



POLITECNICO DI MILANO
Scuola di Ingegneria Industriale e dell'Informazione
LAUREA MAGISTRALE IN INGEGNERIA ELETTRONICA

High-Throughput Electronic Module for Multidimensional TCSPC Instruments

Advisor:
Prof. Ivan Rech

Tutor:
Ing. Luca Miari

Master Dissertation of:
Kaiyu Lin

Register Number:
783401

2012 - 2013

Contents

List of Figures	v
List of Tables	xi
Abstract	xiii
Sommario	xv
Introduction	xvii
1 Time-Correlated Single Photon Counting	1
1.1 TCSPC Principle	1
1.1.1 The Classic TCSPC Setup	2
1.1.2 Multidimensional TCSPC	4
1.2 Performance Evaluation of TCSPC Devices	6
1.3 TCSPC Applications	7
1.3.1 Fluorescence Decay Measurements	8
1.3.2 Diffuse Optical Tomography	9
1.3.3 Laser Scanning Microscopy	10
1.3.4 Time of Flight Measurement	11
1.4 TCSPC Systems State-of-Art	12
1.5 Single-Photon Avalanche Diode	13
1.6 Integrated Time-to-Amplitude Converter	15
2 1024-Channel TCSPC system	17
2.1 TCSPC systems evolution	17
2.2 Single-channel TCSPC System	17
2.3 8-channel TCSPC system	19
2.4 32-channel TCSPC system	21
2.4.1 Performance Comparison	21
2.5 System Throughput	22
2.6 1024-channel TCSPC module	25
2.6.1 Integrated Routing Circuit	25

2.6.2	Detection Head	25
2.6.3	1024-channel TCSPC board	26
2.6.4	Data Management Board	28
2.6.5	Power Management Board	28
2.6.6	Data Processing Board	29
3	Communications Protocol	31
3.1	Introduction to Universal Protocols	31
3.1.1	High-Speed Protocols	32
3.2	Gigabit Ethernet	39
3.2.1	Tri-Mode Ethernet Media Access Controller IP core	41
3.2.2	Ethernet 1000BASE-X PCS/PMA IP core	42
3.2.3	SFP+ Module, Optical Fiber and Network Interface Card	43
3.2.4	1GE Experimental Results	43
3.3	SuperSpeed USB 3.0	45
3.3.1	Cypress EZ-USB FX3	47
3.3.2	USB 3.0 Experimental Results	47
4	Data Management Board	51
4.1	Board Overview	51
4.2	Stop Signal Conditioning Stage	52
4.3	FPGA	54
4.4	USB 3.0 Controller	57
4.4.1	SPI master interface	59
4.4.2	I2C serial communication	60
4.5	SFP+ Daughter Board	61
4.6	Power Delivery Network	63
4.6.1	Ferrite Bead Filter Design	66
4.7	Mechanical considerations	71
5	Firmware and Software	73
5.1	Slave Fifo Interface: VHDL State Machine	73
5.2	1000BASE-X: VHDL implementation	76
5.3	GUI C# Software	78
6	Experimental Results	81
6.1	SuperSpeed USB 3.0 results	81
6.2	10GBASE-X Development Status	83
	Conclusions	85
	Bibliography	87

List of Figures

1.1	The classic TCSPC setup is made of a laser, a specimen and the detection electronics.	2
1.2	Example of how delays are used to build a histogram. (a) Counts and time channels of a typical TCSPC histogram. (b) Fluorescent photons excited by periodic laser pulses.	3
1.3	Reconstructed photon probability distribution and the original analog waveform.	4
1.4	The reversed start-stop configuration, the laser pulse acts as <i>stop</i> and the photon pulse as <i>start</i>	5
1.5	(a) The source time jitter directly impacts the system precision. (b) Exploiting a passive delay line on the source pulse, the source jitter is discarded.	5
1.6	Block diagram of a multi-dimensional TCSPC system. Data processing task is typically engaged by a Field Programmable Gate Array.	6
1.7	Ideal and real transfer functions between the delay time and the output analog voltage. Non-linear behavior leads to a distorted histogram.	7
1.8	Jablonski diagram: fluorescence emission from an excited molecule.	8
1.9	Example of a FLIM measurement (from reference [1]).	9
1.10	Circular arrangement of sources and detectors for optical tomography application.	9
1.11	Classic laser scanning microscope setup with galvanometer mirror.	10
1.12	3D map of a star shaped toy (from reference [2]) using time of flight measurement.	11
1.13	Simplified I–V characteristic of a SPAD, showing the three operating conditions. The x-axis refers to the reversed bias voltage across the SPAD.	14
1.14	(a) Picture of the 8x1 SPAD array developed under the <i>PARAFLUO</i> project. (b) Complete <i>Parafiuo</i> 8-channel module that includes the SPAD array.	15
1.15	Operating phases of the integrated TAC.	16
1.16	Micro-photograph of the manufactured 4-channel TAC array.	16

2.1	Roadmap of the TCSPC systems developed in our research group. . . .	17
2.2	Top and Bottom view of the single-channel TCSPC acquisition board. . .	18
2.3	The 8-channel ADC converts the TAC outputs and the FPGA samples and records the resulted digital values. The collected data are exported via USB to an external PC.	19
2.4	Top and Bottom view of the 8-channel TCSPC acquisition board.	20
2.5	Picture of the internal structure of the 32-channel TCSPC system.	21
2.6	Picture of the 32-channel single-photon detection head: the upper aluminum cover has been removed, to show the signal-processing board and the integrated arrays.	22
2.7	Picture of the original idea of the 32-channel TCSPC system.	24
2.8	Picture of the original idea of the 32-channel TCSPC system.	26
2.9	Rendering of the described detection head architecture with the new TSV technology. The SPAD array is connected to the bottom CMOS circuits through TSV links. Each detector has its own AQC, pick-up circuit, comparator and delay line while control logic and routing circuit are shared.	27
2.10	The two 8-channel ADC converts the TAC outputs and the FPGA samples and records the resulted digital values into two SRAM.	28
2.11	Rendering of the designed 1024-channel TCSPC module. Each board is square shaped of side 12 cm.	29
3.1	7 layers of the OSI model. Moving from one layer to the adjacent one, additional information is encoded/decoded according to the flow direction, e.g. transmission/reception.	32
3.2	Passive QSFP+ cable [3] and PCIe adapter [4] (courtesy of Siemon Co and Mellanox Technologies).	35
3.3	PCIe x4 cable with x4 connectors [5] and switch-switch based cable adapter [6] (courtesy of One Stop Systems).	36
3.4	Thunderbolt connector [7] and PCIe adapter [8] (courtesy of Apple Inc and ASUSTeK Computer Inc). The ThunderboltEX II PCIe adapter is supported only by few ASUS motherboards.	37
3.5	Ethernet frame and packet structure.	39
3.6	Block diagram of the 1000BASE-SR communication channel. One SFP+ optical module is attached to the Spartan 6 GTP transceivers on the SP605 Development Board while the remaining one is connected to the Network Interface Card on the PC.	41
3.7	Overview of the Data Link and Physical sublayers in the OSI model. . .	41

3.8	FPGA internal architecture. The PMA is connected to an external off-the-shelf SFP+ optical transceiver while the GMII of the Ethernet 1000BASE-X PCS/PMA is connected to the Xilinx Tri-Mode Ethernet MAC core.	42
3.9	E10GSFPSR [9] SFP+ module and E10G42BTDA [10] NIC (courtesy of Intel).	43
3.10	1000BASE-SR communication channel setup used to test the actual feasibility of this protocol. Once verified, it will be upgraded to the faster 10GBASE-SR.	44
3.11	Windows 7 network activity monitor shows a $\approx 84\%$ channel utilization.	44
3.12	Comparison between Hi-Speed USB and SuperSpeed USB over two IN transaction requests. (a) 6 packets needed to complete the requests in USB 2.0. (b) 5 packets needed to complete the requests in USB 3.0.	46
3.13	Block diagram of the FX3 internal architecture (courtesy of Cypress Semiconductor).	47
3.14	The Xilinx SP605 board is employed to generate a known pattern and to send it toward the Cypress CYUSB3KIT board through a custom bridge board. Finally, data is received on the PC and stored in a SSD using a C# software.	48
3.15	Custom C# software showing the USB 3.0 throughput in real time. In <i>BULK</i> transfer mode, 256 packets of 16 kB each are transferred per time, while the <i>Xfers to Queue</i> is dependent upon the DLL functions employed to begin each transfer.	49
4.1	Two 120-position Samtec connectors are placed in parallel on each 32-channel TCSPC board.	51
4.2	Block diagram of Data Management Board. The FPGA is in charge of buffering the gathered data toward the USB 3.0 controller and the SFP+ optical transceiver.	52
4.3	Stop conditioning stage: the external single-ended <i>stop</i> signal is converted into a pair of differential signaling pulses, one for each 32-channel TCSPC boards.	53
4.4	Arrangement of the components on the Data Management Board.	56
4.5	USB signals connected on the opposite side of the standard type-A USB receptacle. (a) component arrangement showing the SS TX/RX differential traces and plane cut-outs highlighted with dashed lines; (b) PCB cross-section view: the USB 3.0 type-A through-hole pin acts as a part of the signal trace, thus eliminating the possibility of a stub on the signal line.	59

4.6	SPI bus architecture: this solution allows to address one flash memory per time, using the already embedded SPI interface of the FX3 controller. <i>INIT_B</i> and <i>PROGRAM_B</i> configuration lines are not depicted.	60
4.7	FX3 I ² C bus and its slave devices split into 1.8 V and 3.3 V resources.	61
4.8	Section view of the Data management board. (a) The SFP+ assembly is meant to be installed on the PCB board edge: by doing so, the 1024-channel TCSPC system would have an inconvenient arrangement of connectors. (b) By employing a LC to LC fiber optic adapter, the Ethernet channel plug is placed on the same face of the other connectors. All measurements are in mm unless otherwise indicated.	62
4.9	SFP+ daughter board. (a) Top view. (b) Bottom view.	62
4.10	3D rendered image of the final assembly of the Data Management Board.	63
4.11	Power Delivery Network of the Data Management Board. A single 12 V domain is supplied from the Power Management Board, then filtered through the BNX016 [11] hence down-regulated into 5 V using a compact buck converter module. The inductor sign stands for a noise filtering stage designed with ferrite bead.	64
4.12	(a) Typical ZRX curve of a ferrite bead: the contiguous red line (Z) is the overall impedance behavior versus frequency. (b) First order approximation model of a ferrite bead.	67
4.13	Typical filter configuration: ferrite bead in conjunction with a bypass capacitor. R_{DP} and C_{DP} make up a compensation stage to reduce resonance issues.	67
4.14	Load side impedance versus frequency with one 4.7 μ F bypass capacitor. The resulting Z_{22} red line stays below 1 Ω until 400 MHz.	69
4.15	LTspice simulation results of the design ferrite bead filter. A commercially available value of $C_{DP} = 22 \mu$ F is chosen instead of the 24 μ F, but this slight modification does not influence the filter performance.	71
4.16	Section view of the Data management board: the connectors were carefully chosen to exhibit almost the same height. All measurements are in μ m unless otherwise indicated.	72
5.1	Synchronous Slave FIFO interface diagram implemented between the Cypress EZ-USB FX3 and the Xilinx Kintex-7 FPGA.	74
5.2	Interaction between FPGA and FX3 internal entities while performing a write operation.	75
5.3	FPGA State Machine for Stream IN operation.	76

5.4	Implementation of the 1000BASE-X core: an external 125 MHz clock is fed to the GTP transceiver block that forward out a copy for the all the remaining logic. The Pattern Generator is in charge of hard-coding the MAC addresses and the Ethernet frames. The interface between the two main cores is based on the Gigabit Media Independent Interface (GMII) along with a Management Data I/O control interface.	77
5.5	Scratch of the software flowchart that I intend to develop. The <i>IDLE</i> state is the key point: when the FX3 is in this condition, it awaits for user requests. Note that when the user choose to transfer data through the Ethernet channel, the FX3 is free to accept further instructions by returning in <i>IDLE</i> state.	79
6.1	Snapshot of the custom software I developed to receive and store data from the Cypress FX3. The displayed throughput is comparable with the one obtained during feasibility tests.	82
6.2	Screenshot of the Cypress C++ Streamer software that illustrates the real-time throughput by selecting <i>256 Packets per Xfer</i> and <i>64 Xfers to Queue</i> . (a) software run on Windows 7 64-bit featuring ASMedia Host Controller. (b) software run on Windows 8 64-bit featuring Intel USB 3.0 eXtensible Host Controller.	83
6.3	Implementation of the 1000BASE-X core: an external 125 MHz clock is fed to the GTX transceiver block that forward out a 62.5 MHz clock. A <i>MMCME2_ADV</i> primitive is employed to output two high quality global clocks for the remaining logic.	84
6.4	The Data Management Board and its main hardware component. (a) TOP layer. (b) BOTTOM layer.	86

List of Tables

1.1	State-of-art commercial systems with low number of channels.	12
1.2	State-of-art literature systems with low number of channels.	13
1.3	State-of-art literature and commercial systems with high number of channels.	13
2.1	Performance comparison among the systems develop in our research group.	22
3.1	Performance comparison among the communications protocol that match the 1024-channel TCSPC throughput requirement. Although the USB SuperSpeed 3.0 doesn't reach the minimum speed specification, it has been included in the table for completeness reason.	34
3.2	Variants of the 10GBASE Ethernet protocol.	37
3.3	Comparison between 10GBASE-SR and PCIe External Cabling figure of merit.	38
4.1	Values of the compensated voltage divider components.	53
4.2	Xilinx Kintex-7 XC7K325T-2FFG676C FPGA feature summary table. .	55
4.3	Connections among the FPGA and the four daughter boards. Each GTX transceiver is a combined transmitter and receiver.	55
4.4	The Data Management Board custom stackup and the resulting trace impedance. All measurements are in μm unless otherwise indicated. . .	66
6.1	PC components employed to test the transfer speed. Intel's host controller outperform Asus one by about 20%.	83

Abstract

Photoluminescence is the phenomenon of light emission from a substance after the absorption of photons. In many science fields, recovering the waveform of the original fluorescence pulse is of utmost importance. Time-correlated single-photon counting (TCSPC) technique is the ultimate answer: with picosecond accuracy, it allows the reconstruction of fast and faint decay curves. Time-tag recording mode is a variant of the TCSPC technique that, rather than shaping photon probability distributions, records the arrival time of each detected pulse both from the beginning of the experiment and within each stimulation period. A typical measurement involving the time-tag mode requires up to hundreds of megabytes transfer capabilities that, combined with modern multi-channel instruments, give rise to a demand of extremely fast communications protocols. In this work, a high performance Data Management Board featuring both SuperSpeed USB 3.0 and Ethernet 10GBASE-X links will be presented as part of a 1024-channel TCSPC instrument currently under development. This board acts primarily as the focusing point for two streams of data originated from two twin 32-channel TCSPC modules. Eventually, it is in charge of control tasks and mechanical support to the whole instrument. Nowadays, the USB 3.0 represents the most common connection between peripheral devices and PC. The Cypress EZ-USB FX3 is the controller that manages the SuperSpeed USB 3.0 link on the Data Management Board: in cooperation with a FPGA that gathers data from the TCSPC boards, they implement a FIFO interface allowing to download information at almost 400 MB/s. Despite its simplicity and bitrate, the USB 3.0 does not comply with the 1024-channel TCSPC throughput requirement. Therefore, it has been decided to develop a parallel communication channel featuring higher performance. The 10GBASE-X is a networking technology capable of delivering serialized data at a line rate of 10.3125 Gbit/s. By employing a Small Form-Factor Pluggable (SFP+) optical module in conjunction with high performance FPGA transceivers, the Data Management Board is capable of transferring data over fiber optics toward a PC up to 400 m away from the experiment setup.

Sommario

Per fluorescenza si intende l'emissione di fotoni da parte di un campione sottoposto ad eccitazione con luce nello spettro del visibile. La forma d'onda del segnale ottico riemesso è sfruttato in molte applicazioni, a partire dallo studio delle molecole biologiche fino ad arrivare alla mappatura spaziale di oggetti tridimensionali. La tecnica di misura di Time-Correlated Single-Photon Counting (TCSPC) consente di ricostruire queste curve di fluorescenza rivelando l'istante di arrivo di ogni singolo fotone ed elaborando un istogramma che rappresenta l'intensità del segnale luminoso. La modalità Time-Tag è una variante della tecnica TCSPC che registra l'istante di arrivo dei fotoni rispetto all'inizio dell'esperimento e il ritardo rispetto ad ogni impulso di eccitazione. Applicazioni come la Fluorescence Correlation Spectroscopy (FCS) e la Förster Resonance Energy Transfer (FRET) sfruttano il Time-Tag su un numero elevato di rivelatori, motivo per cui il nostro gruppo di ricerca si sta dedicando allo sviluppo di un sistema TCSPC a 1024 canali. In questo sistema, la grande mole di dati da inviare verso il PC ha richiesto lo studio dei più recenti protocolli di comunicazione veloce. In questo lavoro di tesi, dopo un'attenta analisi dei canali di trasmissione dati ad alte prestazioni, sono stati testati su kit di sviluppo il protocollo SuperSpeed USB 3.0 (5 Gbit/s) e l'Ethernet 1000BASE-X (1 Gbit/s), creando al contempo software appositi per la ricezione dei dati su PC. Infine, avendo verificato le loro integrabilità, sono stati implementati entrambi su un'unica scheda migliorando la trasmissione Ethernet alla versione 10GBASE-X (10 Gbit/s). Il controller EZ-USB FX3 della Cypress unitamente ad una FPGA Kintex-7 della Xilinx si occupa della comunicazione USB 3.0, mentre un modulo ottico Small Form-Factor Pluggable (SFP+) pilotato sempre dalla stessa FPGA si occupa del protocollo Ethernet. Oltre alla gestione del trasferimento dati, la scheda realizzata si occupa anche del controllo da remoto dell'intero sistema TCSPC, garantendone al contempo la stabilità meccanica.

Introduction

The phenomenon of photoluminescence has been known for over a century now, and the technology to employ it for spectroscopic applications is remarkably improving, especially within the last few decades. Biologists and chemists are continuously demanding more accurate electronic instruments, in particular for noninvasive experiments. Preserving the specimen in its natural environment is indeed the best approach to get precise results. This requirement has boosted research toward the development of high-performance photodetectors, along with processing electronics aimed to resolve fluorescent photon signals directly on-site.

Typically, photoluminescence signals last from picoseconds up to few nanoseconds that, associated with low sample concentration, give rise to extremely fast and faint response light. Electronic devices able to detect these signals are single-photon avalanche diodes (SPADs) and photo multiplier tubes (PMTs), however thanks to the higher quantum efficiency and lower jitter noise, the former ones are the most employed.

