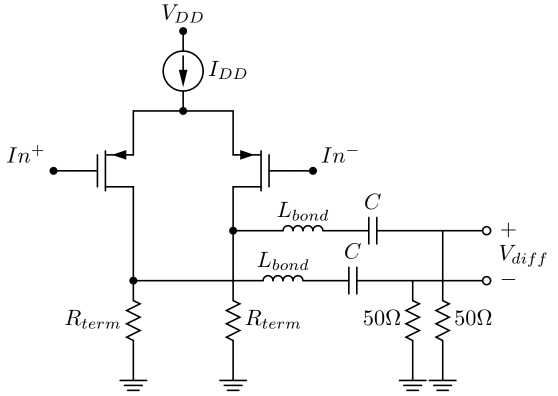


Figure 3.3: Schematic of the simulated 2.5 Gb/s transmitter circuit to extract the single pulse response (Figure 3.4, top graph). L_{bond} is the inductance of the bonding wire and C the decoupling capacitance. The transmitter drives two ideal 50 Ω loads.

each of its ports.

Based on the definition of incident power wave a_i and reflected power wave b_i at port i of the network:

$$\begin{aligned} a_i &= \frac{1}{2\sqrt{Z_0}}\left(V_i + Z_0 I_i\right) \\ b_i &= \frac{1}{2\sqrt{Z_0}}\left(V_i - Z_0 I_i\right) \end{aligned} \tag{3.1}$$

the S parameters are defined as the ratio between the reflected power wave at port i and the incident power wave at port j, with all the incident power waves at the other ports set at zero. The relation between b_i and a_i is thus the following (for a 2-port network):

$$\begin{Bmatrix} b_1 \\ b_2 \end{Bmatrix} = \begin{bmatrix} S_{11} & S_{12} \\ S_{11} & S_{12} \end{bmatrix} \begin{Bmatrix} a_1 \\ a_2 \end{Bmatrix} \quad \text{with} \quad S_{ij} = \left.\frac{b_i}{a_j}\right|_{a_{k,k\neq j}=0} . \tag{3.2}$$

When focusing on links adopting differential signaling schemes, from the single-ended 4-port S parameters describing the channel, the mixed-mode S parameter matrix must be extracted. The mixed-mode S parameters completely describe how the differential and common mode signals travel across the channel or are reflected at each port, and also how much conversion between modes occurs. For the simulation of ISI just the description of the differential behavior is required, so in the following we will focus exclusively on the 2-port differential S parameter matrix S_{DD}, which could be extracted from the 4-port single-ended matrix through the simple arithmetical calculations provided in [13]. Then the transfer function between differential source voltage V_s and differential output voltage across the load impedance V_l (see Figure 3.5) can be found using the following [53]:

$$\mathcal{F} = \frac{V_l}{V_s} = \frac{S_{DD21}(1-\Gamma_l)(1-\Gamma_s)}{2(1-S_{DD22}\Gamma_l)(1-\Gamma_{in}\Gamma_s)} \tag{3.3}$$

where Γ_s, Γ_l and Γ_{in} are the reflection coefficients at the source side, load side and input port

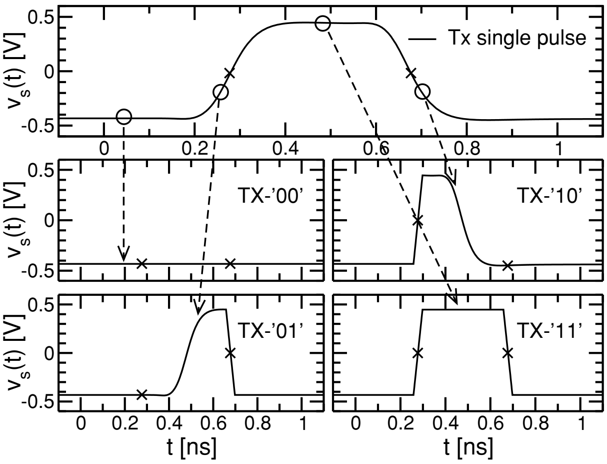


Figure 3.4: Differential single bit response of a 2.5 Gb/s CMOS transmitter obtained from a circuit simulator (Titan) (top). This waveform is used to construct the four transition waveforms which are then input to the channel (bottom). The Unit Interval (UI) ranges from 0.27 to 0.67 ns and is indicated in each graph by the two × symbols. The slope of the linear segments in the TX–'01', TX–'10' and TX–'11' waveforms has been chosen low enough to avoid the introduction of unwanted signal components outside the measured channel S-parameters frequency range. The superposition of two of these linear segments (with opposite slope) yields a null contribution during the forthcoming ISI calculation.

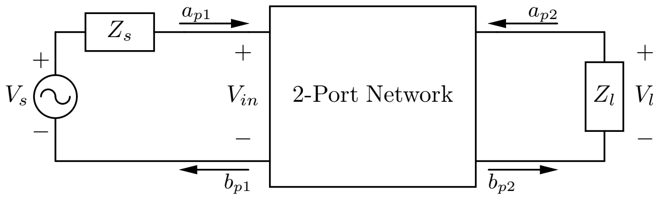


Figure 3.5: Setup for computing the transfer function $\mathcal{F}$.

of the channel, respectively, defined as:

$$\begin{aligned}\Gamma_s &= \frac{Z_s - Z_0}{Z_s + Z_0}\\ \Gamma_l &= \frac{Z_l - Z_0}{Z_l + Z_0}\\ \Gamma_{in} &= S_{DD11} + \left(S_{DD12} \cdot S_{DD21} \frac{\Gamma_l}{(1 - S_{DD22}\Gamma_l)}\right).\end{aligned} \tag{3.4}$$

Notice that Z_l and Z_s denote the differential impedances of the load and source and Z_0 the differential characteristic impedance of the channel. In the case of single-ended channels matched to 50 Ω, all the impedances in eq. (3.4) are thus equal to 100 Ω.

Once the differential channel transfer function is obtained by means of eq. (3.3), the time domain impulse response is extracted, so that the response to the transition waveforms can be calculated by convolving the impulse response with the transition waveform of interest. To do this, the channel transfer function is approximated with a rational function expressed in one of the canonical forms, e.g. the pole-residue form:

$$H(s) = \sum_{m=1}^{N} \frac{r_m}{s - a_m} \tag{3.5}$$

which allows to calculate the impulse response by simply applying the inverse Fourier transform.

The extraction of a rational function $H(s)$ that fits the channel transfer function $\mathcal{F}$ is a non-trivial task. In fact, $\mathcal{F}$ is a complex-valued function containing both magnitude and phase information over frequency thus the fitting must take into account both quantities. Furthermore the number N of poles is not fixed or determined on a-priori basis, but it is a degree of freedom of the fitting process.

The problem of fitting a complex-valued function defined over frequency has been studied in the past to find means of including frequency dependent dispersive effects in time-dependent simulations. References [54, 55] present an accurate algorithm, namely the vector fitting algorithm, that computes a set of N poles, either real or complex conjugate pairs, that are a good approximation of the input function starting from a given set of N complex conjugate poles using a least squares approach[1]. What is important to note here is that the quality of the fitting provided by the new poles strongly depends on the choice of the starting poles. A common approach to define the starting poles is to chose them as uniformly distributed over all the frequency range, so it will be necessary to run the vector fitting algorithm many times using at each step the set of poles identified at the previous step to refine the agreement between $\mathcal{F}$ and $H(s)$. To control the iteration process and consequently stop it when a sufficiently good agreement has been obtained we defined an error function in the form:

$$\epsilon \geq \frac{\sqrt{\sum\limits_{k=1}^{n} |\mathcal{F}\{f_k\} - H(s)|^2}}{\sqrt{\sum\limits_{k=1}^{n} |\mathcal{F}\{f_k\}|^2}} \qquad \text{with} \qquad s = j2\pi f_k \tag{3.6}$$

that is evaluated at each iteration step. When the error ϵ has a value lower than the maximum desired error ϵ_{max}, the set of poles and residues just calculated identify the desired $H(s)$. On the other hand, if after a certain number of iterations, usually in the order of a couple of tens, the error is still above ϵ_{max} the number of poles N is too low to approximate $\mathcal{F}$ with the desired accuracy. To improve the fitting we have then to increase N, calculate a new set of starting poles uniformly distributed over the frequency range and repeat the iteration described above.

As an alternative to the procedure described above, the response of the channel to the transition waveforms can be also obtained using Spice simulation. This approach requires to set up a circuit as the one shown in Figure 3.6, where the 4-port single ended S parameters of the channel are included into the schematic by using a Linear N-port device [56]. Each transition waveform is fed to the circuit by means of the ideal voltage generator v_s, paying attention to the fact that the instantaneous voltage value of the waveform must be doubled to cope with the halving effect to due the voltage divider formed by the R_s-Z_0 pair. Clearly both R_s and R_{load} are equal to $100\,\Omega$ being differential impedances. The disadvantage of this approach is that the implementation depends on the way each Spice simulator handles S-parameters.

[1] The detailed explanation of the complex mathematics behind this algorithm will not be treated here, because it was not object of Ph.D. work. The reader can find all the details about the vector fitting algorithm and the complete mathematical demonstration in [54, 55]. In these references the reader will also find the guidelines on how to chose the set of starting poles.

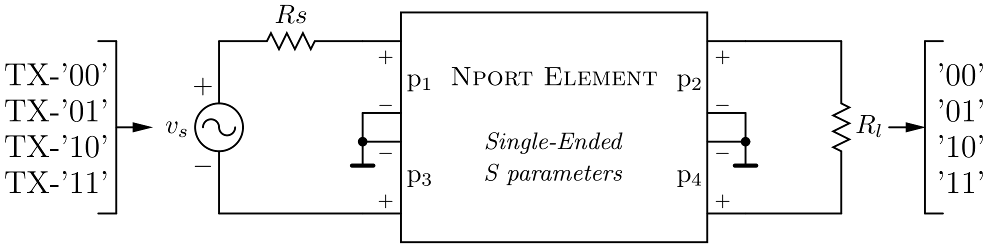


Figure 3.6: Circuit to be used to obtain the channel responses to the transition waveforms with Spice simulations. The channel is represented by its full 4-port S-parameter matrix.

3.3.1 Considerations about Impedance Discontinuities in the Link

The proposed methodology requires circuit simulations. In Section 3.2 and 3.3 we proposed a two steps procedure, which for a matter of clarity is also shown in Figure 3.7, where the TX is simulated first with a differential load of 100 Ω and then the resulting waveform (after being elaborated as in Figure 3.4) is applied to the channel plus RX.

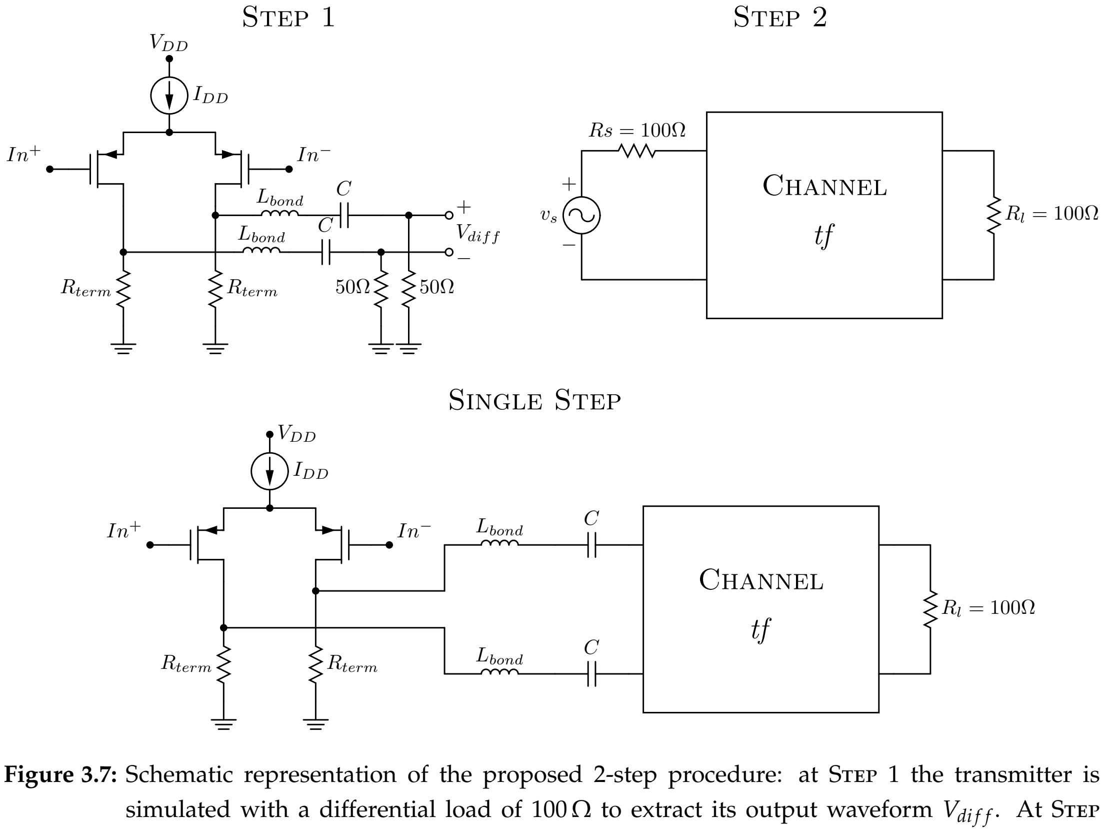


Figure 3.7: Schematic representation of the proposed 2-step procedure: at Step 1 the transmitter is simulated with a differential load of 100 Ω to extract its output waveform V_{diff}. At Step 2 the transition waveforms extracted from V_{diff} as showed in Figure 3.4, are applied to the channel. Note that at Step 2 the transmitter is modeled as an ideal voltage generator applying the desired transition with an ideal 100 Ω internal impedance. For a matter of comparison also the schematic diagram of the Single Step procedure is shown.

This procedure is necessary because we cannot merge the TX with the channel and the RX and then simulate the whole system with ideally sharp input transitions: a simulation including TX, channel and RX (single step in Figure 3.7) would make it difficult, if not impossible, to separate the responses to the '00', '01', '10' and '11' transitions mainly due to the high losses and resonances of the channel. We thus need a simulation of the TX alone (step 1 in Figure 3.7) with as small distortion as possible, to ease the task of separating the four transitions, which are then applied to the channel (step 2 in Figure 3.7).

This approach makes possible to handle arbitrary TX waveforms. Approaches based on the SBR require a single simulation of the whole link, which do not introduce approximations, but are much less accurate in handling arbitrary TX waveforms because the TX is a non linear element as demonstrated in Paragraph 3.1 and thus could not be included as part of the link.

A specific issue of our approach is how to split the communication chain to separate what should be included in the "TX" (i.e. in the simulation step 1) from what should be included in the "channel" (i.e. in the simulation step 2). A simple and viable strategy is to denote as "channel" the portion of the communication chain comprising the board and the receiver, while bonding wires and package should be then included in the TX reference plane, i.e. to set the separation for S parameter measurements at the package to board transition. This choice allows to easily extract the four transitions from the clean pulse waveform as output of simulation step 1. Since in step 1 the TX drives a 100 Ω load, the waveforms are equal to the ones obtained simulating the channel plus RX, when they are exactly matched to 100 Ω. This is enforced by the fact that in step 2 we simulate the TX as a voltage generator producing the waveform at step 1 multiplied by two and loaded by a 100 Ω differential impedance. Reflections due to the bonding wires and the package impedance discontinuities are naturally included in the TX response provided by step 1. Of course, if these are very large, the waveform at the TX output given by step 1 will be degraded and it will be difficult to separate the four transitions. However, in a well designed system this should not happen.

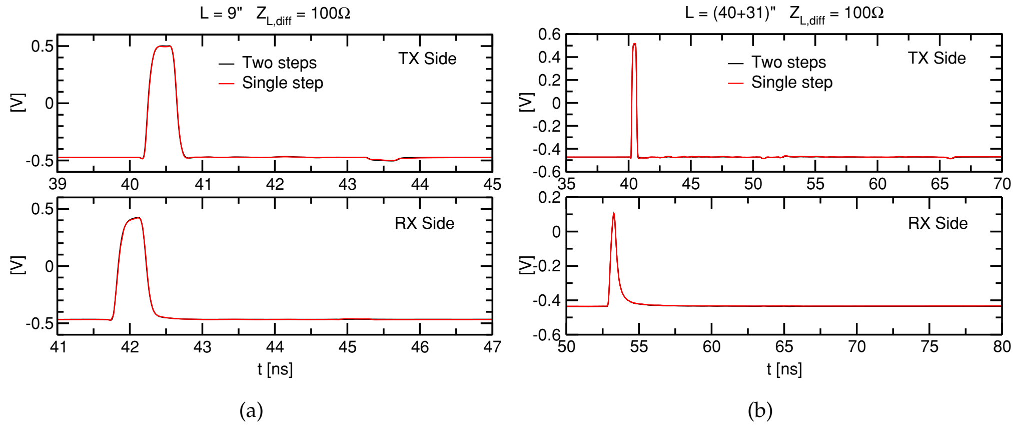
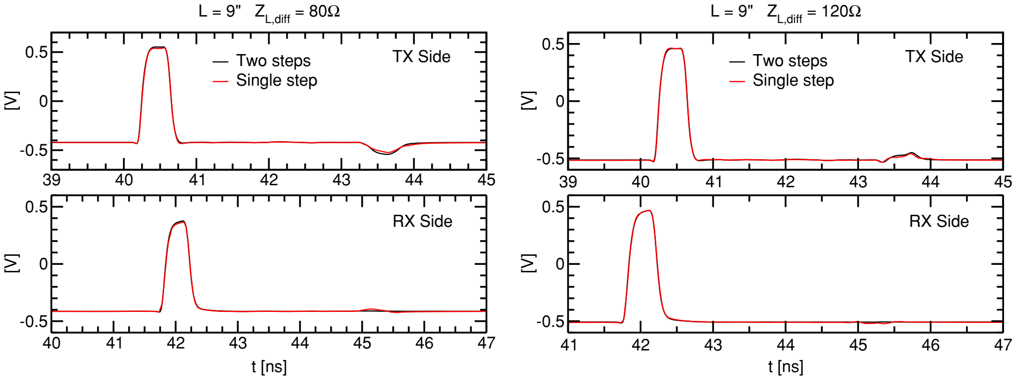


Figure 3.8: Comparison between the two steps approach (ideal voltage generator with ideal 100 Ω differential internal impedance driving the channel and the receiver, using the TX waveform obtained from a Spice simulation of the transmitter, including bonding wires, driving ideal 50 Ω loads) and a single simulation of the whole system. The RX differential impedance is 100 Ω. (a) 9″ channel ($S_{DD21} = -1.2$ dB @ 1.25 GHz) and (b) 71″ channel ($S_{DD21} = -8.9$ dB @ 1.25 GHz).

To investigate quantitatively the limitation of the two steps procedure, the propagation of a single pulse through the whole system has been simulated. Backplane differential channels with lengths up to 71″ have been considered to explore the impact of ISI over data transmission. In the two steps procedure, the bonding wires are considered as part of the TX. By definition, the complete Spice simulations and the proposed two step approach give the same results if channel plus RX are exactly 100 Ω. Figure 3.8 analyzes this case; Figure 3.8(a) for a short 9″ channel and Figure 3.8(b) for a 71″ channel, demonstrating that with a 100 Ω differential RX the two steps and the single step procedure give indeed the same results, as expected.

The cases of realistic RX mismatch (80 Ω and 120 Ω purely real differential impedance) are reported in Figure 3.9 (a and b), where we see that the two steps procedure provides accurate results at the TX and RX side even in the limiting case of a short 9″ board.

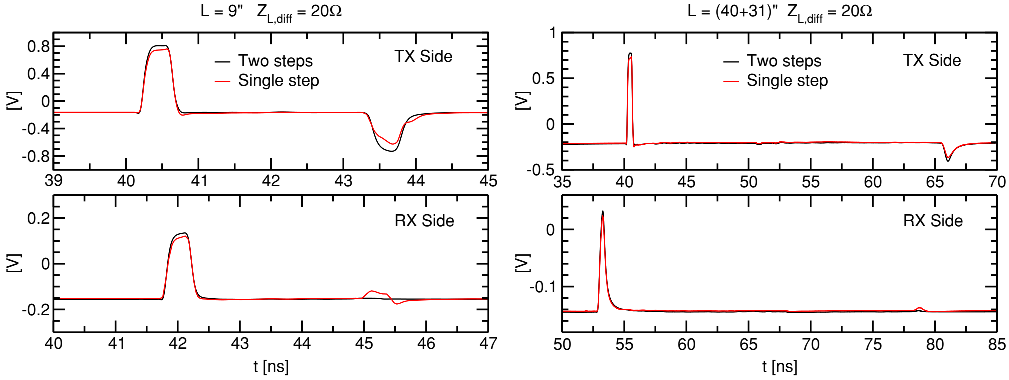


(a)

(b)

Figure 3.9: Same comparison as in Figure 3.8 in the case of the short channel (9″, $S_{DD21} = -1.2$ dB @ 1.25 GHz) with reasonably unmatched RX. (a) Differential impedance of 80 Ω and (b) 120 Ω.

In Figure 3.10 we consider the limit case of RX with a 20 Ω differential impedance as an example of discontinuity in the link. In this case, for the short channel (left plot) the two steps procedure significantly deviates from the single one, meaning that for this case one should devise alternative methods to separate TX and channel for the purpose of the two steps procedure. On the other hand, for long channels (right plot of Figure 3.10) the single and two steps procedure are in good mutual agreement.



(a)

(b)

Figure 3.10: Same as in Figure 3.8 above, but with a strongly unmatched receiver ($Z_{L,diff} = 20\,\Omega$).

These considerations shed new light on the ability of the proposed approach to model the TX-channel-RX system when impedance discontinuities are present along the link. In particular, they show that we can expect reliable ISI and jitter calculations in all the practical situations of a well designed link with a reasonable amount of mismatch (i.e. ±10% as considered in Figure 3.9), even in the case of a very short link. On the other hand, if the link to be analyzed presents very large impedance mismatches, alternative channel response simulation strategies must be put in place.

3.4 Computing the ISI Probability Distribution Function

Once the channel transition responses have been obtained, the ISI is calculated following the same algorithm presented in [38], that is also described in detail in the following.

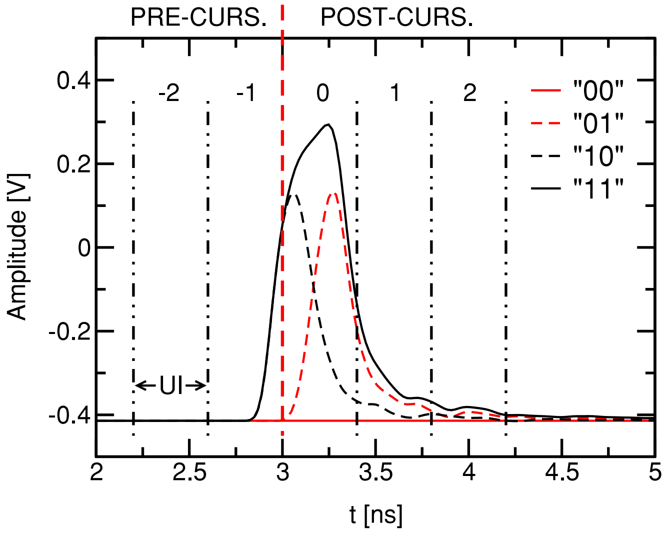


Figure 3.11: Example of channel responses to the four different TX transition waveforms of Figure 3.4, in the case of a channel with $S_{DD} = -4.2\,\text{dB}$. $T = 400\,\text{ps}$.

The first step of the algorithm is to divide all the transition responses into multiples of the UI, as shown in the example of Figure 3.11. Each multiple of the UI is called *cursor*, and the cursor containing the relevant part of the response of the channel is called *central* cursor and is numbered as cursor "0". The cursors taking place before and after the cursor "0" are denoted as *pre*-cursors and *post*-cursors, respectively. Post-cursors are identified with positive indexes while pre-cursors are identified with negative ones, as shown in Figure 3.11 in the case of 2 post-cursors and 2 pre-cursors only. All the relevant cursors before and after the cursor "0", i.e. all the UI intervals in which at least one of the transition responses is above a sufficiently small threshold ϵ, must be considered when computing ISI. Next we can define the phase Φ inside the UI as a normalized time, i.e. a number between 0 and 1 as the time is divided by the period T, and then identify a vector of time instants $t\{k\}$ spaced by multiples of T with respect to Φ, i.e.:

$$t\{k\} = (\Phi + k)T \qquad \text{with} \qquad -m \leq k \leq n. \tag{3.7}$$

This vector groups the $n + m$ time instants whose voltage samples must be accounted for when determining the total ISI contribution corresponding to the same phase Φ of the statistical eye diagram. Therefore the ISI calculation requires to accumulate the ISI caused by all the voltage samples of the channel responses at the time instants $t\{k\}$ and then repeat the calculation for all the different phases Φ inside the UI. If we assume that Φ is sampled over i intervals, the ISI accumulation must be repeated i times.

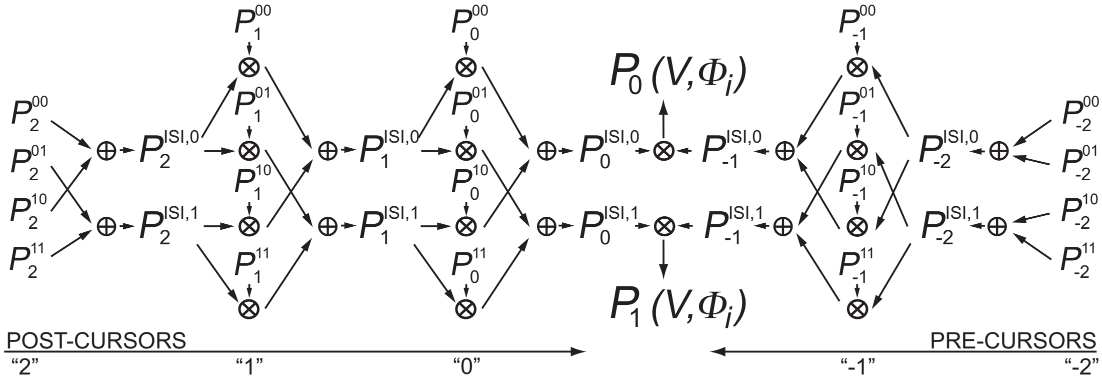


Figure 3.12: ISI calculation algorithm in the case of $n = 2$ postcursors and $m = 2$ precursors. The symbol $\otimes$ indicates the convolution operation between PDFs, while symbol $\oplus$ indicates the average operation.

The ISI accumulation algorithm, which is sketched in Figure 3.12, firstly determines the contribution of the n post-cursors (the cursor "0"). The ISI accumulation starts from the farthest cursor with respect to the "0" cursor (e.g. cursor "2" in Figure 3.12). The ISI contributions of the voltage sample of each one of four transitions at the k-th cursor are expressed by means of a Probability Distribution Function in the form:

$$P_k^{00}(v) = \delta\left[v - V^{00}(\Phi_i T + kT)\right] \tag{3.8}$$

and similarly for the '01', '10' and '11' transitions, which is a Dirac delta function centered at the voltage $V^{00}(\Phi_i T + kT)$. This is the response of the channel to the TX-'00' transition at cursor k centered at phase Φ_i. The ISI contribution of each cursor is accumulated (assuming that all bits are uncorrelated by convolution) between the ISI PDFs [37, 49] but care must be taken since the change of the bit value in the middle of the UI must be avoided, e.g. only a '10' or a '11' transition could follow a '01' transition. For this purpose two temporary PDFs, $P_k^{ISI,1}$ and $P_k^{ISI,0}$, are introduced to accumulate the ISI after operation at the k-th cursor. The k-th ISI accumulation consists in convolution operations alternated by an average operation between PDFs in the voltage domain (indicated by the symbol $\oplus$ in Figure 3.12), i.e.:

$$\begin{aligned} P_k^{ISI,0} &= \frac{P_k^{00} \otimes P_{k-1}^{ISI,0} + P_k^{10} \otimes P_{k-1}^{ISI,1}}{2} \\ P_k^{ISI,1} &= \frac{P_k^{01} \otimes P_{k-1}^{ISI,0} + P_k^{11} \otimes P_{k-1}^{ISI,1}}{2} \end{aligned} \tag{3.9}$$

where also the convolution ($\otimes$ in Figure 3.12) is in the voltage domain.

The same procedure is followed also to determine the contributions to ISI of the m precursors by starting from the farthest cursor ("-2" in Figure 3.12). For each Φ_i, the contribution of all the pre-cursors is convolved with that of all the post-cursors obtaining the final aggregate ISI PDF of the levels '0' ($P_0(V, \Phi_i)$) and '1' ($P_1(V, \Phi_i)$), as shown in Figure 3.12.

The PDFs of levels '0' and '1' at the different phases Φ_i are finally placed side by side to build the eye diagram, which is similar to the one displayed by an oscilloscope or by a Spice-like simulation. Accordingly to eq. (2.11) presented in Section 2.1.3, the final BER is then extracted as a function of the voltage threshold V_{th} used to discriminate between level '0' and level '1' and the phase Φ at which we sample the eye:

$$BER(V_{th}, \Phi) = \frac{1}{2}\left[\int_{-\infty}^{V_{th}} P_1(V, \Phi)dV + \int_{V_{th}}^{+\infty} P_0(V, \Phi)dV\right] \tag{3.10}$$

By contour plotting $BER(V_{th}, \Phi)$ we obtain the so-called *statistical eye*, i.e. the points in the (V_{th}, Φ) plane with the same BER, meaning the eye aperture at a desired BER level.

3.4.1 Validation

The methodology has been tested by comparison with circuit-level simulations (Titan [56]) on a system composed by a differential AC-coupled CMOS transmitter working at 2.5 Gb/s and various channels. Figure 3.13 shows the block diagram of the simulated circuit. A Pseudo Random Binary Sequence (PRBS) generator is used as a source of serial binary data at 2.5 Gb/s. The transmitter is divided into two sub-blocks: a pre-driver and an Off Chip Driver (OCD). The latter is the block which is directly driving the output pads while the former has the general task of generating the driving signals with the proper strength for the big p-MOS switches in the OCD. The simulated link adopts AC coupling (see tha capacitor C_{dec} in Figure 3.13). The parasitic effects of bonding wires between output pads on silicon and chip package are included as simple lumped inductor L_{bond} of 2 nH. The channel S-parameters are included in the simulation using a Linear N-Port device, as previously shown in Figure 3.6. Finally an ideal receiver circuit is considered, using two 50 Ω resistors.