The Time-Correlated Single Photon Counting (TCSPC) technique is able to reconstruct the fluorescent photon probability distribution by arranging the arrival time of each light pulse into an histogram. Moreover, a time-tag mode is also available: in this operating condition, the system records only the arrival time of light pulses which comes in handy for experiments such as Fluorescence Correlation Spectroscopy (FCS) and Förster Resonance Energy Transfer (FRET) measurements.

A TCSPC instrument is characterized with three figures of merit: time resolution, in the order of few tens of picoseconds, differential non-linearity (DNL), typically of few percents of the histogram bin and finally conversion rate, usually up to MHz per channel. There are two main TCSPC architectures known in literature: the first one employs a time-to-digital converter (TDC) and the second one makes use of a time-to-amplitude converter (TAC) in conjunction with an analog-to-digital converter (ADC). Despite of its lower area occupation, the TDC solution shows much worse DNL; for this reason, our research group has chosen to focus on the TAC+ADC structure. Commercially available systems are developed to feature either high performance or high number of parallel channels, indeed due to technological limitations, it is not easy to overcome this trade-off. In this scenario, the work being done in our research laboratories is targeted to develop multichannel TCSPC instruments with outstanding characteristics. So far,

three modules have been developed: a single-channel used to validate the TAC+ADC architecture, an 8-channel to keep up with the existing instruments both cited in literature and commercially available, and at last a 32-channel to demonstrate the joint coexistence of good performances and large amount of channels, all within a compact system.

Accordingly to our development roadmap, the current system under investigation is being designed to feature 1024 SPAD detectors and TCSPC processing electronics, both integrated in a single stand-alone instrument. Due to such a large amount of parallel channels, data throughput is extremely high, especially in time-tag mode, i.e. up to tens of Gbit/s. The goal of this thesis is to handle this bitrate by directly transferring the raw data toward an external PC.

By investigating the fastest communications protocol cited in literature and commercially available, I came up with two solutions: SuperSpeed USB 3.0 and Ethernet 10GBASE-X. Following successful tests on development kits, I finally designed a custom board that implements both communication channels with the support of a Xilinx Kintex-7 FPGA: the SuperSpeed USB 3.0 link is managed by the Cypress EZ-USB FX3, whereas the Ethernet 10GBASE-X is controlled by the FPGA together with a Small Form-Factor Pluggable (SFP+) optical module that features a 850 nm VCSEL diode.

This board will work in conjunction with two twin 32-channel TCSPC modules: by gathering the two raw streams of data, it rearranges them before transferring toward the PC. Moreover, the development of a custom software that manages both data analysis and control tasks is part of this work.

The thesis is organized as follows: principle of the TCSPC technique will be presented in chapter 1, along with practical field applications and state-of-art systems; chapter 2 brings an overview of the current 1024-channel under development; a comparison among the most popular high-speed protocols will be shown in chapter 3, together with feasibility experiment results of the chosen communications link; chapter 4 describes in details the hardware components of the Data Management Board; in chapter 5 its firmware and software developed to manage the system are outlined; at last, chapter 6 shows the experimental results of the Data Management Board. Eventually, conclusions are drawn.

Chapter 1

Time-Correlated Single Photon Counting

The purpose of this chapter is to provide a brief overview of the Time-Correlated Single Photon Counting (TCSPC) technique. Its main objective is to present a broad understanding of the principle of operation and the typical system architecture. Performances and applications are also introduced, along with state-of-art devices, both commercial and reported in literature. The chapter concludes with a discussion of the core components of a TCSPC instrument.

1.1 TCSPC Principle

It is of common knowledge that the light has various natures described with many theories. Since we are interested in the interaction between light and matter, Max Planck's Quantum theory perfectly fits our case. The physicist named these *lumps* of light energy *quanta* (from a Latin word for *how much*) because they travel in finite amounts related to their frequency. These small packets of energy can be detected with proper sensors able to convert light into electrical signals.

Whenever is needed to recover the shape of the original light waveform in the time domain, we can refer to two main detection techniques. The first one is called *Analog Recording* and it carries out the signal amplitude information. Instead, the second one is called *Photon Counting*, and somehow it can be considered as a digital measurement since it provides information about the density of the light pulses. The *Analog Recording* technique takes shorter time to deliver results with respect to the *Photon Counting* one, but has at least two limitations. The first disadvantage is related to low light intensity circumstances, when the signal-to-noise ratio drops far below 1. In order to improve the signal-to-noise ratio (SNR), one could decrease the excitation rate and increase the peak power of the light source, but it is risky because by doing so the specimen could be permanently damaged. The second disadvantage is related to the bandwidth of the detector, e.g. the *instrument response function's* width, or IRF, cannot be shorter than

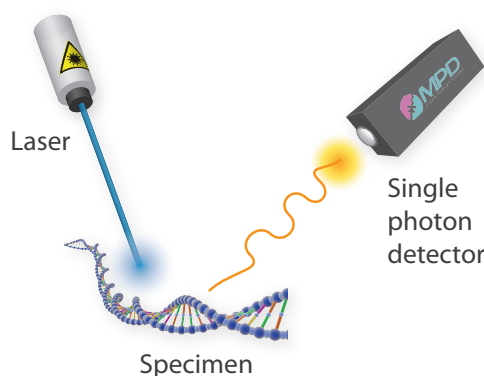


Figure 1.1: The classic TCSPC setup is made of a laser, a specimen and the detection electronics.

the *single electron response's* width, or SER, of the detector.

Since we need to capture information about weak and fast light pulses, the only choice left is the Photon Counting. Each pulse now represents the detection of an individual photon. The original waveform is reconstructed by gathering statistic information about photon arrival time into a histogram. Among the Photon Counting techniques, the Time-Correlated Single Photon Counting (TCSPC) is the one implemented in our detection systems. Limitation in SNR due to weak signals does not matter anymore with TCSPC, since it is possible to reveal every single photon; the only drawback is a longer measurement time. Instead, the device frequency response coincides now with the rise-time of the avalanche of the detector; see section 1.5.

1.1.1 The Classic TCSPC Setup

The standard TCSPC setup consists of an excitation source, typically a laser, a target sample to be excited and an active detection element in conjunction with acquisition electronics 1.1. The measurement starts when the specimen is hit by periodic laser pulses, aware of the fact that it could react by emitting itself a fluorescent photon. The basic assumption made by the TCSPC technique is that the probability of detecting more than one photon in a single cycle is negligible, as consequence it is not necessary to provide for the possibility of detecting several photons in one period. One may use the following rule of thumb to meet the former condition: on average, only one in 20-100 excitation pulses should generate a count at the detector [12]. Choosing the worst case, it simply means that the average count rate at the detector side should be at most 5% of the source rate. Nevertheless, many applications require excitation rates far below the statistic condition limit, hence the probability requirement does not influence the performances.

The delay between laser excitation and photon emission is measured by electronics that act like a stopwatch. If a single photon probability condition is met, there will

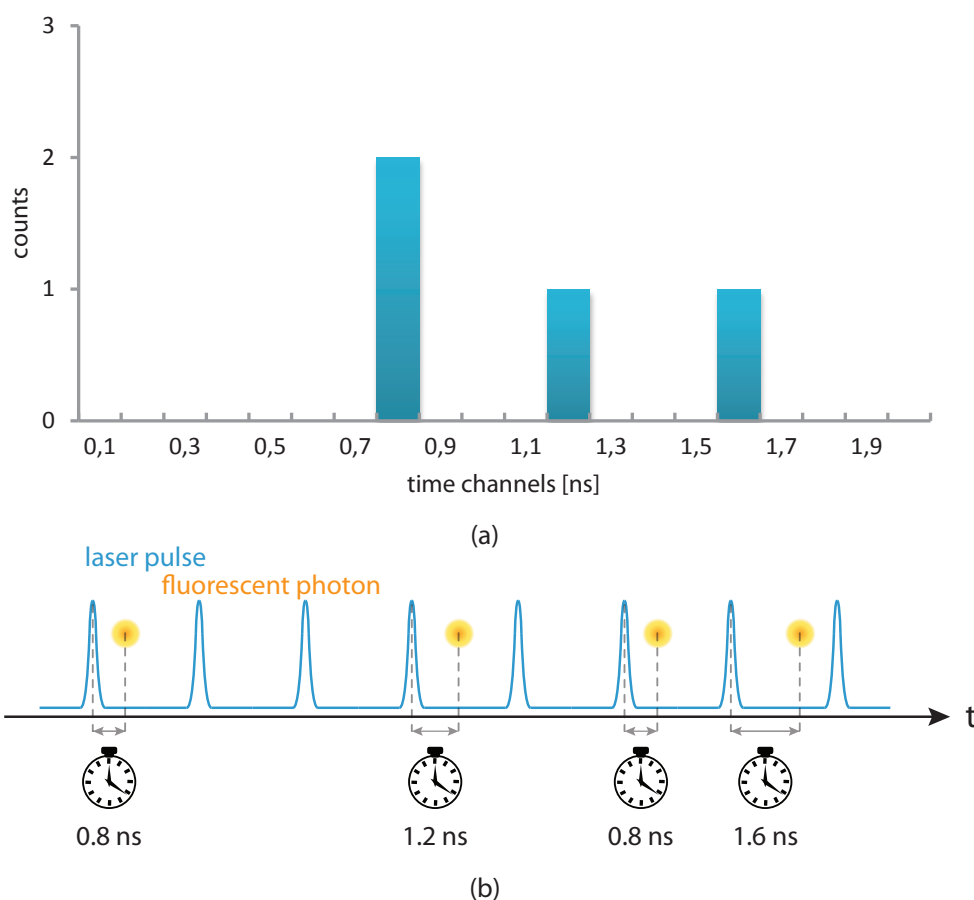


Figure 1.2: Example of how delays are used to build a histogram. (a) Counts and time channels of a typical TCSPC histogram. (b) Fluorescent photons excited by periodic laser pulses.

actually be no photons at all in many cycles [12] as depicted in figure 1.2 (b). Each delay is then stored in a memory, where every cell holds the photons counts for the corresponding time bin. These time bins are often referred to as time channels [13] (figure 1.2 (a)). After many photons, the histogram will replicate the probability distribution of the photon detection times; this distribution corresponds to the shape of the fluorescent optical signal as shown in figure 1.3.

In order to measure the time difference between the source and the fluorescence pulses, we would need at least two trigger signals. Let's suppose that the laser pulse acts as *start* and the photon pulse as *stop*. With this approach, the electronics is triggered for each cycle even if no photon is detected. Before starting a new conversion, a reset is needed which means that the device is unable to reveal any incoming light during this dead time. At high count rates, this reduces the detection efficiency, thus increasing the time needed for the measurement. Therefore, the reversed start-stop configuration [13] is typically employed (figure 1.4). As suggested by the name, the *start* pulse is now given by the emitted photon and the *stop* pulse is given by the reference signal from the

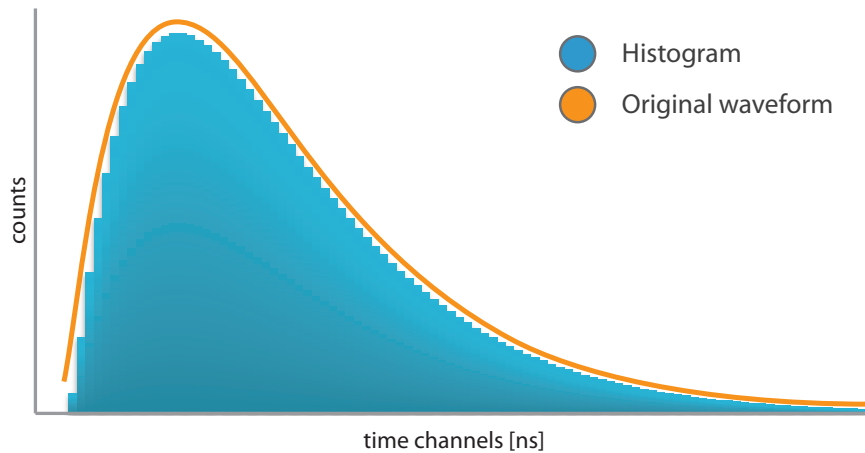


Figure 1.3: Reconstructed photon probability distribution and the original analog waveform.

light source. Now the electronics is triggered only by an incoming photon. However, two issues introduced by the reversed start-stop configuration need to be discussed. The first problem is the reversal of the time axis which can be easily compensated by simply inverting the conversion bits or by reversed readout of the stored data. The second and more tricky problem is that the source jitter directly impacts the system precision. By introducing a passive delay line on the reference pulse, the conversion is readily made insensitive to any frequency noise (figure 1.5).

1.1.2 Multidimensional TCSPC

The system shown so far is able to record information only in one dimension, i.e. determining the fluorescence lifetimes upon optical excitation. New generation of TCSPC devices implement multi-dimensionality, introducing revolutionary features such as detecting the wavelength, spatial coordinates, location within a scanning area or the time from the start of the experiment. Parallel acquisition from more than one detector is achieved by additional routing bits, which are used to address the memory cell that matches the corresponding sensor (figure 1.6). Examples are [14]:

- *Multi-detector operation*: several detector are connected to a router that sends their outputs to a single TCSPC device; the routing bits identify the detector that originated the timing information.
- *Multiplexed detection*: the optical signal is multiplexed in the time domain to a single detector; the routing bits identify the source of the optical signal.
- *Scanning*: the sample is scanned along two axis to obtain a 2D image; the scanner and the TCSPC device are synchronized and the routing bits represent the X-Y coordinates on the sample.

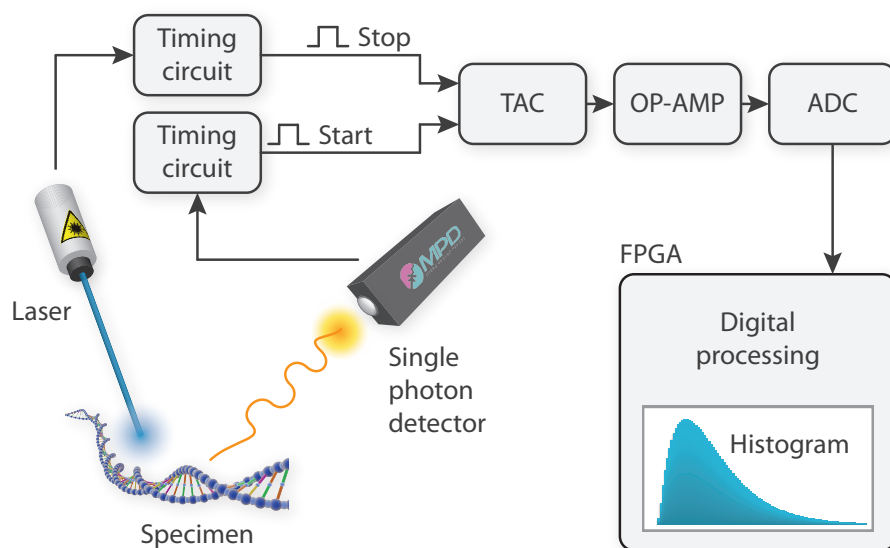


Figure 1.4: The reversed start-stop configuration, the laser pulse acts as *stop* and the photon pulse as *start*.

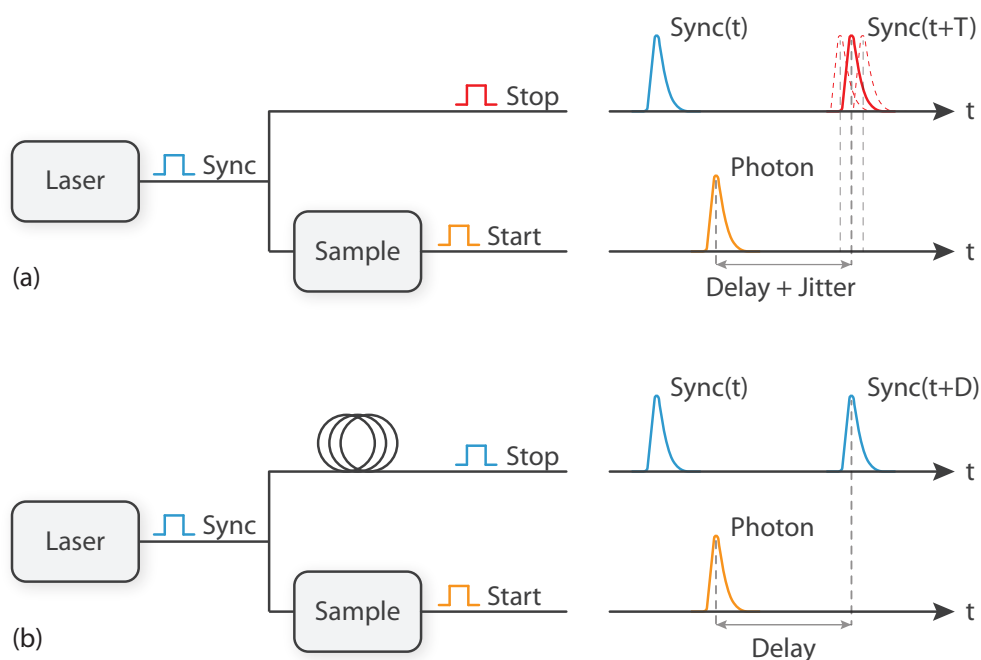


Figure 1.5: (a) The source time jitter directly impacts the system precision. (b) Exploiting a passive delay line on the source pulse, the source jitter is discarded.

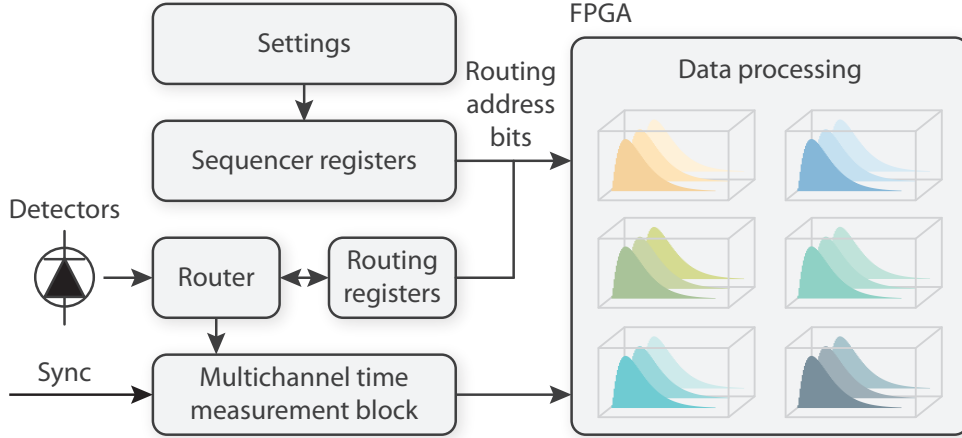


Figure 1.6: Block diagram of a multi-dimensional TCSPC system. Data processing task is typically engaged by a Field Programmable Gate Array.

- *Parametric detection*: several histograms are built up depending on the value of a set of parameters; the routing bits identify a particular set of values.
- *Sequential recording*: an oscillator is used to generate the routing bits, so that a sequence of individual measurement is generated and recorded.

One last TCSPC acquisition mode is the so called *time-tag* or *FIFO mode*. This technique doesn't build up any histogram related to photon density over time, instead it records only the arrival time of light pulses. Information about time is split into two parts, *micro* and *macro* time; the former records the delay in each excitation cycle and the latter records the delay from the beginning of the experiment for each incoming photon. In a standard experiment, data throughput is extremely high, up to tens of Gbit/s. The best way to handle such a high bitrate is to directly transfer the processed data toward an external PC. Typically a First In First Out type of memory is employed which allows to decouple different speed of data flow. This TCSPC technique is often used for experiments where recording the individual photon times makes possible to compute the auto-correlation curve of the light signal, such as in Fluorescence Correlation Spectroscopy measurements [15].

1.2 Performance Evaluation of TCSPC Devices

The performance of a complete TCSPC system can be summarized into three figures of merit: time resolution, differential non-linearity (DNL) and the conversion rate. An ideal electronic instrument should have an infinitely narrow IRF, modeled as a Dirac delta function. Any deviation from this theoretical result causes a broadening of the IRF, which can be quantified by specifying the rms error or the Full Width Half Maximum (FWHM) of the timing error distribution. The main contributions of

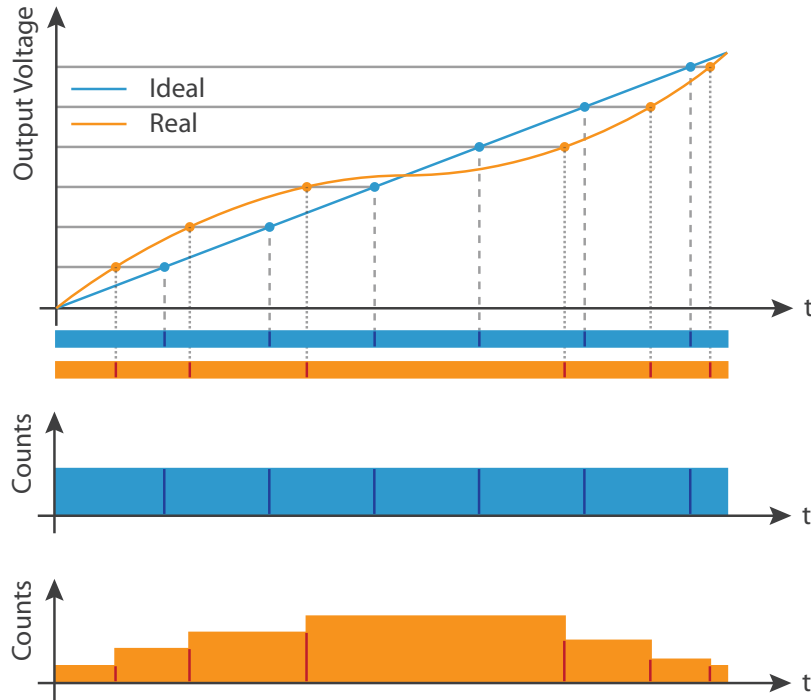


Figure 1.7: Ideal and real transfer functions between the delay time and the output analog voltage. Non-linear behavior leads to a distorted histogram.

inaccuracy are the detector, the laser source and electronic jitter. The final IRF is the convolution of all component IRFs given by the formula 1.1:

$$Resolution_{FWHM} = 2.35 \cdot \sqrt{\sum \sigma_{\Delta t}^2} \quad (1.1)$$

Quantization of analog signals also introduces random errors and it is characterized by the non-uniformity of time channels (figure 1.3). The effects of this kind of non-linear behavior is shown in figure 1.7. Supposing to feed the system with a uniform photon distribution, the output chart should be ideally a perfect rectangle. This is not the case of a non-linear transfer function between the delay time and the output analog voltage, indeed different bin widths means that not every delay slot has the same probability to receive a photon which leads to a distortion of the histogram. Finally, a good TCSPC device should be able to manage a high count rate, which is mostly limited by the reset phase.

1.3 TCSPC Applications

In recent years, a growing interest has arisen in non-invasive optical analysis, in particular, bio-chemists are continuously demanding faster and higher resolution systems. In this scenario, TCSPC technique is being used worldwide for applications such as

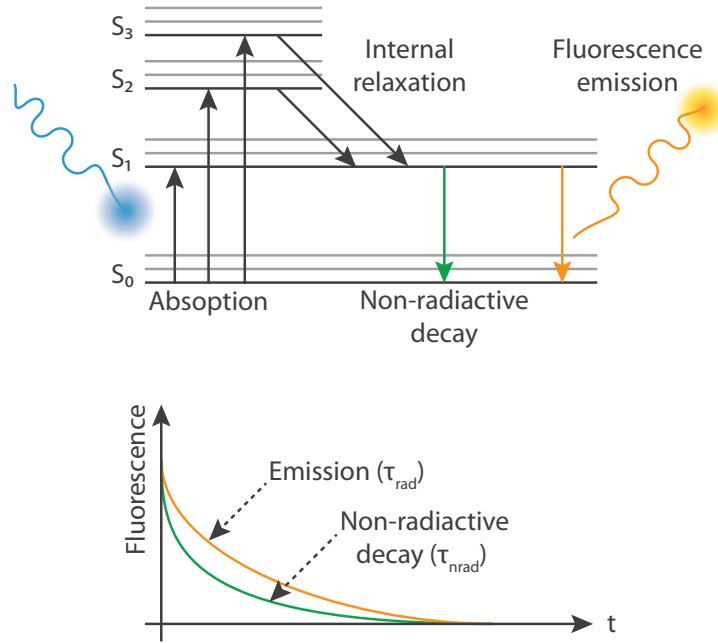


Figure 1.8: Jablonski diagram: fluorescence emission from an excited molecule.

fluorescence diffuse optical tomography (FDOT), laser scanning microscopy and time of flight measurements.

1.3.1 Fluorescence Decay Measurements

The quantum mechanical theory states that a particle can only take on certain discrete values of energy, called energy levels. An electron may jump from the lowest energy level, i.e. ground state, to a higher energy excited state. Fluorescence is the emission of electromagnetic radiation by a substance that has absorbed external energy. In most cases, the emitted light has longer wavelength with respect to the absorbed one. The fluorescence lifetime refers in particular to the average time the molecule stays in its excited state before emitting a photon; first-order kinetics law can be used to describe this behavior:

$$[S1] = [S0] e^{-t/\tau} \quad (1.2)$$

where $[S1]$ is the concentration of excited state particles at time t , $[S0]$ is the initial concentration and τ is the fluorescence lifetime. It is worth noting that energy can be released in various forms besides light, as a matter of fact non-radiative processes convert energy into heat or transfer it to another molecule. The corresponding lifetime of these side effects happen on a much faster timescale than the photon emission. The overall fluorescence time can be calculated from the following expression:

$$\frac{1}{\tau_{tot}} = \frac{1}{\tau_{rad}} + \frac{1}{\tau_{nrad}} \quad (1.3)$$

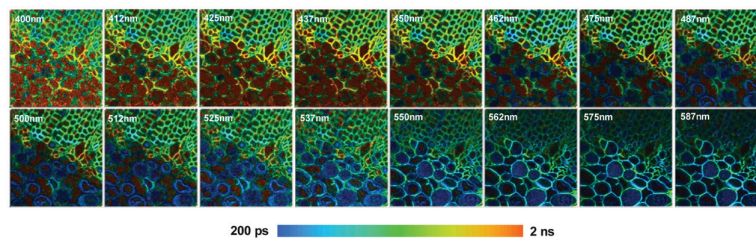


Figure 1.9: Example of a FLIM measurement (from reference [1]).

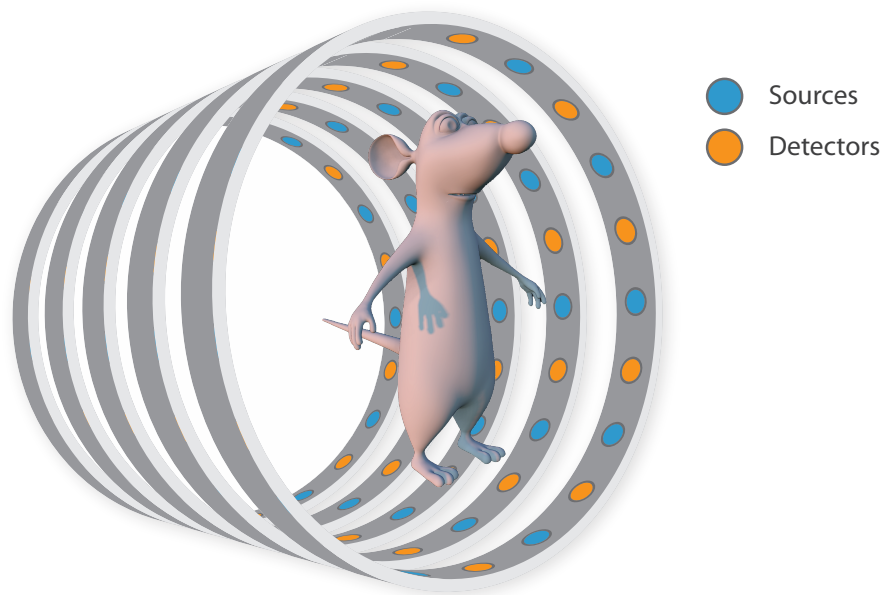


Figure 1.10: Circular arrangement of sources and detectors for optical tomography application.

All the concepts explained so far are shown in the Jablonski diagram [16] in figure 1.8.

1.3.2 Diffuse Optical Tomography

Optical tomography is a form of computed tomography that creates a digital volumetric model of an object by reconstructing images made from light transmitted and scattered through an object [17]. This technique is mostly used in medical imaging. Visible light, in the near infrared spectral region, is employed to measure the optical properties of physiological tissue. The DOT rely on the object under study being at least partially light-transmitting or translucent, so it works best on soft tissues such as breast and brain tissue. By monitoring spatial-temporal variations in the light absorption and scattering properties of tissue, regional variations in oxy- and deoxy-hemoglobin concentration as well as cellular scattering can be imaged [18]. Although the spatial resolution is limited when compared with other imaging modalities, such as magnetic resonance imaging (MRI) or X-ray computerized tomography (CT), DOT provides access to a variety of parameters that otherwise are not accessible. The typical DOT set-up, shown in

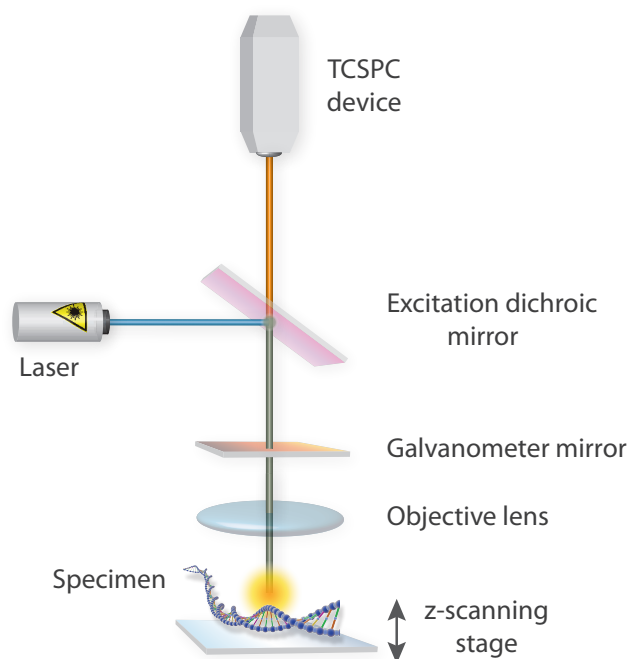


Figure 1.11: Classic laser scanning microscope setup with galvanometer mirror.

figure 1.10, consists in a large number of sources and detectors placed around the test subject. Modern TCSPC devices have all the features required by optical tomography applications [19]; multiplexing of laser excitation and multi-detector operation can be used to obtain few tens of parallel, time resolved acquisition channels and to exploit the high count rate to minimize acquisition times.

1.3.3 Laser Scanning Microscopy

The standard optical setup of a laser scanning microscope is shown in figure 1.11. The laser is fed into the optical path via a dichroic mirror and focused into the specimen by the microscope objective lens. Scanning is achieved by deflecting the beam by a galvanometer-driven mirror. After travelling back through the scanner, the beam of fluorescence light is stationary [1]. To obtain the desired S/N ratio, the scanning process is repeated a large number of times, so that any point of the sample is excited many times and its emission is detected and recorded [20]. Thanks to the efficient suppression of out-of-focus light, these microscopes are mainly used to reconstruct high contrast 3D images.

The fluorescence from organic molecules is characterized by its intensity, lifetime and spectra. By properly combining these three properties, one can retrieve useful information such as protein interaction, fluorophores identification etc. In recent years more features have been introduced to this technique, allowing multi-dimensional imaging, including excitation wavelength scanning, polarisation and second-harmonic imaging.

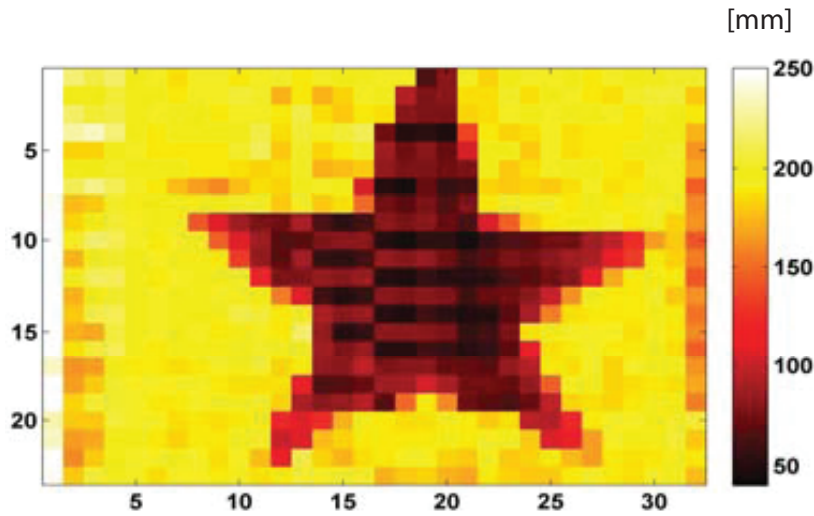


Figure 1.12: 3D map of a star shaped toy (from reference [2]) using time of flight measurement.

The former peculiarities make laser scanning microscopy perfectly suitable for steady-state fluorescence imaging of biological samples [21] [22] [23] [24].

1.3.4 Time of Flight Measurement

In the last few decades, the ability to acquire three-dimensional images and movies of a scene with very low illumination levels has become more and more important in many fields, like ambient surveillance, road safety, identification of people and objects, gaming, biomedical imaging, and studies on physics of materials.

A time-of-flight device is able to record the distance between an energy source and the target, by calculating the round trip time between the emitted and the reflected energy. Light sources are very common since its speed is known, hence the distance of the object can be measured with the following formula:

$$D = \frac{c \cdot T_M}{2} \quad (1.4)$$

Where c is the speed of light and T_M is the measured TOF. However, it must be pointed out that the pulse width determines the maximum range the system can handle, hence high performance lasers and detectors are compulsory.

The TCSPC technique can be easily adapted to TOF applications (figure 1.12). Picosecond-duration laser pulse is directed toward a non-cooperative target and single photon detectors are triggered by the scattered light. Moreover, applying the photon counting mode, the optical power can be considerably lowered yielding to safer measurements.

1.4 TCSPC Systems State-of-Art

	Becker & Hickl SPC-134 [25]	Becker & Hickl SPC-154 [26]	PicoQuant [27] HydraHarp 300	Picoquant [28] HydraHarp 400
Channels	4	4	2	8
FSR (s)	0.1m-2 μ	3.3n-5 μ	260n-33 μ	65n-2.19
Resolution(ps)	8	6.6	< 12	< 12
DNL (rms)	< 0.8%	< 0.5%	< 1%	< 0.2%
Conv. Rate (MHz)	10	10	10	12.5
Power consumption (W)	45	60	25	< 100
System volume (cm ³)	< 2390.6	< 2652	-	-

Table 1.1: State-of-art commercial systems with low number of channels.

1.4 TCSPC Systems State-of-Art

Commercially available TCSPC systems are developed to feature either high performance or high number of parallel channels, even the newest technologies cannot overcome this trade-off. Taking into account time resolution, DNL, conversion rate, power dissipation and area occupation as figure of merits, a brief comparison of state-of-art devices is shown in this section.

The highest performance in terms of resolution (FWHM) and conversion rate is achieved by Becker & Hickl GmbH and by PicoQuant GmbH. The former exploits single-channel [25] and multi-channel [26] TAC/ADC architecture, whereas the latter employs the TDC structure and provides a limited number of channels [27] [28]. The multi-channel option is obtained by connecting single-channel devices in parallel, nevertheless due to power consumption and area occupation, the maximum number of channels achievable is limited. The performance of these instruments are reported in table 1.1. Besides the commercial TCSPC products, there are few systems worth citing in literature (table 1.2).

Multi-channel systems with embedded detection head, lack of good performances, the only commercially available device in table 1.3 is produced by *Princeton Lightwave* [33]. It is also worth speaking about the European *Megaframe project*, which objective is the fabrication of active pixel 2D-arrays exploiting highly scaled technologies. The Megaframe 128 x 128 pixel prototype is being designed to sustain 1,000,000 frames per second with 50 ps time uncertainty.

1.5 Single-Photon Avalanche Diode

	Resnati et al. [29]	Keränen et al. [30]	Jansson et al. [31]	Markovic et al. [32]
Channels	1 (TAC)	1 (TDC+TAC)	7 (TDC)	1 (TDC)
FSR (s)	50n	328 μ	74 μ	160n
Resolution (ps)	60	4	19	36
DNL (rms)	< 0.5%	-	> 70.7%	< 1.5%
Conv. Rate (MHz)	21	-	-	6.6
Power consumption (W)	60m	5.8	85m	80m
System area (mm ²)	< 2.5	< 10 ⁴	8.88	4.2

Table 1.2: State-of-art literature systems with low number of channels.