The run time of the transient simulations has been chosen to simulate enough bits to cover the whole length of one complete PRBS sequence, which is equal to $N \times (2^N - 1)$ bits, where N is the PRBS depth. $N = 15$ has been chosen.

The result of the simulations is the differential voltage waveform observed at the receiver side of the channel which is post-processed with MATLAB, in order to construct an histogram $H(V, \Phi)$ of the number of times a voltage V is observed at a phase Φ inside the UI. This H matrix is normalized to 1 for each normalized time Φ thus obtaining a probability distribution function analogous to the ISI PDF calculated with the statistical approach.

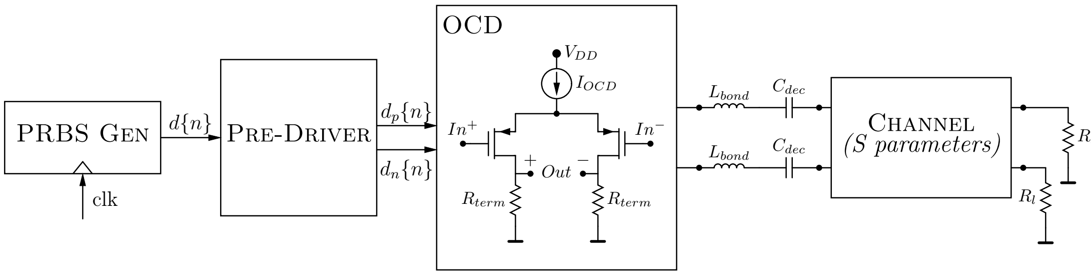


Figure 3.13: Block diagram of the link considered to test the validity of the statistical ISI simulation algorithm. The source of binary data at 2.5 Gb/s is a PRBS generator. The transmitter is composed by the pre-driver and the OCD. Output parasitics are taken into account as a simple lumped inductance L_{bond}. The channel is represented through its 4-port S-parameters by means of a Linear N-Port device, as also done for the channel responses simulation in Figure 3.6.

The comparison between results from our ISI model and the Titan simulations has been done for two different channels whose differential insertion loss S_{DD12} is showed in Figure 3.14. The channel differential S-parameters were measured over the frequency range of 50 MHz – 20 GHz and 10 MHz – 17.5 GHz, respectively for channel (1) and channel (2). These two channels represent two different transmission conditions: channel (1) is an example of a low loss channel with limited link ISI, showing in fact $S_{DD12} = -0.18$ dB at the Nyquist frequency, while channel (2) is an example of a channel with higher loss, showing

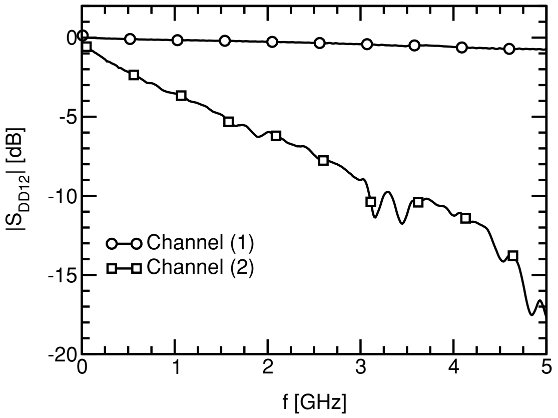


Figure 3.14: Magnitude of the differential transfer function S_{DD12} of the two channels considered respectively in Figure 3.15 and Figure 3.16.

$S_{DD12} = -4.2\,\text{dB}$ at the Nyquist frequency.

Figure 3.15 reports the results obtained considering the low loss channel (-0.18dB at the Nyquist frequency) while Figure 3.16 the results in the case of the high loss channel (-4.2 dB at the Nyquist frequency). As expected the limited ISI introduced by channel (1) results in an wide open eye diagram (Figure 3.15(b)), and the eye closure due to higher ISI of channel (2) is clearly visible comparing the latter with Figure 3.16(b), where the link ISI strongly degrades the eye. Comparing these eye diagrams with the contour plots of the PDFs obtained with the algorithm described in Section 3.4, i.e. Figure 3.15(a) for channel (1) and Figure 3.16(a) for channel (2), we observe a good agreement between the statistical PDF and the Titan simulation in both cases, thus validating the ISI accumulation procedure.

The remarkable advantage of our approach with respect to traditional Spice approaches resides in the small computation time required (a couple of minutes) compared with the many hours required by the time domain circuit simulation of a few hundreds of thousand bits ($N = 15$ has been used, thus the number of simulated bits is approx. 5×10^5). This short simulation time is crucial during the design and evaluation phases of a HSSI transceiver.

3.4.2 Effect of the Edge Steepness

It is important to qualitatively understand the improvements made to the statistical algorithm towards a realistic description of the transmitter pulse shape. In other words, we would like to compare the eye apertures obtained for different shapes of the rise and fall edges of the transmitter waveforms and identify under which circumstances having a realistic description of the transmitter waveform helps in improving the prediction capability of the link simulation tool.

Figure 3.17 shows the three different shapes of the rise and fall edge of the transmitter pulse that have been considered to this purpose. They are, respectively:

1. a realistic TX edge shape as from Spice-like simulations;
2. a trapezoidal transition having a slope at zero crossing equal to the one of the real waveform;
3. an ideally sharp edge transition with $t_{rise} = t_{fall} = 10\,\text{ps}$ that mimics the step function assumed in many approaches available in the literature [33,36,37,49]).

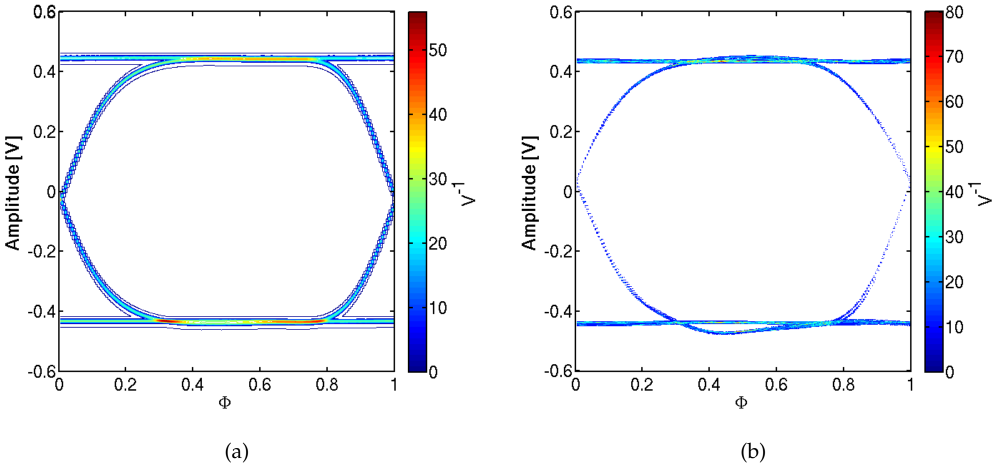


Figure 3.15: (a) Contour plot of $P(V,\Phi) = (P_0 + P_1)/2$ for channel (1) in Figure 3.14 ($S_{DD12} = -0.18\,\mathrm{dB}$ at the Nyquist frequency). (b) Titan simulation of the eye for the same channel. The eye is constructed simulating a PRBS15 sequence of $2^{15} - 1$ bits and the CMOS transmitter at 2.5 Gb/s.

The eye diagram apertures, for a BER level of 10^{-12}, obtained with these three edge shapes in the two cases of the channels adopted in Section 3.4.1 (whose S_{DD12} are plotted in Figure 3.14) are reported in Figure 3.18. Plot (a) reports the results for the low loss channel and shows that there are limited differences between the eye calculated with the real TX waveform and the one calculated with the 'trapezoidal' edges, while the eye produced with the sharp edges is definitely wider than the other two. Plot (b) reports the results for the high loss channel and depicts a different situation, in which all the three eyes are similar. In the latter circumstance, it is possible to affirm that the shape of the edges is not that relevant for the modeling of the link performance because the actual signal transition at the receiver is mostly determined by the link response. On the contrary the former case indicates that the link performance, i.e. the eye aperture, is mainly degraded by the transmitter performance.

The advantage of a realistic TX pulse shape description therefore is visible when considering channels characterized by a low attenuation, typically short channels then, where the bandwidth limitation of the transmitter circuit are still able to manifest at the end of the channel.

3.5 Introducing Jitter in the Model

The modeling approaches available in literature to introduce transmitter jitter in the ISI statistical simulation framework have been described in Section 2.2.3. Among all these approaches, we have decided to use the *Receiver Sampling Distribution* because this approach has the remarkable advantage of being of straightforward implementation into a statistical ISI calculation tool [49]. The disadvantage of this approach is that all the jitter sources are considered as independent from each other, meaning that tracking of the transmitter jitter by the receiver clock recovery circuitry is ignored. In addition to that, also the possible effects of the channel over the transmitter jitter distribution are neglected. Nevertheless, the choice of the methodology must be driven not only by its predictive ability but also by the implementation

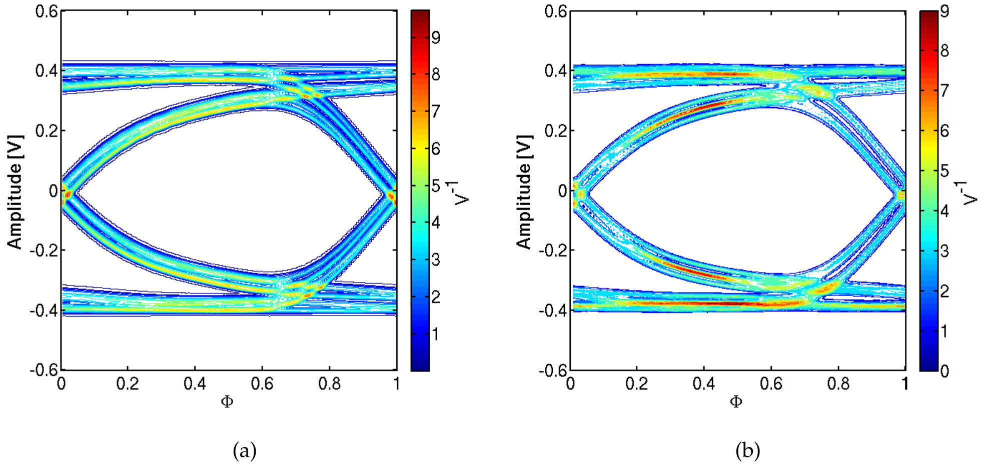


Figure 3.16: Same as above but for channel (2) in Figure 3.14 ($S_{DD12} = -4.2\,\text{dB}$ at the Nyquist frequency).

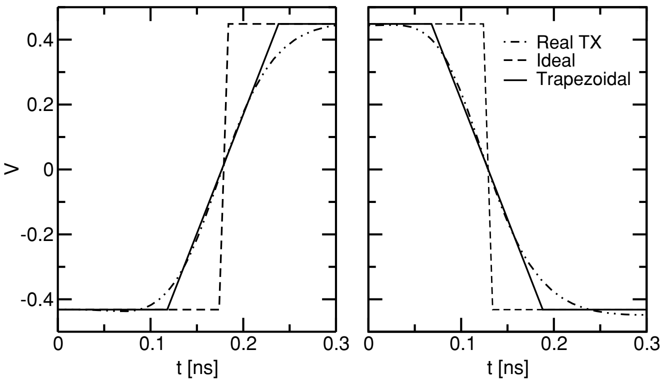


Figure 3.17: Comparison of the three different shapes of the rise and fall edge considered for the simulations of Figure 3.18.

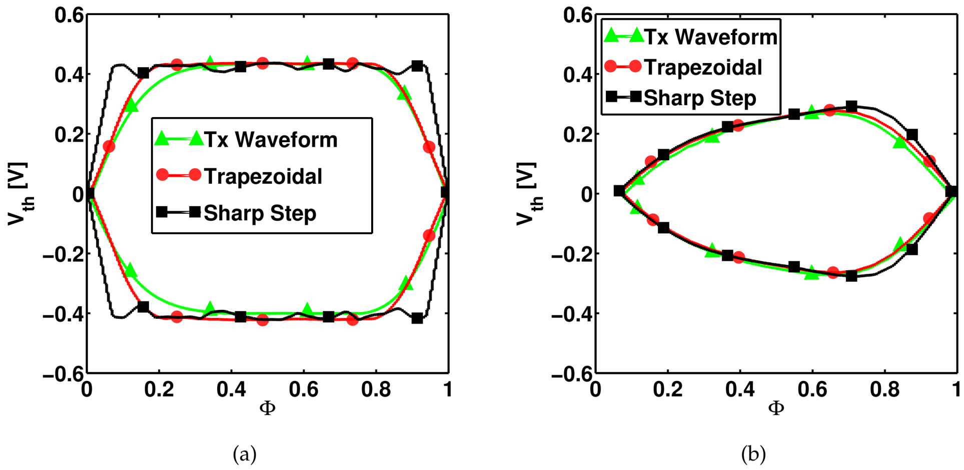


Figure 3.18: Eye aperture corresponding to BER=10^{-12} for different rise and fall edges of the transmitter waveforms (Figure 3.17): real transmitter edges, trapezoidal edges with a slope equal to the transmitter edges and sharp edges with $t_{rise} = t_{fall} = 10\,\text{ps}$. Plot (a): channel (1) of Figure 3.14; plot (b): channel (2) of Figure 3.14.

effort. We have then preferred the Receiver Sampling Distribution with respect to other more accurate but more complex methods because it is robust and of simple implementation.

Given the PDFs of the ISI ($P_0(V,\Phi)$ and $P_1(V,\Phi)$ in Figure 3.12) and the PDF of the jitter $P_{jitter}(\Phi)$, this model gives the combined jitter and ISI PDFs for '0' and '1' by:

$$\begin{aligned} P_0'(V,\Phi) &= \int_{-\infty}^{\infty} P_0(V,\Phi-\tau)\cdot P_{jitter}(\tau)d\tau \\ P_1'(V,\Phi) &= \int_{-\infty}^{\infty} P_1(V,\Phi-\tau)\cdot P_{jitter}(\tau)d\tau \end{aligned} \tag{3.11}$$

Eq. (3.11) introduces the jitter effect by means of a convolution in the time domain. Note that the ISI PDF has to be defined also outside the UI, i.e. for $\Phi < 0$ and $\Phi > 1$, otherwise errors at the two UI edges may be introduced. Since in all practical cases the relation $7\sigma < T$ is valid (i.e. the 10^{-12} tail of the jitter PDF is within a UI), in the implementation of the ISI calculation algorithm we have decided to calculate the ISI considering three consecutive transitions, i.e. three consecutive UIs (see Figure 3.19), and then convolve the ISI PDF thus determined with the jitter PDF. Among all the different definitions of jitter (period, accumulated, long-term, absolute, etc. [41]) the one that enters (3.11) is the absolute jitter because it represents the deviation of actual transition with respect to an ideal clock. We thus assume that the receiver clock is ideal and does not track the transmitter jitter.

The absolute jitter is calculated as [41]:

$$\sigma_{abs} = \frac{T}{2\pi}\sqrt{2\int_0^{\infty} L(f)df} \tag{3.12}$$

where $L(f)$ is the phase noise. Thus:

$$P_{jitter}(\tau) = \frac{1}{\sqrt{2\pi}}\cdot\frac{1}{\sigma_{abs}}\cdot exp\left(-\frac{\tau^2}{2\sigma_{abs}^2}\right) \tag{3.13}$$

Eq. (3.12) assumes that $L(f)$ is entirely due to random noise. Spurious tones in the clock spectrum introducing deterministic jitter components must be accounted for separately, for example using a dual Dirac distribution, as we will see in Chapter 4.

3.6 Validation of the Jitter Model

Verifying the accuracy of eqs. (3.11), (3.12) and (3.13) with the help of time domain Spice–like simulations, as done in Section 3.4.2, is not feasible since, to consider the combined effect of ISI and jitter, transient noise simulations would have to be performed, taking into account at the transistor level all the noise sources of the whole transmitter system, including PLL, serializer and driver circuits. It has been then decided to employ Simulink [57] to perform time domain simulations with the advantage of using different levels of abstraction for different parts of the systems, thus reducing the complexity of the model and the simulation time. As an example, to model the jitter of the system PLL, we have used a behavioral model [58] of the phase error of the clock signal instead of a detailed description of the whole PLL down to the transistor level (i.e. Spice schematic).

The Simulink model is reported in Figure 3.20. The core is formed by the "Channel S Parameters" block which models the differential serial channel by means of the rational function of eq. (3.5) (see Figure 3.21(b)). The clock triggers a PRBS generator of variable length realized as a Linear Feedback Shift Register (LFSR) as illustrated in [59]. The PRBS

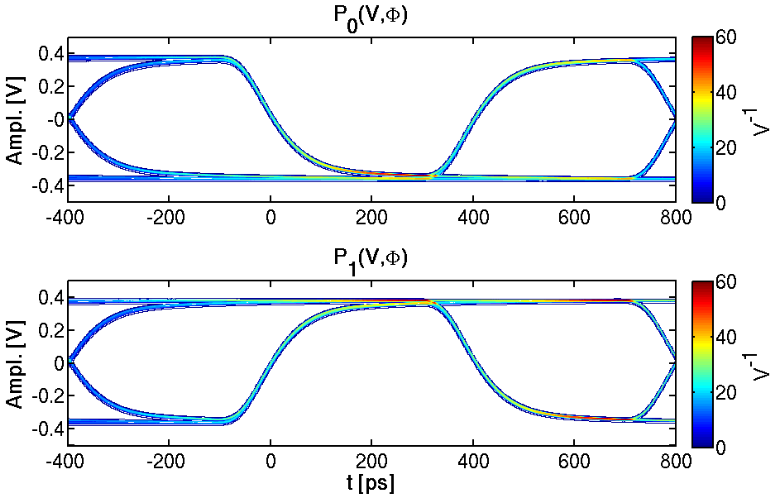


Figure 3.19: Contour plot of $P_0(V,\Phi)$ and $P_1(V,\Phi)$ resulting from the ISI calculation algorithm of Section 3.4. The ISI calculation spans over three UIs, thus three following transitions, to avoid problems when computing the convolution with the jitter PDF (eq. (3.11)) at the UI edges. Same channel as in Figure 3.16.

generator internal structure is showed in Figure 3.21(a) in the case of a PRBS sequence of length $n = 17$. The output levels are then shifted between +1 and -1. To take into account the real rise and fall transient waveform we have used the "Rate Limiter" block, which limits the maximum derivative of the transmitted symbols waveform.

The full-rate clock signal can be ideal or jittered: to produce this signal we have followed the approach presented in [58], where the Power Spectral Density of the random frequency deviation is set to reproduce the realistic Phase Noise Spectrum at the output of a PLL.

The output distribution of the Simulink Random Number Source block used to produce the random phase deviations has been tested over a population of 10^9 samples with excellent agreement with respect to an ideal Gaussian distribution (see Figure 3.22).

Although they can be included in the study, for a sake of simplicity we have not considered the influence of flicker noise components and of spurious tones of the PLL phase noise spectrum.

The output of the model in Figure 3.20 is the time-domain waveform $V_{OUT}(t)$ received at the end of the channel. The vector with all the voltage and time samples is saved as a workspace variable at the end of the simulation and it is then used to construct a two–dimensional histogram $H(V,\Phi)$ which expresses the number of times that a pair (V,Φ) is observed in the received waveform. This H matrix is normalized to 1 for each normalized time Φ thus obtaining a probability distribution function analogous to the combined PDF of ISI and jitter given by eq. (3.11) of our model. In other words, if our model is correct we should find:

$$\frac{P_0'(V,\Phi) + P_1'(V,\Phi)}{2} = H(V,\Phi). \tag{3.14}$$

As a first step we have verified (Figure 3.23) that the results produced by the Simulink model in absence of jitter are in agreement with the results of the procedure for ISI of Section 3.4, as expected according to equation (3.14). Note that in this simple case $P_0' = P_0$ and $P_1' = P_1$.

Jitter has been then introduced in the clock signal of the Simulink model, by specifying a

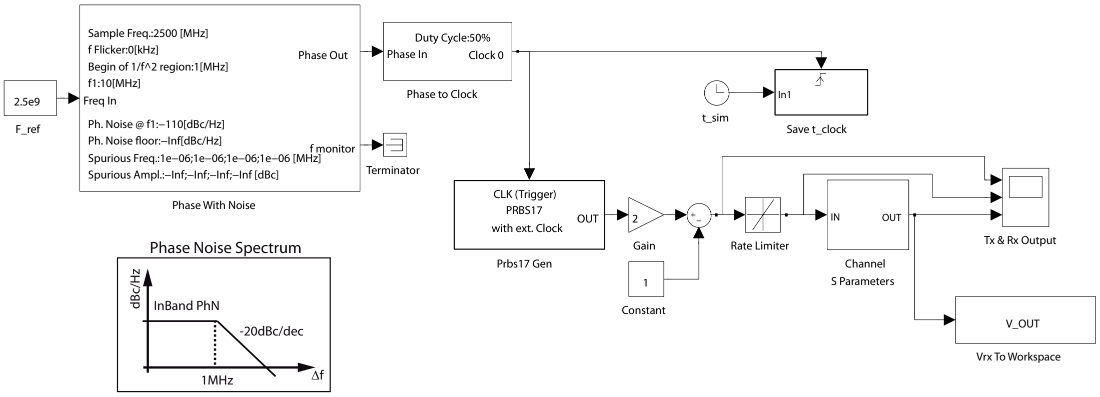


Figure 3.20: Schematic representation of the HSSI implemented in Simulink. The clock driving the PRBS generator is produced by adding to an ideal clock signal a phase noise (mask parameters of the block *Phase With Noise*). The jittered clock is then injected into the differential binary data transmitted and then through the channel. The channel itself is represented by means of eq. (3.5). Output of the model is the time-domain waveform V_OUT observed at the receiver side of the channel. The inset shows the phase noise spectrum.

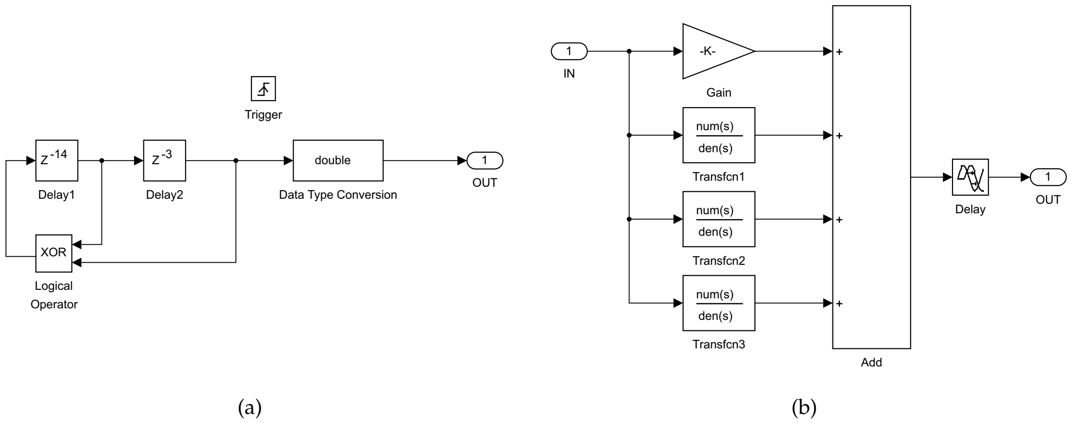


Figure 3.21: (a) Block diagram of the internal structure of the PRBS generator of the Simulink HSSI model and (b) of the channel block implementing the rational function of eq. (3.5). For a matter of space, only the direct term K and three single-pole rational terms are showed.

phase noise in the form shown as an inset in Figure 3.20, that is representative of the phase noise of a typical PLL, namely, it is constant up to 1 MHz and then decreases with a slope of -20 dB per decade.

To extensively test the accuracy of eq. (3.11) of our tool we have then simulated three different data rates, respectively 2.5, 5 and 8 Gb/s. For each data rate we have studied the data transmission for two different phase noise levels of the clock driving the PRBS generator: -90 dBc/Hz and -80 dBc/Hz respectively (see inset of Figure 3.20). For each combination of data rate and phase noise level the received eye has been extracted. Figures 3.24, 3.25 and 3.26 report the results for three of all the tested cases. The very good qualitative agreement between Simulink and our model visible here has been equally obtained in all other tested data rate and phase noise combinations.

Figure 3.27 plots the PDFs at $V = 0$ for the same cases of Figures 3.24 and 3.26. We can see that good mutual agreement is obtained over more than three orders of magnitude.

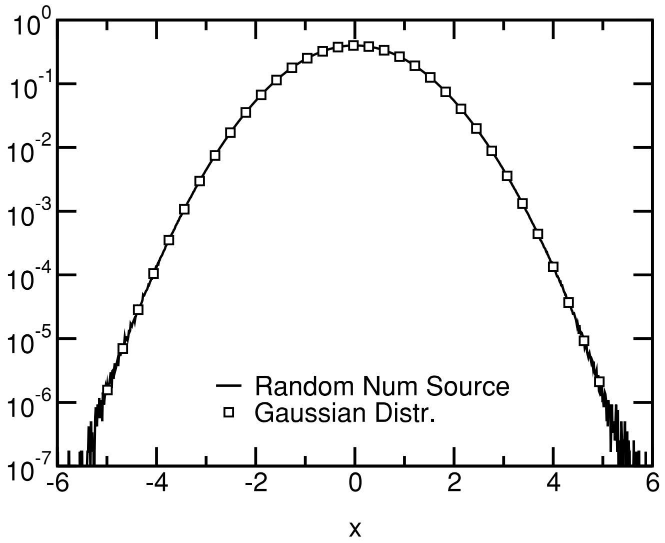
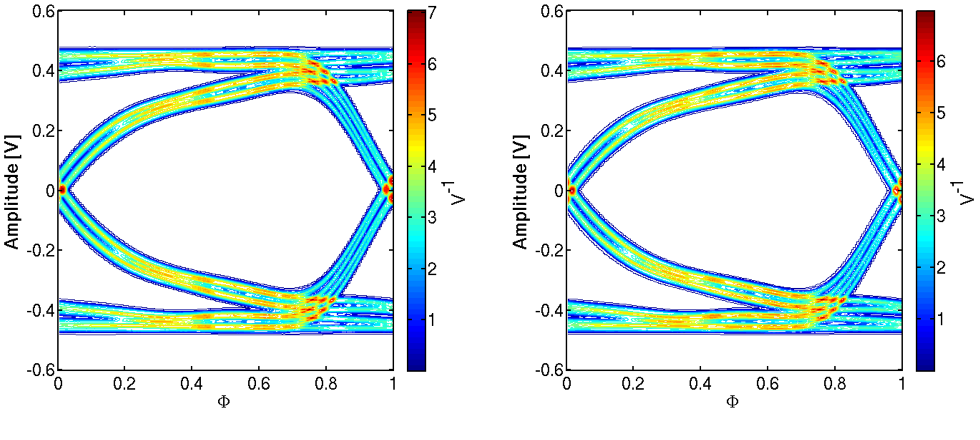


Figure 3.22: Comparison between the distribution of the samples generated by a Random Number block in Simulink and an ideal Gaussian distribution with zero mean and $\sigma = 1$.

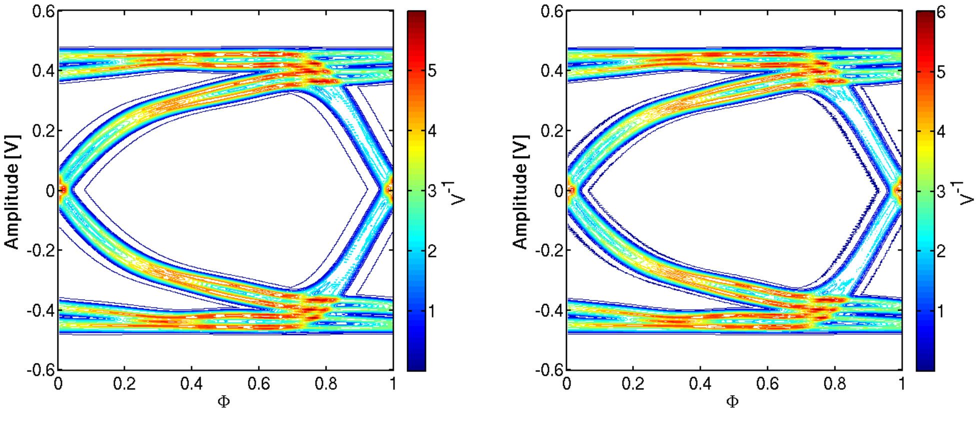
Considering that Simulink calculations took on average 100 minutes it becomes clear that the simulation time to reach the 10^{-12} threshold would be prohibitive even for Simulink.

As a further verification, two Titan simulations of the transmitter driven by a PRBS13 sequence at 1.25 Gb/s and 2.5 Gb/s were run. To account for jitter in the serial data we have generated the time-domain clock waveforms with MATLAB and then used this signal as an input of the Titan simulations. Each rising and falling edge of the clock signal has been jittered according to a Gaussian distribution with zero mean and different σ. The agreement with the results from our approach (see Figure 3.28) is as good as in the case of Simulink. However the computation time of the full Titan simulation is more than 20 times larger than using Simulink and the tails of the PDFs can be explored only over 2 decades as opposed to more than 5 decades in Figure 3.27.



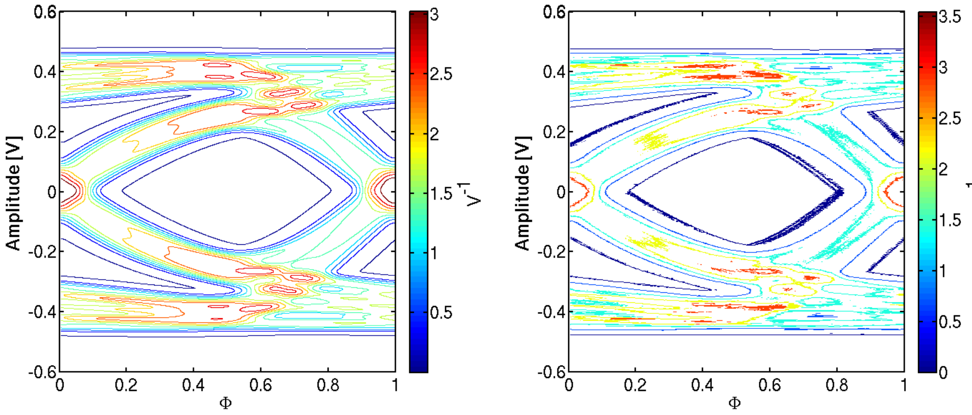
(a) (b)

Figure 3.23: Verification of the ISI model by comparison of the contour plot of the PDFs resulting from our model ($(P'_0 + P'_1)/2$) (a) and from Simulink simulations (H histogram) (b) in the case of an un–jittered clock signal at 2.5 GHz driving the PRBS generator. Since there is no jitter $P'_0 = P_0$ and $P'_1 = P_1$.