	Princeton Lightwave [33]	Stoppa et al. [34]	Veerappan et al. [35]	Niclass et al. [36]	Villa et al. [37]
Channels	128x32	32x32	160x128	32 (TDC)	32x32
FSR (ns)	4–40 · 10 ³	20	55	100	320
Resolution (ps)	< 1.1 · 10 ³	600	140	LSB = 97	413
DNL (peak)	-	35%	30%	8%	< 4.9% rms
Conv. Rate (MHz)	72 kframe/s	500 kframe/s	10	10	20 · 10 ⁻³
Power consumption (W)	20	300 μ / pixel	550m	150m	3
System area (mm ²)	< 900 (cm ³)	2.56	135.3	40	-

Table 1.3: State-of-art literature and commercial systems with high number of channels.

1.5 Single-Photon Avalanche Diode

So far, the working principles of a TCSPC system have been explained from a general point of view, however the actual conversion from light signal into electrical one needs to be described more in details.

In any signal acquisition chain, the most critical part is the transducer. Whatever the energy source (mechanical, chemical, optical etc.), the sensor must be able to provide a reasonable signal to noise ratio to the attached processing block. In a TCSPC device, the photodetector is the front-element of the chain. Two kind of detector are usually employed, both characterized by single photon sensitivity, low transit-time and high quantum efficiency: Photomultiplier tubes (PMTs) and Single Photon Avalanche Diodes (SPADs). PMTs are able to multiply the current produced by incident light by as much

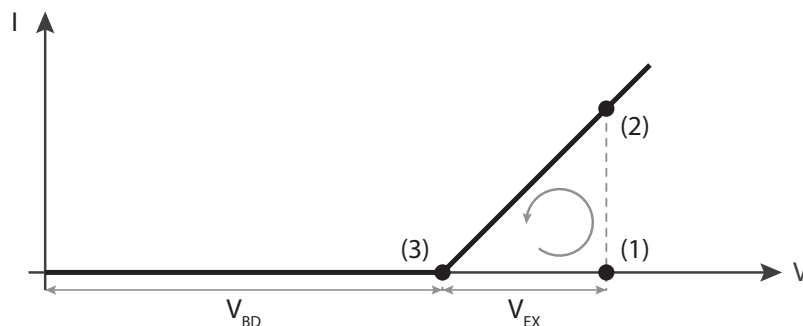


Figure 1.13: Simplified I–V characteristic of a SPAD, showing the three operating conditions. The x-axis refers to the reversed bias voltage across the SPAD.

as 100 million times (i.e. 160 dB). The combination of low noise, high gain and frequency response has earned photomultipliers an essential place in many applications. Nevertheless, random amplitude jitter of their output signal makes them not suitable for timing purposes, quite apart from the fact that they are fragile and energy wasting. SPADs operating in Geiger-mode provides instead a digital signal per photon detected [38]; however, they require demanding electronics such as a suitable quenching circuit to reset the avalanche. This device can be described as a p-n junction reverse biased above the breakdown voltage.

Whenever a carrier reaches the multiplication region during this quiescent state, a high probability of sequential impact ionization mechanism can occur due to the high electric field. As a result, a self-sustained avalanche is triggered and a macroscopic current can be read out.

Once the avalanche is on, the SPAD is not able to detect further incoming photons. To restore the quiescent state, a quenching circuit [38] is employed which task is to bias the device below its breakdown voltage. Ideally, the SPAD should be sensitive only to electron-hole pairs generated by photon absorption. Unfortunately, this is not the case because also thermally generated and trapped pairs can trigger avalanches.

The operation phases of a SPAD are illustrated in figure 1.13. During the quiescent phase (1), the device is biased with a reverse voltage equal to $V_{BD} + V_{EX}$, being V_{EX} the excess bias voltage with respect to the breakdown one V_{BD} . Once the avalanche is triggered, current flows through the detector (2), until the quenching circuit reacts by lowering the bias voltage below breakdown (3). Finally, the quiescent condition (1) is restored again. Custom technologies for SPAD fabrication are currently under investigation in our research group. Results of our studies can be seen in the *Parafluo project* [39] [40], which module features 8-channel and extremely high time resolution (figure 1.14).

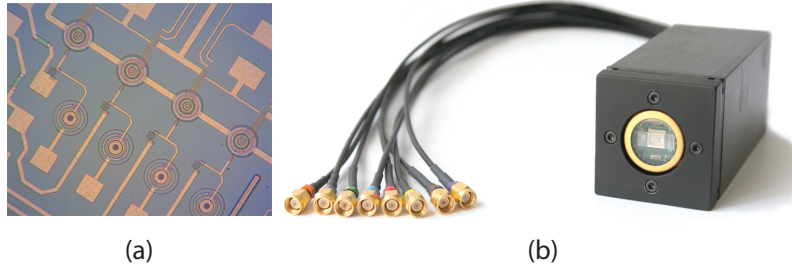


Figure 1.14: (a) Picture of the 8x1 SPAD array developed under the *PARAFLUO* project. (b) Complete *Parafluo* 8-channel module that includes the SPAD array.

1.6 Integrated Time-to-Amplitude Converter

The time interval between the arrival instant of a photon and the reference pulse from the light source (e.g. the trigger signal of laser diode) is measured by means of a timing block that may exploit two main architectures: TAC+ADC [41] or TDC [31]. Due to the better performances, especially in terms of DNL, the TCSPC systems developed in our research group employ the first structure. Few prototypes have been already designed in a SI-Ge BiCMOS 0.35 μm technology [41]. The idea behind the TAC is extremely simple: a constant current (I_{CONV}) triggered by the *start-stop* signals charges a capacitor (C_{CONV}), thus the voltage (V_{OUT}) across the capacitor increases linearly over time:

$$V_{\text{OUT}} = V_0 + \frac{I_{\text{CONV}}}{C_{\text{CONV}}} \cdot (T_{\text{stop}} - T_{\text{start}}) \quad (1.5)$$

After the conversion, the capacitor is discharged and the TAC is reset. Although the operating principle is straightforward, the actual structure of the time-to-amplitude converter is much more complicated since high performance are requested, such as temporal resolution, differential non-linearity, stability against temperature and supply voltage noise. The operating principle can be divided into four stages, as shown in the diagram reported in figure 1.15. The entry point of the *Finite State Machine* is the *idle phase*, where the converter is awaiting for the arrival of a *start* signal. When this arrives, the *conversion phase* is reached. From this point on, three possible conditions can occur: if the conversion interval is too short (e.g. the *stop* signal arrives too early, *under-range* condition) or too long (e.g. the *stop* signal does not appear within the FSR of the TAC, *over-range* condition), the control logic triggers the internal reset (*reset phase*); instead, if the *stop* signal arrives within the valid range, the output voltage is held constant (*wait phase*) allowing the ADC to sample it properly. When a valid conversion is achieved, an external trigger brings the state machine into the *reset phase*. Finally, the entry point is reached again making the TAC ready for a new conversion. A micro-photograph of the integrated 4-channel TAC array (presented by Crotti et al. in [41]) is shown in figure 1.16. From the picture, you can also notice the presence of a D/A converter and four adder stages. The integrated digital-to-analog

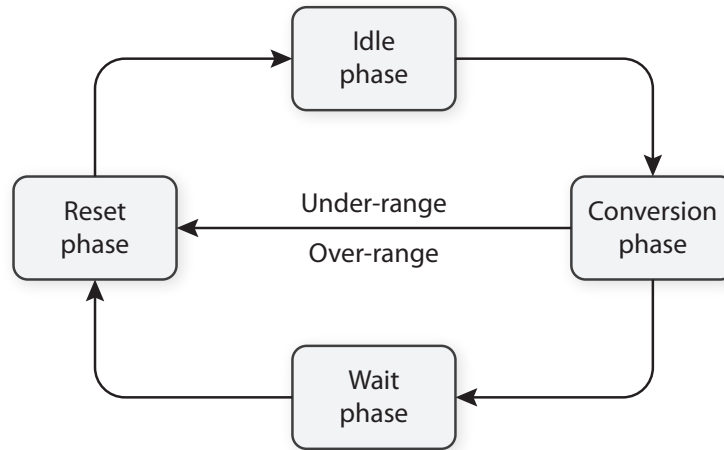


Figure 1.15: Operating phases of the integrated TAC.

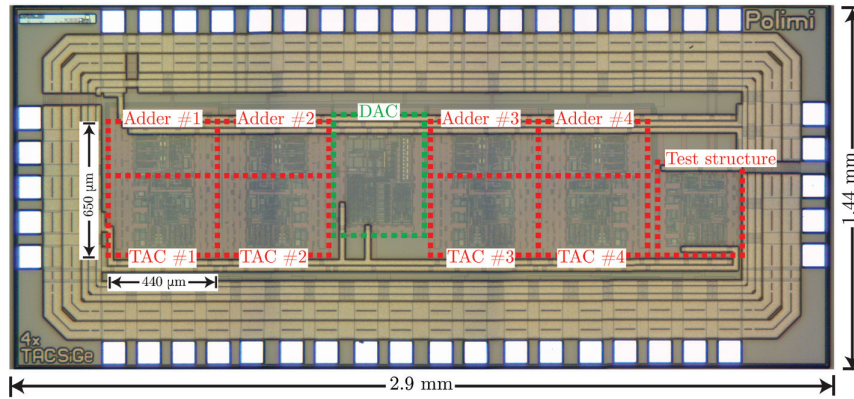


Figure 1.16: Micro-photograph of the manufactured 4-channel TAC array.

convert is employed to generate a dithering signal to be summed to the TAC output by means of the adders, thus reducing the overall system DNL. The 4-channel TAC is characterized by an excellent time resolution (less than 50 ps FWHM), low DNL (less than 0.3% LSB rms), high counting rate (16 MHz), low and constant power dissipation (50 mW) and low area occupation ($2.58 \times 1.28 \text{ mm}^2$).

Chapter 2

1024-Channel TCSPC system

The intent of this chapter is to present the fascinating evolution of the TCSPC systems developed by our research group. The first three sections are dedicated to show an overview of the single, 8 and 32-channel instruments along with their features, components and mating boards. Thereafter a performance comparison is made among them, highlighting the key figure of merits. The chapter concludes with an introduction to the 1024-channel TCSPC system currently under development.

2.1 TCSPC systems evolution

In the previous Chapter, it has been pointed out that TCSPC systems are stuck in the trade-off between multi-dimensionality and performance. Indeed, even with the most advanced technologies, commercial devices do not feature good characteristics blended into a multi-channel instrument. The work done by our research group is aimed to overcome this trade-off: so far a 32-channel TCSPC system has been developed and tested in field applications, the next big step is toward a 1024-channel stand-alone module (figure 2.1).

2.2 Single-channel TCSPC System

The single-channel system was developed to test the first integrated TAC circuit [42] in conjunction with the ADC-FPGA architecture. It is implemented on a 95 x 40



Figure 2.1: Roadmap of the TCSPC systems developed in our research group.

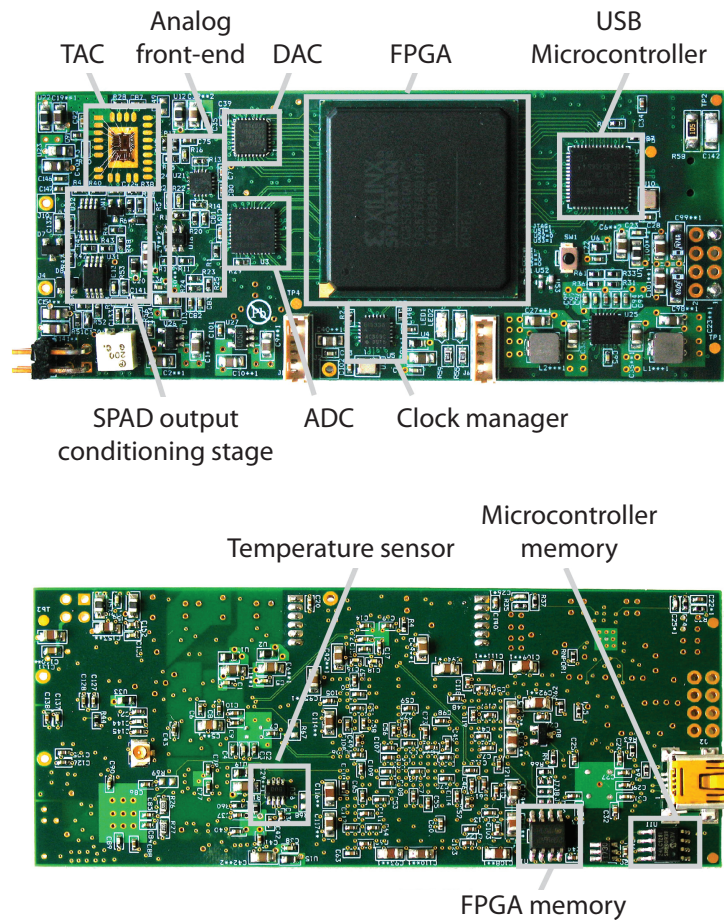


Figure 2.2: Top and Bottom view of the single-channel TCSPC acquisition board.

mm², 8-layer PCB board on which several power planes are used to avoid electrical crosstalk between the analog conditioning stages of the SPADs and the digital processing blocks [14]. The dithering [43] technique was also exploited to lower non-linearity effects (see section 1.2) of the TAC and the ADC through a commercial DAC, in particular the *sliding scale* [44] one. Processed data are downloaded via Hi-Speed USB 2.0 and managed by a custom Visual C# software on external PC.

The system dimensions and autonomy were prerogatives for the design, hence the FPGA size, in terms of memory capability, was chosen such that the device could work as a stand-alone system. The Hi-Speed USB 2.0 interface is managed through a microcontroller that embeds a dedicated USB transceiver. The chosen device is the CY7C68013A, from Cypress Semiconductor [45].

The single-channel system works with a single 5.5 V DC power supply and features a very low power consumption (< 2.5 W). A picture of the described single-channel board is shown in figure 2.2 in which the main on-board components are highlighted.

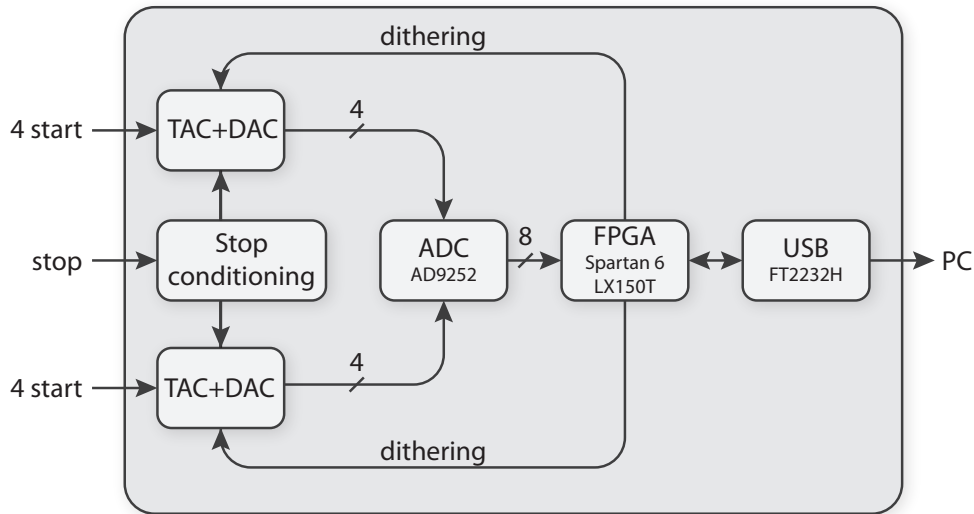


Figure 2.3: The 8-channel ADC converts the TAC outputs and the FPGA samples and records the resulted digital values. The collected data are exported via USB to an external PC.

2.3 8-channel TCSPC system

Starting from the architecture developed in the single-channel device, an 8-channel TCSPC system has been engineered keeping the same area occupation and power consumption (6 W). Once again, it is made on an 8-layer PCB board, mounting two 4-channel TAC arrays and a single 8-channel commercial ADC. On the contrary, the commercial DAC in charge of implementing the dithering technique has been replaced now with a integrated one. Due to a slower transfer rate and challenging firmware implementation for the CY7C68013A, communication tasks are now left to the FT2232H, from FTDI Chip [46].

Besides the histogram mode, this new TCSPC module is capable of operating in time-tag mode. However, this feature comes with a limitation: Hi-Speed USB 2.0 reaches effective maximum signaling rate of 280 Mbit/s , instead the FT2232H it is limited to 240 Mbit/s which corresponds to 625 kcps per channel. Whenever this throughput boundary is hit, any additional data is lost. It is clear that the USB 2.0 transfer rate represents a strong limitation to the SPADs maximum count rate.

The 8-channel device works with a single (8–16) V DC power supply and absorbs less than 6 W. A picture of the described 8-channel board is shown in figure 2.4. An additional interface board has been designed to mate the 8-channel system. The main task of this board is to convert single-ended, NIM/TTL standard, timing signals to the differential TAC inputs. Auxiliary functions, such as temperature control and power management are also included.

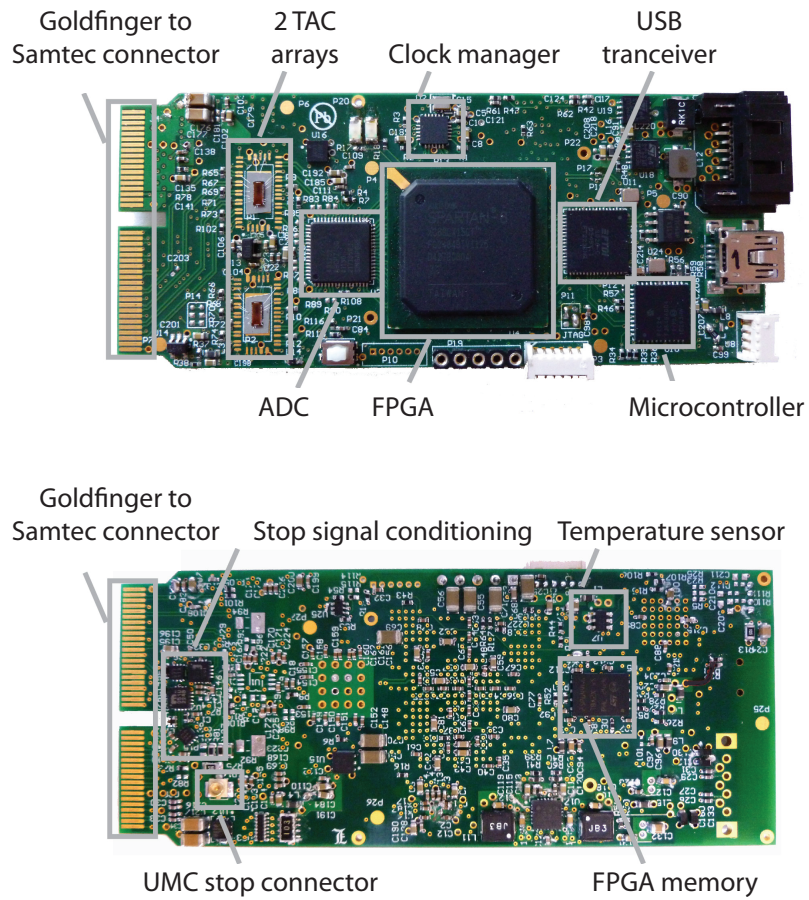


Figure 2.4: Top and Bottom view of the 8-channel TCSPC acquisition board.