(a) (b)

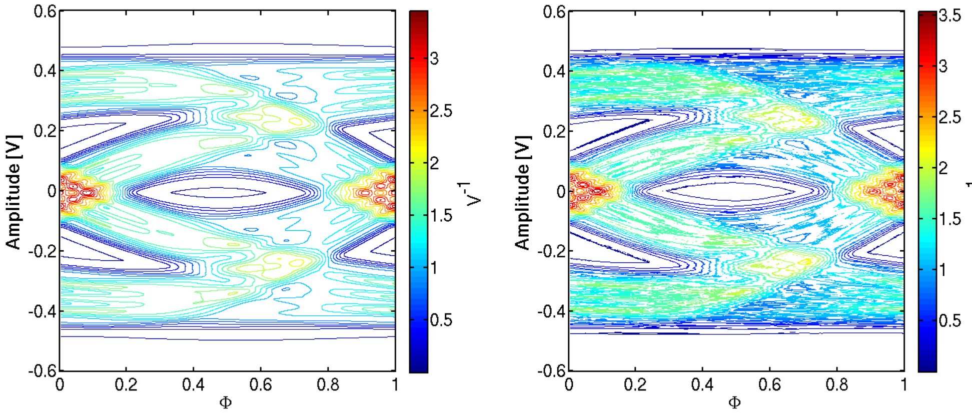
Figure 3.24: Contour plot of the PDFs resulting from our model ($(P'_0 + P'_1)/2$ as from eq. (3.11)) (a) and from Simulink calculations (H histogram) (b) in the case of a 2.5 GHz jittered clock driving the PRBS generator. The Phase Noise spectrum is equal to -90 dBc/Hz up to $f = 1$ MHz and then decreases with a slope of 20 dB per decade. The resulting σ_{abs} is equal to 4.02 ps. The channel has $S_{DD12} = -4.2$ dB at the Nyquist frequency.



(a)

(b)

Figure 3.25: Same as in Figure 3.24 but with a clock frequency $f_0 = 5\,\text{GHz}$. The Phase Noise spectrum is equal to $-80\,\text{dBc/Hz}$ up to $f = 1\,\text{MHz}$ and then decreases with a slope of 20 dB per decade. The resulting σ_{abs} is equal to 6.3 ps. The channel has $S_{DD12} = -11.2\,\text{dB}$ at the Nyquist frequency.



(a)

(b)

Figure 3.26: Same as in Figure 3.24 but with a clock frequency $f_0 = 8\,\text{GHz}$. The resulting σ_{abs} is equal to 1.25 ps. The channel has $S_{DD12} = -11.2\,\text{dB}$ at the Nyquist frequency.

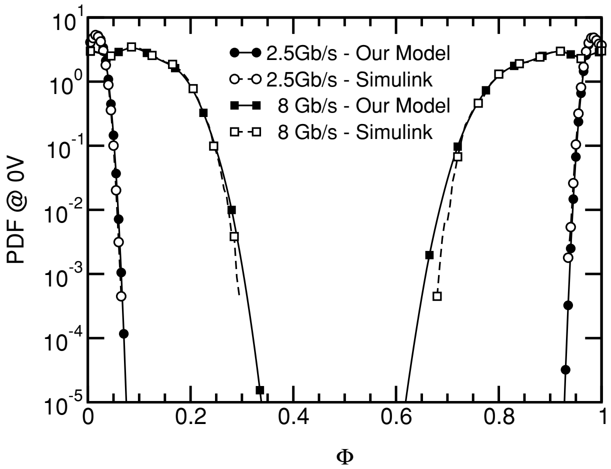


Figure 3.27: Comparison of the PDFs at $V = 0$ for the same cases as in Figures 3.24 and 3.26. Excellent agreement with the reference Simulink model is observed.

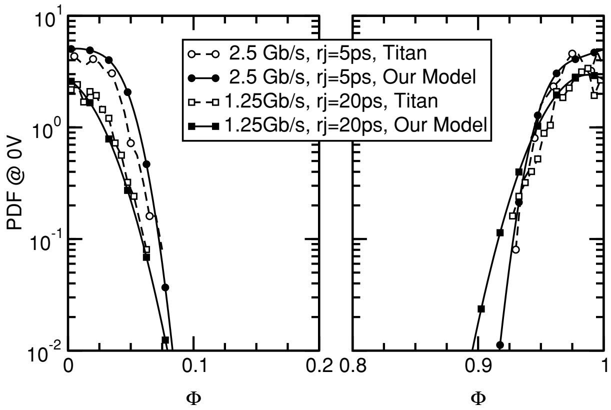


Figure 3.28: Comparison between the PDFs at 0V extracted from two Titan simulations (at 1.25 Gb/s and 2.5 Gb/s) of a transmitter driving the channel with a PRBS13 sequence and as obtained with our model. The clock signal in the Titan simulation is jittered according to a Gaussian distribution with zero mean and two different σs corresponding to $rj = 5$ ps and $rj = 20$ ps.

Chapter 4

Experimental Verification

In the previous chapter we have described in detail the principles of the developed modeling framework for ISI and jitter. We have also shown how the ISI algorithm has been tested with the help of Spice-like simulations, and the jitter model by means of Simulink simulations. The agreement unveiled by this comparison is more than satisfactory.

Despite the importance of these tests, the key element for the predictive capability of our simulation method tool is the comparison with real world systems. For this reason the present chapter will describe the experimental activity focused on measuring the performance of a high-speed link and will report on the comparison between the results of the measurements and the simulations performed with our ISI and jitter simulation framework.

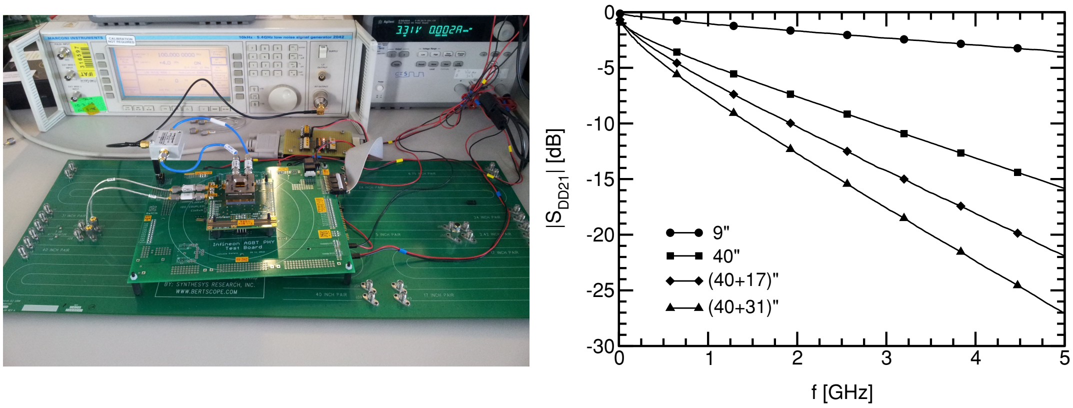
4.1 Test System

As a test system we have considered a high speed link consisting of a CMOS differential transmitter and various backplane channels. The available data rates are 2.5 Gb/s and 1.25 Gb/s. To study the impact of ISI on the transmitted data a BERTScope Differential ISI Board [60] has been used as physical media. This special purpose test board implements differential backplane channels with variable physical lengths from a minimum of 2.42" up to 40". While all the available channels have been experimentally characterized over frequency, just two of these channels have been considered in the study, specifically the 9″ and 40″ channels. In addition, to better test the developed simulating framework in presence of even higher signal attenuation than that provided by the 40″ channel, the cascade of the 40″ channel with respectively the 17" and 31" channels has been considered too. For these four channels the 4-port S-parameters have been measured in the 12.5 MHz – 20 GHz frequency band with a Vector Network Analyzer. Their differential insertion loss S_{DD21} is reported in Figure 4.1(b).

4.2 Transmitter Characterization

The transmitter used for this set of measurement is implemented in CMOS technology. The silicon chip is bonded on a general purpose ceramic package and connection of the chip to the test board is done using a solder-less spring-pin socket.

The first step in transmitter characterization is the measure of the single pulse waveform at the board output connectors using a 20 GHz oscilloscope. To this purpose we have forced the transmission of a repeating pattern of 20 bits: 19 ′0′ and only one ′1′.



(a)

(b)

Figure 4.1: (a) Picture of the high speed link considered as test system. At the bottom the BERTScope Differential ISI Board that implements the backplane channels is visible. Image courtesy of Infineon Technologies AG. (b) Magnitude of the differential insertion loss S_{DD21} of the communication channels of the Test Board; the labels (40+17)" and (40+31)" denote the cascade connection of two channels.

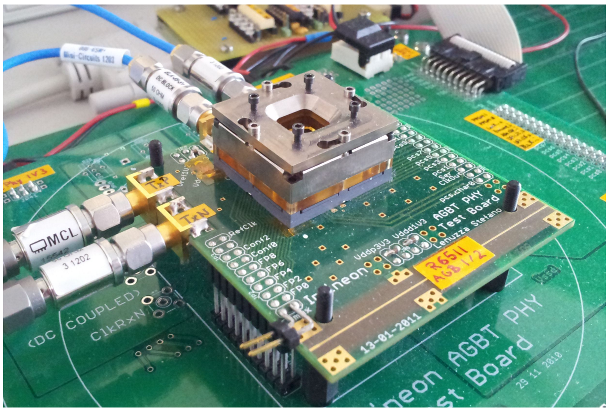


Figure 4.2: Detail of the transmitter part of the link. Clearly visible are the solder-less spring-pin socket and the transmitter test board. Image courtesy of Infineon Technologies AG.

We have then compared the measured waveform to the results of circuit simulations (Titan [56]) including on-chip parasitics as extracted from post-layout simulations and a simple lumped inductance for the bonding wires (L=2 nH) between silicon and package (Figure 4.3). We see that the chip package, the socket hosting the chip and the test board (Figure 4.2), which are not included in the simulation, have a non-negligible impact on the signal that is measured at TX output. In particular a loss of the signal amplitude and the presence of reflections between chip pads and connectors are visible. Unfortunately an accurate characterization and modeling of these transitions would require special purpose test boards and packages which are not available. To mimic at best the actual system, it has been decided to consider as input TX waveform for our ISI tool (Figures 3.4, 3.6, 3.12) the one obtained with Spice and include an additional 2 dB voltage loss when computing the channel responses. The loss value has been determined as the ratio between the peak-to-peak differential voltage swing observed measuring a pulse of eight ’1’ bits and the same differential swing obtained from the Spice simulation of the TX pulse (Figure 4.3). This workaround is expected to affect mostly the simulations for short channels, where the TX pulse is not much degraded by the low pass characteristic of the channel.

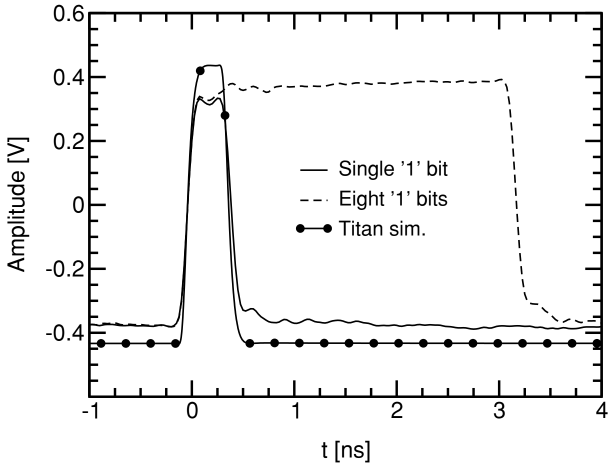
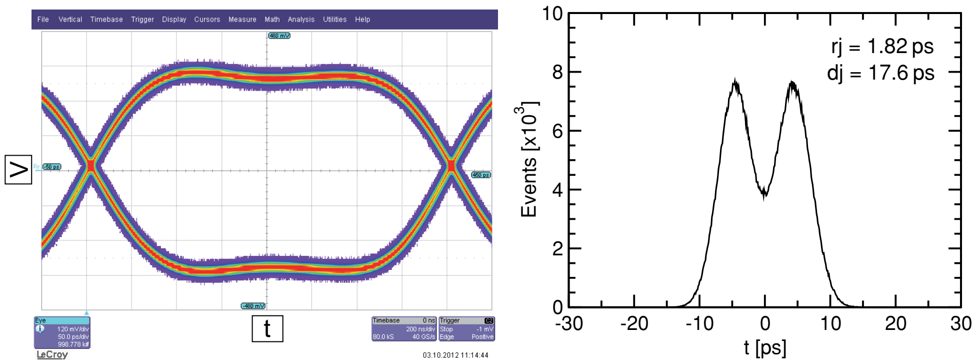


Figure 4.3: Differential single bit pulse waveforms at the output of the transmitter as obtained with a Spice-like simulation (solid line with symbols) and measured on chip (solid line). The Figure also reports the waveform of eight consecutive '1' bits (dashed).

The second step concerned the TX jitter characterization: this has been done by forcing at the output of the transmitter a clock-like pattern of alternating '1' and '0' bits. This allowed to isolate the random and deterministic components of the jitter which are not pattern dependent. The corresponding eye diagram and jitter histogram have been measured with the help of a Serial Data Analyzer. Results are reported in Figure 4.4 for a bit rate of 2.5 Gb/s. The random jitter component is $r_j = 1.82\,\text{ps}$ and the deterministic jitter component is $d_j = 17.6\,\text{ps}$. We have then considered a dual Dirac jitter distribution:

$$P_{jitter}(\tau) = \frac{1}{2r_j} \cdot \frac{1}{\sqrt{2\pi}} \left(e^{-\left[\frac{\tau - \frac{d_j}{2}}{2r_j^2}\right]} + e^{\left[\frac{\tau + \frac{d_j}{2}}{2r_j^2}\right]} \right) \tag{4.1}$$

in place of the Gaussian distribution when introducing the jitter effects in eq. 3.11.



(a) (b)

Figure 4.4: Jitter measurement of the differential transmitter producing a 'clock-like' pattern at 2.5 Gb/s: eye diagram (a) and corresponding jitter histogram (b).

4.3 Comparison with Simulations

Comparison between simulations and measurements focused on the eye diagram and on the eye opening, i.e. the bathtub plot, obtained with the Serial Data Analyzer. These have been compared to the contour plot of the joint ISI and jitter PDF (eq. 3.11) and the bathtub extracted from the statistical BER eye.

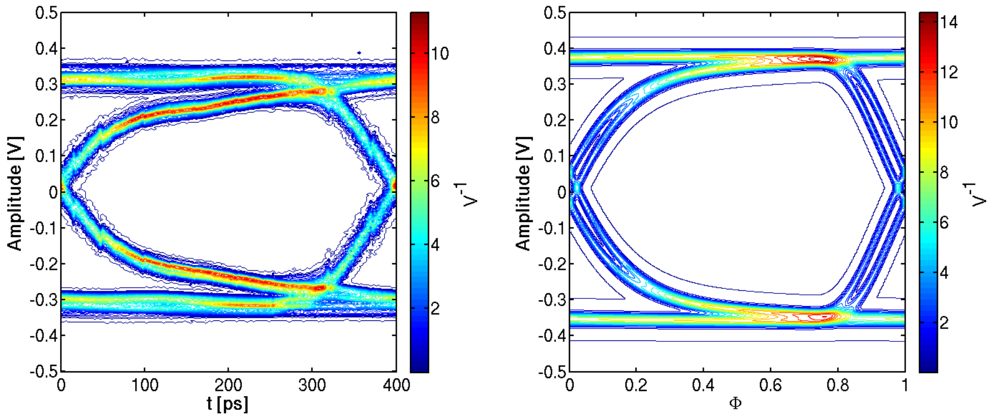
Voltage noise effects have been accounted for by measuring the standard deviation of the persistent trace of the oscilloscope when the output of the TX was turned into a high-impedance condition. The measured value $\sigma_n = 9\,\mathrm{mV}$ has been included in the joint ISI and jitter PDF with the following:

$$\begin{aligned} P_0''(V,\Phi) &= \int_{-\infty}^{\infty} P_0'(V-u)\cdot P_n(u)du \\ P_1''(V,\Phi) &= \int_{-\infty}^{\infty} P_1'(V-u)\cdot P_n(u)du \end{aligned} \tag{4.2}$$

where P_n is the PDF of the voltage noise expressed as a Gaussian distribution with zero mean:

$$P_n(v) = \frac{1}{\sqrt{2\pi}}\cdot\frac{1}{\sigma_n}\cdot exp\left(-\frac{v^2}{2\sigma_n^2}\right) \tag{4.3}$$

and P_0', P_1' are the PDFs obtained with eqs. 3.11 and 4.1. Eq. 4.2 is in fact a convolution between PDFs in the voltage domain, analogous to the convolution in the time domain we have employed to consider the jitter.



(a)

(b)

Figure 4.5: Eye at 2.5 Gb/s measured at the end of the 9″ channel (a) compared with the contour plot of the PDF simulated with our approach ($(P_0'' + P_1'')/2$) (b).

Figure 4.5 compares the measured eye and the statistical PDF for the case of the 9″ channel, while Figure 4.6 does the same for the channel consisting of the series of the 40″ and the 31″ channels. As expected, the longer channel exhibits a better agreement between model and experiments. This is due to the fact that the channel ISI worsens the performance of the link up to a point where the need to accurately model the socket hosting the package and the test board is not critical.

The good agreement between measurements and simulations is also confirmed in Figure 4.7 that compares the measured bathtub for the 9″ and (40+31)″ channels with that extracted

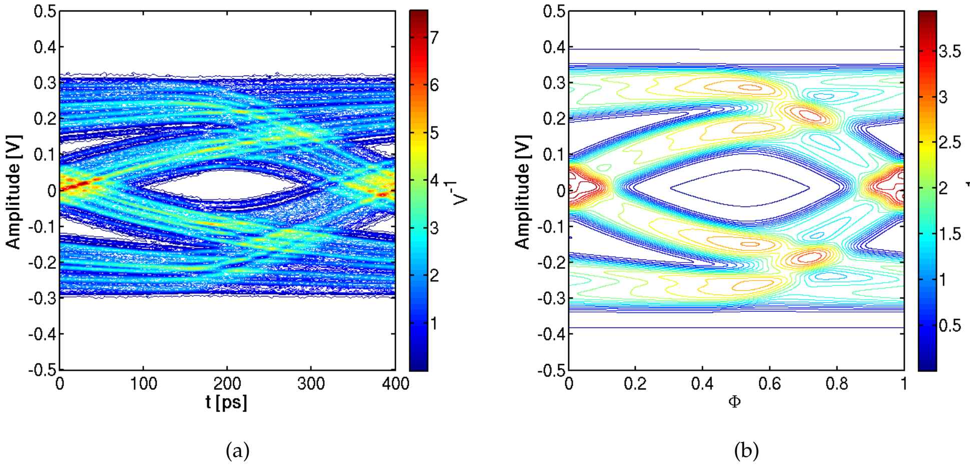


Figure 4.6: Same as in Figure 4.5 but for the cascade of the 40″ and 31″ channels.

from the statistical BER eye (eq. 3.10 with P_0'' and P_1'' from eq. 4.2). Figure 4.8 compares the bathtub opening at a BER = 10^{-12} for all the tested channels at 1.25 Gb/s and 2.5 Gb/s: a very good agreement is clearly visible for all the considered channel lengths, thus validating our modeling approach.

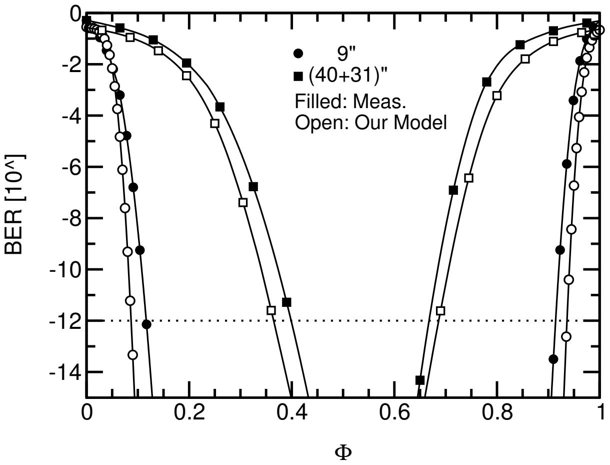


Figure 4.7: Comparison of the bathtubs at 2.5 Gb/s obtained from measurements and our model (eqs. 3.10, 3.11, 4.1 and 4.2) for the same cases as in Figure 4.5 and 4.6.

4.3.1 Data Transmission with De-Emphasis

The comparison of the eye diagram and bathtub has been also performed for data transmission in presence of de-emphasis. De-emphasis (as anticipated in Section 1.4.1) underdrives any consecutive bit of the same value after the first of each transition, with the aim of de-emphasize the low frequency signal content with respect to high frequency content, thus compensating for channel losses. De-emphasis is employed when transmitting over channels with high losses, thus we have considered in these measurements only the channel lengths above 40″.

The inclusion of the de-emphasis in the transition-based statistical algorithm is not trivial,

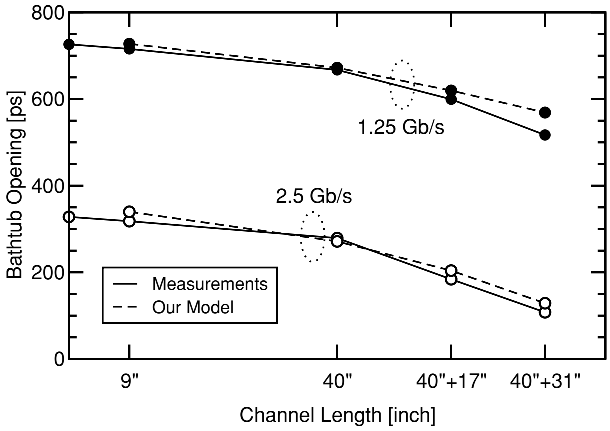


Figure 4.8: Comparison of the bathtub openings at 1.25 Gb/s and 2.5 Gb/s for BER = 10^{-12} obtained from the measurements and simulation over all the considered channel lengths.

because would require handling eight different transitions instead of four as done in Section 3.4. We have then decided to implement a simple SBR-based statistical simulation tool analogous to the one in [37,49] and then implement de-emphasis into it. In this way, we neglect the detailed modeling of the transmitter waveform but the de-emphasis implementation is quite straightforward. In fact, considering that the transmitted symbols are $+1$ or -1, de-emphasis is a linear operation and can be modeled as a FIR filter acting on the binary data. In the Z-transform domain we can represent the relation between the discrete filter output $Y(z)$ and the input symbols $X(z)$ as:

$$Y(z) = [b(1) + b(2)z^{-1}]X(z) \tag{4.4}$$

where the filter coefficients $b(i)$ are calculated from the desired de-emphasis ratio. The transmitter used for the measurements implemented a de-emphasis ratio of -3.5 dB. In terms of voltage swings we have (see eq. 1.1): $V_{deemph-pp} = 0.67V_{pp}$. Then, simple arithmetical calculations provide the FIR coefficients giving this de-emphasis ratio, that are $b = [\frac{5}{6}; -\frac{1}{6}]$.

Figure 4.9 (left plot) shows the NRZ pulse used as channel input in the SBR-based statistical algorithm and the same pulse at the FIR filter output, as it is applied to the channel. Figure 4.9 (right plot) compares the channel responses to the NRZ pulse and to the de-emphasized pulse. Note that the channel response with de-emphasis is the sum of two channel responses to NRZ pulses whose amplitude is weighted by $b(i)$. Therefore linearity is not violated when considering the de-emphasized channel response as input of the SBR-based statistical algorithm.

Figure 4.10 compares the bathtub openings at 1.25 Gb/s and 2.5 Gb/s for the 40″, (40+17)″ and (40+31)″ channels activating the de-emphasis: a very good agreement between measurements and simulations is visible.

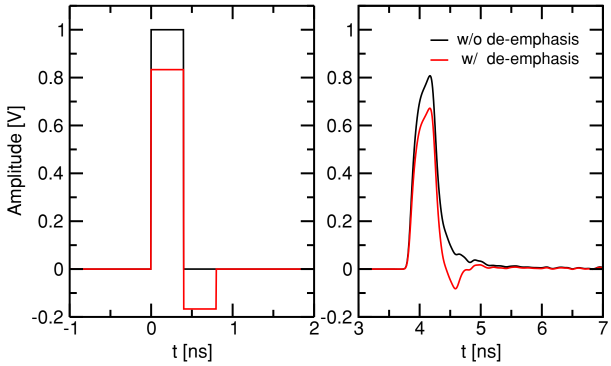


Figure 4.9: Comparison of the 2.5 Gb/s NRZ pulse considered in the SBR-based statistical algorithm and the same pulse at the output of the FIR filter that models the de-emphasis (left). Comparison of the channel response to the two pulses (right). We have considered here a channel with $S_{DD12} = -4.2\,\text{dB}$ at the Nyquist frequency (channel (2) in Figure 3.14).

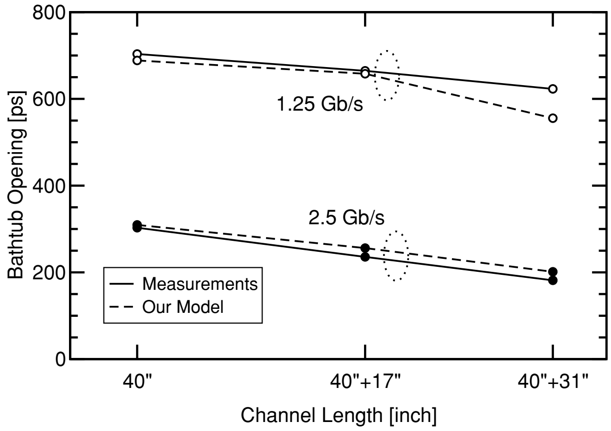


Figure 4.10: Same as Figure 4.8 but with de-emphasis activated.

Chapter 5

Transmitter Architectures for High Speed Links

In this Chapter and the in following, we will examine the design of a novel high speed transmitter for serial links in an integrated CMOS technology. In particular this Chapter discuss general concepts of transmitter architectures and it is preparatory to the next one, where the details of the specific design will be given. Firstly, the two main transmitter architectures for high speed applications and some of the concepts related to transmitter design will be introduced. Then, the state-of-the-art of voltage-mode drivers will be reviewed, by describing the details of the Source Series Terminated (SST) architecture that will be selected for implementation in the next Chapter.

It must be noted that here we will focus exclusively on differential signaling, as it is the technique adopted by all the more recent high-speed standards: PCIe, SATA, XAUI and 10 Gigabit Ethernet. This widespread adoption of differential over single-ended signaling resides in its superior robustness to noise, and represents a relatively straightforward path towards data rates above few Gb/s [21]. In differential links, serial data is transmitted using a dedicated pair of transmission lines: the signals over these two transmission lines are 180° out of phase, and the difference between them at the receiver side is used to recover the transmitted symbol. In this way the signal swing that can be achieved is twice that achievable in the single-ended mode, with a significant improvement in terms of signal to noise ratio. Other important advantages of differential signaling are the decisive reduction of the impact of common-mode noise on the performance of the link and the reduction of noise injection into the supplies. Furthermore, at the receiver side the threshold voltage to discriminate between ′0′ and ′1′ bits can be set at 0 V, thus avoiding the need to generate an additional voltage reference to sample the data. On the other hand, the use of differential signaling has the drawback of increasing the cost, due to the increase in the package, socket and connector pin count. There is also a potential increase of the required silicon area and an additional effort is required in the design phase to check and contain the mismatch and the small asymmetries in the differential signal paths, that can have a large impact on the overall signal integrity.

5.1 Current Mode vs. Voltage Mode Differential Drivers

There are two main ways of implementing high speed output drivers: Current Mode (CM) and VM [11,21].

The CM architecture, sometimes also referred as Current Mode Logic (CML), is schematically depicted in the left plot of Figure 5.1: it consist of a pair of FETs connected in a differential configuration and two resistive loads. The voltage signal applied to the transmission line is generated by steering a fixed current source (I_S in Figure 5.1(a)) over the two termination resistors. The differential input signals, $D+$ and $D-$, should have a large enough swing to ensure that only one side of the circuit is in conducting state at a given time. The fixed current I_S is steered between the two sides of the circuit by the input signals. The current flow creates a voltage drop across the source termination resistor R_{term} on one side of the circuit, while on the other side the output pin is pulled up to V_{DD} due to the absence of current flow. In this way the output signals v_{out} and $\bar{v}_{out}$ toggle between two voltage levels given by:

$$\begin{aligned} V_H &= V_{DD} \\ V_L &= V_{DD} - R_{term} * I_S \end{aligned} \tag{5.1}$$

From the relations above the differential output swing comes straightforward:

$$V_{diff} = 2 \cdot (V_H - V_L) = 2 \cdot R_{term} \cdot I_S \tag{5.2}$$

CM transmitters are frequently employed because they support high data rates and have an inherently low susceptibility to power supply noise. These advantages, however, come along with some drawbacks. The first in order of importance is the poor power efficiency. There are two reasons behind this: first of all, the transmitter consumes static power, as it is based on a constant current source. Secondarily, it only uses a quarter of this current to drive the load [61]. This factor will be demonstrated later in Section 6.2, where power dissipation of CM and VM implementations will be analyzed to help in the choice of the transmitter topology. A limitation comes also in the maximum achievable output swing, because a minimum voltage drop is needed over the current source to assure correct operation in the FET saturation region, and also over the differential FET pair to assure operation in the saturation region. Finally, as the CML output always refers to one of the two power supply rails, a CM driver is unable to support arbitrary DC termination voltages.

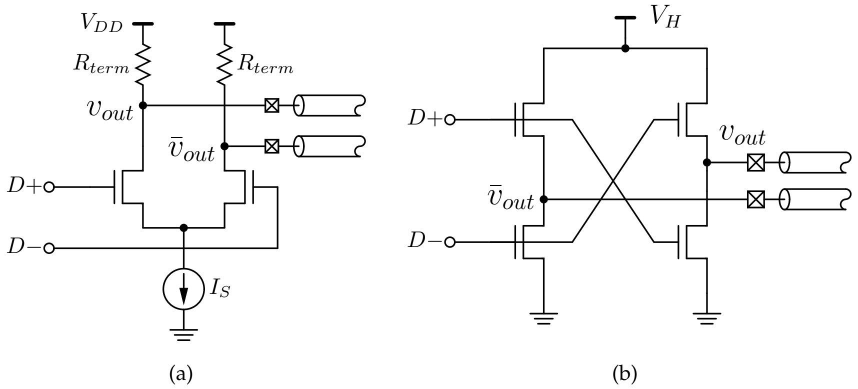


Figure 5.1: (a) Differential current mode simple transmitter architecture. (b) Differential voltage mode architecture [11].

Opposite to CM output stages, a voltage mode driver acts as a switch selectively connecting the line to two voltage references with very low impedance, without the use of a current source. The simplest way to accomplish this function is to use a "Push-Pull" structure, as schematically shown in Figure 5.1(b). A differential VM driver is made by two inverter structures driven by the complementary input signals. Each inverter has only one transistor active at any given instant, thus the output voltage V_{out} toggles between two voltage levels, $V_{O,H}$ and $V_{O,L}$, that can be calculated as:

$$\begin{aligned} V_{O,H} &= V_H - R_{pull-up} I_{out} \\ V_{O,L} &= R_{pull-down} I_{out} \end{aligned} \tag{5.3}$$

where $R_{pull-up}$ and $R_{pull-down}$ are the equivalent output resistance of the pull-up and pull-down branches, that must be matched to Z_0 (usually $50\,\Omega$). The differential voltage swing achievable is thus $V_{diff} = V_H$. We immediately identify here two advantages of VM topology: no static power consumption, as at any instant no direct path from V_H to ground is present, and possibility to achieve high swings due to the fact that V_H can be in principle as high as V_{DD}.

The desired voltage swing at the output drives the implementation of a VM driver. As shown in Figure 5.2 there are two possible declinations of the VM architecture: one employing only NMOS devices (NMOS-over-NMOS) and one employing both NMOS and PMOS (PMOS-over-NMOS). The structure with only NMOS does not allow for high swing operation ($V_H \cong V_{DD}$) due to the fact that the gate-source voltage overdrive of the pull-up NMOS goes to zero when the output node must be pulled up, thus switching off the FET. In this case the complementary structure of the PMOS-over-NMOS implementation must be adopted. On the contrary, when low-swing operation is required, the NMOS-only structure is preferable because in this case the NMOS in the pull-up branch can be much smaller than a PMOS with the same impedance. Nevertheless, the NMOS in the pull-up branch has to be bigger than the NMOS in the pull-down branch due to the reduced voltage overdrive applied to its gate node.

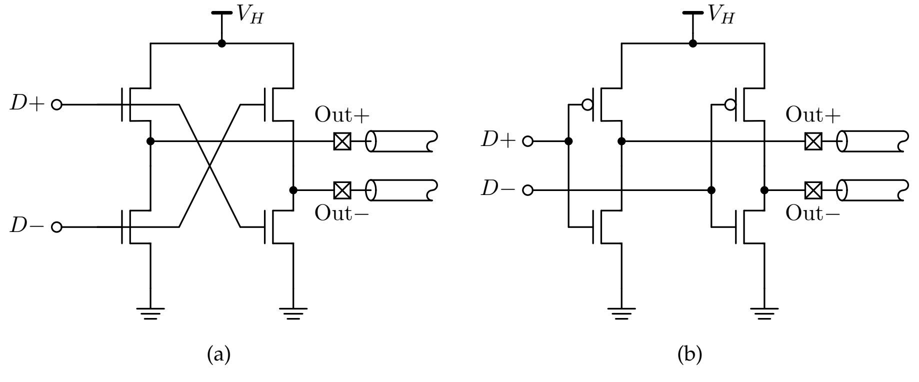


Figure 5.2: (a) NMOS-over-NMOS VM driver implementation. (b) PMOS-over-NMOS implementation [11].

5.2 Termination

One of the most important aspects to consider when designing a transmitter is the output impedance. Limiting the reflections at the transmitter/line interface imposes in fact the output impedance to be matched to the characteristic transmission line impedance Z_0. Thus, given that most of the transmission lines used for high speed links have a Z_0 equal to $50\,\Omega$, each driver must match this value. However the actual impedance value seen by the transmitter is sometimes spoiled by parasitics in the chip/package interface.

In the past, the termination of a transmitter circuit was implemented by means of discrete resistors directly soldered on the board. For example, in a CM driver like the one shown in Figure 5.1(a), the output pads where directly connected to the drain of the FETs and the termination resistors R_{term} where outside the silicon package. While going for external resistors allows for a more accurate control of the resistance value, it increases the cost and represents an additional source of reflections. The connection between the driver chip and the off-chip termination employes a piece of transmission line (stub) across which signal waves travel freely. This determines additional signal integrity issues and threatens high speed capabilities of the link. The methodology of choice is thus the on-chip termination.

At silicon level, the termination can be implemented using resistors, typically poly-silicon resistor due to their better linearity over diffused or well resistors [62], or active devices [11]. In using FETs to make the termination two common structures are mainly used: triode connection or pass-gate. The first structure consists of a triode connected FET paired to a diode connected one (see Figure 5.3(a)): this structure is quite effective in providing a sufficiently linear resistor [11]. The second structure is based on a complementary pass-gate structure (Figure 5.3(b)) in which the NMOS and PMOS gates are driven such that the transistors are in triode region, to make the current-voltage relationship as linear as possible.

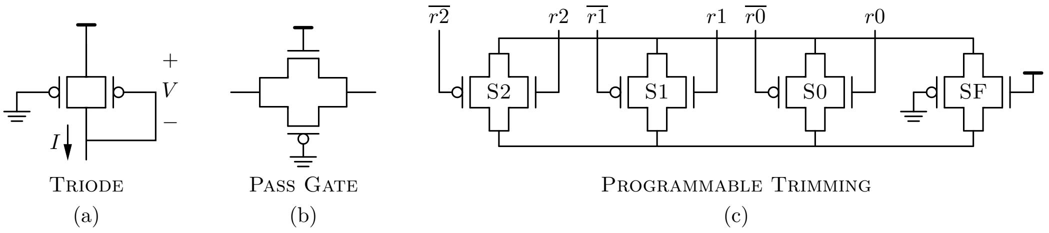


Figure 5.3: (a) Termination resistor implemented as a triode connected FET paired with a diode connected one. (b) Termination resistor implemented as a pass-gate. (c) Termination structure with digital trimming capability [11].

In both configurations the resistance of the active termination is a function of the threshold voltage through the gate overdrive $V_{OD} = V_{GS} - V_t$, as can be easily shown by recalling the FET expression of the small signal resistance in triode region:

$$R_{FET} = \frac{1}{\mu C_{OX}(W/L)(V_{GS} - V_t)}. \tag{5.4}$$

We thus see that variations of the threshold voltage due to fabrication parameters and temperature can affect the resistance of the termination. It is thus common to introduce techniques to adjust the termination resistance value in order to achieve a well-matched termination. Figure 5.3 (c) reports an example of a pass-gate structure implementing a termination resistance with digital trimming capability [11]. A number of pass gates are connected in parallel and selectively activated by the digital code on $r\{2:0\}$. One of the pass-gates in parallel is always active: in this way the trimming range is restricted to values close to the resistance of the fixed element.

Besides the trimming structure described here, one can achieve the regulation of the termination resistance in a number of different ways depending on the particular architecture chosen for the transmitter.

The techniques here described to realize a termination fully apply to the design of a CM transmitter, where the output impedance is fully determined by R_{term} (see Figure 5.1(a)). On the contrary, in VM implementations trimming of the output impedance to the characteristic line impedance Z_0 is achieved by precisely controlling the gate voltage of the FETs (e.g. as done in [63, 64]) or by segmentation of the output driver (as done in [61]) such that the desired impedance is obtained by the parallel connection of a number of drivers with the same topology but with different sizing of the FETs.

5.3 Source-Series Terminated Transmitter Architecture

While CM output stages have been frequently preferred in the past over VM architectures or the design of high speed transmitters, the quest for containing the power consumption, achieve high signal swings and continue pushing to higher data rates has renewed the interest in the VM architectures thanks to their potential for lower power operation and reduced area occupation, as also demonstrated by recent works on VM implementations [63,64].

SST drivers have been recently proposed [65] to exploit the advantages of VM architectures over CM stages and, at the same time, overcome the increasing challenge in achieving acceptable analog performance in aggressively scaled CMOS technologies. The SST driver principle in fact is based entirely on a CMOS design style, with digital switching FETs that are optimized for high-speed operation. As shown in Figure 5.4, the basic topology of a Source-Series-Terminated driver [65–67] is a Push-Pull structure made of a pull-up and a pull-down branch consisting of a FET switch (NMOS for the pull-down and PMOS for the pull-up) in series with a linearization resistor. Pull-up and pull-down branches are designed to match the transmission line impedance Z_0, typically 50 Ω. The usage of the expression "source terminated" arises from the following characteristic: given that the driver already presents an impedance matched to Z_0, it is possible to consider it as *self-terminated* and no dedicated

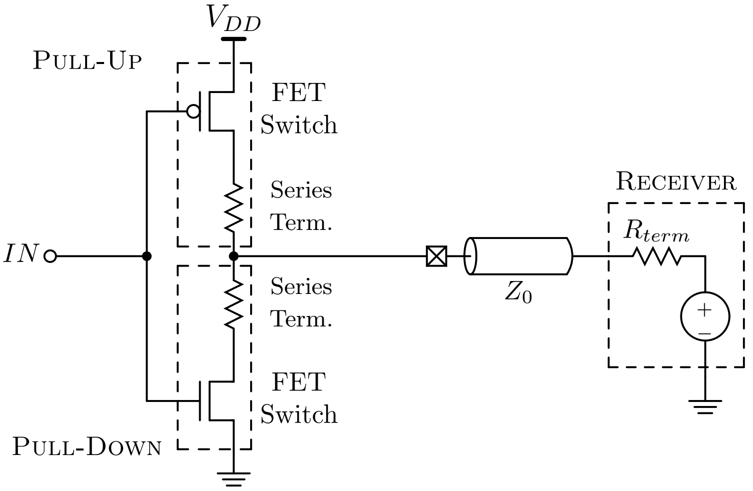


Figure 5.4: SST structure [66].

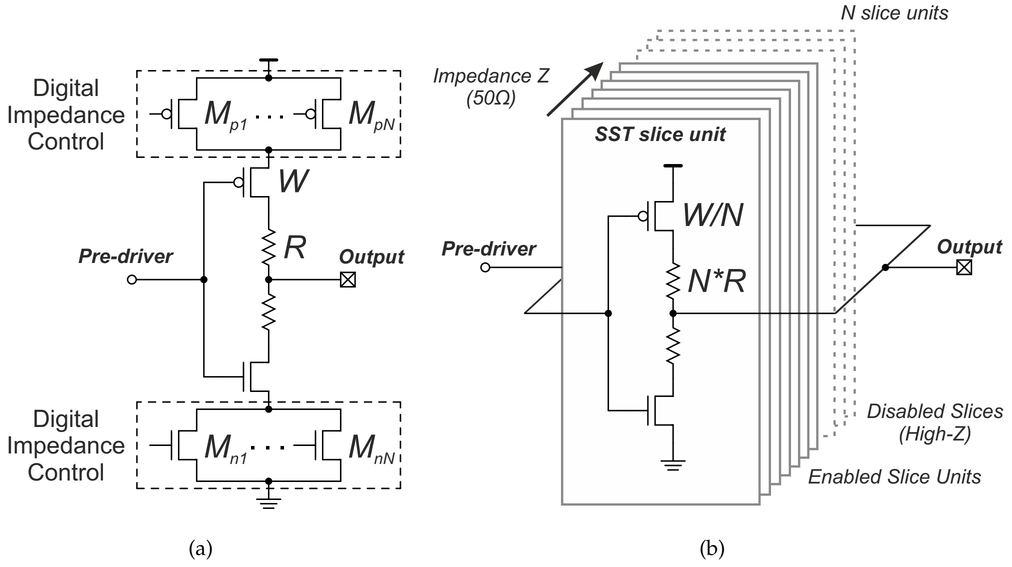
termination structures external to the driver or at the receiver side are needed. Thanks to this, SST transmitters are independent from the topology of receiver and its termination architecture thus making them prime candidates for multi-standard TX implementations.

Furthermore, as the active devices of the output stage are entirely operated as digital switches optimized for high speed, technology scaling can have a beneficial effect on the speed performance of the SST driver.

5.3.1 Impedance Tuning

The necessity to balance process variations which are becoming increasingly relevant in modern nano-electronic technologies imposes the adoption of impedance tuning solutions. Figure 5.5 shows two proposed ways of achieving it [66, 67].

The first concept (Figure 5.5(a)), modifies the SST architecture introducing a set of FETs in series to each branch of the driver. This set of stacked FETs are PMOS for the pull-up and NMOS for the pull-down and are sized using a binary weight criterion. It is possible to control the equivalent resistance of these banks of FETs through a digital word and, as a consequence, adjust the driver impedance to the desired value with a suitable calibration phase. This approach allows for a good impedance tuning granularity around the nominal value. Its major limit is the voltage headroom required to operate four stacked banks of FETs: as the trend towards reduction of supply voltage for scaled CMOS technologies reaches 1 V or even below, implementing four stacked FETs is very challenging and can impose large area penalties to keep satisfactory values of the on-resistance.



(a) (b)

Figure 5.5: Possible implementations of the impedance tuning capability in the SST transmitter architecture [66].

An interesting alternative to this approach is the scheme reported in Figure 5.5(b), which has been proposed for the first time in [65]. In this case, the driver is realized by replicating N times the basic SST structure: the impedance of each single structure, called "slice", is scaled such that the desired overall driver impedance is obtained with a certain number K of identical active slices in parallel. In this way, if the driver impedance is too low, some slices are disabled thus increasing the overall impedance, while if it is too high, some more slices are activated, thus decreasing the driver impedance. The choice of the maximum number (N) of available slices is driven by the worst-case process variation that has to be tolerated. For this reason, during nominal condition some slices are kept disabled with some overhead in terms of silicon area.

This impedance trimming concept has the remarkable advantage that the basic driver architecture is kept simple and small. Moreover, the voltage headroom requirement is much more relaxed compared to the structure reported in Figure 5.5(a) because the slice disable function is operated through the data path signals and not through additional pass transistors in series to V_{DD} and ground. On the other side, drawback of this choice is that the driver strength depends on the number of active slices.

Some considerations must be devoted to the parasitic capacitance at the output node of the driver. Each linearization resistor in the pull-up and pull-down branches comes with its parasitic capacitances between poly-silicon and bulk. Thus these capacitances sum across all N slices of the driver, as also tri-stated slices contribute to the parasitic capacitance. The big metal routing between slice outputs and pads and the ESD protection structures as well contributes to the computation of the total capacitance connected to the output node. These devices are quite large (in order to conduct massive static discharge currents) and contribute by a significant amount to the overall capacitance at the transmitter output. It is clear then that the parasitic capacitance can reach easily very high values of several hundreds of fF. In the small-signal regime, this parasitic capacitance couples with the equivalent driver impedance forming a RC circuit, and has a two-fold effect: from one side it is responsible of transient performance penalties because at each bit transition this capacitance must be charged or discharged with a time constant given by $\tau = RC$, where $R = 50\,\Omega$. On the other hand the capacitance is responsible of a deviation of the output impedance from the ideal $50\,\Omega$ at high frequencies. this deviation can end in the transmitter being strongly unmatched at the frequencies of the data stream. This effect must be faced by the designer through Return Loss estimation and optimization. An interesting Return Loss optimization technique was proposed in [66, 68, 69]: an L-C matching circuit in the form of a T-coil structure [70, 71] has been connected to the differential output of the SST driver. This technique allowed to improve the output impedance matching of the transmitter in the whole frequency range up to the data rate frequency.

5.3.2 Equalization

As anticipated in Section 1.4.1, transmit equalization is a widely employed technique to compensate for frequency-depended channel losses. Performing pre-emphasis is then a must for modern transmitters. In [67] the authors proposed the technique shown in Figure 5.6 (described in more detail in [66]) to implement equalization in the SST architecture. The main idea is to exploit the parallelism concept illustrated in Section 5.3.1 for the impedance tuning to introduce also the equalization capability. As can be seen in Figure 5.6, in [66] the very simple structure of the slice unit has been divided in three subslices, each one implementing a separate SST driver. The target is to operate each subslice driver independently such that some pull-up branches are active at the same time of some pull-downs. In this way, it is possible to obtain at the driver output a voltage level different than the maximum and minimum levels corresponding to all pull-ups or all pull-downs active, respectively. The binary weighting of the subslice driver size is chosen based on the equalization settings.

In the example of Figure 5.6, the transmitter equalization scheme implements a 3-bit

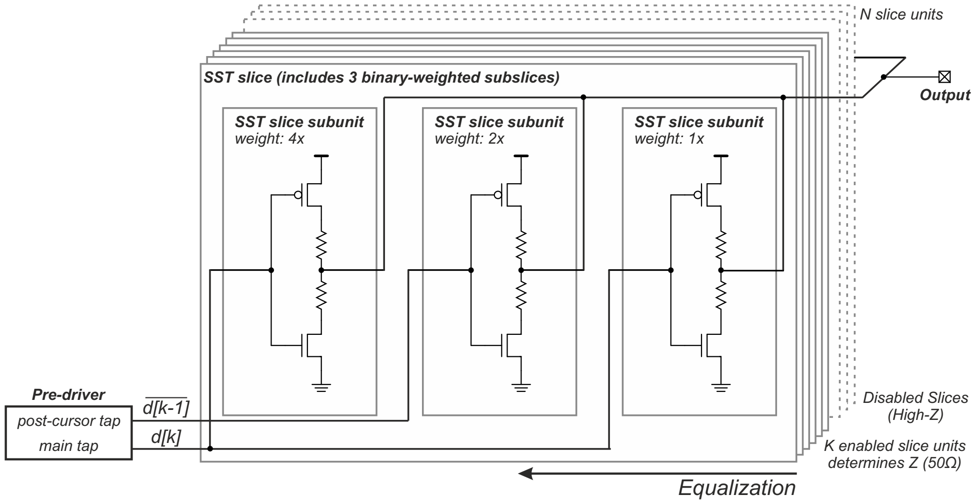


Figure 5.6: Implementation of equalization in the SST transmitter architecture [66]: the transmitter shown here implements a 3-bit amplitude resolution for a 2-tap equalization scheme.

amplitude resolution for a 2-tap equalization. The largest and the smallest subslices are driven by the bit sequence $d[k]$ (that correspond to the main tap) and the middle sized driver is driven by the 1-bit delayed and inverted data sequence $\overline{d[k-1]}$ (that correspond to the post-cursor tap). The unequalized output is obtained after a bit transition in the data sequence: in this case $d[k]$ and $\overline{d[k-1]}$ are equal and all the subslice drivers are pulling the output voltage in the same direction as a unique SST driver with equivalent size of 7× the smallest driver size. On the contrary the equalized output is obtained when the data presents a sequence of two or more equal bits: in this case $d[k]$ and $\overline{d[k-1]}$ are different, and the subslice drivers are pulling the output node to opposite voltages. If, as a matter of example, $d[k] = 1$ and $\overline{d[k-1]} = 0$, the largest and smallest driver will pull the voltage up while the middle-sized driver will pull it down, thus behaving like a SST driver with equivalent size of $(5-2)\times$ the smallest driver size. Therefore the equalized voltage will be $(5-2)/7$ of the unequalized amplitude, thus resulting in an equalization level of (see eq. 1.1):

$$-20 \cdot \log_{10}((5-2)/7) \approx 7.4dB.$$

The choice to implement the equalization introducing a further level of parallelism with respect to impedance trimming has the advantage that the desired equalization setting can be maintained regardless of the number of active slices. In this way, impedance and equalization can be trimmed independently, a characteristic that was not present in early implementations of SST transmitters [65].

5.3.3 Performance comparison of recent publications

To conclude this overview on SST transmitters, Table 5.1 collects some of the most relevant figures of merit for SST transmitters presented in recent publications and compares them to CML implementations. From this comparison it is possible to see that, given similar data rate and voltage swing, in general SST implementations allow for remarkable reduction of the power dissipation per transmitter lane. Another interesting trend that is visible in Table 5.1 is that the high-swing capability of SST drivers entails output swings in the order of 1 V even in the case of the latest technology nodes.

Table 5.1: Performance summary of state-of-the-art SST transmitters and comparison with previous work based on CM architecture.

Ref.	TX Arch.	Technology	Data Rate [Gb/s]	Diff. Eye Height [Vpp]	Efficiency [mW/Gb/s]	Ret. Loss @ $f_0/2$ [dB]
[63]	SST	180 nm	3.6	0.25	2.7	-
[65]	SST	65 nm SOI	16	0.5	3.6	-7
[64]	SST	90 nm	6.25	0.125	0.8	-12
[66, 67]	SST	65 nm	8.5	1	11.3	-16
[72]	SST	45 nm SOI	7.4	0.8	4.32	-
[68, 69]	SST	32 nm SOI	28	0.95	7.75	-
[73]	CML	130 nm	6.4	0.35	15.6	-10
[19]	CML	90 nm	10	0.9	17.4	-
[74]	CML	90 nm	10	1	7	-
[75]	CML	130 nm	8	0.65	20.5	-10
[76]	CML	65 nm	15	0.16	2.3	-

Chapter 6

Design of a High Speed CMOS Transmitter

As happens in all electronic markets, like mobile communications and IT equipment, the struggle for cost reduction and profitability increase is a common factor also in the automotive field. These driving forces impose a continuous improvement of the current products and the development of newer, more efficient and performing ones.

As an application of the concepts, theory and models developed in previous chapters here we report the design of a high speed CMOS transmitter. The opportunity to engage in this challenging practical follow-up of the previous work was given by the competition framework shortly discussed above. In particular, the push towards better performance and connectivity for future generations of microcontrollers specifically dedicated to the control of diverse functions in a vehicle. This new generation of microcontroller will be equipped with a number of different connectivity solutions, one of which will be a high speed serial transmitter, in order to allow for communication with peripherals external to the microcontroller requiring large data bandwidths. With respect to the available implementations this new transmitter design is characterized by a number of innovations and in particular:

New technology node: the push for better energy efficiency and cost reduction of the new products imposed the adoption of a more advanced silicon technology. With respect to the 65 nm node of the current implementation, the new transmitter will be implemented using a 40 nm integrated CMOS technology. Given the very recent adoption of this technology node, the technology is not optimized and thus also the agreement between actual DC and AC characteristics of the devices and compact model predictions is not assessed.

Power dissipation reduction: high speed data transmission demands high current consumption. Any achievable reduction is thus most welcome. Given that the current transmitter implementation is based on a CM architecture, there are margins for improvement adopting a VM topology, as anticipated in Section 5.1.

Performance increase: an improvement of the transmitter speed is desired, overcoming the 2.5 Gb/s data rate capability of the current solution.

In the following the design in the 40 nm technology node of a new transmitter, employing state-of-the-art topology and solution to improve data rate and contain power dissipation, is described. This design has been implemented into a first prototype. Only a reduced set of experimental results is available at the time of completion of this thesis: they will be presented at the end of the Chapter.

6.1 Design Requirements

The main specifications and requirements to be fulfilled by the design of the high speed transmitter are listed here in the following.

Voltage Swing: the differential voltage swing at the output of the transmitter must be not lower than 800 mV peak-to-peak.

Output Impedance: each transmitter lane needs to have a 50 Ω output impedance. This is to match with the standard 50 Ω characteristic impedance of a vast majority of communication channels employed nowadays. The differential output impedance must thus be 100 Ω.

Coupling: to guarantee interoperability with receivers of different vendors, the signaling scheme adopted by the link employing the transmitter requires AC coupling.

Data Rate: a mandatory data rate of 2.5 Gb/s is required. 5 Gb/s capability is not mandatory, but highly desirable.

Power Dissipation: a VM implementation is required, in order to take advantage of the power dissipation reduction capability with respect to the old CM implementation.

De-Emphasis: the data can be transmitted un-equalized or de-emphasized. In the latter case, when multiple bits of the same polarity are streamed at the output, subsequent bits are driven at a differential voltage level of 3.5 ± 0.5dB below the first bit.

Return Loss: Common-Mode and Differential Return Loss are defined as [3]:

$$RL_{TX} = 20\log_{10}\left(\left|\frac{Z_{TX} - Z_0}{Z_{TX} + Z_0}\right|\right) \quad [\text{dB}] \tag{6.1}$$

where $Z_0 = 50\,\Omega$ for the Common-Mode Return Loss (RL_{TX-CM}) and $Z_0 = 100\,\Omega$ for the Differential Return Loss ($RL_{TX-DIFF}$). The corresponding masks are reported in Figure 6.1.

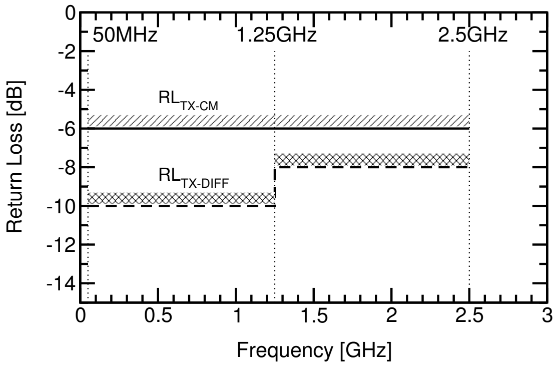


Figure 6.1: Transmitter Common-Mode and Differential Return Loss masks.

The design requirements for the target application specify the supply voltage as $V_{DD} = 1.2 \pm 10\% \mathrm{V}$ and a temperature range from $-40°$ to $170°$. Nevertheless, as for this first prototype implementation only models up to 130° are available, this lower limit will be considered throughout the design.

A few words must be spent here about the design requirement regarding the Electrostatic Discharge (ESD) robustness level. Although compliance to automotive grade specifications for integrated application would require to withstand ESD events up to 500 V – Charged Device Model (CDM) – and 2 kV – Human Body Model (HBM) – [77], the requirements for the transmitter design are more relaxed. In fact, it will be implemented only on a family of products intended as a support for the development phase of automotive applications and not used directly on-board of the vehicle. The required ESD level of robustness is then 250 V (CDM) and 1 kV (HBM). ESD protection of the transmitter output pads is guaranteed by the use of Transient Triggered Silicon Controlled Rectifiers (TTSCRs) [78,79]. The design of these protection devices, as the rest of the discharge protection structures on the chip supplies, has been provided by the ESD group of Infineon Technologies Munich, and has not been object of work of the candidate.

6.2 Choice of the Transmitter Topology

Two topologies have been considered to implement the high speed transmitter: a NMOS-over-NMOS and a PMOS-over-NMOS SST architecture. The decision among these two structures has been driven by the results of a simple study of the voltage levels and FET gate-source voltage overdrives achievable in the two cases. We have also compared these results with a traditional CM architecture. This step has been useful in order to compare the power dissipation of CM and VM schemes and thus for a quick estimate of the achievable reduction in power dissipation.

For the purpose of this study, some simplifying assumption have been considered: a desired differential output swing of 1 V peak-to-peak has been assumed and all the active devices have been considered as ideal switches, thus with null voltage drop between drain and source, and with a fixed threshold voltage $V_t = 0.45\,\mathrm{V}$. As said the supply voltage is 1.2 V. Transmitter and receiver are AC coupled, and each single-ended receiver lane is represented as a 50 Ω resistor to ground, thus achieving a differential receiver impedance of 100 Ω.

Current-Mode Architecture

The traditional CML driver architecture considered as reference is shown in Figure 6.2: since the target application requires a common mode voltage of 0 V the adopted signaling scheme takes the ground level as reference. The current source is then connected to the high supply (V_{DD}) and the differential pair is made using PMOS. The terminating resistor are connected between the drain of the FETs and ground.

Whatever the working condition of the driver (transmitting data or idle) the current drawn by the CM driver is always equal to I_{DD}. The magnitude of I_{DD} is determined considering the current to be supplied to the load (i.e. the receiver) to get the desired differential swing:

$$I_{MAX,load} = \frac{V_{MAX,diff}}{100\Omega} = 5\,\mathrm{mA} \tag{6.2}$$

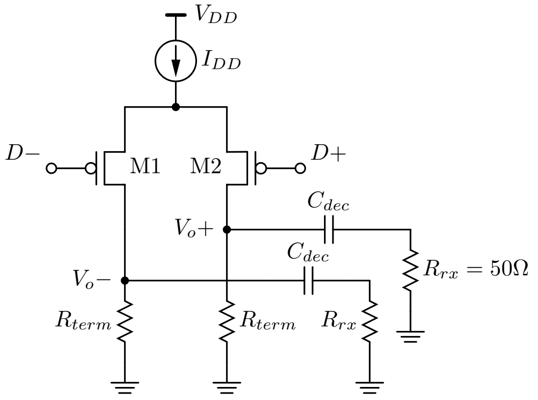


Figure 6.2: Circuit schematic of the CM architecture.

where 100 Ω is the receiver differential input impedance and $V_{MAX,diff}$ is the maximum differential output voltage, which is equal to:

$$\frac{V_{pkpk,diff}}{2} = 500\,\text{mV}. \tag{6.3}$$

Assuming now that the signals $D+$ and $D-$ are such that M1 is off and M2 is on, we have that at the node V_o+ the source current I_{DD} splits between the R_{term} branch and the load. Inverting the current divider relationship we can determine I_{DD}:

$$I_{DD} = \frac{2 * R_{rx} + 2 * R_{term}}{R_{term}} \cdot I_{MAX,load} = 4I_{MAX,load}. \tag{6.4}$$

where $R_{rx} = R_{term} = 50\,\Omega$. As extensively anticipated in Chapter 5, we see here that CM architecture, from the point of view of the current consumption, is not very efficient as only 1/4 of the current drawn from the supply is provided to the load. Thus to get a differential swing of 1 V peak-to-peak the current source must be designed to provide 20 mA.

NMOS-over-NMOS Voltage-Mode Architecture

Secondly we consider a NMOS-over-NMOS VM architecture, which has been slightly modified with respect to the NMOS-over-NMOS architecture of Figure 5.2(a) by adding two terminator resistors R_{term} to ground and to V_{DD}: in this way M1-4 can be operated as digital switches, as it happens in the SST architecture.

As anticipated in Chapter 5, the VM architectures have a null static current consumption: in fact if the driver is in idle condition, either pair M1-M3 or M2-M4 are switched off, thus removing any direct path between V_{DD} and ground.

The current that has to be provided to the load to get the desired $V_{pkpk,diff} = 1\,\text{V}$ is:

$$I_{MAX,load} = \frac{V_{MAX,diff}}{100\Omega} = 5\,\text{mA}. \tag{6.5}$$

This current coincides with the current drawn from the supply V_H. No other current flow is present in the driver. It must be noted that the high supply V_H is not equal to V_{DD}: when M2 is on, we have $V_o+ = 750\,\text{mV}$, thus V_H is equal to 1 V. We see that this architecture imposes $V_{pkpk,diff} = V_H$.