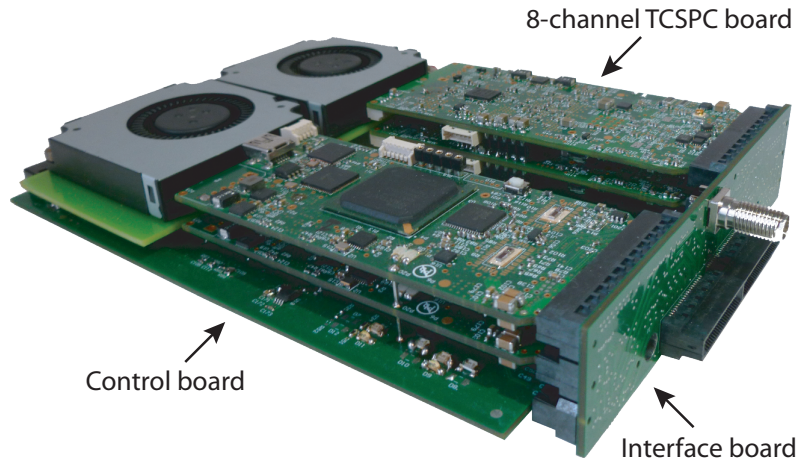


Figure 2.5: Picture of the internal structure of the 32-channel TCSPC system.

2.4 32-channel TCSPC system

The small form factor of the 8-channel TCSPC board allowed us to parallelize four of them in one compact ($160 \times 125 \times 30 \text{ mm}^3$) stand-alone system. Input photon-timing signals are routed through an interface board and each TCSPC device is able to work independently; finally an USB 2.0 hub (USB2517, from Microchip [47]) placed on a control unit gathers the 4 independent data streams and convey them to an external PC through a single port (figure 2.5). The USB digital bandwidth is now shared among 4 TCSPC boards, limiting even more the count rate per single channel which results equal to 156.25 kcps. This module is also equipped with a internal SATA interface and a 1.8" SSD slot for fast and independent data storage. Furthermore, a custom PC application has been made for this system which is able to work either in histogram or time-tag mode.

The 32-channel system has been designed to work with a single (8–16) V DC power supply, consuming less than 30 W. Together with this device, a photon detection head shown in figure 2.6 was also developed, featuring a custom-technology 32×1 linear SPAD array. The two modules are paired by a 32-differential-pairs cable, making up a complete and stand-alone TCSPC system.

2.4.1 Performance Comparison

The former systems have been tested, characterizing their TCSPC performance. Each testing setup was build carefully, considering the modules as stand-alone TCSPC systems, hence avoiding any possible additional jitter and noise sources. Among the parameters reported in table 2.1, the conversion rate has been calculated taking into account the TAC and the FPGA lag time, which results from the following formula:

$$T_{\text{rate}} = T_{\text{FSR}} + T_{\text{strobe}} + T_{\text{FPGA}} + T_{\text{reset}} \quad (2.1)$$

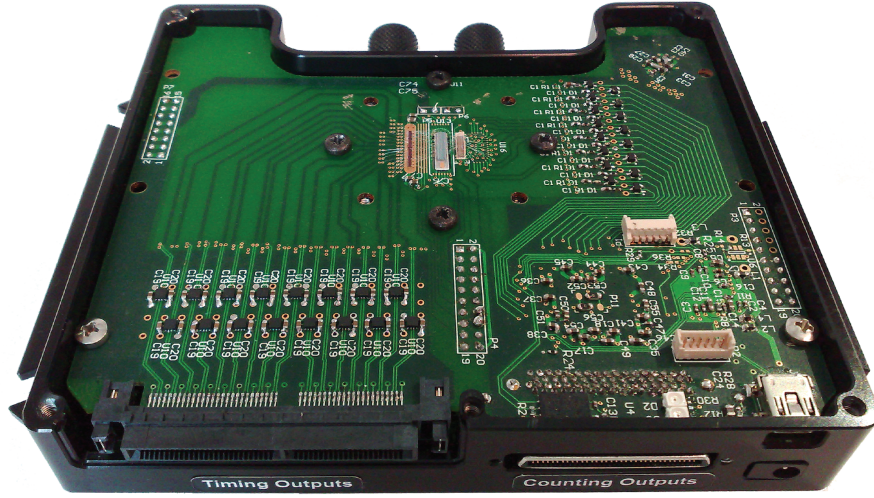


Figure 2.6: Picture of the 32-channel single-photon detection head: the upper aluminum cover has been removed, to show the signal-processing board and the integrated arrays.

	Single-channel [48]	8-channel [49]	32-channel [50]
FSR (s)	50n	(11–88)n	(11–88)n
Resolution (ps)	46	18	18
DNL (rms)	< 0.1%	< 0.1%	< 0.15%
Conv. Rate (MHz)	4	5	5
Power consumption (W)	< 2.5	< 6	< 30
System area (mm ²)	$3.8 \cdot 10^3$	$< 20 \cdot 10^3$	$20 \cdot 10^3$
Throughput (MB/s)	< 10	30	30

Table 2.1: Performance comparison among the systems develop in our research group.

where T_{FSR} is the maximum *start-stop* delay time, T_{strobe} the wait time before the assertion of the *strobe* for a valid conversion, T_{FPGA} the time needed for the FPGA to sample the *strobe* and manage the *reset* signal and finally T_{reset} being the TAC reset time [14]. The figures in table 2.1 are related to the best performing conditions.

2.5 System Throughput

In a TCSPC system characterized by many parallel channels, the throughput bottleneck is mainly due to the digital processing electronics downstreaming the TAC. The three devices shown in the previous sections, implemented the Hi-Speed USB 2.0 interface which, as mentioned before, is a big limitation to the system bandwidth. For sake

of clarity, few bitrate calculations related to the 32-channel device are shown in the following part, both for histogram and time-tag mode. The following formula works for histogram mode:

$$FrameSize = (n + k \cdot 2^m) \cdot N \quad (2.2)$$

where $FrameSize$ is the dimension of the overall frame to be sent to the PC, n is the number of bits required to address the source channel, k is the number of the bits that determines the maximum counts per ADC time channel, m is the number of the ADC bits and finally N is the system number of channels. By properly replacing these parameters with the ones used in the 32-channel system, we obtain:

$$FrameSize = (5 + 24 \cdot 2^{14}) \cdot 32 \approx 1.6 \text{ MB} \quad (2.3)$$

which corresponds to a refresh rate of $(30 \text{ MB} / 1.6 \text{ MB}) \approx 19 \text{ fps}$ using the FT2232H. This result tells us that the Hi-Speed USB 2.0 is adequate to histogram mode operations. Instead, time-tag mode requires much more bandwidth for transferring data as demonstrated by the following equation:

$$Throughput = (n + m + M) \cdot f \cdot N \quad (2.4)$$

where n is the number of bits required to address the detector, m is the number of the ADC bit related to the *microtime*, M is the number of bit related to the *macrotime* which is arbitrary, f is the TAC conversion rate and N is the overall number of available channels. Substituting the values, we get:

$$Throughput = (5 + 14 + 38) \cdot 5\text{MHz} \cdot 32 = 1.14 \text{ GB/s} \quad (2.5)$$

It is worth noting that M can be set arbitrary accordingly to the measurement, indeed it determines the maximum duration allowed per experiment:

$$Duration = 2^M / (F) = 2^{38} / (80\text{MHz}) \approx 3436 \text{ s} \quad (2.6)$$

where F is the laser synchronism frequency. However, such a high bitrate can be improved by applying a smart compression: supposing to have a high count rate, e.g. high probability of detecting a fluorescent photon per laser excitation, we could retrieve the *macrotime* information by counting the time difference among sequential photons, rather than providing the *macrotime* itself in every time-tag frame. Hence, by establishing a count rate threshold (Δt), we can either include the *macrotime* when the threshold is not reached or we can provide the time difference between two adjacent pulses [51]. The former idea is summarized into the following equations:

$$Throughput1 = (D + n + m + M) / \Delta t \quad (2.7)$$

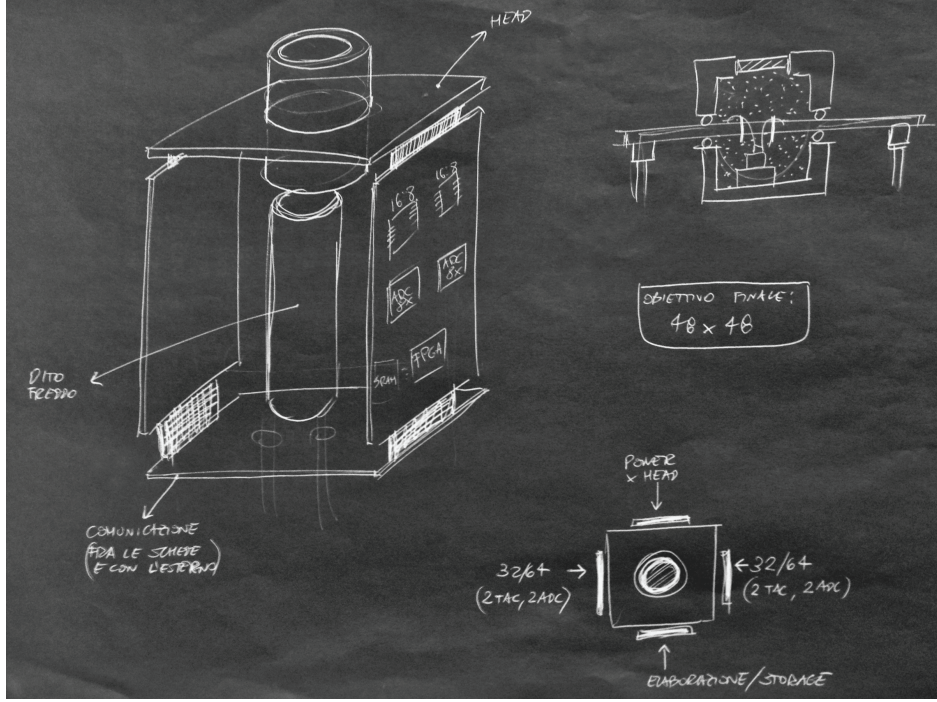


Figure 2.7: Picture of the original idea of the 32-channel TCSPC system.

$$\text{Throughput2} = (D + n + m + j) \cdot f \cdot N \quad (2.8)$$

where Throughput1 applies for count rates higher than the threshold, vice versa Throughput2 is used when the threshold is overcome. The D bit denotes which Throughput is being used, while j is the number of bits necessary to count a period equal to Δt based on the laser frequency. For example, if we set the threshold to 640 ns and supposing to work with a laser synchronism frequency of $F=100$ MHz:

$$j = \log_2(\Delta t \cdot f_L) = \log_2(640\text{ns} \cdot 100\text{MHz}) = 6 \text{ bit} \quad (2.9)$$

Solving equations 2.7 and 2.8, we get:

$$\text{Throughput1} = (5 + 14 + 38 + 1) / 640\text{ns} \approx 11.3 \text{ MB/s} \quad (2.10)$$

$$\text{Throughput2} = (5 + 14 + 6 + 1) \cdot 5\text{MHz} \cdot 32 = 520 \text{ MB/s} \quad (2.11)$$

The bitrates obtained after compression are effectively lower with respect to the one from equation 2.5, nevertheless the FT2232H is still unable to reach this throughput. Hence, we can conclude that faster communication protocols are compulsory to manage time-tag mode.

2.6 1024-channel TCSPC module

A lot of effort has been made to develop the 32-channel TCSPC system, which features two independent modules (e.g. detection head and signal processing) as explained in section 2.4. Following the successful tests, an insane idea of engineering a compact module with a paramount number of custom technology SPAD detectors has taken hold of our brain. The first sketch of what was going to be the next generation TCSPC system is shown in figure 2.7. An exponential increase of the number of channels, e.g. 48 x 48 SPADs, was conceived at first but it turned out to be infeasible. The target was then lowered to 32 x 32 array.

So far, in the TCSPC instruments developed in our research laboratories, every single photon detector had its own dedicated conversion channel. Applying the 1:1 architecture also to the 1024-channel module, it would result in a unmanageable routing issues, not to mention the electrical crosstalks and overall module dimensions. As an example, supposing a mean count rate in the order of some megacounts per second, 1000 pixels and two bytes to record the result of a TAC conversion, the resulting bitrate to be handled (i.e. stored or transferred) is in the order of 10 GB/s [52]. Given these considerations, a fully integrated routing circuit has been developed to handle N detectors with M conversion channels, being $N \gg M$.

2.6.1 Integrated Routing Circuit

A first prototype, made in 0.18 μm HV CMOS technology by IBM, is being currently tested in our laboratories. It provides the routing of 64 detector signals toward 8 conversion channels. Since the prototype was conceived to be modular, it is straightforward to replicate its architecture and match the new 1024-channel requirements.

Each timing signal is fed to a low-jitter comparator, which output is split into two parallel paths: one is connected to the control logic, whereas the second is associated to a delay line. As soon as the logic stage detects a trigger from the comparator, it sets the MUX-DEMUX block to route this signal toward an available TCSPC conversion channel (figure 2.8). Since the logic block requires a finite amount of time to properly route the timing signals, passive delay lines are employed to skew their arrival instant, indeed this device is capable of determining different levels of priority. Since this architecture allows each SPAD to be connected to any of the available TCSPC channel, it would be more appropriate to define an array count rate, rather than the single detect one.

2.6.2 Detection Head

The design of a 2-D SPAD array in custom technology is a great issue, due to the limitation of the signal routing capability by a non-standard process. Internal routing is made challenging by the limited number of available metal layers, while off-chip bondings are

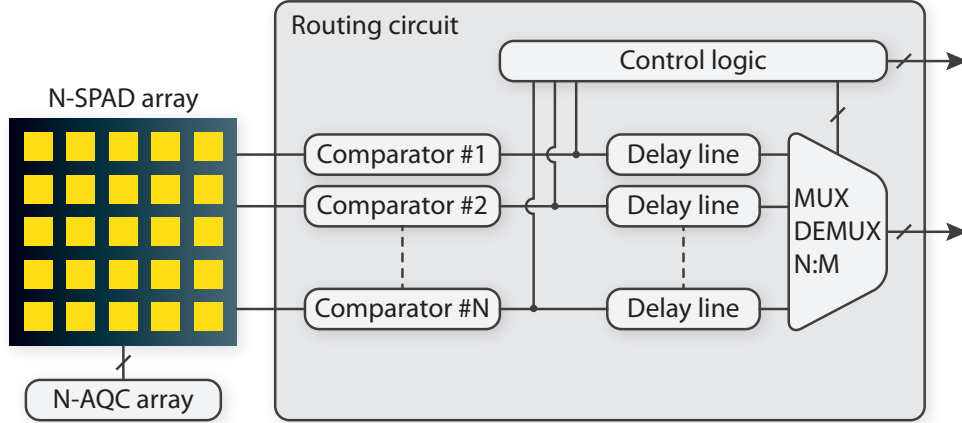


Figure 2.8: Picture of the original idea of the 32-channel TCSPC system.

constrained to the perimeter length of the detector array. Indeed, concerning a square geometry form factor, given n^2 the number of SPADs, the number of pads that has to be placed around the perimeter grows as n^2 , whereas the perimeter length grows a n [52].

Planar fabrication technologies have always been a limitation to electronics, thus new type of connections are currently under investigation. Through Silicon Via (TSV) is a recent technology that allows to contact two independent integrated chips by stacking them one above the other. This architecture will finally make feasible the superposition of custom technology detectors above standard CMOS chips as shown in figure 2.9.

2.6.3 1024-channel TCSPC board

The 1024 timing signals coming out from the detection head are fed to the routing circuit, which produces 64 *start* triggers to be split equally toward two 32-channel TCSPC boards. Hence, throughout two multiplexed 16-channel TACs per board, 16 analog timing signals are finally delivered toward two 8-channel A/D converters. Conversely, the *stop* signal is brought to the TACs by the data management board as shown in figure 2.10. It is worthy of note though, that external memories are now added to the FPGA. Indeed, with 1024 SPADs as many as 1024 histograms have to be build and stored; estimation of the required memory size is done from the following equation:

$$MemorySize = (k \cdot 2^m) \cdot N = (24 \cdot 2^{14}) \cdot 1024 \approx 402.7 \text{ Mbit} \quad (2.12)$$

where k is the number of bits which determines channel depth, m is the number of the ADC bits and finally N is the system number of channels.