This structure is particularly critical from the point of view of the voltage overdrive for M1 and M2. In fact, the output nodes V_o+ and V_o- toggle during data transmission between 250

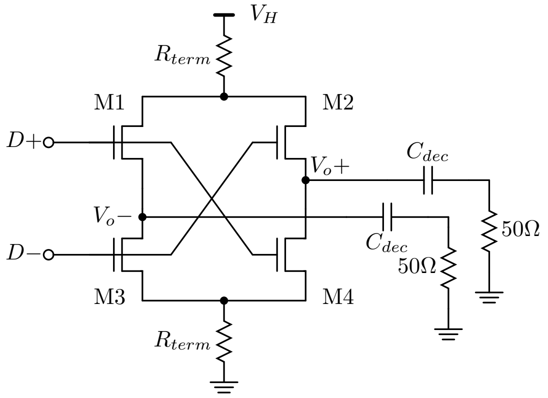


Figure 6.3: Circuit schematic of the VM architecture with NMOS only.

and 750 mV. Assuming that the digital signal driving the gate of M2 has low and high levels equal to 0 and V_{DD} respectively, when M2 is on, the gate voltage is equal to $V_{DD} = 1.2\,\mathrm{V}$ and the source voltage is 750 mV. Under this condition M2 must conduct a current equal to $I_{MAX,load}$ with a gate overdrive of $V_{GS} - V_t = (1.2 - 0.75) - 0.45 = 0\mathrm{V}$, which is clearly not feasible. As expected this structure is thus critical regarding the turn-on condition of the pull-up NMOS.

PMOS-over-NMOS SST Voltage-Mode architecture

Finally a PMOS-over-NMOS SST structure has been considered. The load current necessary to obtain the desired 1 V swing is the same as for the NMOS-over-NMOS VM structure considered in the previous paragraph, as reported in eq. (6.5). Again, no static power consumption is present, as PMOS and NMOS are never active at the same time. As opposed to the NMOS-over-NMOS topology, the structure considered here is optimal regarding the gate voltage overdrive: NMOS and PMOS are always driven with the maximum overdrive possible ($V_{GS} = V_{DD}$ when active). This is quite promising in view of the circuit implementation with real FETs, as it can avoid over-sizing pull-up FETs to counteract the small overdrive, as it is needed in the NMOS-over-NMOS topology.

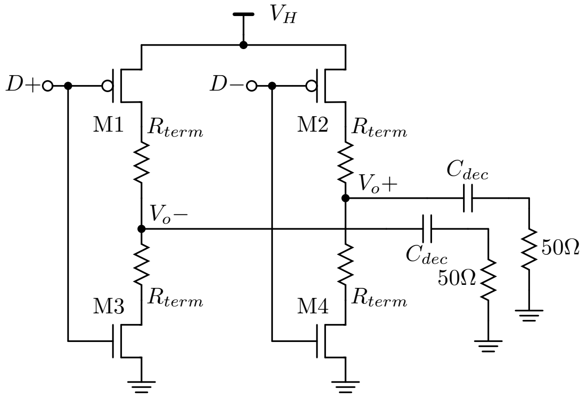


Figure 6.4: Circuit schematic of the VM SST architecture with PMOS and NMOS.

Again, comparing the present topology with the NMOS-over-NMOS one, it is possible to see that in view of the circuit implementation of this architecture concerns may arise about R_{term} resistors implementation. In the previous structure one terminal of R_{term} was always connected to a fixed supply, either ground or V_{DD}, thus allowing for a simple implementation of the trimming capability using a combination of NMOS (or PMOS) and resistors. In the structure of Figure 6.4, instead, both R_{term} terminals are subject to voltage variation from bit to bit; thus, resistance trimming should make use of pass gate structures, which is a less efficient design solution. It is possible to circumvent this difficulty by adopting the impedance trimming concept of [66] (see Section 5.3.1), that splits the driver into N slices that can be conditionally activated to increase or decrease the overall output impedance.

Finally, from the point of view of the protection from ESD of the output circuit, a driver scheme with a resistor between pad and active devices is preferable [77], because it helps containing the overvoltage at the source and drain contacts of the FETs during the discharge phenomenon. This is a further advantage of the PMOS-over-NMOS topology with respect to the NMOS-over-NMOS one as it inherently includes part of the ESD protection structures.

In the light of the reasoning exposed above, the chosen driver architecture is the PMOS-over-NMOS SST topology, as it is the most advantageous from the point of view of power dissipation and switching capability among the considered architectures.

6.3 Transmitter Design

Figure 6.5 shows the complete block diagram of the transmitter.

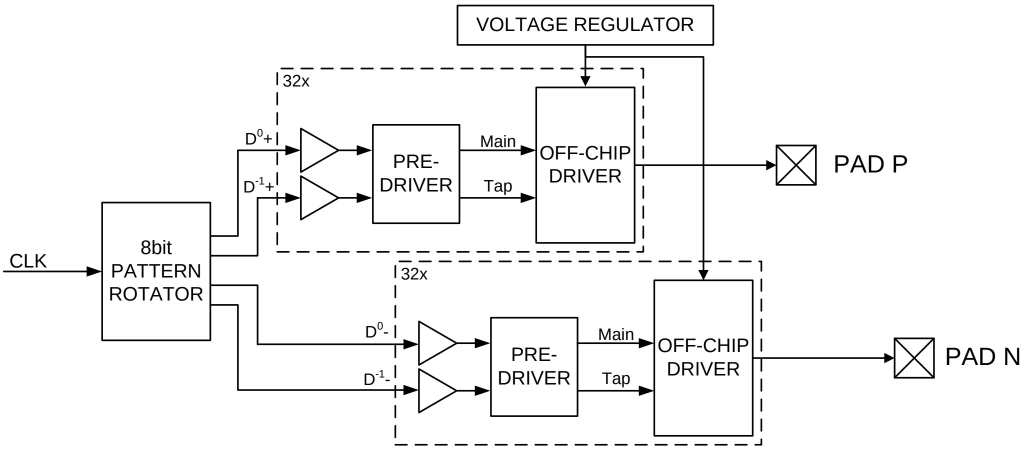


Figure 6.5: Block diagram of the complete transmitter architecture.

It consists of two parallel data paths driving the "P" pad and "N" pad, respectively. From the left side we see that the input of the transmitter is the stream of serial data D^0+ and its 180° out-of-phase replica D^0-. To allow for de-emphasis, the data streams $D^{-1}+$ and the previous bit and its 180° out-of-phase copy $D^{-1}-$ are also needed. All these streams are CMOS compatible: '0' and '1' bits are coded as ground and V_{DD} voltage levels, respectively.

The transmitter inputs are generated by a 8 bit pattern rotator: it consist of a 8 bit shift register in which the pattern rotates at each clock edge by 1 bit. The four signals input of the transmitter are then generated from one of the bits of the register by means of combinatory logic and flip-flops for re-timing.

The core of the transmitter is the Off-Chip Driver (OCD), which essentially consists in the part of the transmitter circuit that drives the output pads, and the Pre-Driver, which has the task of driving the FETs of the OCD with the proper signal levels and power. The transmitter core has been implemented as a bank of 32 basic transmitter slices, each one including one OCD slice and its pre-driver. This structure has been chosen as it best fits with the concept of splitting the overall OCD into slices for impedance trimming (see Figure 5.5(b)), as originally proposed in [66] and described in Section 5.3.1. Further details will be given when treating the design of the off-chip driver (Section 6.3.1).

Each transmitter slice also includes CMOS buffers at its input. This is necessary because the outputs of the pattern rotator are forwarded to 32 slices which represent a huge load, with the risk of unacceptably slowing down the signal switching.

Finally, a voltage regulator is present: this block generates a stable supply voltage for the OCD from the 1.2 V supply.

In the following paragraphs design details will be provide for each transmitter block except for the pattern rotator, as it was not directly designed by the author.

6.3.1 Off-Chip Driver

The first block to be designed is the OCD. As anticipated, the impedance matching concept proposed in [66] has been exploited here. Therefore the first step is to define the number N of slices each transmitter lane will contain. Here $N = 32$, while the number of slices active under nominal PVT conditions is 16. In this way, the equivalent impedance of one slice must be:

$$R_{eq,TX} = \frac{R_{eq,slice}}{16} \rightarrow R_{eq,slice} = 800\Omega. \tag{6.6}$$

Having targeted the equivalent impedance of a single slice, the internal slice structure must be additionally segmented in order to implement the de-emphasis. To this purpose, the approach described in Section 5.3.2 has been followed: each slice of the driver is split into multiple drivers, according the number of desired equalization levels that must be achieved. In this way, when all the subslice drivers are driven with the same polarity, the maximum magnitude of the output voltage is achieved, while if a fraction of the subslice drivers is driven with opposite polarities, the level magnitude is a fraction of the maximum one. This fraction is given by the size ratio between subslice drivers.

Figure 6.6 shows the slice architecture implementing the de-emphasis: since only one equalization level is required, only two independent subslice drivers are needed. The ratio between them is determined from the de-emphasis ratio: to achieve the desired de-emphasis level of −3.5 dB the equalized magnitude of the output voltage must be:

$$V_{deemph-pp} = (10^{-3.5/20})V_{pp} = 0.67V_{pp}. \tag{6.7}$$

This ratio can be achieved assigning to the subslice driver corresponding to the current symbol to transmit (*main* driver) an equivalent size of $5/6$ of the overall slice driver size, and to the subslice driver corresponding to the previous symbol (*tap* driver) an equivalent size of $1/6$ of the overall slice driver size. In this way, when the two subslice drivers are driven with opposite polarities the equivalent driver size is $(5/6 - 1/6) = 4/6 = 0.67$ of the unequalized case.

At this point, the equivalent impedance of each pull-up ($R_{eq,PU}$) and pull-down branch ($R_{eq,PD}$) is determined as:

$$R_{eq,slice} = \frac{R_{eq,PU}}{6} = \frac{R_{eq,PD}}{6} \rightarrow R_{eq,PD} = R_{eq,PU} = R_{eq,slice} * 6 = 4.8\,\mathrm{k}\Omega. \tag{6.8}$$

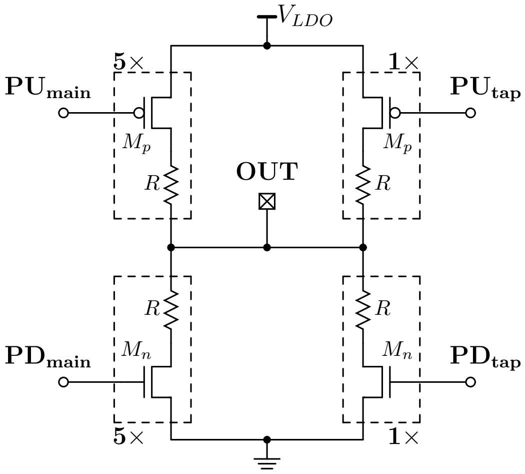


Figure 6.6: Slice circuit schematic of the Off-Chip Driver: the structure is organized as two independent drivers to implement the de-emphasis.

This value is the equivalent series resistance of a polysilicon resistor and a FET. To allow for a convenient modularity of the design, the desired 4.8 kΩ pull-up and pull-down series resistance has been achieved using in both pull-up and pull-down branches a 3 kΩ resistor and consequently sizing NMOS and PMOS to achieve the remaining 1.8 kΩ equivalent resistance under operating conditions.

6.3.2 Pre-Driver

The main task of the pre-driver is to drive the slice FETs of the OCD with adequate signal strength. The second important function is to provide a mean to disable the slice output by forcing the OCD in a high impedance state. For this purpose a tristate pre-driver is needed.

The simplest way of implementing a tristate pre-driver is using logic gates driving the FET gates, as shown in Figure 6.7(a) [11]. When the slice enable signal EN is active, the serial data is forwarded to the OCD. When $EN = 0$, the PMOS gate is forced to '1' and the NMOS gate to '0', switching off both pull-up and pull-down branches and forcing a high impedance condition at slice output. Despite the simplicity of this architecture, it is not the most convenient from the point of view of the total number of transistors required. In fact, AND and OR gates common implementations require both six FETs, a total of 14 FETs are necessary (including the inverter). Switching to logic gates with negated output, thus a NOR and a NAND, that require 4 transistors each, it is possible to reduce the total number of FETs to 10. It is also possible to further reduce this number down to 8 by the adoption of a folded tristate pre-driver topology [11], which is shown in Figure 6.7(b). With this topology the pull-up and pull-down disable function is achieved through M3-6. For $EN = 0$ the pass gate formed by M5 and M6 is open, M3 and M4 are active and forcing a '1' and a '0' to Mp and Mn gates, respectively. When the enable signal is asserted this circuit operates as an inverted with a resistive transmission gate between its two outputs, with the proper driving strength guaranteed by M1-M2 pair. In addition to reducing the transistor count, this circuit helps also in giving a *break-before-make* action. The RC delay of the transmission gate resistance coupled to the gate capacitances causes one of the two nodes to switch later than the other one, thus switching e.g. the PMOS M_p on only after the NMOS M_n is at least partly off. M5 and M6 are therefore sized to trade off delay against pull-up and pull-down overlap current (short

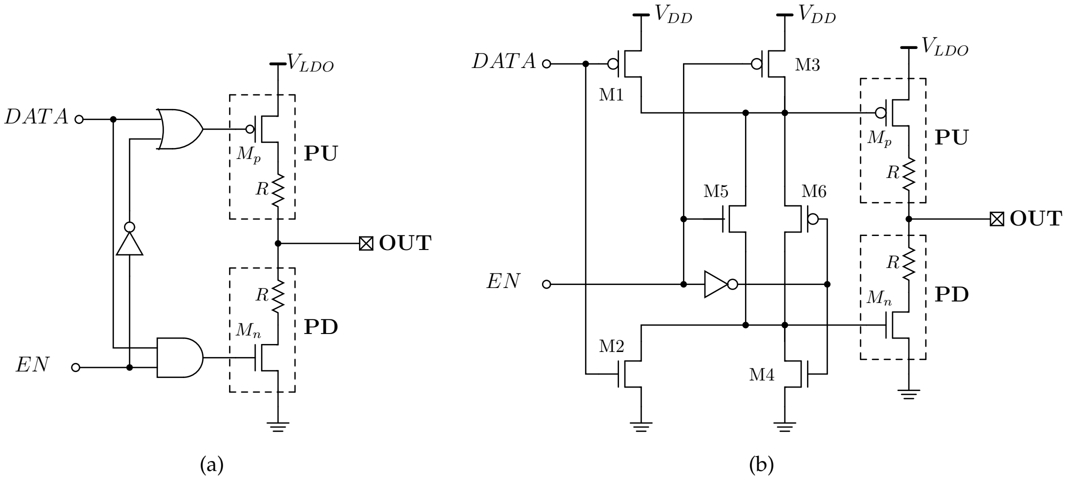


Figure 6.7: The two considered topologies for the design of the pre-driver block: (a) simple and (b) folded tristate pre-driver [64].

circuit current). Given all these advantages the choice for our design fell on the folded tristate pre-driver circuit.

6.3.3 Buffer

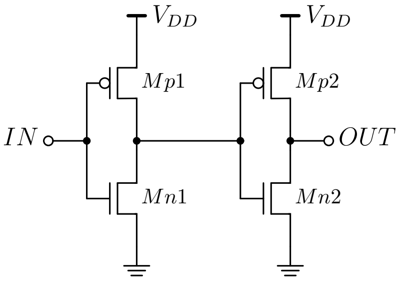


Figure 6.8: Schematic diagram of the simple CMOS inverter buffers providing the required driving strength to the pre-driver inputs.

The buffer stages connecting the pattern rotator to the pre-driver inputs (see Figure 6.5) have a very simple topology. As can be seen in Figure 6.8, each buffer consists of two CMOS inverter stages. In determining the W of all the FETs we proceeded as follows:

- The first inverter stage is of minimum size: W of the NMOS is equal to the W_{min} allowed by the technology, 40 nm in our case.
- The W of PMOS is chosen 3× that of the NMOS.
- A multiplication factor of 3 is also chosen to determine the W of the second stage from those of the first one.

Titan [56] simulations proven that such a design is adequate across all PVT corners.

6.3.4 Voltage Regulator

For the generation of the voltage needed to supply the OCD a Low Dropout (LDO) linear voltage regulator is employed. This is a feedback system that generates a desired voltage from a higher input voltage, in our case V_{DD}, and it is able to operate with a very small drop (*Low Dropout*) between input and output. The advantage of such a circuit resides in the low minimum operating voltage at its input which allows to achieve high regulator efficiency [80].

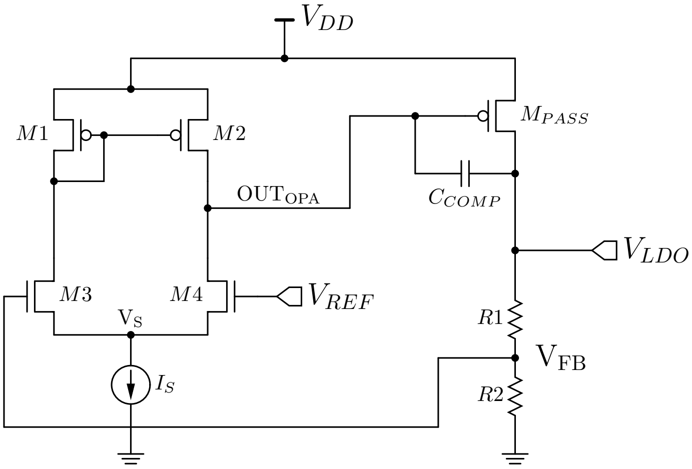


Figure 6.9: Schematic diagram of the LDO voltage regulator.

Figure 6.9 shows the schematic diagram of the LDO. Its main building blocks are: 1) a large pass device M_{PASS} (here a PMOS), 2) a differential amplifier (M1 - M4) and 3) a voltage divider (R1 - R2). One of the inputs of the differential amplifier monitors the voltage signal from the voltage divider connected at the LDO output, thus a fraction of V_{LDO}. The other input comes from a stable voltage V_{REF} that is generated by a bandgap reference circuit [81]. The differential amplifier compares the two input signals, i.e. extracts the error signal, and acts on the gate-source voltage of M_{PASS} to regulate the output voltage until the error signal goes to zero. Given the high forward gain of the differential and M_{PASS} stages, the expression that relates the regulated voltage V_{LDO} to the reference voltage V_{REF} is given by the inverse of the feedback block transfer function (i.e. the inverse expression of the voltage divider):

$$V_{LDO} = \left(1 + \frac{R_1}{R_2}\right) V_{REF}. \tag{6.9}$$

Thanks to the regulating property of the LDO, the output voltage is kept constant in spite of variations in the current drawn by its load.

The first step in the LDO design is the definition of the desired magnitude of the output voltage. Titan time-domain simulations of the OCD and pre-driver showed that the minimum OCD supply voltage that allows for a differential peak-to-peak swing of at least 800 mV at the output across all PVT variations, plus some safety margin, is 900 mV. This is then the voltage that the LDO must provide to the OCD. As a consequence, in nominal conditions the current the LDO must provide is equal to 4.5 mA.

It is then possible to determine V_{REF}: on one side it must be smaller than or equal to V_{LDO} due to the presence of the voltage divider. On the other side, it can not be chosen arbitrarily small because it must allow for an adequate voltage overdrive at the gate of $M4$, also considering that the current source I_S will be implemented by a current mirror that also requires a minimum voltage headroom for operation in saturation. The chosen I_S generator is based on a cascoded current mirror topology, as shown in Figure 6.10. This structure allows to achieve high output impedance of the current source, which has a beneficial effect in limiting the mirror current deviation against PVT variations. The price to pay is a slightly higher voltage headroom. Simulations indicated a minimum voltage at mirror output of 0.2 V. This value, summed to the NMOS $V_{th} \approx 0.45$V leads to calculate $V_{REF} = 0.7$V. The ratio between values of $R1$ and $R2$ is determined from eq. (6.9). To implement it we eventually choose $R1 = 1.5\,\mathrm{k}\Omega$ and $R1 = 5.2\,\mathrm{k}\Omega$

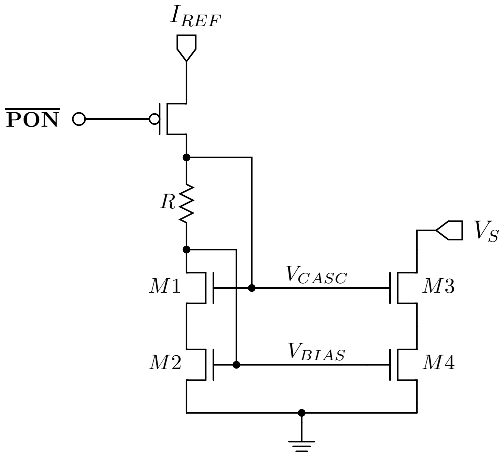
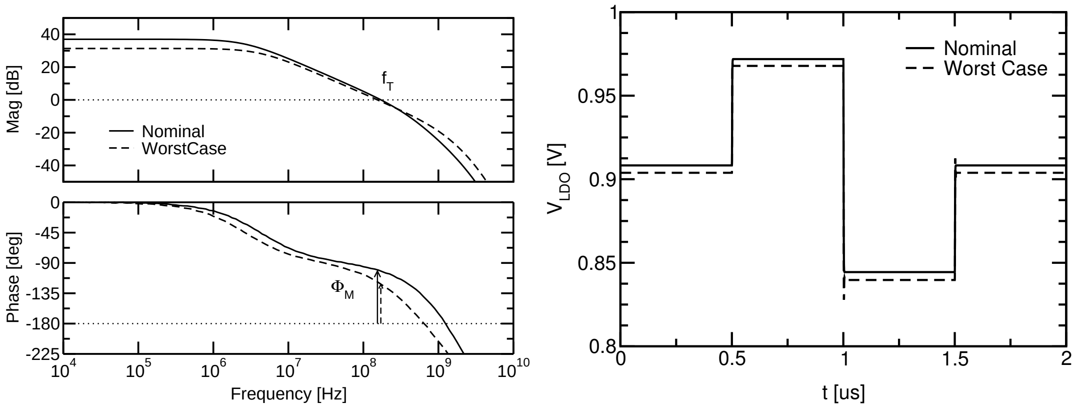


Figure 6.10: Schematic diagram of the cascoded current mirror implementing the current source I_S.

To size M_{PASS}, we considered the condition of minimum dropout voltage between V_{DD} and V_{LDO} and maximum current drawn by the load. In this situation, that occurs when the supply voltage drifts at its minimum (V_{DD} − 10%), M_{PASS} must be big enough to conduct the maximum load current assuring that its V_{DS} will not be bigger than $(0.9V_{DD} - V_{LDO})$. In our case, the minimum drop is equal to $(1.08 - 0.9) = 0.18V$. The maximum current value has been set to 10 mA: this is to allow for a safety margin over the current peaks observed in transient simulations in correspondence of OCD switching. Given these constraints, the width of M_{PASS} has been set to 500 µm and the length to $L_{min} = 40$ nm.

The last step in the LDO design consist in assuring the stability of the feedback loop. For this purpose all possible combinations of minimum and maximum values of the PVT design parameters have been considered. To mimic the condition of the LDO driving the OCD, a resistor load (R_{LOAD}) has been considered. Under nominal PVT condition and assuming the OCD perfectly matched to 50 Ω, the load seen by the LDO is equivalent to 200 Ω[1]. Besides this nominal condition, we have already seen that the current drawn by the OCD could reach 10 mA, which corresponds to a minimum $R_{LOAD} = 90\,\Omega$. On the other hand the maximum value of R_{LOAD} has been assumed to be 400 Ω, which corresponds to a minimum output current of 2.25 mA, to allow for a safety margin in the case of poor receiver 50 Ω matching. The analysis of the open loop transfer function [80] of the system revealed instability problems. These have been compensated inserting the capacitance C_{COMP} in the loop, as shown in Figure 6.9. In this way, the Miller effect can be exploited and the capacitor can be kept small as the capacitance value required for the stability hence the capacitor area are smaller.

[1] In nominal conditions, $V_{LDO} = 0.9$ V and $I_{OCD} = 4.5$ mA. Therefore, at the output node of the LDO the OCD is equivalent to a load resistance $R_{LOAD} = 0.9/4.5 \times 10^{-3} = 200\,\Omega$.



(a) (b)

Figure 6.11: (a) Magnitude and phase of the LDO open loop transfer function, highlighting the nominal and worst-case phase margin Φ_M. (b) Transient simulation of the LDO output voltage in the presence of a 3-step variation of the reference voltage V_{REF}: the absence of oscillation at the LDO output is the proof of robust stability of the feedback. The plots refer to the simulation results under nominal PVT corner and in the corner corresponding to the worst-case Φ_M.

As can be seen in Figure 6.11(a), with $C_{COMP} = 300\,\text{fF}$, a satisfactory worst case phase margin of 60° is obtained ($\approx 90°$ nominal). C_{COMP} has been implemented as a poly-poly capacitor. The Gain Bandwidth Product (GBP) is $> 100\,\text{MHz}$.

As a further confirmation of the stability of the LDO output, a transient simulation has been run to observe the V_{LDO} variation due to a step change of V_{REF}. Figure 6.11 shows the V_{LDO} behavior resulting from these simulations, for both nominal PVT condition and the PVT corner responsible of the worst case Φ_M, which correspond to the case of $V_{DD} = 1.32\,\text{V}$, $T = -40\,°\text{C}$, slow technology corner and maximum R_{LOAD} (minimum LDO output current). As can be seen in the plot, V_{LDO} variations are free from ringing or oscillations, thus confirming the stability of the voltage regulator under non-linear transient conditions.

Figure 6.12 shows a more detailed schematic diagram including the digital signals and the corresponding FET switches needed to implement the power on of the regulator (**PON**), the by-pass of the regulator (i.e. M_{PASS} fully switched on, **VMAX**) and the activation of an additional current path from the LDO output towards ground (**PAR_TEST**). The by-pass function has been implemented as a countermeasure to a possible failure of the LDO design in the prototype: in fact, in such a condition, OCD experimental verification would be compromised. Allowing for LDO by-pass reduces this risk. The activation of the additional current path is needed instead, when testing the LDO with the OCD switched off, thus with zero current drawn at its output. This condition, in fact, is the most detrimental for the stability of the feedback loop and may cause the LDO to become unstable, making impossible the proper experimental verification of the circuit. By activating an additional current path, we can avoid this dangerous situation and then allow for regulator testability under all conditions.

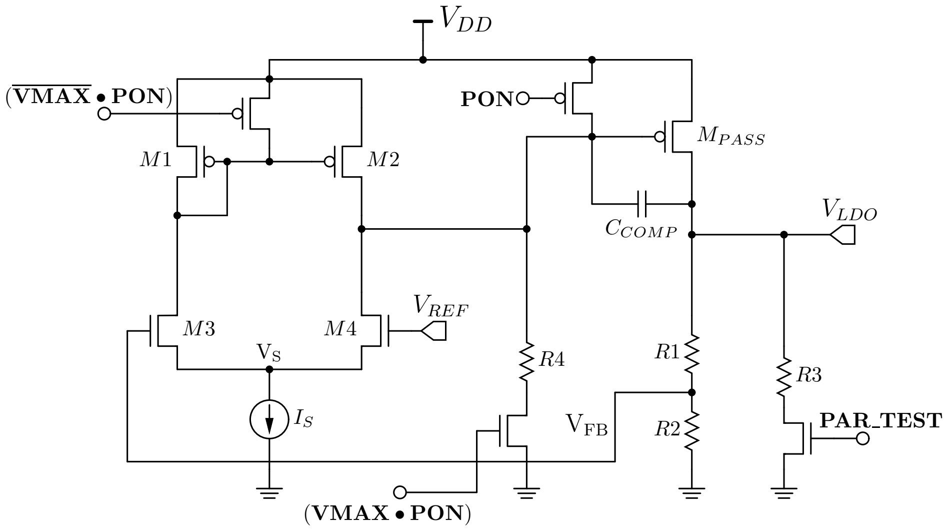
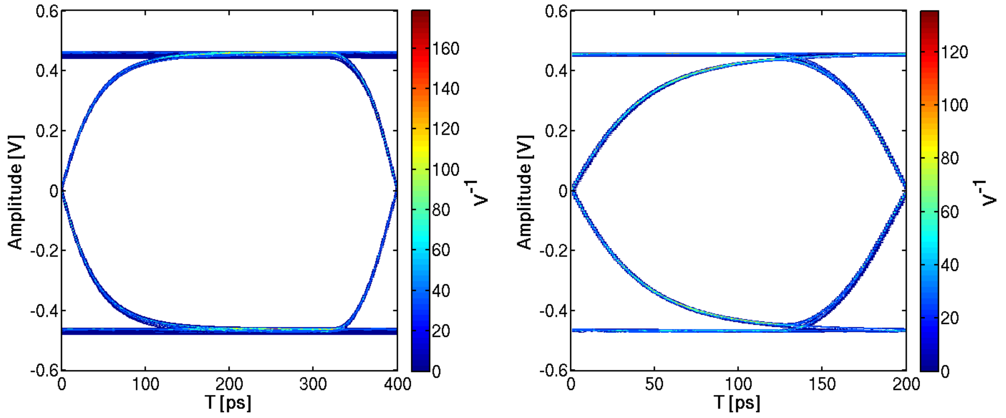


Figure 6.12: Schematic diagram of the voltage regulator including the digital control signals for the power-on (**PON**), the by-pass of the LDO (**VMAX**) and the activation of the test current (**PAR_TEST**).

6.3.5 Eye Diagram

As a final verification of the complete transmitter, time-domain Titan simulations have been run, with the transmitter driven by a PRBS generator to reproduce a random stream of data. The length of the PRBS sequence has been set to $2^n - 1$, with $n = 7$. Figure 6.13 reports the differential eye diagrams at the output of the driver resulting from these simulations, for the 2.5 and 5 Gb/s cases. The transmitter model considered here includes complete post-layout parasitics and a simple lumped elements model for the bonding wires connecting the pads on the silicon chip to the package pins ($L_{bonding} = 0.5\,\text{nH}$). As can be seen in Figure 6.13, both eye diagrams at 2.5 and 5 Gb/s are wide open and guarantee a maximum differential swing greater than 800 mV.



(a) (b)

Figure 6.13: Simulated eye diagram at the differential output of the driver for the (a) 2.5 and (b) 5 Gb/s data rate.

6.4 Comparison with Literature

At this point it is possible to summarize the overall design reported above, and to compare it with similar works available in literature. Table 6.1 reports the most significant figures of merit for a high speed transmitter, i.e. technology node, data rate, differential eye height, efficiency and supply voltage, comparing the work presented in this thesis and three SST transmitters recently published. As a matter of comparison, the performance of two current mode implementations are reported too; one is from the literature and the other one is the previous high speed transmitter implementation available at the Infineon Technologies Design Center. The latter one, a 2.5 Gb/s transmitter in a 65 nm CMOS technology, is reported to better highlight the performance improvement of the work presented here.

Table 6.1: Simulated transmitter performance compared to similar works available in literature. Performance of the previous transmitter implementation designed at the Infineon Technologies Design Center is also given for comparison.

	This Work	Poulton [64] (2007)	Kossel [66] (2008)	Bulzacchelli [68] (2012)	Bulzacchelli [74] (2006)	Previous Implem. (2011)
TX Arch.	SST	SST	SST	SST	CM	CM
Technology [nm]	40	90	65	32/SOI	90	65
Data Rate [Gb/s]	5	6.25	8.5	28	10	2.5
Eye Height [$V_{\mathrm{pkpk-DIFF}}$]	0.9	0.125	1	1.05	1	0.38
Efficiency [mW/Gb/s]	4.5	0.8	11.3	7.75	7	10.4
V_{DD} [V]	1.2	1	1.5	1.1	1.2	1.3

The work developed during the Ph.D. provides a differential eye height that is more than double that of the previous implementation, at double the data rate with a much better power efficiency. The advancement is then quite clear. Comparing now the performance with the literature it is possible to see that while from one side the achieved data rate is in general lower with respect to the state-of-the-art, the power efficiency is quite good if compared with transmitters providing a similar differential eye height. In fact, the best efficiency number reported in Table 6.1 has been achieved with a much smaller eye height.

6.5 Signal Integrity Study

This Section describes the results of a study aimed at understanding the performance of the transmitter when coupled to backplane channels. For this purpose the ISI and jitter simulation tool previously developed during the Ph.D. activity and described in the details in Chapter 3 has been employed.

To study the signal integrity properties of the transmitter we have started from the backplane channel measurements described in Chapter 4. Random jitter effects have been taken into account and, for increasing values of σ_{rj} we identified the maximum length of the channel that it is possible to drive imposing a minimum horizontal eye aperture equal to half of the bit period.

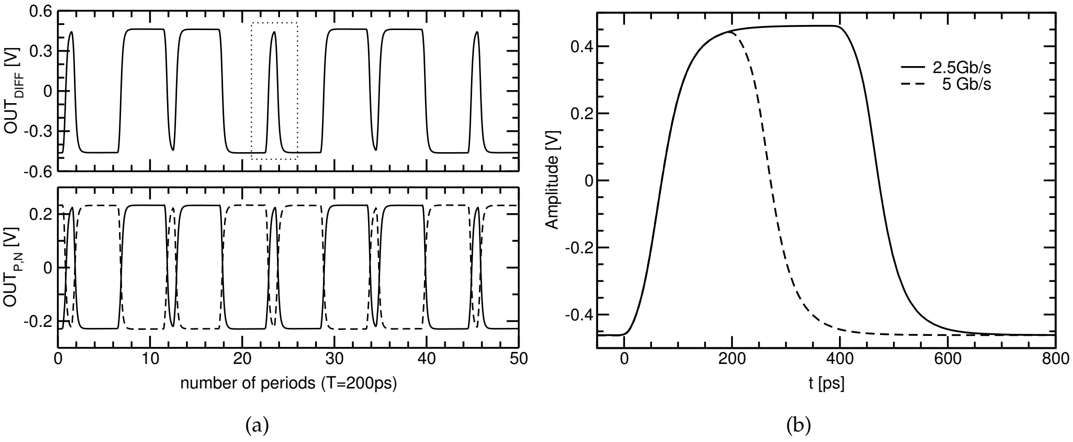
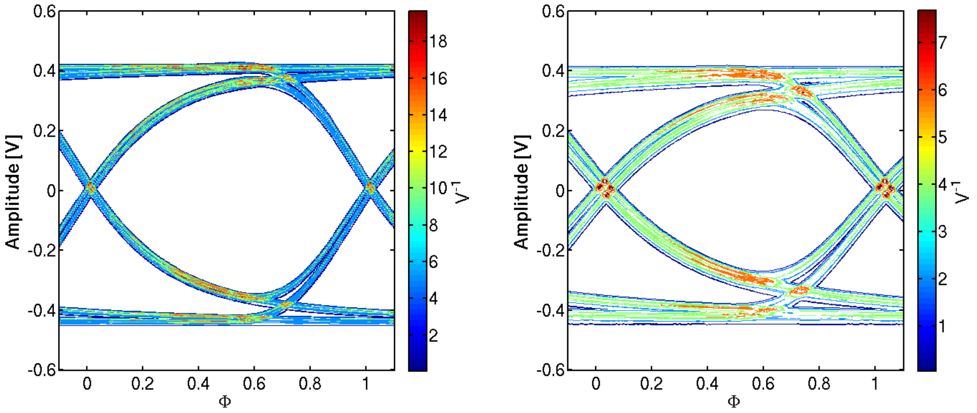


Figure 6.14: (a) Differential (top) and Single-Ended (bottom) output waveforms from Titan transient simulations at 5 Gb/s used to extract the driver single bit pulse. The dashed box indicates the portion of the waveform used as input of our ISI and jitter simulation tool. (b) Magnification of the single bit pulse for both considered data-rates (2.5 and 5 Gb/s).

The transmitter single pulse waveform, which is one of the inputs requested by our ISI and jitter tool, has been determined from a Titan time-domain simulation. The driver is forced to transmit a fixed pattern of 22 bits with the same number of ′0′s and ′1′s, in which a ′1′ bit is preceded and followed by five ′0′ bits, as shown in Figure 6.14(a). This pattern has been chosen because it allows to isolate a single ′1′ bit, and thus extract correctly the single pulse waveform, while at the same time has null DC content as it contains the same number of ′1′s and ′0′s. The latter property is particularly important because the transmitter is AC coupled and the output waveform accumulates the DC content of the data, which results in a slow DC voltage shift in high and low logic levels.

Figure 6.14(b) shows a magnification of the single pulse for the 2.5 and 5 Gb/s data rates, input to our signal integrity tool. Figures 6.15 and 6.16 show the statistical eye diagrams obtained with our approach in the cases of a 5 Gb/s unjittered data stream that goes through 9″, 17″, 31″ and 40″ channels, respectively.

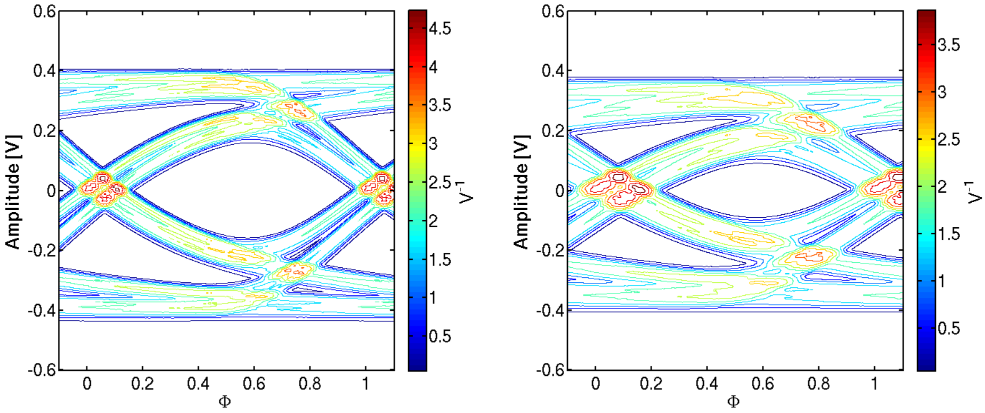
Figure 6.17 summarizes the results of this analysis: the bathtub opening corresponding to BER= 10^{-12} is plotted as a function of σ_{rj} for various channel lengths. For each channel length, the maximum σ_{rj} that can be tolerated having a minimum horizontal eye opening equal to 0.5UI is straightforwardly determined.



(a)

(b)

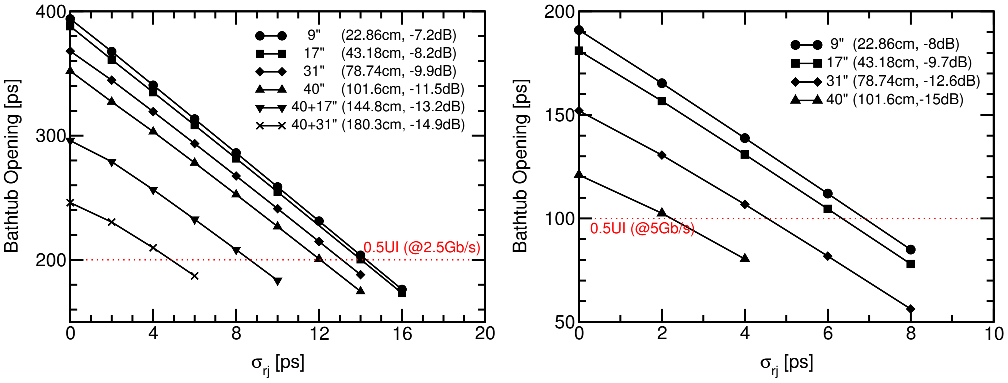
Figure 6.15: Statistical eye diagrams at the end of the channel, obtained with our ISI and jitter simulation tool. The data-rate is equal to 5 Gb/s, jitter effects have not been considered. (a) 9″ channel ($S_{DD12} = -8$ dB @ 2.5 GHz) and (b) 17″ channel ($S_{DD12} = -8$ dB @ 2.5 GHz).



(a)

(b)

Figure 6.16: Same as Figure 6.15 but for (a) the 31″ channel ($S_{DD12} = -12.6$ dB @ 2.5 GHz) and (b) the 40″ channel ($S_{DD12} = -15$ dB @ 2.5 GHz).



(a) (b)

Figure 6.17: Bathtub openings corresponding to BER $= 10^{-12}$, obtained with our ISI and jitter simulation tool, as a function of σ_{rj} for all the considered channel lengths. (a) 2.5 Gb/s and (b) 5 Gb/s. For each channel, the corresponding length in cm and S_{DD12} at Nyquist frequency are reported.

6.6 Experimental Results

The experimental characterization of the fabricated lot of test chips is reported in the following. All the test chips available were fabricated on silicon material corresponding to the nominal process corner.

6.6.1 Voltage Regulator Output

Measurement results of the magnitude of the voltage regulator output are reported in Table 6.2. The agreement with the values obtained from Titan simulations, also reported in Table 6.2 is very good.

Table 6.2: Measurement results of the voltage regulator output compared with simulation results. Measurements include VT variations only.

$[V]$	Measurement			Simulation		
	Min	Nom	Max	Min	Nom	Max
V_{LDO}	0.907	0.911	0.932	0.908	0.913	0.927

6.6.2 Transmitter Output Impedance

Figure 6.18 reports the measured transmitter output impedance as a function of the number of active slices N. The pull-up and pull-down impedance must be measured independently: to do so, we must avoid the OCD to switch between '0' and '1' bits during the measurements. This is possible by setting a pattern of all '1' bits in the pattern rotator: in this way the positive transmitter channel will always output a '1' (pull-up active) and the negative channel a '0' (pull-down active). At the two pads of the transmitter it is therefore possible to measure separately the pull-up and pull-down impedances. The measurement has been carried out with an Impedance Meter by setting the AC stimulus signal to a low frequency value of 300 kHz. The real part of the complex impedance value is then plotted. Measurements were taken over the whole supply voltage and temperature variation range. As can be seen in the

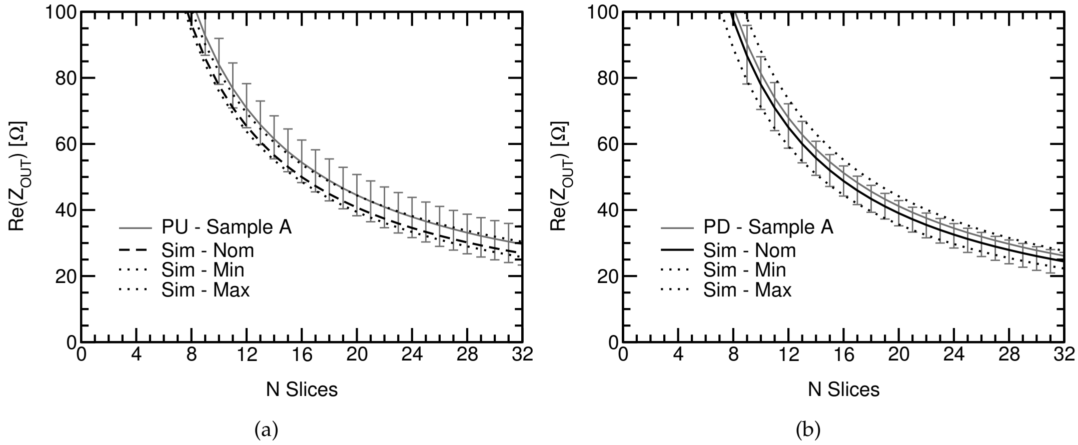


Figure 6.18: Measurement results (error bars) of the transmitter output impedance over VT variations compared to Titan simulation results (lines). (a) Pull-up and (b) pull-down branches.

plots of Figure 6.18, the agreement between the measurements and Titan simulations is very satisfactory for both pull-up and pull-down branches.

6.6.3 Transmitter Return Loss

Return loss has been measured programming the transmitter to the same settings as for the output impedance measurements. The common mode and differential return loss, RL_{CM} and RL_{DIFF} respectively, have been extracted from a 2-port S-parameter measurement performed with a Vector Network Analyzer (VNA), as shown in Figure 6.19. The expressions relating RL_{CM} and RL_{DIFF} to the 2-port S-parameters are the following [13] (see also the S-parameters definition at eq. 3.1 and 3.2):

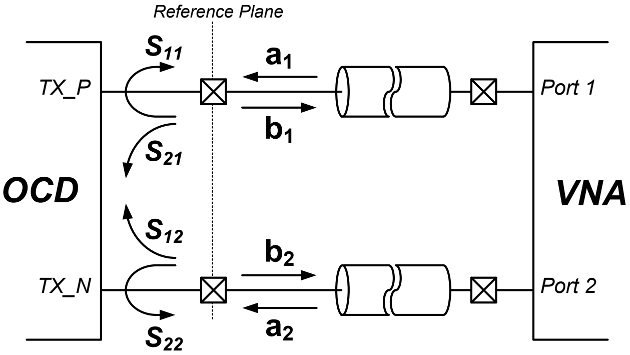


Figure 6.19: Instrument set-up for the OCD RL_{CM} and RL_{DIFF} measurement using a VNA. The instrument calibration has been performed including connectors and cable, in order to pose the measurement reference plane at transmitter output connectors.

$$\begin{aligned} RL_{CM} &= 20\log_{10}\left(\left|\frac{(S_{11}+S_{22}+S_{12}+S_{21})}{2}\right|\right) \ \text{[dB]} \\ RL_{DIFF} &= 20\log_{10}\left(\left|\frac{(S_{11}+S_{22}-S_{12}-S_{21})}{2}\right|\right) \ \text{[dB]}. \end{aligned} \tag{6.10}$$

Figure 6.21 reports the results of the measurements for the different combinations of supply and temperature corners considered. Due to the very poor agreement of these measurement results with simulations done during the design phase, we have repeated those simulations with a more detailed model for the bonding wires. In fact, a microscope inspection of the inner portion of the package hosting the prototype silicon chips (see Figure 6.25) revealed that the bonding wires are much longer than expected, approximately 5 mm versus the expected < 1 mm. The simple lumped 0.5 nH inductance is not adequate to include all the parasitic effects of such a long bonding. A more detailed model [82] that takes into account the pair of wires (one for V_o+ and one for V_o-) connecting the differential transmitter output to the package pins has thus been considered. As shown in Figure 6.20, this model includes the wire resistance and inductance effects, and also the wire coupling in the form of a capacitance and a mutual inductance between the two wires. Each bonding wire has a length of 5 mm and a diameter of 30 µm. Resistance and inductance terms are modeled at both DC and at the maximum frequency of the output signal, which is equal to the half of the data rate (1.25 GHz). At DC the total series resistance of a bonding wire is $R_{bonding} = R_{DC}$ and the total inductance is $L_{bonding} = L_1 + L_2$, while at high frequency $R_{bonding} = R_{DC} + R_{AC}$ and $L_{bonding} = L_2$, to model the skin effect. The capacitance term C and a mutual inductance M

accounts for the coupling between the bonding wires of the differential outputs (see Figure 6.20).

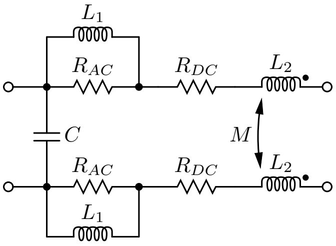
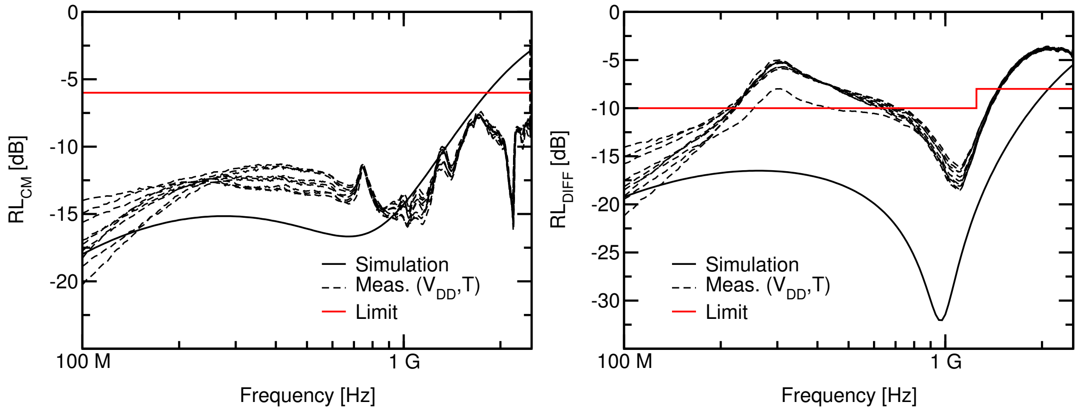


Figure 6.20: Schematic diagram of the bonding wires model considered for the return loss simulations of Figure 6.21.

The return loss extracted from these simulations is reported in Figure 6.21. Qualitative agreement between simulations and measurements is visible, but further effort is needed in order to improve the model/hardware correlation and understand how to improve the design, considering that the return loss specification is not completely fulfilled. To this aim, a better insight of the package parasitics would be particularly useful, but could not be obtained yet.



(a) (b)

Figure 6.21: Measured transmitter return loss for all the considered VT corners, compared with Titan simulations accounting for the 5 mm bonding wire through the more accurate model for the bonding wires. (a) Common Mode Return Loss RL_{CM} and (b) Differential Return Loss RL_{DIFF}.

6.6.4 Current Consumption

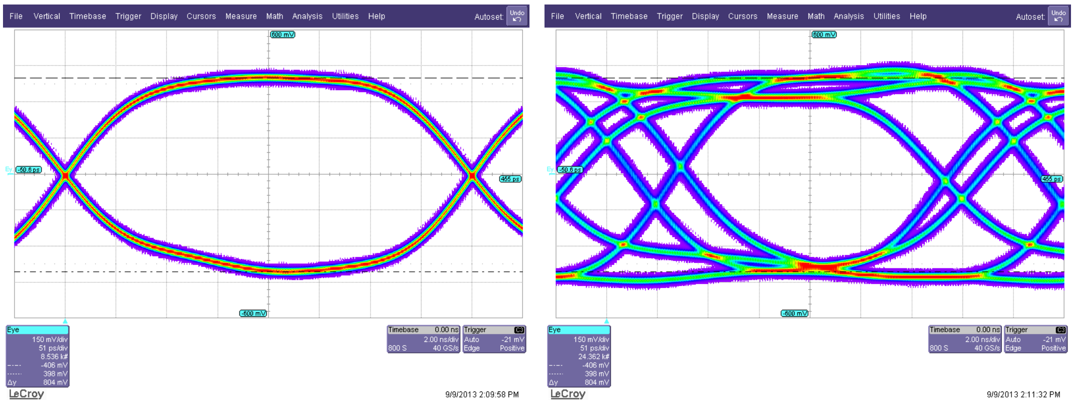
For testing purposes in the fabricated test chip the V_{DD} domain supplying the LDO has been kept separated from that supplying the driving circuitry (pre-driver, see fig 6.7(b), buffers and pattern rotator). In this way it has been possible to measure separately the currents drawn by the LDO ($I_{VDD,LDO}$, which is also the current flowing through the OCD) and by the rest of the transmitter circuitry ($I_{VDD,DIG}$). Table 6.3 reports the results measured at room temperature and nominal supply voltage value (i.e. 1.2 V) for the transmitter driving data at the 2.5 Gb/s bit rate. The table also reports the values obtained from Titan schematic simulations, run over PVT variations. A satisfactory agreement between measured values and simulations is visible.

Table 6.3: Measurement results of the transmitter current consumption compared with simulation results. Measurements include VT variations only. The considered bit rate is 2.5 Gb/s.

[mA]	Measurement	Simulation		
	Nom	Min	Nom	Max
$I_{VDD,LDO}$	3.57	4	4.2	4.38
$I_{VDD,DIG}$	10.71	8.35	9.4	10.8

6.6.5 Eye Diagram

As a final step, the transmitter output waveform have been characterized by means of the eye diagram. The 2.5 Gb/s data rate has been considered first. To observe the transmitter eye diagram, two patterns of 8 bit length have been considered: '01010101' and '11001010'. The first one produces a serial stream in the same form of a clock-like pattern and the second one reproduces a stream of data in which all the possible bit transition between two consecutive bits (e.g. '00', '01', '10' and '11') are present. The eye diagrams that have been observed for these two patterns are shown in Figure 6.22(a) and 6.22(b), respectively. Measurements show the strong presence of Data Dependent Jitter (DDJ): the eye diagram constructed from the clock-like pattern (Figure 6.22(a)), which is by definition unaffected by DDJ, is wide open. On the contrary, in the eye diagram constructed from the data-like pattern, Figure 6.22(b), it is possible to observe that multiple (3) waveform edges are visible for both '01' and '10' transitions. The horizontal eye opening is equal to 245 ps, which corresponds to 0.6 UI as the measurement clock frequency is 2.47 GHz. This indicates that DDJ is severely hampering the transmitter performance. The reasons behind the very poor DDJ performance have been investigated and are described in the following Section.



Eye 150 mV/div 51 ps/div 8.536 k# -406 mV 398 mV Δy 804 mV
Timebase 0.00 ns 2.00 ns/div 800 S 40 GS/s
Trigger C3 Auto -21 mV Edge Positive
9/9/2013 2:09:58 PM

(a)

Eye 150 mV/div 51 ps/div 24.362 k# -406 mV 398 mV Δy 804 mV
Timebase 0.00 ns 2.00 ns/div 800 S 40 GS/s
Trigger C3 Auto -21 mV Edge Positive
9/9/2013 2:11:32 PM

(b)

Figure 6.22: Measured transmitter eye diagram at room temperature (25 °C) and nominal supply voltage (1.2 V). (a) Clock-like pattern (i.e. '01010101'). (b) Data-like pattern (i.e. '11001010'). DDJ is clearly visible, leading to a horizontal eye opening of 245 ps (0.6 UI at the clock frequency of 2.47 GHz).

Eye Diagram Debugging

The cause of the huge amount of DDJ that is observed in Figure 6.22(b) has been identified as the combination of three factors.

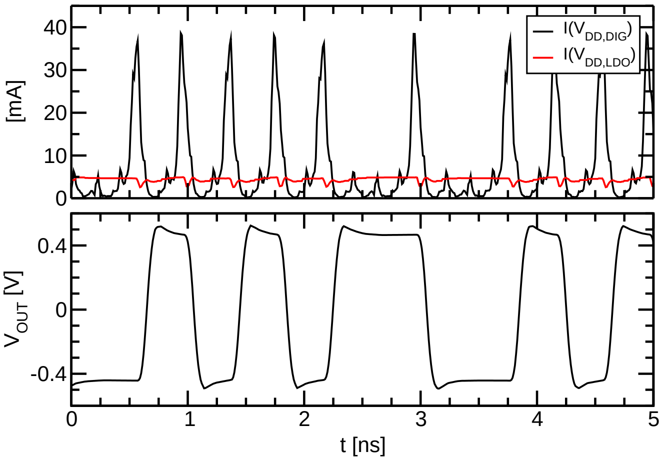


Figure 6.23: Circuit time-domain simulation of the transmitter driving the data-like pattern at 2.5 Gb/s. (Top) Currents drawn from the $V_{DD,DIG}$ and $V_{DD,LDO}$ supply, considered as ideal. (Bottom) Differential transmitter output voltage V_{OUT}.

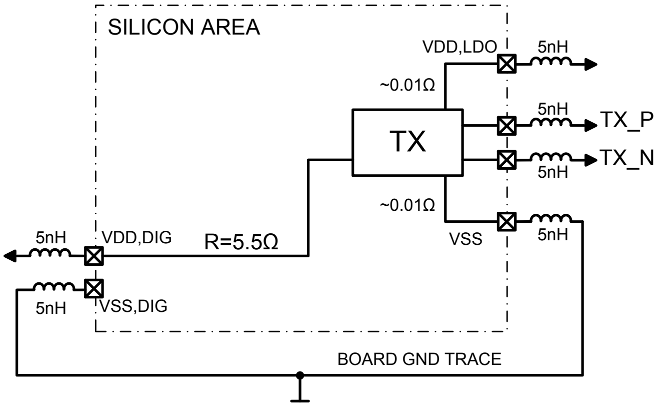


Figure 6.24: Sketch of the on-silicon physical routing of the supply lines from the pads to transmitter macro. The chip area is approximately 1 mm × 1 mm. The $V_{DD,DIG}$ supply line has been routed across all the chip width, thus forming a parasitic resistance of 5.5 Ω. The parasitic resistances due to metal routing at the $V_{DD,LDO}$ and V_{SS} supplies on the contrary can be estimated to be approximately 0.01 Ω.

Pre-driver current consumption: a careful analysis of the circuit time-domain simulations of the transmitter reveled the presence of significant current peaks at the $V_{DD,DIG}$ supply domain. As can seen in Figure 6.23, the current drawn by the transmitter from the $V_{DD,DIG}$ supply features huge peaks each time the output bit makes a transition between two symbols. These peaks can reach 40 mA although for a short duration. The simulation of Figure 6.23 has been done considering nominal PVT conditions. Considering all the different corners higher current peaks values can be found.

Supply lines parasitic resistance: having the hint of high supply current peaks, the analysis followed by identifying the possible presence of resistance and inductance parasitic effects on the $V_{DD,DIG}$ supply line. In fact, the presence of parasitic resistance and inductance can cause non negligible supply voltage drops on the pre-driver circuitry thus determining DDJ. The analysis of the physical silicon design of the fabricated test chip revealed that the $V_{DD,DIG}$ supply pad has been place on the opposite edge of the silicon chip with respect to the the edge where the transmitter macro has been place. As it is possible to see in Figure 6.24, the $V_{DD,DIG}$ line has to cross all the chip, being 1 mm × 1 mm its dimensions. A metal line of such length is quite likely to be subject of consistent parasitic series resistance. The estimation of the series resistance of the line is possible, given that the physical dimensions (section of the line and total length) and metal resistivity are known. The resulting series resistance is approximately 5.5 Ω. As we have seen above, the current on this line could be as high as 40 mA: the voltage drop over the series resistance is thus $5.5\text{ohm} \times 40\text{mA} = 220\text{mV}$, which corresponds to the 18% of the nominal supply voltage value.

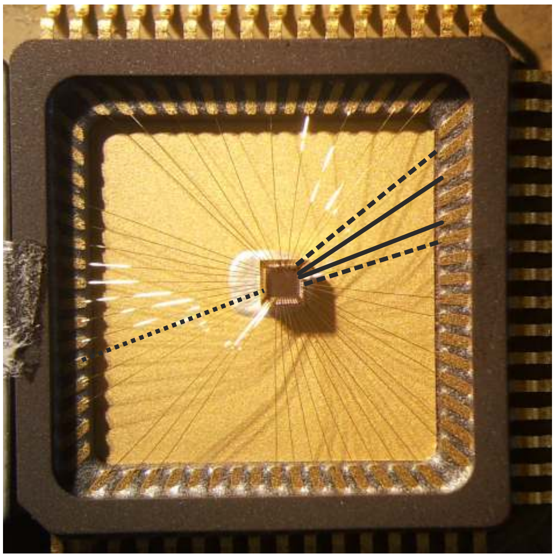


Figure 6.25: Picture showing the internal cavity of the CQFP64 package used to bond the fabricated silicon chips. The wire bonding is clearly visible. The solid lines highlight the bonding wires connecting the transmitter output from the silicon pads to package pins (TX_P/TX_N). The dashed lines highlight the $V_{DD,LDO}$ (top) and V_{SS} bonding wires. The dotted line highlight the $V_{DD,DIG}$ bonding wire. All the wire lengths are approximately 5 mm long. The ceramic package enclosure has a dimension of 14 mm × 14 mm. Image courtesy of Infineon Technologies AG.

Bonding wire induced parasitics: as already mentioned when presenting the return loss measurement results, the silicon chip has been bonded with much longer bonding wires than expected. Figure 6.25 reports the photograph of the package and silicon chip. A CQFP64 package with an internal cavity with dimensions of 9.75 mm × 9.75 mm is used to bond the chip. What can be immediately seen is that 1 mm × 1 mm die is centered in the cavity thus all the bonding wire connecting the silicon pads to the package pins are equally 5 mm long. As a consequence, all the connections between pads and package are affected by a 5 nH inductance. These inductance terms cause an additional voltage drop on the supply in correspondence of each current peak due to the almost instantaneous current variation on the supply.

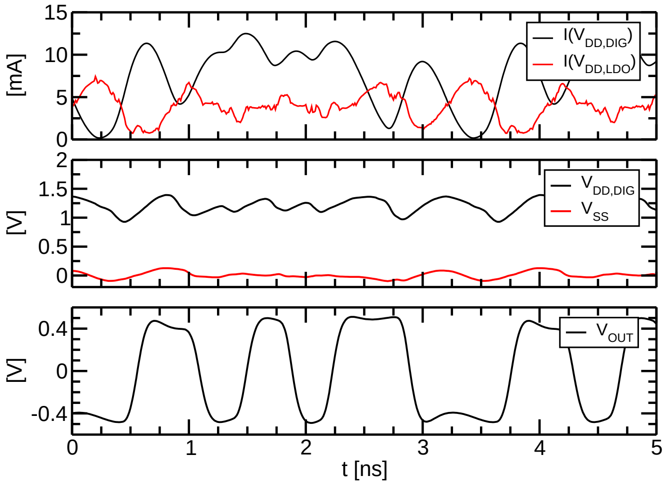


Figure 6.26: Circuit time-domain simulation of the transmitter driving the data-like pattern ('11001010') at 2.5 Gb/s, considering the $V_{DD,DIG}$ 5.5 Ω series resistance and 5nH of inductance on all the supplies. (Top) Currents drawn from the $V_{DD,DIG}$ and $V_{DD,LDO}$ supply. (Center) Instantaneous value of $V_{DD,DIG}$ supplying the transmitter. (Bottom) Differential transmitter output voltage V_{OUT}.