The top commercially available FPGA (Virtex UltraScale XCVU160 [53]) features only 115.2 Mbit of BlockRAM, which led us to exploit additional external memory. There are essentially two kind of memories which could accomplish this purpose: *Dy-*

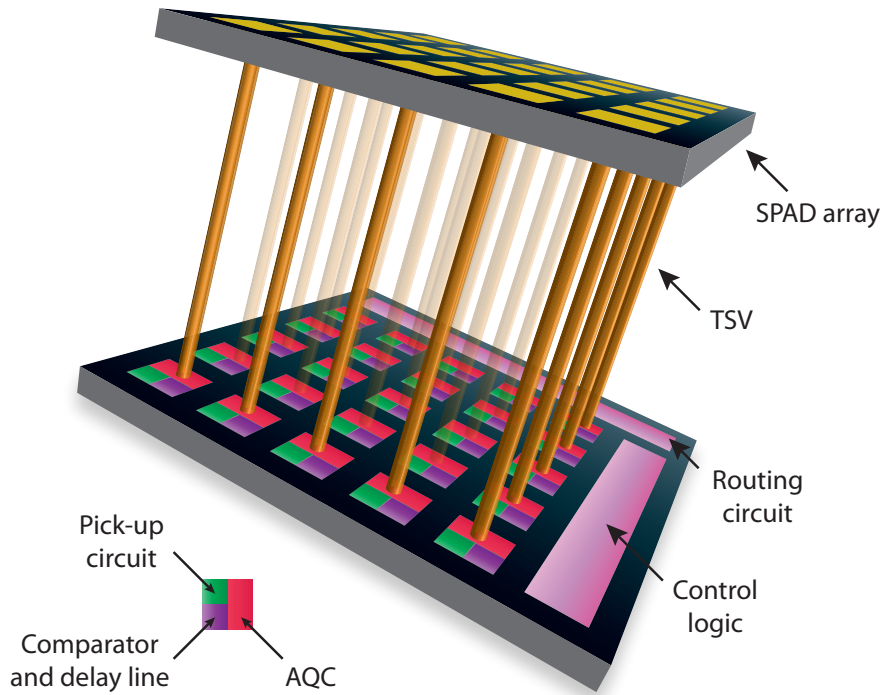


Figure 2.9: Rendering of the described detection head architecture with the new TSV technology. The SPAD array is connected to the bottom CMOS circuits through TSV links. Each detector has its own AQC, pick-up circuit, comparator and delay line while control logic and routing circuit are shared.

dynamic RAMs (DRAM) and *Static RAMs* (SRAM). Due to the refresh latency of first one, it cannot be employed as primary storage units while the latter is pretty suitable. The CY7C1625KV18-333BZXC [54], from Cypress Semiconductor, has been chosen for our application. It is a QDR II synchronous pipelined SRAM that features 333 MHz clock and a storage capability of 144 Mbit. The device is organized in 16 M x 9-bit words, enough to store all the 1024 histograms. However, with 8-bit channel depth (1-bit is typically used for checksum) we would obtain a poor S/N, reason why two SRAMs are placed in parallel reaching an overall 16-bit resolution. These 16-bit are then split equally, the 8 MSB conveyed toward one SRAM and the 8 LSB toward the remaining one.

In order to enhance even more the maximum counts per channel, a 2 Gbit DDR2 synchronous DRAM (MT47H256M8EB [55], from Micron Technology) has also been attached to the FPGA. Indeed, in the same way the 16-bit have been split into two SRAMs, we could for example split 24-bit into two SRAMs and one DRAM, where the bulky bits are assigned to the DRAM. Moreover, it could be also used to backup the SRAMs data in between different measurements.

Conversely, when the system is employed in time-tag mode, none of the former memory is used. Note that, increasing the detector number from 32 to 1024, has not changed linearly the required throughput for this operating mode. Actually, solving

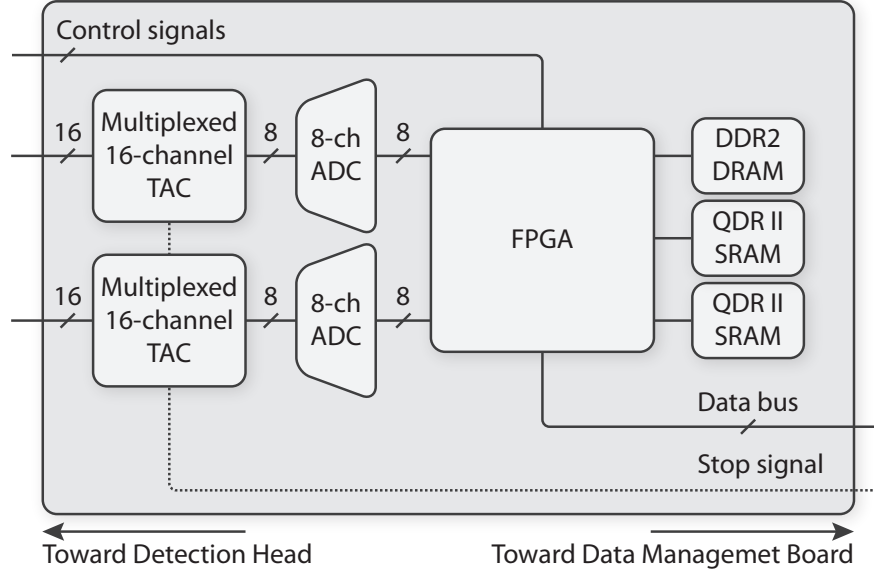


Figure 2.10: The two 8-channel ADC converts the TAC outputs and the FPGA samples and records the resulted digital values into two SRAM.

equations 2.4 and 2.8, the maximum bitrates per board are:

$$\textit{Throughput} = (10 + 14 + 38) \cdot 4\text{MHz} \cdot 32 = 992 \text{ MB/s} \quad (2.13)$$

$$\textit{Throughput2} = (10 + 14 + 6 + 1) \cdot 4\text{MHz} \cdot 32 = 496 \text{ MB/s} \quad (2.14)$$

where the first result is related to the uncompressed data and the second to the compressed one. Also note that the conversion rate of the new multiplexed TAC is 4 MHz. As expected, the found transfer rate cannot be handled by the USB 2.0, hence a dedicated data management board is implemented for the 1024-channel TCSPC module.

2.6.4 Data Management Board

The main purpose of this board is to gather and transfer data from the two TCSPC boards toward an external PC, eventually routing the *stop* signal toward the TACs. This board acts as the only communication interface to the outside world, hence it must handle control signals as well as providing mechanical support to the whole module.

Since the Data Management Board is the main topic of the work done in this thesis, an entire chapter (see chapter 4) has been dedicated to.

2.6.5 Power Management Board

The Power Management Board is crucial in the design of the whole module, indeed it is in charge of providing power to every single board. The Power Distribution Network has to deal with many different requirements that increase the overall system complexity.

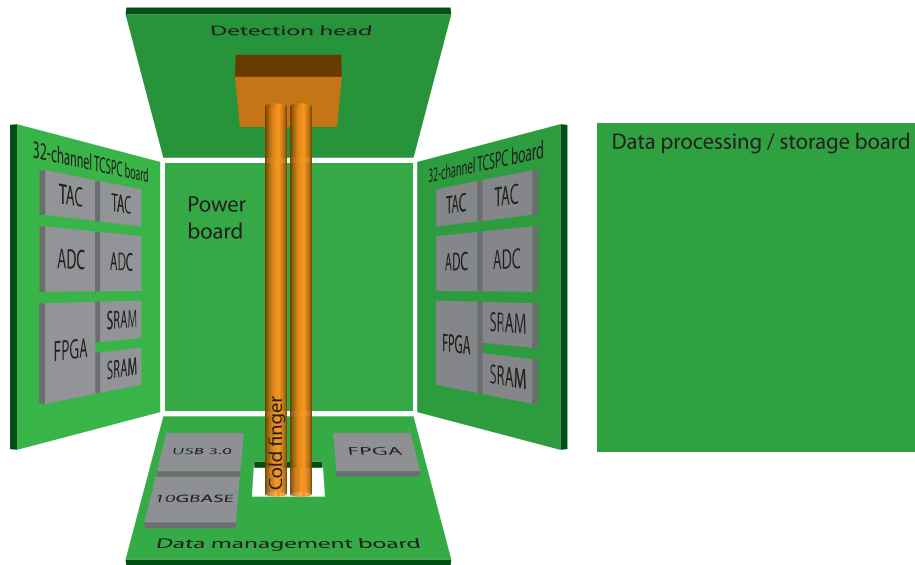


Figure 2.11: Rendering of the designed 1024-channel TCSPC module. Each board is square shaped of side 12 cm.

Close attention is being paid to supply proper bias voltage to the detection head. First of all, the SPAD array demands high negative bias voltage (tens of Volts) and, at the maximum count rate, they draw up to hundreds of milliamperes of mean current. Furthermore, to stabilize the detectors operating temperature, a thermoelectric cooler (e.g. Peltier cell) needs to be placed underneath the diodes substrate. At last, the produced heat is carried away by means of cold fingers. The 1024-channel TCSPC module is supposed to work with a single 48 V DC power supply.

2.6.6 Data Processing Board

A rendered picture of the whole 1024-channel TCSPC module is shown in figure 2.11. This board has not a predefined role yet and it may be designed depending on the specific field application. For instance, it could work as an interface between the two TCSPC boards and a solid-state drive, thus increasing the data storage capability rather than implementing a data-processing unit for on-board FCS elaboration or FLIM time-constant extraction [52].

Chapter 3

Communications Protocol

The following chapter is aimed to provide a brief analysis of the existing communications protocol framework and the reasons why Super-Speed USB 3.0 and 10 Gigabit Ethernet have been chosen for the 1024-channel TCSPC module. A general description of these two transmission standards will be given, along with their test setup and achieved bitrates.

3.1 Introduction to Universal Protocols

Communications Protocol, in computer science, is a set of rules or procedures for transmitting data between electronic devices. In order to let different devices to exchange information, there must be a preexisting agreement, such as how the information will be structured and how each side will send and receive it. In general, much of the following must be addressed [56]:

- *Data formats for data exchange*: a message usually consists of bitstrings which are divided into two fields, typically called *header* and *data area*.
- *Address formats for data exchange*: addresses are used to identify both the sender and the receiver(s) and they are part of the *header*.
- *Address mapping*: sometimes protocols need to map addresses of another scheme.
- *Routing*: in case the sender and the receiver(s) are not directly connected, intermediary systems are necessary.
- *Detection of transmission errors*: typically *cyclic redundancy check* (CRC) is employed allowing the receiver(s) to detect errors.
- *Acknowledgments*: correct data exchange flags are required.
- *Loss of information - timeouts and retries*: whenever the acknowledgement flag is not set, the sender must assume the packet was not correctly received and retransmit it.

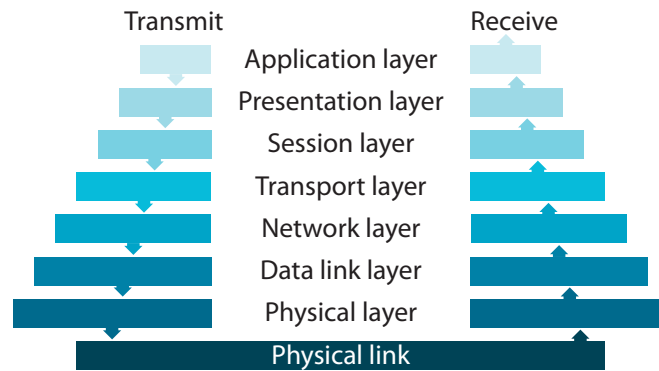


Figure 3.1: 7 layers of the OSI model. Moving from one layer to the adjacent one, additional information is encoded/decoded according to the flow direction, e.g. transmission/reception.

- *Direction of information flow*: in case of half-duplex communication channels, the direction of transmission must be clearly specified.
- *Sequence control*: whenever the data is split into many packets and sent randomly, the receiver needs to reorganize them by checking the sequence bits.
- *Flow control*: in case sender and receiver(s) communicates with different speed, the data flow must be managed properly.

The most relevant protocols are established by the *Open Systems Interconnection* (OSI) project at the *International Organization for Standardization* (ISO).

The OSI model characterizes internal functions of a communication system by partitioning it into abstraction layers. The first concept of a seven-layer model (figure 3.1) was provided by the work of Charles Bachman. Within each layer, one or more *entities*, also known as active elements, implement its functionality. Then each entity interacts directly only with the layer beneath it, and provides facilities for use by the layer above it. It works just like an assembly line in a manufacturing process in which parts are added as the semi-finished product moves from work station to work station where the parts are added in sequence until the final assembly is produced. In our case, the final product is a digital data packet or frame.

3.1.1 High-Speed Protocols

Despite their numbers, networking protocols show little variety, because all networking protocols use the same underlying principles and concepts, in the same way [57]. Transmission and reception of data is performed in four steps [58]:

1. The data is coded as binary numbers at the sender end
2. A carrier signal is modulated as specified by the binary representation of the data

3. At the receiving end, the incoming signal is demodulated into the respective binary numbers
4. Decoding of the binary numbers is performed

The rate of successful message delivery over a communication channel is named *throughput*. Basically, information may be transported from point-to-point over a physical or logical link, or switched through a certain network node. When different interfaces are connected together, data exchange will be limited to the throughput of the slowest one (referred to as the bottleneck). For instance, SATA 6G controllers on one PCIe 5G channel will be limited to the 5G rate.

Maximum throughput is essentially a synonym of digital bandwidth capacity and it is measured in bits per second (bit/s). This number is closely related to the channel capacity of the system [59], and is the maximum possible quantity of data that can be transmitted under ideal circumstances (e.g. asynchronous technologies, data compression etc.). Real scenario bitrates go under the following two parameters: *peak measured* throughput and *maximum sustained* throughput. Peak measured throughput is the one measured by a real, implemented or a simulated system. The value is obtained over a short period of time. This number is useful for systems that rely on burst data transmission. Instead, maximum sustained is the throughput averaged or integrated over a long time.

Furthermore, a differentiation between *channel utilization* and *channel efficiency* can be made. The first one, also known as *bandwidth utilization efficiency*, is the achieved throughput in percentage related to the net bitrate of a digital communication channel. For example, if the throughput is 60 Mbit/s in a 100 Mbit/s Ethernet connection, the channel efficiency is 60%. Instead, the second term is related to the use of the channel disregarding the throughput. It takes into account, besides the data bits, also the overhead which consists of preamble sequences, frame headers and acknowledge packets. It is worthy of note that, in real scenarios, the throughput is limited by many factors such as bandwidth and S/N of the analog physical medium. The maximum achievable bitrate in this case is the *channel capacity*. This upper bound is reached only in few circumstances, one of them is a point-to-point communication link where the channel utilization can be almost 100%, except for small inter-frame gaps. For example, the maximum frame size of the Ethernet protocol is 1518 byte (14 byte header + 1500 byte payload + 4 byte trailer). In addition, a minimum interframe gap corresponding to 12 byte and a frame preamble of 8 byte are inserted. This corresponds to a maximum channel utilization of $1518/(1518+12+8) \cdot 100\% \approx 98.69\%$ and a maximum throughput of $1500/(1518+12+8) \approx 97.5$ Mbit/s on a 100 Mbit Ethernet link.

Having introduced the digital communication basics, let's now define the requirements that the 1024-channel TCSPC module demands from the transmission link:

- *Throughput higher or equal than 992 MB/s*: from equation 2.14, the minimum

Technology	Rate (kB/s)
RapidIO Gen2 2x	1.25 GB/s
Infiniband FDR-10 1x	1.29 GB/s
PCI Express 2.0 x4	2 GB/s
Thunderbolt	1.25 GB/s x 2
10 Gigabit Ethernet	1.25 GB/s
USB SuperSpeed 3.0	0.625 GB/s

Table 3.1: Performance comparison among the communications protocol that match the 1024-channel TCSPC throughput requirement. Although the USB SuperSpeed 3.0 doesn't reach the minimum speed specification, it has been included in the table for completeness reason.

throughput requirement resulted to be set by the time-tag mode. Indeed, in the worst case, the data management board should be able to merge and buffer the compressed data from two 1024-channel TCSPC boards toward an external PC

- *Commercially available hardware*: both for implementing the logical part and for building the physical link
- *Accessible code for FPGA*: building up a protocol in HDL from scratch is time wasting and not the objective of this thesis
- *Flexibility*: the protocol must be hardware independent and ease of use for the end-user of the TCSPC module
- *Compact dimensions*: hardware overhead must be as low as possible in order to keep the instrument volume at minimum

These specifications put an ultimate filter to the wide landscape of available protocols, the most likely candidates are listed in table 3.1. The differences among these communication standards are based on the trade-offs between flexibility and extensibility versus latency and overhead. As an example, shrinking the frame size to decrease latency means that the header bits are not negligible with respect to the packet length, thus decreasing the effective bandwidth.

The RapidIO developers made this choice after all to support low latency links. It is a high-performance packet-switched, interconnect technology. It can be used as a chip-to-chip, board-to-board, and chassis-to-chassis communication channel. At first it seemed satisfactory, indeed IP cores are available for implementing a RapidIO capable device on a FPGA. Problem is that this protocol does not have its own connector nor cable, and as a matter of fact, the electrical specifications are based on industry-standard Ethernet and Optical Interconnect Forum standards such as XAUI or 10GBASE-KR.

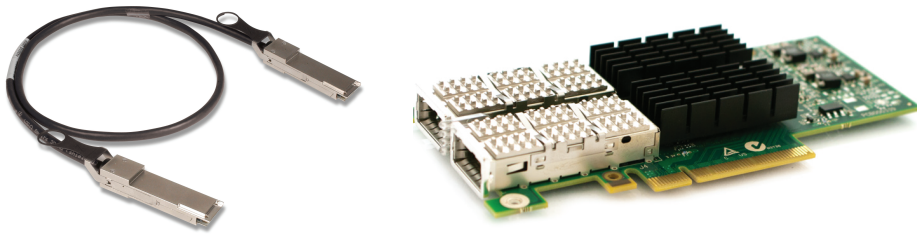


Figure 3.2: Passive QSFP+ cable [3] and PCIe adapter [4] (courtesy of Siemon Co and Mellanox Technologies).

Furthermore, since the 1024-channel TCSPC module has to be connected to a PC, the latter must be equipped with a RapidIO-to-PCIe card which could not be found if not as a general purpose development board (e.g. FPGA based). For all these reasons, the RapidIO option was discarded.

On the contrary, still concerning the performance trade-offs, the Infiniband transmission protocol adds puzzling header information to allow for complex routing, giving up higher throughput capabilities. This standard is a switched fabric computer network communications link used in high-performance computing and enterprise data centers. Its architecture makes it perfect for point-to-point bidirectional serial links intended for the connection between processors and storage devices. Unlike the RapidIO protocol, Infiniband host bus adapters and network switches are commercially available while the physical interconnections relies on Quad Small Form-Factor Pluggable (QSFP) connectors and can be copper or fiber, depending on the length required (figure 3.2). So far so good. What made this protocol difficult to be implemented is the lack of documentations and source codes for FPGA, not to mention the required hardware specifications and costs of enterprise PCIe adapter boards.

The Peripheral Component Interconnect Express, officially abbreviated as PCIe, falls somewhere in between the former two protocol mainstays. Targeted by design a local bus rather than linking independent systems, it is a high-speed point-to-point serial computer expansion bus. This standard defines slots and connectors for multiple lane widths in backplane solutions: x1, x4, x8, x16, x32 and its throughput increases proportionally with it. Each lane is composed of two differential signaling pairs: one pair for transmission and one pair for reception. Since the TCSPC module has to be placed close to the measurement setup, e.g. as near as possible to the specimen, the backplane direct connection inside a PC is not feasible and external cabling (figure 3.3) is compulsory. In 2007 the PCI-SIG industry consortium approved the PCI Express External Cabling Specification that defines how PCI Express can be implemented over a standard cable. This new capability allows the full bandwidth of the PCIe bus to be utilized within multiple chassis systems and small local networks. Four connector and copper cable sizes are defined: x1, x4, x8, x16 where the length of the latter is

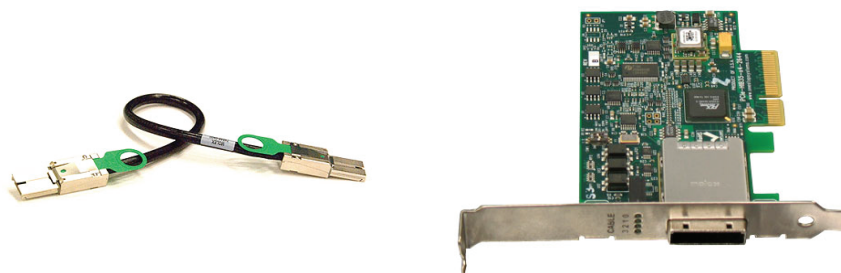


Figure 3.3: PCIe x4 cable with x4 connectors [5] and switch-switch based cable adapter [6] (courtesy of One Stop Systems).

unspecified but commercially available from 0.5 m up to 7 m. PCIe downstream host adapters are available on the market, however only few manufactures produce them. Taking into account these considerations, the PCIe solution seems reasonable and it will be later on subject of further comparisons.

Let's keep analyzing the remaining options listed in table 3.1. Released in its finished state in 2011, Thunderbolt (also know as Light Peak) is a hardware interface that allows for the connection of external peripherals to a computer. Developed by Intel and commercialized by Apple, it encodes data with a 64b/66b scheme while the interface controller takes existing PCI-Express and DisplayPort and multiplexes them together into a single data stream. The Thunderbolt specification uses the mini-DisplayPort connector and it is capable of accepting a standard DisplayPort connection as well. A single port supports up to six devices via hubs or daisy chains architecture. The first revision ensures a maximum bitrate up to 10 Gbit/s per channel in full-duplex mode, although copper based cables are limited to 3 m length. Despite of its high-throughput and small connector form-factor, there are few issues that make this candidate not suitable, at least not yet. Indeed, this technology is not widespread and employed mostly only by Apple; PCIe adapters are not available on the market but only as demonstration prototypes or as specific daughter boards (figure 3.4). Furthermore, logic cores for implementing the protocol on a FPGA are not accessible nor any documentation. For all these reasons, Thunderbolt has been discarded from the possible options.

The 10 Gigabit Ethernet refers to a various technologies for transmitting Ethernet frames at a rate of 10 Gbit/s. Defined by the IEEE 802.3ae-2002 standard, it is a full-duplex point-to-point link and supports both copper and fiber cabling. Since different physical layer (PHY) standards are available, a communication device may use any one of them through pluggable PHY modules; the newest one is the enhanced Small Form-Factor Pluggable transceiver, or SFP+, which became the also the most popular socket on 10GE systems. The SFP+ module serves only as optical to electrical converter and is connected to the host by either XAUI, XFI or SFI interface. Depending on the type of medium and on the communication range, many interconnect are defined and few of them are reported in table 3.2. As you can see, optical fiber solutions allow a much



Figure 3.4: Thunderbolt connector [7] and PCIe adapter [8] (courtesy of Apple Inc and ASUSTeK Computer Inc). The ThunderboltEX II PCIe adapter is supported only by few ASUS motherboards.

Interconnect	Medium	Max Range
10GBASE-USR	fiber	100m
10GBASE-SR	fiber	400m
10GBASE-LR	fiber	10km
10GBASE-ZR	fiber	80km
10GBASE-CX4	copper	15m
10GBASE-T	copper	55m (cat 6), 100m (cat 6a or 7)
10GBASE-KR	copper	1m
SFP+ Direct Attach	copper	15m

Table 3.2: Variants of the 10GBASE Ethernet protocol.

	10GBASE-SR	PCIe External Cabling
Commercially available hardware	✓	✓
Accessible code for FPGA	✓	✓
Communication range	400m	7m
Maximum throughput	10Gbit/s	128Gbit/s

Table 3.3: Comparison between 10GBASE-SR and PCIe External Cabling figure of merit.

longer range of connections but it is way more expensive than traditional copper cables. Considering that the TCSPC module has to work with high-power lasers (up to few W) and in extremely dark conditions, it may come in handy having the PC far away from the measurement setup, hence fiber is the best choice. Among these interconnections, the 10GBASE-SR (*Short Range*) seems a reasonable trade-off. It is a port type for multi-mode fiber and uses 850 nm Vertical-Cavity Surface-Emitting Lasers (VCSEL), delivering serialized data at a line rate of 10.3125 Gbit/s. 10GE *Network Interface Cards* (NICs) are commercially available from several manufacturers, which makes this protocol quite attractive. Table 3.3 summarizes both the 10GE and the PCIe performances for a quick comparison. In terms of bandwidth, PCIe spans a much broader performance range: 2.5 Gbit/s for a x1 Gen 1, up to 128 Gbit/s for a x16 Gen 3. Instead, from a hardware point of view, both are commercially available; however, PCIe over cable still represents a niche technology, hence it would result in a more challenging development. It is worth of note also the fact that the FPGA implementing the PCIe must be provided with a number of transceivers at least twice (TX/RX) the number of the PCIe lane width, e.g. 32 transceivers for a x16 solution. Nevertheless, it should be pointed out that the higher the throughput the more data has to be stored at the same speed. The fastest commercially available mass storage device is the Solid State Drive, however top performance consumer SSD (Samsung 840 Pro 512 GB [60]) hits only 520 MB/s sequential write, with limited lifetime with respect to the much slower HDD. In order to store the incoming data at GB/s speed, RAID configuration of multiple SSD in parallel is necessary. From a cost point of view, we should make a conscious trade-off between obtainable throughput and number of SSDs, keeping also in mind that a commercially available RAID controller has to be included among the hardware overhead. In the the worst case, 128 Gbit/s for a x16 Gen 3 PCIe means at least 32 SSDs in parallel (no manufacturer produces such a controller), not to consider the additional drives in case of implementing redundancy control.

Concerning the range of physical connection, 10GE fiber is the winner: thanks to its thin jacket diameter ($< 3\text{mm}$) and customizable length, optical fiber gives much more freedom and flexibility to the experiment setup. For all these reasons, 10 Giga-

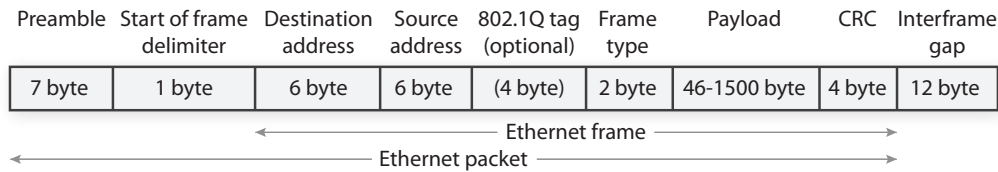


Figure 3.5: Ethernet frame and packet structure.

bit Ethernet has been chosen to implement the high-speed communication link on the 1024-channel TCSPC link, provided that the PCIe solution could be also developed in future in case of more demanding requirements.

With reference to the table 3.1, it is worth noting one last communication protocol: the Universal Serial Bus or USB. It was designed to standardize the connection of peripheral devices to a PC, both for communication and power supply. The latest commercial release, SuperSpeed USB 3.0, delivers a maximum signaling speed of 5 Gbit/s on full-duplex mode. Since 8b/10b encoding is employed, the payload throughput reaches 4 Gbit/s while maximum sustained throughput is around 3.2 Gbit/s. What makes this protocol very attractive is its wide employment, indeed it comes baked into chipsets from both AMD and Intel, and even when it's not native, host PCIe adapters controllers cost is negligible. Provided this, for a ready-to-use TCSPC module (e.g. no fancy 10GE PCIe adapter to install), it has been decided to upgrade the old FT2232 into a newer USB 3.0 controller. Indeed, with its 3.2 Gbit/s bandwidth, a better performance in terms of refresh rate of histograms can be appreciated (equation 2.2):

$$FrameSize = (10 + 16 \cdot 2^{14}) \cdot 64 \approx 2 \text{ MB} \quad (3.1)$$

which corresponds approximately to 190 fps.

3.2 Gigabit Ethernet

Ethernet is a family of computer networking technologies for local area networks; it includes several wiring and signaling variants of the OSI physical layer (figure 3.1). The original commercially available variant of this technology was the 10BASE5, where 10 refers to its transmission speed of 10 Mbit/s, BASE is short for baseband signaling as opposed to broadband, and the 5 stands for the maximum link range which is 500 m. Since the physical medium is shared among many stations, in order to avoid collision of simultaneous transmissions, a scheme know as *carrier sense multiple access with collision detection* (CSMA/CD) governs the protocol. Data are sent over Ethernet network in packets, which transports an Ethernet frame. Ethernet frames are of variable length, with no frame shorter than 64 byte or longer than 1526 byte as shown in figure 3.5. A typical packet starts with a 7 byte *preamble* and 1 byte *start of frame delimiter*:

0x55 0x55 0x55 0x55 0x55 0x55 0x55 0xD5

after which comes the 12 byte frame header, made of *destination address* and *source address*. The IEEE defined a 48-bit addressing scheme: each computer attached to an Ethernet network is assigned a unique 48-bit number known as its Ethernet address or *media access* (MAC) address. To assign an address, hardware manufacturers purchase blocks of Ethernet addresses and assign them in sequence as they manufacture interface devices. Thus, no two hardware interfaces have the same address [57]. The optional IEEE 802.1Q tag indicates Virtual Lan membership and IEEE 802.1p priority [61]. The *Frame type* field can be used for two different purposes: values equal or below 1500 announces that it represents the size in byte of the payload, while values equal or higher than 1536 indicate that it stands for Ethernet type, e.g. which protocol is encapsulated inside the payload (ARP, IPv4, IPv6 etc.). The maximum load size (*payload*) is 1500 byte, however non-conventional *jumbo frames* can be employed in Gigabit Ethernet which can carry up to 9000 byte. Instead, error detection is accomplished by the frame check sequence, which is a 4-byte *cyclic redundancy check* (CRC). The last field, *interframe gap*, is idle time between packets which corresponds to be 96-bit times (9.6 μ s for 10 Mbit/s, 0.96 μ s for 100 Mbit/s, and 96 ns for 1 Gbit/s).

Having introduced briefly the work principle of the Ethernet technology, let's see how to implement it on a real setup. With respect to the OSI model shown in figure 3.1, the first three layers (e.g. physical, data link and network) are named *Media layers*, while the others (e.g. transport, session, presentation and application) are called *Host layers*. The latter are responsible for accurate data delivery between connected nodes, instead the former are responsible for ensuring that information does indeed arrive at the destination for which it was intended. Since the main goal of the data management board is to carry out data from the 32-channel TCSPC boards as fast as possible, there is no need to envelop the raw information into any fancy high-level protocol. Under this assumption, we can neglect all the Host layers and use only the Media layers to lighten the demand of resources. The *data link* layer provides a reliable link between two directly connected nodes by detecting and possibly correcting errors that may occur during transmission, while the *physical* layer defines the electrical and physical specifications of the connection. The *network* layer instead is in charge of routing the payloads by adding the destination and source addresses while putting all together into one organized frame; since the Ethernet link is used as a point-to-point connection, this layer has been bypassed by *hard-coding* the frame structure. By the time it was decided to implement the Ethernet link, the FPGA development kit available in our laboratories was not able to implement the 10GE, hence the slower 1GE was chosen to test the actual feasibility of the connection. Figure 3.6 shows the block diagram of the intended communication channel. The Xilinx Spartan-6 FPGA [62] features up to 3.2 Gbit/s GTP transceivers which can be used to drive the VCSEL optical module and has enough resources to accommodate the Ethernet logics. Two Xilinx IP cores have been generated to manage

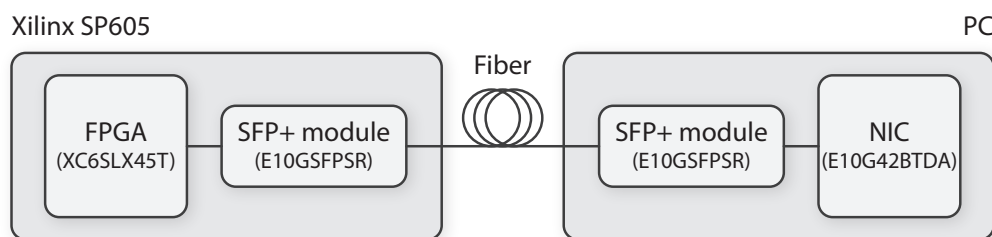


Figure 3.6: Block diagram of the 1000BASE-SR communication channel. One SFP+ optical module is attached to the Spartan 6 GTP transceivers on the SP605 Development Board while the remaining one is connected to the Network Interface Card on the PC.

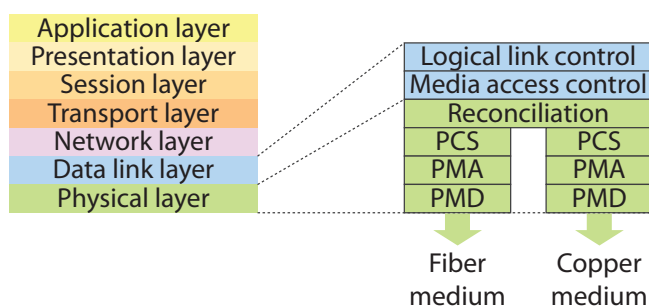


Figure 3.7: Overview of the Data Link and Physical sublayers in the OSI model.

the Ethernet link: the *Tri-Mode Ethernet Media Access Controller* [63] (TEMAC) and the *Ethernet 1000BASE-X PCS/PMA or SGMII* [64].

3.2.1 Tri-Mode Ethernet Media Access Controller IP core

The TEMAC solution comprises the 10/100/1000 Mbit/s option, the 1 Gbit/s option and finally the 10/100 Mbit/s option. Generally speaking, the Logical Link Control (LLC) and the Media access control (MAC) sublayers are part of the Ethernet architecture, in the OSI model they go along the data link layer (figure 3.7). Accordingly to IEEE 802.3-2002 section 4.1.4, the functions required of a MAC are [65]:

- Receive/transmit normal frames
- Half-duplex retransmission and backoff functions
- Append/check frame check sequence
- Interframe gap enforcement
- Discard malformed frames
- Append(TX)/remove(RX) preamble, SFD (start frame delimiter), and padding
- Half-duplex compatibility: append(tx)/remove(rx) MAC address

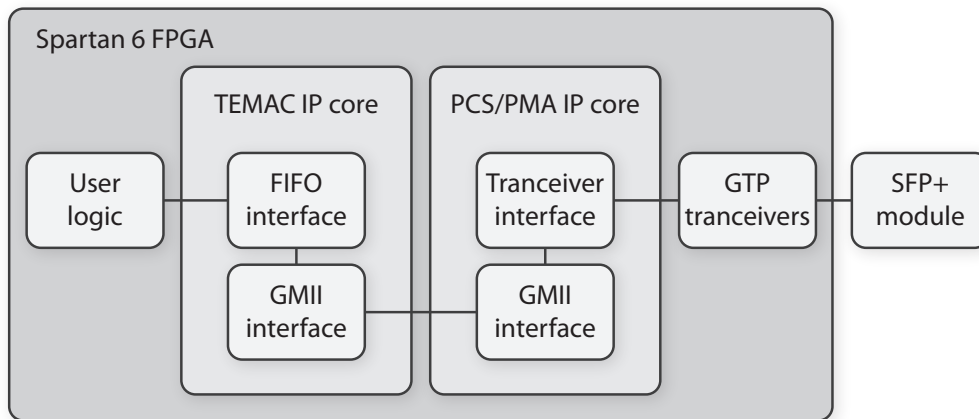


Figure 3.8: FPGA internal architecture. The PMA is connected to an external off-the-shelf SFP+ optical transceiver while the GMII of the Ethernet 1000BASE-X PCS/PMA is connected to the Xilinx Tri-Mode Ethernet MAC core.

The uppermost sublayer, LLC, multiplexes protocols running atop the data link layer, and optionally provides flow control, acknowledgment, and error notification. Since the Host layers are not being employed in our case, the LLC can be neglected. Concerning the MAC sublayer, it interacts directly with the lower physical layer which consists of the *Physical Coding Sublayer* (PCS), the *Physical Medium Attachment* (PMA), and the *Physical Medium Dependent* (PMD) sublayers (see section 3.2.2). Two main physical standards are specified:

- *BASE-T* PHYs provide a link between the MAC and copper mediums
- *BASE-X* PHYs provide a link between the MAC and fiber optic mediums

3.2.2 Ethernet 1000BASE-X PCS/PMA IP core

This IP core interfaces directly to the FPGA GTP tranceivers and provides some of the PCS layer functionality such as 8b/10b encoding/decoding, the PMA Serializer/Deserializer (SerDes), and clock recovery. On the other side, it is connected to the TEMAC core through internal Gigabit Media Independent Interface (GMII) as shown in figure 3.8. The Physical Coding Sublayer (PCS) performs the following operations:

- Encoding/decoding of GMII data to form a sequence of ordered sets
- 8B/10B encoding/decoding
- Auto-Negotiation for information exchange with the link partner

The Physical Medium Attachment (PMA) instead, is in charge of these tasks:

- Serialization and deserialization of data for transmission and reception on the underlying PMD sublayer (SFP+ optical transceiver)

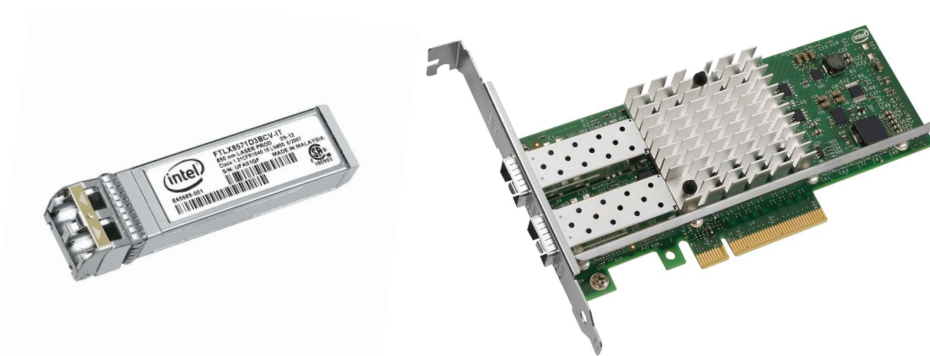


Figure 3.9: E10GSFPSR [9] SFP+ module and E10G42BTDA [10] NIC (courtesy of Intel).