To confirm these observations circuit time-domain simulations of the transmitter have been repeated, including all the parasitic effects reported above. In particular, distributed parasitic elements extracted from the physical silicon design have been considered which, included the $V_{DD,LDO}$ and V_{SS} parasitic series resistances. An ideal 5.5 Ω resistance has been included in series to the $V_{DD,DIG}$ supply. The parasitic inductance due to bonding has been taken into account for each supply. As done for the return loss, the bonding wire model of Figure 6.20 has been considered to reproduce bonding wire effects at the transmitter outputs. The transmitter has been driven with the data-like pattern '11001010' used in the measurements. As can be seen in Figure 6.26, the supply parasitics combined to the huge current peaks on $V_{DD,DIG}$ supply cause significant variations of both the high and low supply ($V_{DD,DIG}$ and V_{SS}, respectively). The resulting eye diagram is reported in Figure 6.27. DDJ is now visible also in the simulations, allowing to reproduce quite well the 3 different edges for both the '0' to '1' and '1' to '0' transitions.

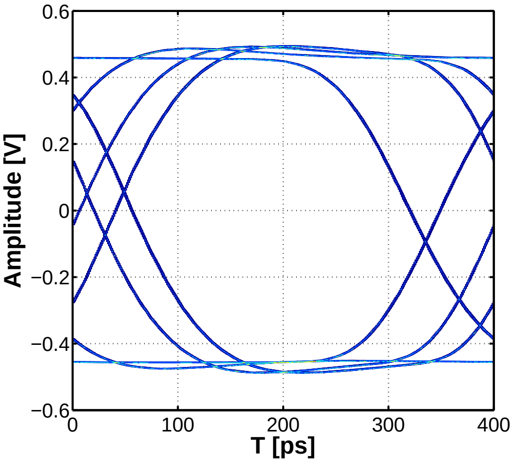


Figure 6.27: Eye diagram obtained from a circuit time-domain simulation of the transmitter driven with the data-like pattern ('11001010'). The simulation accounts for distributed parasitics as extracted from the physical silicon design, the 5.5 Ω series resistance on the $V_{DD,DIG}$ supply and a 5 nH series inductance on all supplies. The bonding wire parasitic effects at transmitter outputs have been accounted for using the bonding wire model of Figure 6.20.

6.6.6 Improved Chip Bonding

In order to reduce part of the parasitic effects due to bonding wires for a second lot of test chips, a different bonding scheme has been tested. The different bonding structure is shown in Figure 6.28: in this case a CQFP64 package with a smaller cavity (6.35 mm × 6.35 mm) has been used. This reduces the distance between the die and the package pins that has to be covered by bonding wires. Furthermore the die has not been placed at the center of the cavity but closer to the top-right corner of the cavity, to limit the length of the bonding wires at the transmitter output. To limit the parasitic inductance at the $V_{DD,DIG}$ supply, the bonding has been done with a double wire to halve the parasitic inductance and the $V_{DD,DIG}$ pin has been shifted from the left side to the right side of the package, as shown in Figure 6.28. With this second bonding configuration it has been possible to reduce the wire lengths to 2.2 mm for the transmitter outputs, 2.35 mm for $V_{DD,LDO}$, 1.75 mm for V_{SS} and 3 mm for $V_{DD,DIG}$, the latter with double connection.

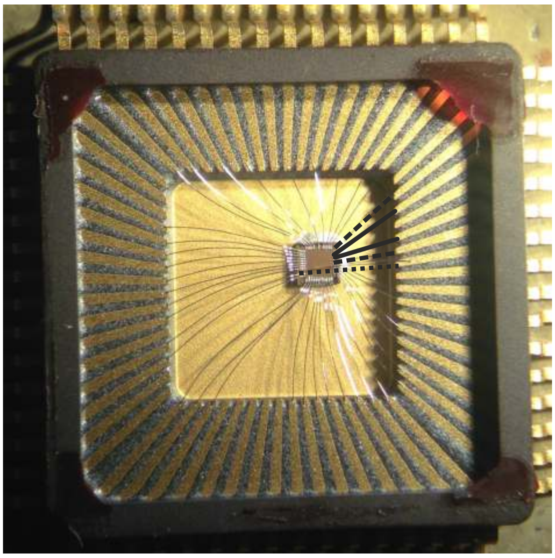
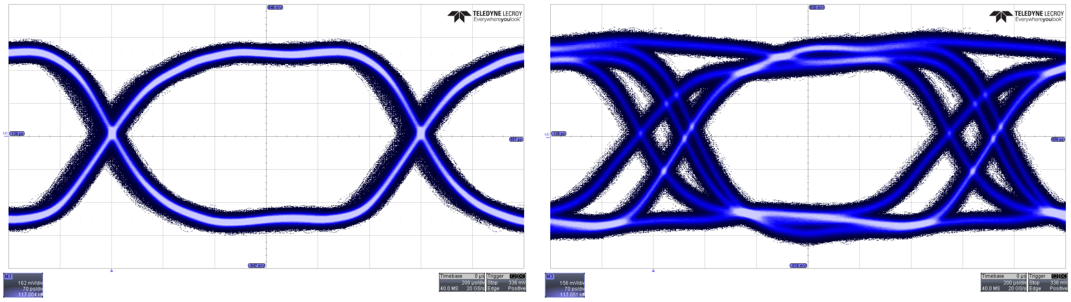


Figure 6.28: Picture showing the internal cavity of the second CQFP64 package used to bond the fabricated silicon chips. The difference in the silicon chip positioning inside the package cavity, that now has a dimension of 6.35 mm × 6.35 mm, and the different internal length of the package pins is clearly visible. The solid lines highlight the bonding wires connecting the transmitter output from the silicon pads to package pins (TX_P/TX_N). The dashed lines highlight the $V_{DD,LDO}$ (top) and V_{SS} (bottom) bonding wires. The dotted line highlight the $V_{DD,DIG}$ bonding wire pair passing above the silicon die. All the wire lengths are approximately 3 mm long. The ceramic package enclosure has an edge dimension of 14 mm. Image courtesy of Infineon Technologies AG.

The eye diagram measurement obtained from this improved bonding configuration are reported in Figure 6.29 for both the clock-like and the data-like pattern. As can be seen, the amount of DDJ in the eye diagram is now reduced, as the horizontal eye opening is now equal to 335 ps, corresponding to 0.81 UI as the measurement clock frequency is 2.41 GHz, due to a slightly lower center frequency of the oscillator providing the clock signal in the measured sample.

A time-domain circuit simulation has been run reproducing the reduced parasitics guaranteed by the second bonding scheme. Series inductance terms of 1.75 nH at V_{SS} supply, of



(a)

(b)

Figure 6.29: Measured transmitter eye diagram at room temperature (25 °C) and nominal supply voltage (1.2 V). (a) Clock-like pattern (i.e. ’01010101’). (b) Data-like pattern (i.e. ’11001010’). The horizontal eye opening is 335 ps (0.81 UI at the measurement clock frequency of 2.41 GHz). Bonding as in Figure 6.28.

2.35 nH at $V_{DD,LDO}$ supply and of $3/2 = 1.5$nH at $V_{DD,DIG}$ supply have been considered. Distributed parasitics extracted from the physical silicon design and the 5.5 Ω series resistance at $V_{DD,DIG}$ due to metal line routing are included as well. The eye diagram resulting from the simulation is shown in Figure 6.30. Comparing it with the eye diagram of Figure 6.27 it is clearly possible to see that the horizontal eye opening is now improved as it has been observed in the measurements. A satisfactory qualitative agreement with the measured eye diagram of Figure 6.29(b) is achieved.

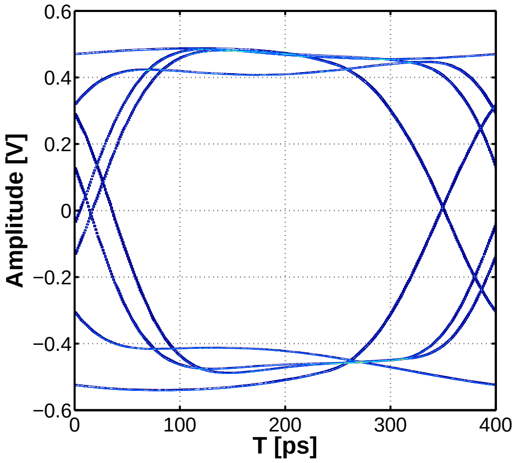


Figure 6.30: Eye diagram obtained from a circuit time-domain simulation of the transmitter driven with the data-like pattern (’11001010’) and the reduced bonding wire parasitics (i.e. supply series inductance) as in the improved bonding scheme of Figure 6.28. The simulation accounts for distributed parasitics as extracted from the physical silicon design, the 5.5 Ω series resistance on the $V_{DD,DIG}$ supply. The bonding wire parasitic effects at transmitter outputs have been accounted for using the bonding wire model of Figure 6.20.

Despite the improvement in the amount of DDJ with this improved bonding scheme, the eye diagram performance is still not satisfactory. As mentioned above, the causes of DDJ are not limited to parasitic inductance due to long bonding wires, thus just improving the bonding will not solve the problem. The other two causes, parasitic series resistance at

$V_{DD,DIG}$ supply and high pre-driver current consumption, must be tackled. If, on one side, the $V_{DD,DIG}$ supply series resistance can be reduced at the layout phase to values in the order of $1\,\Omega$ (or even less) with a more careful routing of the supply lines at layout phase, on the other side limiting the pre-driver current consumption require a significant redesign of the pre-driver circuitry and fabrication of new prototypes.

As first step in the redesign, a significant reduction of the current consumption may be immediately achieved by disabling the buffer stages that drive all the disabled transmitter slices. In fact, these buffers still switch at the interface data rate even if the slice they are required to drive is not in use, thus consuming power. A simple solution would be to put the buffer chain inside each slice and use a NAND gate as a first buffer stage, with one input the serial data and the other input the slice enable signal. In this way disabling the slice will also make the buffer idle, thus with null contribution to the overall current consumption.

A further possibility that should be investigated is the reduction of the number of slices: if the required granularity of the transmitter output impedance is lower than what achieved here with 16 active slices, one could think of using a lower number of active slices (e.g. $N_{active} = 12$) to achieve the same $50\,\Omega$ impedance. This will lead to a lower number of active pre-driver circuits, with a potential reduction of the overall current consumption. On the other side this choice will also lead to increase the W of the pull-up and pull-down MOSFETs on the OCD to achieve a lower $R_{eq,slice}$ and, as a consequence, also the W of the active devices in the pre-driver may require to be increased. The trade-off between N_{active} and the minimization of the current consumption must be carefully evaluated with the help of simulations.

Finally the transmitter redesign will also have to focus on a further optimization of the number of buffer stages required to drive each slice input, and their optimum sizing, in order to reduce the current throughout the data-path.

6.7 Final Remarks

The design activity of the high speed transmitter has been described in this Chapter. The fabricated lot of test chips has been evaluated with measurements of the main performance figures. The picture drawn by the experimental results is twofold: some of the performance have shown a very good qualitative and quantitative agreement with pre-silicon simulations. This is the case of the LDO output voltage, the output impedance of the transmitter and the overall current consumption of the circuit. On the other hand a poor performance level has been observed for the return loss, and especially for the eye diagram that is heavily affected by DDJ. The causes of this have been investigated by the analysis of the physical silicon design, of the bonding scheme of the fabricated chips and of the pre-silicon simulations. An improved bonding scheme has been arranged and implemented, with promising improvements of the eye diagram performance of the transmitter. The final balance of the design activity is anyway positive, as the main problems affecting the first test lot have been identified, highlighting the points where the design activity has to focus in the future development of the circuit. Unfortunately, due to the poor eye diagram performance at already 2.5 Gb/s, was not possible to test the capability of 5 Gb/s operation, which had to be postponed to a future redesign.

Chapter 7

Conclusions

In this dissertation we have investigated serial data transmission systems with bit rates in the order of Gb/s. In the first phase, the focus has been on the modeling of high speed links and the development of a simulation framework to allow for a fast estimation of the link performance. In the second part the circuit implementation of a transmitter employing an aggressive deca-nanometer CMOS technology has been described.

We have proposed a statistical model for ISI and jitter in HSSI and demonstrated its implementation into a MATLAB program. This approach allows for an accurate modeling of the transmitter pulse shape, a feature that is missing in other statistical techniques due to the non-trivial problem of dealing with transmitter non-linearity. Our approach has been widely tested by comparison with Spice-like time-domain simulation. The model proved to be advantageous, as it dramatically reduces the simulation time over traditional Spice-like techniques. Limitations, instead, arose in the two step procedure proposed to simulate the channel response to the transmitter waveform, in handling impedance discontinuities. We have demonstrated that the limitations do not have much impact on the prediction capability of the technique in the usual case of well-designed links, with common tolerances in terms of impedance matching (e.g. $50\,\Omega \pm 10\%$).

A more conservative approach has been chosen, instead, to model jitter: balancing the pros and cons of various approaches, the Receiver Sampling Distribution technique has been preferred and implemented. Despite the fact that the assumption of uncorrelation between all jitter source is clearly quite simplistic, the approach demonstrated to provide reliable results in the explored range of data rates.

The link model has been further validated by comparison with experimental data from a high-speed test system (1.25 and 2.5 Gb/s). While achieving a precise agreement in terms of the eye diagrams was not possible, due to a lack of accurate models for the package and board at transmitter side, the prediction of the eye opening reduction for long channels was more than satisfactory.

About the design activity of a high speed transmitter circuit in an aggressively scaled CMOS technology, the review of recent works pictured a clear improvement in the power efficiency of transmitters when adopting the Voltage Mode topologies in place of traditional Current Mode implementations, thanks to their potential for low power consumption and high swing capability. The Source Series Terminated (SST) architecture appears attractive as it combines the advantage in power reduction of a Voltage Mode driver to a design completely based on digital switching techniques that cope well with nowadays deca-nanometer technologies. The SST architecture has been chosen for the transmitter implementation. Targets of the design are to achieve a data rate of 5 Gb/s with a minimum voltage swing of 800 mV, thus doubling the data rate of the latest transmitter designed at the Infineon Technologies Design Center Villach. The experimental characterization of the fabricated lot draws a twofold picture, with some of the performance figures showing a very good qualitative and quantitative agreement with pre-silicon simulations, and others revealing a poor performance level, especially for the eye diagram. The investigation of the root causes by the analysis of the physical silicon design, of the bonding scheme of the prototypes and of the pre-silicon simulations has been reported. It revealed that significant current peaks drawn by pre-driver, buffer and pattern rotator circuitry, combined with high parasitic series resistance of the on-chip supply line and high parasitic inductance of the poor bonding scheme, caused large voltage drops on the $V_{DD,DIG}$ supply and, as a consequence, huge Data Dependent Jitter (DDJ) in the measured eye diagram. An improved bonding scheme has been tested, with promising improvements of the eye diagram, indicating that a successful redesign must go in the direction of minimizing the parasitic series resistance and inductance on the supplies, together with a marked reduction of the current consumption of the pre-driver and buffer circuits, which has been overlooked in the first transmitter design described here. The final balance of the design activity is anyway positive, as the main problems affecting the first fabricated test lot have been identified, and guidelines for the redesign are clear. Unfortunately, due to the poor eye diagram performance at already 2.5 Gb/s, it was not possible to test the capability of 5 Gb/s operation, which had to be postponed to the future redesign.

Appendix A

Matlab Implementation

This appendix firstly reports two simple MATLAB [53] scripts developed for this thesis as preliminary work towards the implementation of the statistical ISI and jitter modeling approach, described in detail in Chapter 3. The first one implements the calculation of the worst-case eye diagram from the channel pulse response as described in Section 2.1.1. The second script implements the calculation and plotting of the statistical eye diagram based on the SBR algorithm [36,37,49], which is described in Section 2.1.2.

In the second part of this appendix, an overview of the MATLAB implementation of the combined ISI and jitter modeling tool is given. The complete scripts are not reported for a matter of space.

A.1 Scripts

The only input that must be provided by the user to the two script is a `.txt` file containing the SBR of the channel, calculated e.g. with the help of a time-domain circuit simulation as explained in Section 3.3. To be correctly imported by the script the file must be in a two column format:

```
2.04000E-08  3.83620E-01
2.04010E-08  3.83826E-01
2.04020E-08  3.84027E-01
2.04030E-08  3.84223E-01
2.04040E-08  3.84414E-01
2.04050E-08  3.84600E-01
2.04060E-08  3.84781E-01
2.04070E-08  3.84958E-01
2.04080E-08  3.85129E-01
2.04090E-08  3.85297E-01
...
```

On the first column there are the time samples (in s), and on the second column the voltage samples (in V). At the end of the computation the two scripts produce the contour plots of the worst case eye and statistical eye as shown in Figure 2.4(b).

A.1.1 Worst-Case Eye Diagram

```

%%%%%% WORST - CASE %%%%%%

% UI discretization
m=200;
% Number of cursors
n=40;
p=12;
% Number of bins
bins=1000;
width=2*m;

bitrate=2.5e9;
UI=1/bitrate;

%% Reading the Pulse Response from an external text file
a=importdata('pulse_resp.txt',' ');

if isstruct(a)
   t_plsrsp=a.data(:,1)';
   plsrsp=a.data(:,2)';
else
   t_plsrsp=a(:,1)';
   plsrsp=a(:,2)';
end;

%%%%%%% RESAMPLING THE PULSE RESPONSE %%%%%%%%
%Finding the base UI & resampling the pulse resp
Max_plsrsp=max(plsrsp);  %Reference is t @ pulse_resp=Vmax/2
t0=t_plsrsp(find(plsrsp≥max(plsrsp)/2,1,'first'));
%Vector of time instants of interest for the resampling
t=((t0-(p+0.5)*UI):UI/m:(t0+(n+1.5)*UI));

%ERROR CHECK: is the input pulse response long enough?
if t(end)>t_plsrsp(end)
    error('Input pulse response too short: impossible to extract n POST-cursors.')
end;
if t(1)<t_plsrsp(1)
    error('Input pulse response too short: impossible to extract p PRE-cursors.')
end;

%Resampling
pulse_resp=interp1(t_plsrsp,plsrsp,t);

%ERROR CHECK: the resampling produced NaN or Inf?
if (max(isnan(pulse_resp))≠0 || max(isinf(pulse_resp))≠0)
    error('Re-sampling of the pulse response FAILED. EXECUTION TERMINATED!')
end;

%% WORST-CASE CONTOUR CALCULATION %%
pulseresp_worstcase=zeros(width,p+n+1);
pulse_resp=pulse_resp(2:end);
for i=1:width
   pulseresp_worstcase(i,:)=pulse_resp(i:m:(p+n)*m+i);
end;
```

```

worstcase_1=pulseresp_worstcase(:,p+1);
worstcase_0=-1*worstcase_1;
for j=1:width
    for i=1:p
        if (pulseresp_worstcase(j,i)>0)
           worstcase_0(j)=worstcase_0(j)+pulseresp_worstcase(j,i);
           worstcase_1(j)=worstcase_1(j)-pulseresp_worstcase(j,i);
        else
           worstcase_0(j)=worstcase_0(j)-pulseresp_worstcase(j,i);
           worstcase_1(j)=worstcase_1(j)+pulseresp_worstcase(j,i);
        end;
    end;
    for i=p+2:p+n+1
       if (pulseresp_worstcase(j,i)>0)
           worstcase_0(j)=worstcase_0(j)+pulseresp_worstcase(j,i);
           worstcase_1(j)=worstcase_1(j)-pulseresp_worstcase(j,i);
       else
           worstcase_0(j)=worstcase_0(j)-pulseresp_worstcase(j,i);
           worstcase_1(j)=worstcase_1(j)+pulseresp_worstcase(j,i);
       end;
   end;
end;

worstcase_0(worstcase_0≥0)=NaN;
worstcase_1(worstcase_1≤0)=NaN;

%% Plotting the countour
figure;
set(gcf,'OuterPosition',[1,1,700,700]);
set(gcf,'defaultaxesfontsize',15,'defaultaxeslinewidth',2,...
        'defaultlinelinewidth',3,'defaultpatchlinewidth',2)
plot((((1:2*m)-m/2)/m),worstcase_1,'k','LineWidth',3,'LineStyle','--');
plot((((1:2*m)-m/2)/m),worstcase_0,'k','LineWidth',3,...
           'LineStyle','--','HandleVisibility','off');
xlim([0 1])
ylim([-0.6-1e-12 0.6+1e-12])
xlabel('\Phi','FontName','Helvetica','FontSize',24,'FontWeight','bold')
ylabel('Amplitude [V]','FontSize',24,'FontWeight','bold')
set(gca,'FontName','Helvetica','FontSize',20,'FontWeight','normal')
```

A.1.2 Single Bit Response ISI Algorithm

```

% UI discretization
m=200;
% Number of cursors
n=40;
p=12;
% Number of bins
bins=1000;
width=2*m;

bitrate=2.5e9;   %StatEye: 11.1e9
UI=1/bitrate;

```

```

% Reading the Pulse Response from an external text file
a=importdata('pulse_resp.txt',' ');

if isstruct(a)
   t_plsrsp=a.data(:,1)';
   plsrsp=a.data(:,2)';
else
   t_plsrsp=a(:,1)';
   plsrsp=a(:,2)';
end;

%%%%%%% RESAMPLING THE PULSE RESPONSE %%%%%%%%
%Finding the base UI & resampling the pulse resp
Max_plsrsp=max(plsrsp);  %Reference is t @ pulse_resp=Vmax/2

t0=t_plsrsp(find(plsrsp≥max(plsrsp)/2,1,'first'));
%Vector of time instants of interest for the resampling
t=((t0-(p+0.5)*UI):UI/m:(t0+(n+1.5)*UI));

%ERROR CHECK: is the input pulse response long enough?
if t(end)>t_plsrsp(end)
    error('Input pulse response too short: impossible to extract n POST-cursors.')
end;
if t(1)<t_plsrsp(1)
    error('Input pulse response too short: impossible to extract p PRE-cursors.')
end;

%Resampling
pulse_resp=interp1(t_plsrsp,plsrsp,t);
%pulse_resp=interp1(t_plsrsp,plsrsp,t,'linear','extrap');

%ERROR CHECK: the resampling produced NaN or Inf?
if (max(isnan(pulse_resp))≠0 || max(isinf(pulse_resp))≠0)
    error('Re-sampling of the pulse response FAILED. EXECUTION TERMINATED!')
end;

%%%%%%%  ISI PDF Construction  %%%%%%%
tStart=tic;
Max_pulseresp=max(pulse_resp);

% Construction of vectors [0 ... 0 1 0 ... 0 1 0 ... 0] using quantization ...
    function
% STEP 1: definition of the positive thresholds and extension also to negative ...
    amplitudes
partition_pos=((Max_pulseresp/bins)/2:...
                Max_pulseresp/bins:...
                                (Max_pulseresp-(Max_pulseresp/bins)/2));
partition=[fliplr(-partition_pos) partition_pos];

% STEP 2: obtaining the number of the bin in which each sample of the pulse ...
    response goes
indx=quantiz(pulse_resp,partition);

%STEP 3: identification of the samples needed for each set of convolutions
indx_samples=zeros(width,p+n+1);
indx=indx(2:end);
for i=1:width
    indx_samples(i,:)=indx(i:m:(p+n)*m+i);
```

```
end;

% STEP 4: separation of the base vector of samples (2UI long)
base_column=p+1;
indx_base=indx_samples(:,base_column);  %Array containing the ampl indeces of ...
    central vector
indx_cursors=indx_samples(:,[(1:base_column-1) (base_column+1:(n+p+1))]); %All ...
    other ampl indeces

% STEP 5: creation of the vectors (1 0 0 ... 0 0 1) and convolution to obtain ...
    m ISI sets
maxPdfbins=max(sum(abs(indx_samples-bins),2)); %maxPdfbins=max amplitude ...
    observed after all the convolutions

Pdf=zeros(width,2*maxPdfbins+1);        %Pdf must be large enough to contain ...
    the max amplitude due to ISI

for k=1:width
    if (indx_base(k)-bins)≥0     %Check if sample is positive or negative
        Pdf_temp=zeros(1,2*(indx_base(k)-bins)+1);  %Minimum length of ...
            Pdf_temp is used, it varies after each convolution
        Pdf_temp(end)=1; %First convolution = first unit vector
    else
        Pdf_temp=zeros(1,2*abs(indx_base(k)-bins)+1);
        Pdf_temp(1)=1;
    end;
    for j=1:(p+n)       %All other cursors can be pos or neg, doesn't care
        unit_vec=zeros(1,2*abs(indx_cursors(k,j)-bins)+1);  %Each unit_vector ...
            has minimum length
        unit_vec([1 length(unit_vec)])=1; %unit_vec has 1s only in first and ...
            last position (bin)
        Pdf_temp=conv(Pdf_temp,unit_vec);
    end;
    %Copying Pdf_temp taking care of the length difference with Pdf
    offset=(length(Pdf)-length(Pdf_temp))/2;
    Pdf(k,(offset+1:offset+length(Pdf_temp)))=Pdf_temp;
end;

tElapsed=toc(tStart);    %Plotting compute time for convolutions
disp(sprintf('Elapsed time for computing ISI convolutions: %g s\n', tElapsed));

%Translating bins into amplitudes
ampl=((1:2*maxPdfbins+1)-(maxPdfbins+1)).*(Max_pulseresp/bins);

% TEST: smoothing the Pdf
% StatEye intrisically produces a Pdf in which a lot of 0s are present.
% This slowers a lot the contour plotting of the Pdf. This code substitutes
% the 0 with a mean of the nearby elements (only for plotting the Pdf)
smoothing_en=1;
if smoothing_en
    for i=1:width
        for j=2:size(Pdf,2)-1
            if Pdf(i,j)==0
                Pdf(i,j)=(Pdf(i,j-1)+Pdf(i,j+1))/2;
            end;
        end;
    end
end;

```

```
% NORMALIZATION
ind_int=sum(Pdf,2)*Max_pulseresp/bins; %integrating the Pdfs from -inf to inf
Pdf_norm=bsxfun(@rdivide,Pdf,ind_int);

%%%%%% CDF + EYE %%%%%%%

Cdf=zeros(size(Pdf_norm));
for i=1:width
    Cdf(i,:)=(cumsum(Pdf_norm(i,:))*Max_pulseresp/bins);%./ind_int(i);
end;
Cdf_log=log10(Cdf);

%% BER
figure;
set(gcf,'OuterPosition',[1,1,700,700]);
set(gcf,'defaultaxesfontsize',15,'defaultaxeslinewidth',2,...
        'defaultlinelinewidth',3,'defaultpatchlinewidth',2)
contour((-(m-1):m)/m,ampl,Cdf_log',(-16:-2),'LineWidth',2)
hold on;
grid on;
contour((-(m-1):m)/m,-ampl,Cdf_log',(-16:-2),'LineWidth',2,...
        'HandleVisibility','off')
xlabel('\Phi','FontName','Helvetica','FontSize',24,'FontWeight','bold')
ylabel('Amplitude [V]','FontSize',24,'FontWeight','bold')
ylabel(colorbar, 'contour for BER = 10e ...','FontSize',24,'FontWeight','bold');

%% Statistical PDF
figure;
set(gcf,'OuterPosition',[1,1,700,700]);
set(gcf,'defaultaxesfontsize',15,'defaultaxeslinewidth',2,...
        'defaultlinelinewidth',3,'defaultpatchlinewidth',2)
[C h]=contour(([1:m]/m),ampl,Pdf_complete',...
        [0:max(max(Pdf_complete))/5:max(max(Pdf_complete))],...
                'HandleVisibility','on');
hold on;
xlim([0 1])
ylim([-0.6-1e-12 0.6+1e-12])
xlabel('\Phi','FontName','Helvetica','FontSize',24,'FontWeight','bold')
ylabel('Amplitude [V]','FontSize',24,'FontWeight','bold')
ylabel(colorbar, 'V^{-1}','FontSize',24,'FontWeight','bold');
```

A.2 ISI and Jitter Modeling Tool

The MATLAB implementation of the ISI and jitter tool described in Chapter 3 is organized as shown in Figure A.1. The top level is constituted by a graphical user interface, which allows the user to set the parameters for the simulation and visualize the resulting plots. The graphical user interface also has the task to memorize the data resulting from the various elaborations as, for example, the transition waveforms generated from the transmitter pulse or the rational function $H(s)$ which fits the channel transfer function $\mathcal{F}$. The main data computations are four (indicated by the green boxes in Figure A.1):

- generation of the transition waveforms from the transmitter pulse response;
- extraction of the channel differential insertion loss S_{DD21}. The data path to simulate can be formed by up to four different passive elements (package, connectors, PCB/cable,

etc.), each one modeled by means of a 4-port S parameter matrix. The cascade connection of the various elements is performed at the S parameter level using the expressions from [83]. The S_{DD21} is extracted from the S parameter matrix of the cascade and then the channel transfer function $\mathcal{F}$ is obtained using eq. 3.3.

- fitting of $\mathcal{F}$ by means of a rational function $H(s)$ with a finite number of poles. This step can be performed using the built-in MATLAB function `rationalfit` for the rational fitting (requires the availability of the *RF Toolbox* license) or alternatively the rational fitting function developed by the candidate and based on the work reported in [54, 55], which does not require any additional license;
- calculation of the statistical ISI and jitter PDF using the algorithm developed in this PhD thesis.

Additionally to these main functions, the tool also includes a few functions to plot the final results (ISI and jitter PDF, BER) as well as the intermediate computation data like the transition waveforms and the comparison between $\mathcal{F}$ and $H(s)$ resulting from the fitting process. Figure A.2 shows a picture of the graphical user interface of the developed tool.

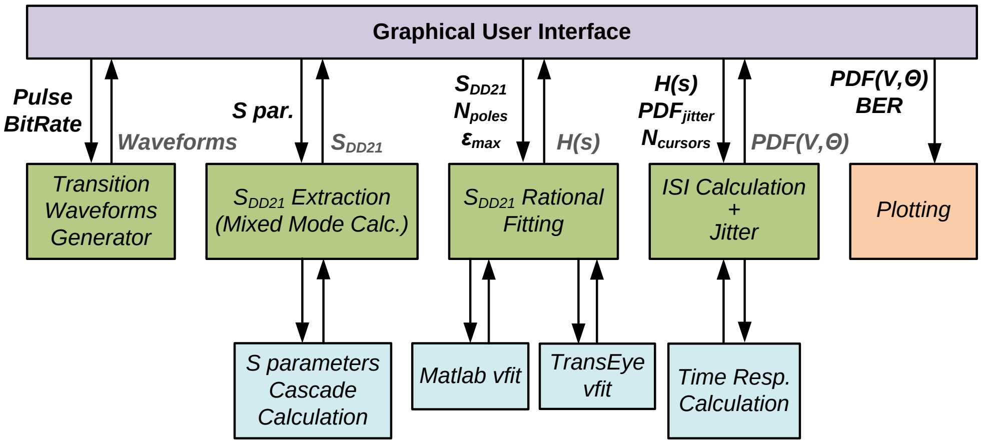
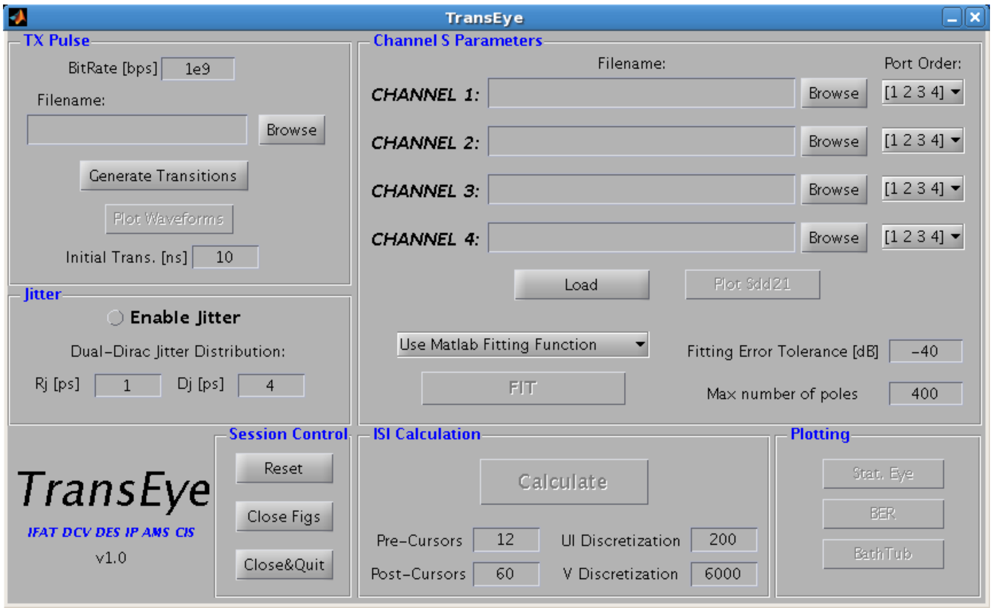


Figure A.1: Block diagram of the various functions that compose the ISI and jitter modeling tool and their interactions with the graphical user interface, which is the part of the tool that is visible to the user. The most important parameters provided to the main computation functions and their results are also reported. In particular, N_{poles} indicates the maximum number of poles that must be used to fit $\mathcal{F}$ with $H(s)$ (m in eq. 3.5), ϵ_{max} is the maximum fitting error that can be tolerated (see eq. 3.6) and $N_{cursors}$ is the number of pre-cursors and post-cursors to consider in the statistical computation of ISI.



BitRate [bps] 1e9

Initial Trans. [ns] 10

Rj [ps] 1 Dj [ps] 4

CHANNEL 1: [1 2 3 4]

CHANNEL 2: [1 2 3 4]

CHANNEL 3: [1 2 3 4]

CHANNEL 4: [1 2 3 4]

Use Matlab Fitting Function

Fitting Error Tolerance [dB] −40

Max number of poles 400

Pre−Cursors 12 UI Discretization 200

Post−Cursors 60 V Discretization 6000

Figure A.2: Picture of the graphical user interface of the developed ISI and jitter tool.

Bibliography

[1] "FPGAs at 40 nm and >10 Gbps: Jitter-, Signal Integrity-, Power-, and Process-Optimized Transceivers," White Paper WP-01092-1.1, Altera Corporation, Apr. 2013. [Online]. Available: http://www.altera.com/literature/wp/wp-01092-stratix-iv-gt-10gbps-transceivers.pdf

[2] "Interlaken Protocol Definition - A Joint Specification of Cortina Systems and Cisco Systems," Revision 1.2, Cortina Systems Inc. and Cisco Systems Inc., Oct. 2008. [Online]. Available: http://www.interlakenalliance.com/Interlaken_Protocol_Definition_v1.2.pdf

[3] "PCI Express Base Specification Revision 3.0," Nov. 2010. [Online]. Available: http://www.pcisig.com/specifications/pciexpress/specifications

[4] "HyperTransport I/O Link Specification Revision 3.10," Technical Document HTC20051222-0046-0026, HyperTransport Technology Consortium , Jul. 2008. [Online]. Available: http://www.hypertransport.org/docs/twgdocs/HTC20051222-00046-0028.pdf

[5] "Common Public Radio Interface (CPRI) Specification V6.0," Aug. 2013. [Online]. Available: http://www.cpri.info/downloads/CPRI_v_6_0_2013-08-30.pdf

[6] "RapidIO Interconnect Specification Rev. 3.0," Oct. 2013. [Online]. Available: http://www.rapidio.org/specs/current

[7] "Backplane Applications with 28nm FPGAs," White Paper WP-01185-1.1, Altera Corporation, Dec. 2012. [Online]. Available: http://www.altera.com/devices/fpga/stratix-fpgas/stratix-v/transceivers/stxv-transceivers.html

[8] "Extending Transceiver Leadership at 28 nm," White Paper WP-01130-2.1, Altera Corporation, Oct. 2012. [Online]. Available: http://www.altera.com/education/webcasts/all/wc-2010-transceiver-leadership-28nm.html

[9] S. Palermo, "High-Speed Serial I/O Design for Channel-Limited and Power-Constrained Systems," in *CMOS Nanoelectronics Analog and RF VLSI Circuits*, K. Iniewski, Ed. McGraw-Hill, 2011.

[10] A. Athavale and C. Christensen, "High-Speed Serial I/O Made Simple - A Designer's Guide with FPGA Applications," XILINX, Apr. 2005. [Online]. Available: http://www.xilinx.com/publications/archives/books/serialio.pdf

[11] W. J. Dally and J. W. Poulton, *Digital Systems Engineering*. New York, NY, USA: Cambridge University Press, 1998.

[12] M. Horowitz, C.-K. K. Yang, and S. Sidiropoulos, “High-Speed Electrical Signaling: Overview and Limitations,” *IEEE Micro*, vol. 18, no. 1, pp. 12–24, 1998.

[13] E. Bogatin, *Signal and Power Integrity - Simplified*, 2nd ed. Upper Saddle River, NJ, USA: Prentice Hall, 2009.

[14] J. Fan *et al.*, “Signal Integrity Design for High-Speed Digital Circuits: Progress and Directions,” *IEEE Trans. Electromagn. Compat.*, vol. 52, no. 2, pp. 392–400, 2010.

[15] B. Young, *Digital Signal Integrity: Modeling and Simulation with Interconnects and Packages*, 1st ed. Upper Saddle River, NJ, USA: Prentice Hall, 2000.

[16] P. K. Hanumolu, G.-Y. Wei, and U.-K. Moon, “Equalizers for High-Speed Serial Links,” *International Journal of High Speed Electronics and Systems*, vol. 15, no. 02, pp. 429–458, 2005.

[17] J. Liu and X. Lin, “Equalization in high-speed communication systems,” *IEEE Circuits and Systems Magazine*, vol. 4, no. 2, pp. 4–17, 2004.

[18] W. Dally and J. Poulton, “Transmitter Equalization for 4Gb/s Signaling,” *IEEE Micro*, vol. 17, no. 1, pp. 48–56, 1997.

[19] A. Rylyakov and S. Rylov, “A Low Power 10 Gb/s Serial Link Transmitter in 90-nm CMOS,” in *IEEE Compound Semiconductor Integrated Circuit Symp. (CSIC)*, 2005, pp. 4 pp.–.

[20] B. Casper *et al.*, “A 20Gb/s Forwarded Clock Transceiver in 90nm CMOS,” in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2006, pp. 263–272.

[21] S. H. Hall and H. L. Heck, *Advanced Signal Integrity for High-Speed Digital Designs*. Wiley-IEEE Press, 2009.

[22] C.-F. Liao and S.-I. Liu, “A 40 Gb/s CMOS Serial-Link Receiver With Adaptive Equalization and Clock/Data Recovery,” *IEEE J. Solid-State Circuits*, vol. 43, no. 11, pp. 2492–2502, 2008.

[23] S. Gondi and B. Razavi, “Equalization and Clock and Data Recovery Techniques for 10-Gb/s CMOS Serial-Link Receivers,” *IEEE J. Solid-State Circuits*, vol. 42, no. 9, pp. 1999–2011, 2007.

[24] H. Wang and J. Lee, “A 21-Gb/s 87-mW Transceiver With FFE/DFE/Analog Equalizer in 65-nm CMOS Technology,” *IEEE J. Solid-State Circuits*, vol. 45, no. 4, pp. 909–920, 2010.

[25] “Ready For Tomorrow,” Annual Report, Infineon Technologies AG, Dec. 2013. [Online]. Available: http://www.infineon.com/cms/en/corporate/investor/reporting/reporting.html

[26] R. Ploss, A. Mueller, and P. Leteinturier, “Solving automotive challenges with Electronics,” in *International Symposium on VLSI Technology, Systems and Applications (VLSI-TSA)*, 2008.

[27] “Future Advances in Body Electronics,” White Paper BODYDELECTRWP, Freescale Semiconductor, Inc., 2013. [Online]. Available: http://www.freescale.com/files/automotive/doc/white_paper/BODYDELECTRWP.pdf

[28] B. Fleming, "Microcontroller Units in Automobiles," *IEEE Vehicular Technology Magazine*, vol. 6, no. 3, pp. 4–8, 2011.

[29] K. Matheus and S. Sturm, "The Car: Transformation from Mechanical to Electronic Device," presented at the 2013 IEEE Int. Solid-State Circuits Conf. (ISSCC) - ES3: High-speed communications on 4 wheels: What's in your next car?, Feb. 2013.

[30] "In-Vehicle Networking," White Paper BRINVEHICLENET, Freescale Semiconductor, Inc., 2006. [Online]. Available: http://cache.freescale.com/files/microcontrollers/doc/brochure/BRINVEHICLENET.pdf

[31] N. J. Endo, "Wireless Communication in and around the car - Status and Outlook," presented at the 2013 IEEE Int. Solid-State Circuits Conf. (ISSCC) - ES3: High-speed communications on 4 wheels: What's in your next car?, Feb. 2013.

[32] C. Schmidt, "Automotive Electronics - Enabling the future of individual mobility," in *Proc. IEEE International Electron Devices Meeting (IEDM)*, 2007, pp. 3–8.

[33] B. Casper, M. Haycock, and R. Mooney, "An Accurate and Efficient Analysis Method for Multi-Gb/s Chip-to-chip Signaling Schemes," in *IEEE Symp. Very Large Scale (VLSI) Circuits*, 2002, p. 54.

[34] P. Hanumolu *et al.*, "Analysis of PLL Clock Jitter in High-Speed Serial Links," *IEEE Trans. Circuits Syst. II*, vol. 50, no. 11, pp. 879–886, 2003.

[35] J. Proakis and M. Salehi, *Digital Communications*, 5th ed. McGraw-Hill, 2008.

[36] V. Stojanovic and M. Horowitz, "Modeling and Analysis of High-speed Links," in *Proc. IEEE Custom Integr. Circuits Conf.*, 2003, p. 589.

[37] A. Sanders, M. Resso, and J. D'Ambrosia, "Channel Compliance Testing Utilizing Novel Statistical Eye Methodology," in *DesignCon*, Santa Clara, 2004.

[38] B. Casper *et al.*, "Future Microprocessor Interfaces: Analysis, Design and Optimization," in *Proc. IEEE Custom Integr. Circuits Conf.*, 2007, p. 479.

[39] G. Balamurugan *et al.*, "Modeling and Analysis of High-Speed I/O Links," *IEEE Trans. Adv. Packag.*, vol. 32, no. 2, p. 237, 2009.

[40] R. Nonis, "Modeling And Design Of Low Power Integrated Circuits For Frequency Synthesis In CMOS Technology," Ph.D. dissertation, Università degli Studi di Udine, Udine, Italy, 2007.

[41] N. Da Dalt, "Tutorial T5: Jitter: Basic and Advanced Concepts, Statistics, and Applications," in *2012 IEEE International Solid-State Circuits Conference (ISSCC)*, Feb. 2012.

[42] B. Ham, "Fibre Channel - Methodologies for Jitter and Signal Quality Specification - MJSQ," INCITS, Tech. Rep. REV 14, June 2004. [Online]. Available: http://www.t11.org/ftp/t11/member/fc/mjsq/04-101v4.pdf

[43] "Understanding and Characterizing Timing Jitter Primer," Tech. Rep. 55W-16146-5, Tektronix, 2012. [Online]. Available: http://www.tek.com/primer/understanding-and-characterizing-timing-jitter-primer

[44] "Jitter Analysis - Basic Classification of Jitter Components using Sampling Scope," App. Note MP2100A-E-F-3-(1.00), Tektronix, Jun. 2012. [Online]. Available: http://www.anritsu.com/en-US/Downloads/Application-Notes/Application-Note/DWL9658.aspx

[45] "Jitter Analysis: The dual-Dirac Model, RJ/DJ, and Q-Scale," White Paper 5989-3206EN, Agilent Technologies, Dec. 2004. [Online]. Available: http://cp.literature.agilent.com/litweb/pdf/5989-3206EN.pdf

[46] J. Ren, D. Oh, and S. Chang, "High-Speed I/O Jitter Modeling Methodologies," in *Proc. IEEE 19th Conf. on Electr. Perform. Electron. Packag. and Syst. (EPEPS)*, 2010, pp. 113–116.

[47] S. Chaudhuri *et al.*, "Jitter Amplification Characterization of Passive Clock Channels at 6.4 and 9.6 Gb/s," in *Proc. IEEE 15th Electr. Perform. Electron. Packag.*, 2006, pp. 21–24.

[48] C. Madden *et al.*, "Jitter Amplification Considerations for PCB Clock Channel Design," in *Proc. IEEE 16th Electr. Perform. Electron. Packag.*, 2007, pp. 135–138.

[49] A. Sanders, "Statistical Simulation of Physical Transmission Media," *IEEE Trans. Adv. Packag.*, vol. 32, no. 2, p. 260, 2009.

[50] K. S. Oh *et al.*, "Accurate System Voltage and Timing Margin Simulation in High-Speed I/O System Designs," *IEEE Trans. Adv. Packag.*, vol. 31, no. 4, p. 722, 2008.

[51] Y. Chang, D. Oh, and C. Madden, "Jitter Modeling in Statistical Link Simulation," in *Proc. IEEE Int. Symp. Electromagn. Compat.*, 2008, p. 1.

[52] B. Analui, J. Buckwalter, and A. Hajimiri, "Data-Dependent Jitter in Serial Communications," *IEEE Trans. Microw. Theory Tech.*, vol. 53, no. 11, pp. 3388–3397, 2005.

[53] *Matlab*, The MathWorks, Inc, 2010, release R2010b.

[54] B. Gustavsen and A. Semlyen, "Rational Approximation of Frequency Domain Responses by Vector Fitting," *IEEE Trans. Power Del.*, vol. 14, no. 3, pp. 1052–1061, 1999.

[55] B. Gustavsen, "Improving the Pole Relocating Properties of Vector Fitting," *IEEE Trans. Power Del.*, vol. 21, no. 3, pp. 1587–1592, 2006.

[56] *Titan User's Manual*, Infineon Technologies AG München, 2011, version 8.0.

[57] *Matlab Simulink*, The MathWorks, Inc, 2010, release R2010b.

[58] L. Bizjak *et al.*, "Comprehensive Behavioral Modeling of Conventional and Dual-Tuning PLLs," *IEEE Trans. Circuits Syst. I*, vol. 55, no. 6, pp. 1628 –1638, 2008.

[59] P. Alfke, "Efficient Shift Registers, LFSR Counters, and Long Pseudo-Random Sequence Generators," Xilinx, Tech. Rep. XAPP 052, Jul. 1996. [Online]. Available: www.xilinx.com/bvdocs/appnotes/xapp052.pdf

[60] Tektronix Inc. [Online]. Available: http://www.tek.com/bit-error-rate-tester/bertscope

[61] K. Fukuda *et al.*, "A 12.3-mW 12.5-Gb/s Complete Transceiver in 65-nm CMOS Process," *IEEE J. Solid-State Circuits*, vol. 45, no. 12, pp. 2838–2849, 2010.

[62] B. Ankele, "CMOS Devices - from Technology to CAD," Internal Training, Infineon Technologies Austria, 2013.

[63] K.-L. Wong *et al.*, “A 27-mW 3.6-Gb/s I/O transceiver,” *IEEE J. Solid-State Circuits*, vol. 39, no. 4, pp. 602–612, 2004.

[64] J. Poulton *et al.*, “A 14-mW 6.25-Gb/s Transceiver in 90-nm CMOS,” *IEEE J. Solid-State Circuits*, vol. 42, no. 12, pp. 2745–2757, 2007.

[65] C. Menolfi *et al.*, “A 16Gb/s Source-Series Terminated Transmitter in 65nm CMOS SOI,” in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2007, pp. 446–614.

[66] M. Kossel *et al.*, “A T-Coil-Enhanced 8.5 Gb/s High-Swing SST Transmitter in 65 nm Bulk CMOS With < -16 dB Return Loss Over 10 GHz Bandwidth,” *IEEE J. Solid-State Circuits*, vol. 43, no. 12, pp. 2905–2920, 2008.

[67] ——, “A T-Coil-Enhanced 8.5Gb/s High-Swing Source-Series-Terminated Transmitter in 65nm Bulk CMOS,” in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2008, pp. 110–599.

[68] J. Bulzacchelli *et al.*, “A 28-Gb/s 4-Tap FFE/15-Tap DFE Serial Link Transceiver in 32-nm SOI CMOS Technology,” *IEEE J. Solid-State Circuits*, vol. 47, no. 12, pp. 3232–3248, 2012.

[69] C. Menolfi *et al.*, “A 28Gb/s Source-Series Terminated TX in 32nm CMOS SOI,” in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2012, pp. 334–336.

[70] L. Selmi, D. Estreich, and B. Ricco, “Small-signal MMIC amplifiers with bridged T-coil matching networks,” *IEEE J. Solid-State Circuits*, vol. 27, no. 7, pp. 1093–1096, 1992.

[71] S. Galal and B. Razavi, “Broadband ESD Protection Circuits in CMOS Technology,” *IEEE J. Solid-State Circuits*, vol. 38, no. 12, pp. 2334–2340, 2003.

[72] W. Dettloff *et al.*, “A 32mW 7.4Gb/s Protocol-Agile Source-Series-Terminated Transmitter in 45nm CMOS SOI,” in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, 2010, pp. 370–371.

[73] V. Balan *et al.*, “A 4.8-6.4-Gb/s Serial Link for Backplane Applications Using Decision Feedback Equalization,” *IEEE J. Solid-State Circuits*, vol. 40, no. 9, pp. 1957–1967, 2005.

[74] J. Bulzacchelli *et al.*, “A 10-Gb/s 5-Tap DFE/4-Tap FFE Transceiver in 90-nm CMOS Technology,” *IEEE J. Solid-State Circuits*, vol. 41, no. 12, pp. 2885–2900, 2006.

[75] J. Kim, H. Hatamkhani, and C.-K. Yang, “A Large-Swing Transformer-Boosted Serial Link Transmitter With $>$ V_{DD} Swing,” *IEEE J. Solid-State Circuits*, vol. 42, no. 5, pp. 1131–1142, 2007.

[76] G. Balamurugan *et al.*, “A Scalable 5-15 Gbps, 14-75 mW Low-Power I/O Transceiver in 65 nm CMOS,” *IEEE J. Solid-State Circuits*, vol. 43, no. 4, pp. 1010–1019, 2008.

[77] E. Amerasekera and C. Duvvury, *ESD in Silicon Integrated Circuits*, 2nd ed. John Wiley & Sons, Ltd, 2002.

[78] G. Langguth *et al.*, “A Self Protecting RF Output with 2 kV HBM Hardness,” in *29th Electrical Overstress/Electrostatic Discharge Symposium (EOS/ESD)*, Sep. 2007, pp. 1A.2–1–1A.2–10.

[79] ——, “ESD Challenges in Advanced CMOS Systems on Chip,” in *IEEE International Conference on IC Design and Technology (ICICDT)*, Jun. 2010, pp. 29–34.

[80] B. Schaffer, "Design of linear voltage regulators," Internal Training, Infineon Technologies Austria.

[81] T. Carusone, D. Johns, and K. Martin, *Analog Integrated Circuit Design*, 2nd ed. John Wiley & Sons, 2012.

[82] *Bond Wire Modeling Standard*, EIA/JEDEC Std. EIA/JESD59, 1997. [Online]. Available: http://www.jedec.org/sites/default/files/docs/jesd59.pdf

[83] J. Frei, X.-D. Cai, and S. Muller, "Multiport S-Parameter and T-Parameter Conversion With Symmetry Extension," *IEEE Trans. Microw. Theory Tech.*, vol. 56, no. 11, pp. 2493–2504, Nov. 2008.

Acronyms

ADAS	Advanced Driver Assistance Systems
ADC	Analog to Digital Converter
BER	Bit Error Rate
BUJ	Bounded Uncorrelated Jitter
CAN	Controller Area Network
CDF	Cumulative Distribution Function
CDM	Charged Device Model
CDR	Clock and Data Recovery
CM	Current Mode
CML	Current Mode Logic
CPRI	Common Public Radio Interface
CTLE	Continuous Time Linear Equalization
DCD	Duty Cycle Distortion
DDJ	Data Dependent Jitter
DFE	Decision Feedback Equalization
DJ	Deterministic Jitter
ESD	Electrostatic Discharge
EVN	Equivalent Voltage Noise
FEXT	Far-End Crosstalk
FIR	Finite Impulse Response
GBP	Gain Bandwidth Product
HBM	Human Body Model
HSSI	High Speed Serial Interface
IIR	Infinite Impulse Response

ISI	Intersymbol Interference
LDO	Low Dropout
LFSR	Linear Feedback Shift Register
LIN	Local Interconnect Network
LTI	Linear and Time Invariant
MCU	Microcontroller Unit
NEXT	Near-End Crosstalk
NRZ	Non-Return to Zero
OCD	Off Chip Driver
OIF	Optical Internetworking Forum
PAM	Pulse Amplitude Modulation
PCB	Printed Circuit Board
PCIe	Peripheral Component Interconnect Express
PDF	Probability Distribution Function
PLL	Phase Locked Loop
PRBS	Pseudo Random Binary Sequence
PJ	Periodic Jitter
RJ	Random Jitter
SATA	Serial Advanced Technology Attachment
SBR	Single Bit Response
SRIO	Serial Rapid IO
SST	Source Series Terminated
TJ	Total Jitter
TTSCR	Transient Triggered Silicon Controlled Rectifier
UI	Unit Interval
VM	Voltage Mode
VNA	Vector Network Analyzer
XAUI	X Attachment Unit Interface

Candidate's Bio and Publications

Andrea Cristofoli was born in San Daniele del Friuli, Italy, in 1984. He received the *Laurea Magistrale* degree (Summa cum Laude) in Electronic Engineering from the University of Udine, Italy, in 2009, with a thesis focusing on radiation effects on the electrical performance of silicon particle detectors for high energy physics experiments.

In 2010 he was with the Department of Electric, Management and Mechanical Engineering (DIEGM) of the University of Udine as a contract researcher, working on radiation effects on silicon particle detectors for high energy physics. This activity partially developed in the framework of the "ATLAS 3D pixel R&D Collaboration", whose focus is on the silicon detector upgrade of the ATLAS-LHC experiment at CERN, Geneva, and in close collaboration with the Italian Nuclear Physics Institute (INFN) and the Center for Materials and MicroSystems at Fondazione Bruno Kessler (FBK), Trento.

The scientific activity focused on:

- measurement of the Impact Ionization coefficients of irradiated silicon using bipolar transistors irradiated in the TRIGA research nuclear reactor of Jozef Stefan Institut in Ljubljana;
- analysis of the breakdown performances of full-3D and 3D-DDTC detectors, before and after irradiation, using 2d/3d TCAD simulations and measurements on prototypes.

These investigations have been published in:

- **A. Cristofoli**, A. Dalla Costa, M. Boscardin, V. Cindro, G.F. Dalla Betta, F. Driussi, G. Giacomini, M.P. Giordani, P. Palestri, M. Povoli, S. Ronchin, E. Vianello, L. Selmi, "Simulations and electrical characterization of Double-side Double Type Column 3D detectors," *2011 IEEE Nuclear Science Symposium and Medical Imaging Conference (NSS/MIC)*, pp. 518-522, 23-29 Oct. 2011, Valencia (Spain);
- **A. Cristofoli**, P. Palestri, M.P. Giordani, V. Cindro, G.-F. Dalla Betta, L. Selmi, "Experimental Determination of the Impact Ionization Coefficients in Irradiated Silicon," *IEEE Transactions on Nuclear Science*, vol.58, no.4, pp.2091-2096, Aug. 2011.

As member of the 3D Pixel collaboration, the activity also contributed to the results in the following publications:

- P. Grenier, *et al.*, "Test beam results of 3D silicon pixel sensors for the ATLAS upgrade," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spectrometers, Detectors and Associated Equipment*, vol. 638, no. 1, pp. 33-40, May 2011.
- A. Micelli, *et al.*, "3D-FBK pixel sensors: Recent beam tests results with irradiated devices," *Nuclear Instruments and Methods in Physics Research Section A: Accelerators, Spec-*

trometers, Detectors and Associated Equipment, vol. 650, no. 1, pp. 150-157, September 2011.

Since 2011, he started working towards the Ph.D. degree in Electronic Engineering with DIEGM at University of Udine. The thesis focuses on high speed serial interfaces, with a particular attention to the requirements of electronics application for the automotive environment. The Ph.D. activity is done in close cooperation with Infineon Technologies A.G, Development Center Villach, Austria. Focus of the research is on the performance analysis of high-speed serial interfaces (Intersymbol Interference and Jitter) with the statistical simulation approach, the design of an innovative high speed CMOS transmitter and the experimental signal integrity characterization of prototypes. Results of the research have been published at international level in:

- **A. Cristofoli**, P. Palestri, L. Selmi, N. Da Dalt, "Improved Modeling of Intersymbol Interference in High Speed Serial Links," *8th Conference on Ph.D. Research in Microelectronics and Electronics (PRIME)*, 12-15 June 2012, Aachen, Germany.

- **A. Cristofoli**, P. Palestri, N. Da Dalt, L. Selmi, "Efficient Statistical Simulation of Intersymbol Interference and Jitter in High-Speed Serial Interfaces," *IEEE Trans. on Components, Packaging and Manufacturing Technology*, vol. 4, no. 3, pp.480-489, March 2014.

and at national level in:

- **A. Cristofoli**, P. Palestri, N. Da Dalt, L. Selmi, , "Improved Modeling of Intersymbol Interference in High Speed Serial Links," *Proceedings of GE2012, 44th Conference*, 20-22 June 2012, Marina di Carrara, Italy.

- **A. Cristofoli**, P. Palestri, N. Da Dalt, L. Selmi, "Experimental Verification of ISI and Jitter Modeling in High Speed Links," *Proceedings of GE2013, 45th Conference*, 19-21 June 2013, Udine, Italy.

He also collaborated with the ESD group of Infineon Technologies Münich to the validation of a novel ESD compact model for high speed I/Os. Results have been presented in:

- G. Langguth, W. Soldner, A. Ille, **A. Cristofoli**, M. Wendel, "Thyristor Compact Model for ESD, DC, and RF Simulation," *35th Annual EOS/ESD Symposium*, Las Vegas, USA.

Finally, he presented results from his research activity at:

- "Capacitance and Breakdown Measurements on Virgin and Neutron Irradiated DDTC and STC diodes," given at the *6th "Trento" Workshop on Advanced Silicon Radiation Detectors (3D and P-type Technologies)*, 2-4 March 2011, Trento, Italy.

- "Efficient Statistical Simulation of Intersymbol Interference and Jitter in High-Speed Serial Interfaces," given at the *2nd Infineon Technologies Austria AMP-S Design Symposium*, 11-12 April 2013, Villach, Austria.

- "Analysis and Design of High Speed Serial Interfaces," given at the *2013 PhD conference*, 27th June 2013, University of Udine, Udine, Italy.

Acknowledgments

The very first to thank are my tutors at University of Udine, Prof. Pierpaolo Palestri and Prof. Luca Selmi. I am profoundly grateful to Prof. Selmi for giving me the possibility to join the challenging world of the R&D in microelectronics. His constant spur helped me not to loose momentum along the way, and to keep as high as possible the technical content of my research. A huge thank goes in particular to Prof. Pierpaolo Palestri, whose guidance has been fundamental for my work during these three years. He has been of outstanding support for me with everyday enthusiasm and brilliant ideas. I will be always grateful to him.

I am grateful also to my supervisor at the Infineon Technologies Design Center Villach, Dr. Nicola Da Dalt. Thanks to his uncommon technical expertise, he has been a fundamental adviser during each step of the PhD work. His vision of the research directions has always been a source of motivation for my work.

This thesis would not have been possible without the enthusiastic support and help from all the members of the CIS team of Infineon Technologies Austria. I would like to thank in particular Peter Thurner, Dr. Roberto Nonis and Thomas Santa for the support and many fruitful discussions during the design of the high speed transmitter. I big thank also goes to Stefan Petschar and Roberto Aberjido, for the layout of the design, and to Dmytro Cherniak, Lino Alves and Florin Bulhac for the support during measurements in the lab. I want also to thank Dr. Paolo Toniutti of the DTI group of Infineon Technologies, for the useful insights about CMOS technology given during the design activity. To all of them I am grateful for their availability in sharing their technical knowledge with me and their patience in answering to all my questions.

Besides the technical support and guidance, I want to thank all the people I shared my days with in Villach and at the University of Udine. In particular I want to thank all the people of the "SelmiLab": they were able to make the lab an amusing place to work. Among all, a special thank goes to Alan, Marco, Francesco, Paolo, Daniel, Patrick, Federico, Alberto and Francesco. I am profoundly grateful to all of them for making me enjoying the PhD life, and to make the hard times easier to face.

A final thank to Roberto, Davide and Luca. I am grateful to them for helping me making my first steps in Villach and enjoy the living in Carithia. Their greatest merit is to have made me feel like home.