- Recovery of the clock from the 8b/10b-coded data supplied by the PMD

Note that, as mentioned in section 3.2, the network layer is hard-coded: Ethernet frames are built by the *User logic* block and fed directly to the MAC core. For further details about the Gigabit Ethernet VHDL implementation, refer to section 5.2.

3.2.3 SFP+ Module, Optical Fiber and Network Interface Card

The SFP+ module serves only as optical-to-electrical converter and it is connected to the FPGA transceivers by SFI interface. It has a built-in 256 byte EEPROM which describes the module's characteristics and operating status, all accessible by an I²C interface. The E10GSFPSR [9], manufactured by Intel, has been chosen for the TCSPC system. This module features an uncooled 850 nm VCSEL laser, power dissipation lower than 1 W and duplex LC connector. According to the receptacle plug type, the optical fiber has been chosen: 50/125 μm multi-mode, OM3-standard laser optimized cable. On the PC side, optical signal is back-converted into electrical one by an identical SFP+ module attached to the E10G42BTDA [10], which is a network interface card (NIC). The mentioned hardware are shown in figure 3.9.

3.2.4 1GE Experimental Results

A whole picture of the 1GE link setup is shown in figure 3.10. Once the FPGA is successfully programmed with its firmware, start of transmission is triggered by a push-button on the SP605: it is the *update_speed* signal which allows the MAC core to return to its initial state and to set the *mac_speed* to 1000 Mbit/s. A pattern generator in the firmware is in charge of creating Ethernet frames: it starts with the minimum frame size and after each frame is sent, increments the frame size until the maximum value is reached, it then starts again at the minimum frame size. On the PC side, *Wireshark* software is employed to analyze the incoming packets while channel utilization is verified through Windows 7 network activity monitor (figure 3.11). The average sustained throughput over three minutes of successful transmission is 105 MB/s however this

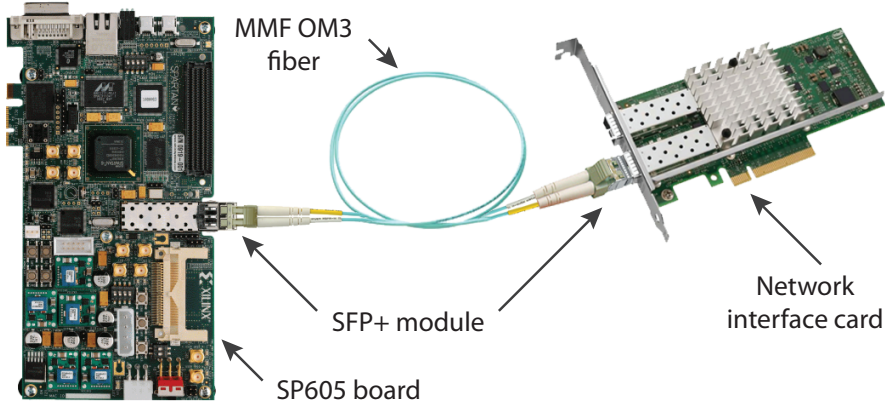


Figure 3.10: 1000BASE-SR communication channel setup used to test the actual feasibility of this protocol. Once verified, it will be upgraded to the faster 10GBASE-SR.

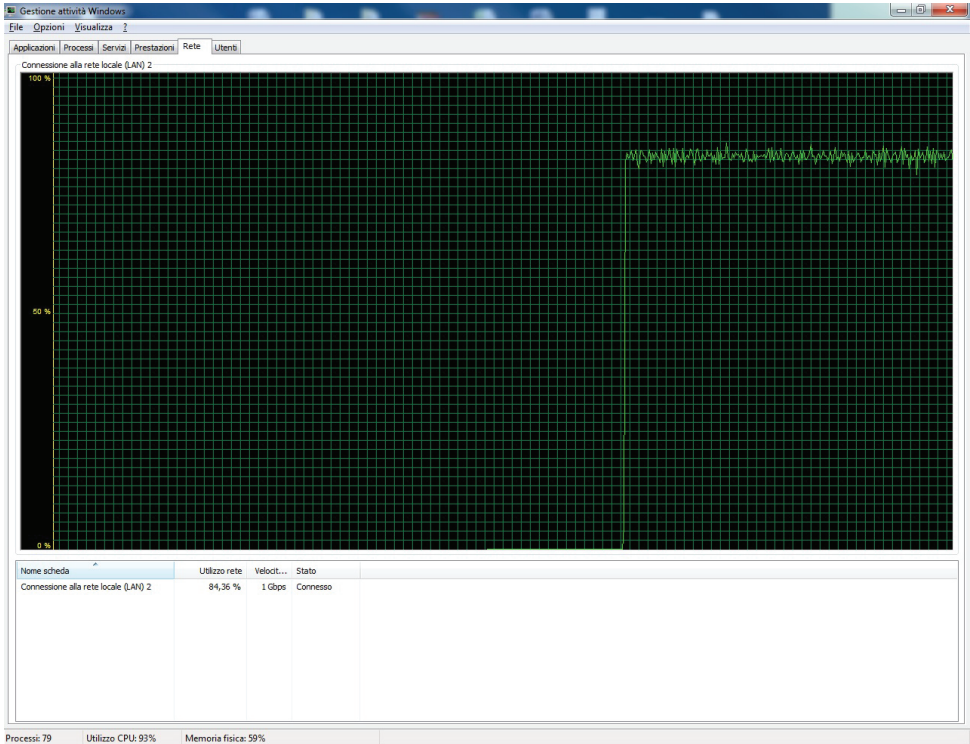


Figure 3.11: Windows 7 network activity monitor shows a $\approx 84\%$ channel utilization.

measurement is setup-dependent in the sense that the PC resources are shared among several running processes and managed by a specific operating system.

3.3 SuperSpeed USB 3.0

The Universal Serial Bus (USB) is a standard that defines both hardware specifications and communications protocol aimed to link PC with peripheral devices. So far, four data transfer speed has been officially approved: Low Speed (USB 1.0, 1.5 Mbit/s), Full Speed (USB 1.1, 12 Mbit/s), Hi-Speed (USB 2.0, 480 Mbit/s) and finally SuperSpeed (USB 3.0, 5 Gbit/s). Typically, a USB device may consist of several logical sub-devices that are referred to as *device functions*. Physical devices which provide different functions are named *composite devices*. The communication link is based on *pipes* that link the host to the device *endpoints*. Each device could in principle have up to 32 endpoints, defined and addressed during initialization. Basically, there are two types of endpoints: message and stream. The former is bi-directional and employed for control transfers, whereas the latter is uni-directional and can transfer data using a *bulk*, *interrupt*, or *isochronous* transfer. Endpoint are grouped into *interfaces* and each interface is associated with a single device function, except for endpoint zero, which is used for device configuration.

Up to the latest 32-channel TCSPC system [14], only the Hi-Speed USB 2.0 has been exploited using the CY7C68013A [45] controller in the first place, then upgrading to the faster FT2232H [46]. As mentioned in section 2.5, the 1024-channel TCSPC system demands much higher throughput capabilities, reason why it has been decided to engage the SuperSpeed USB 3.0. While still being backward compatible, the newest revision features full-duplex mode, optimized power efficiency and improved bus utilization with respect to the USB 2.0. Asynchronous notification between the host and the device using packets *Not Ready* (NRDY) and *Endpoint Ready* (ERDY) enables links that are not active to be put into idle state, allowing for better power and bandwidth management. SuperSpeed channel efficiency is dependent on a number of factors including 8b/10b symbol encoding, packet structure and framing, link level flow control, and protocol overhead. When all these aspects are considered, it is realistic for 400 MB/s or more to be delivered to an application [66]. The packet protocol is derived from the same *Token/Data/Handshake* model employed by USB 2.0. SuperSpeed packets start with a header and some packets consist of a header only. All headers begin with one of the four possible Packet Type (Link Management, Transaction, Data and Isochronous Timestamp) and ends with a 2 byte link control word. As an example, let's consider the IN transaction in figure 3.12. An overall of 6 packets are required to perform two *Token/Data/Handshake* transactions in the Hi-Speed link:

1. Host broadcast an *IN Token* to initialize the transaction
2. Hi-Speed device returns the requested *Data* packet

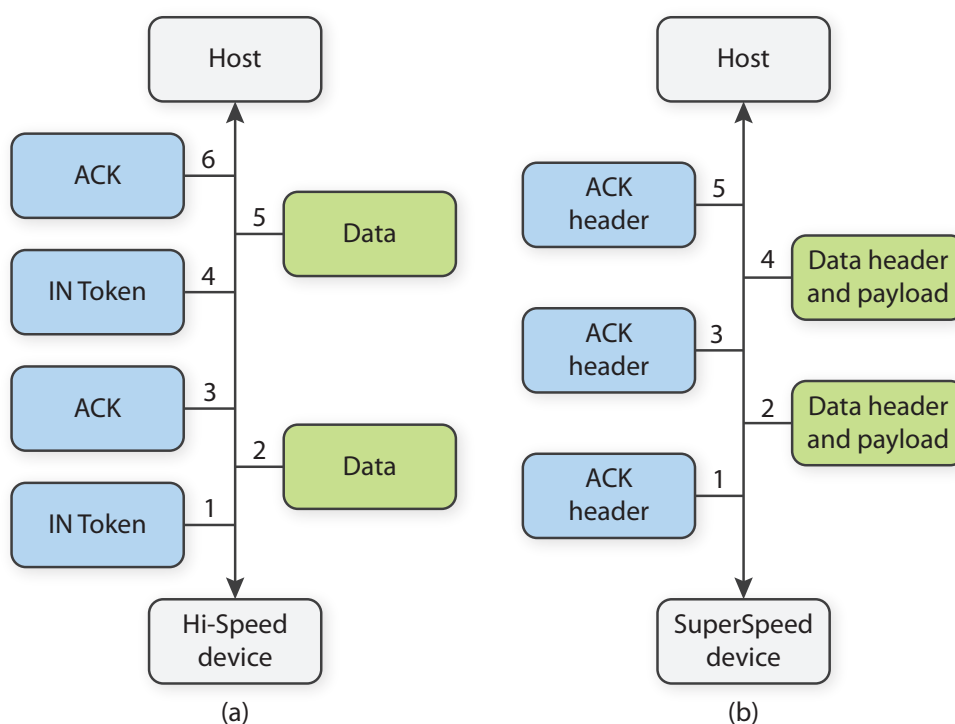


Figure 3.12: Comparison between Hi-Speed USB and SuperSpeed USB over two IN transaction requests. (a) 6 packets needed to complete the requests in USB 2.0. (b) 5 packets needed to complete the requests in USB 3.0.

3. Host acknowledge receipt of data with an *ACK* handshake packet
4. Steps 1–3 are repeated

Whereas, only 5 packets are needed to perform the same operation in the SuperSpeed link:

1. Host broadcast an *ACK header* to initialize an IN transaction
2. SuperSpeed device returns the *Data* packet
3. Host acknowledge receipt of data with an *ACK header* which now contains also a second transaction request
4. SuperSpeed device delivers the *Data* packet
5. Host acknowledge receipt of data with an *ACK header*

USB 3.0 architecture is inspired by the layered PCIe architecture and the OSI model, indeed it implements the physical layer, the link layer and the protocol layer. The physical layer includes the PHY and makes use of 8b/10b encoding/decoding, data scrambling/descrambling, and serialization/deserialization. Similar to the Ethernet, the link layer is in charge of preserving data integrity and error detection. Finally, the

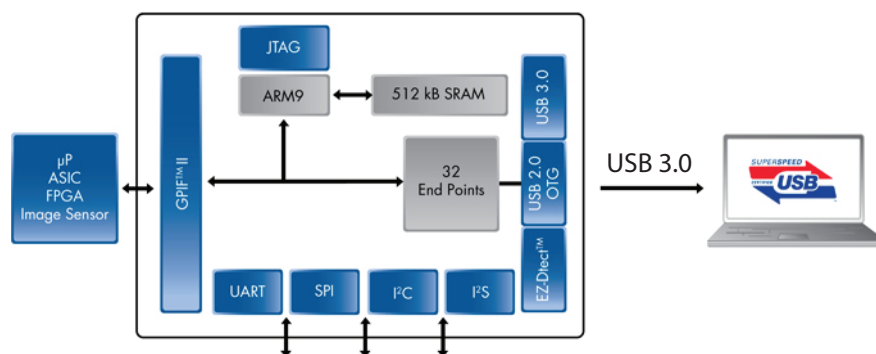


Figure 3.13: Block diagram of the FX3 internal architecture (courtesy of Cypress Semiconductor).

protocol layer manages end-to-end data flow between host and device. Coming to the actual implementation of the USB 3.0 communication channel, it was decided to test its feasibility on a development board. By the time, there was only one manufacturer that provided its USB 3.0 controller embedded on a development kit: Cypress Semiconductor with its CYUSB3KIT-001 [67]. This board is a combination of hardware, software, and documentation that enables the evaluation of the FX3 [68] SuperSpeed USB controller.

3.3.1 Cypress EZ-USB FX3

Figure 3.13 shows the block diagram of the FX3 controller. Its main features are:

- Integration: Full USB 3.0 Peripheral Controller with built-in PHY
- High-Performance: ARM9 with 512 kB RAM for data processing
- Connectivity: I2S, SPI and UART peripherals
- Small Footprint: 10x10mm 121 ball 0.8mm pitch BGA package

This controller embeds a fully configurable, parallel, general programmable interface called GPIF II, which can be connected to any processor, ASIC, or FPGA. On the Data Management Board, the FX3 functions as a coprocessor and is connected to the FPGA through the GPIF II; example implementation of the GPIF II is the synchronous slave FIFO [69] interface which allows the FPGA to directly access buffers internal to FX3.

3.3.2 USB 3.0 Experimental Results

The setup used to test the SuperSpeed USB 3.0 channel is shown in figure 3.14. The FPGA board used to implement the slave FIFO interface is the same as the one used in the 1GE experiment, in addition a custom 2-layer bridge board has been suitably designed to join the SP605 FMC connector to the CYUSB3KIT-001 Samtec connector.

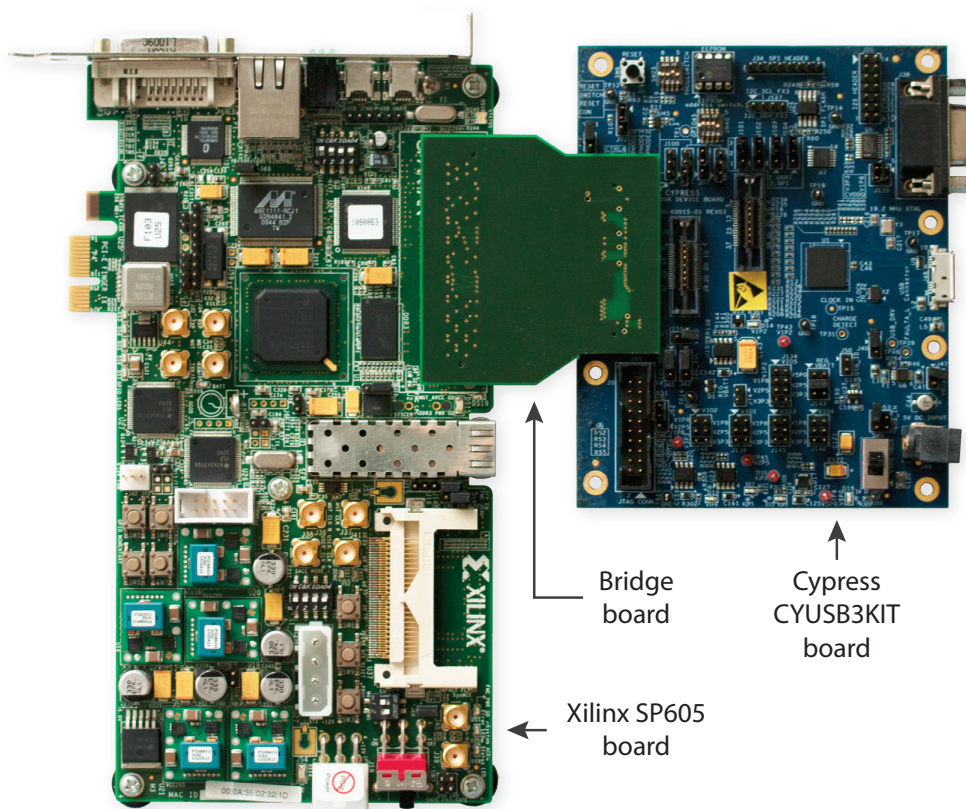


Figure 3.14: The Xilinx SP605 board is employed to generate a known pattern and to send it toward the Cypress CYUSB3KIT board through a custom bridge board. Finally, data is received on the PC and stored in a SSD using a C# software.

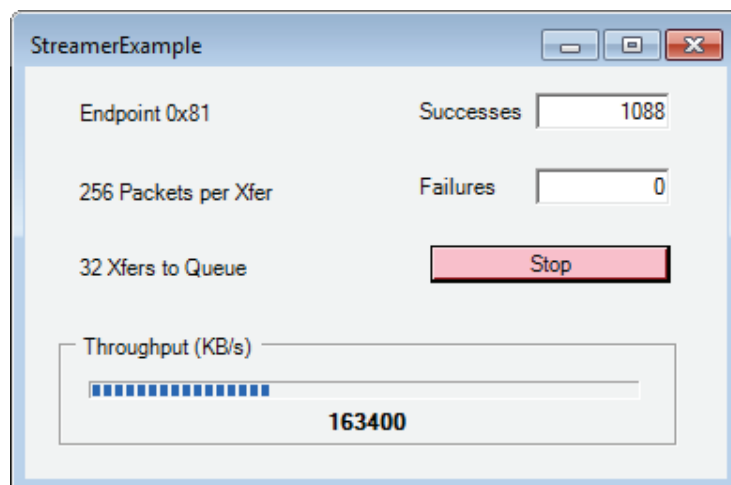


Figure 3.15: Custom C# software showing the USB 3.0 throughput in real time. In *BULK* transfer mode, 256 packets of 16 kB each are transferred per time, while the *Xfers to Queue* is dependent upon the DLL functions employed to begin each transfer.

The FPGA is programmed with a state machine firmware that checks on the *FLAG* signals before writing a known bit pattern into the FX3 memory. Instead, the FX3 controller is programmed with the *Stream IN* firmware (provided by Cypress) that takes the content of its buffers and transfer it toward the PC which mounts a USB 3.0 adapter and a SSD. A custom software has been developed in C# to save the incoming data into a *.dat* file while displaying the throughput in real time as shown in figure 3.15. The bitrate is limited to 180 MB/s due to the host PC, indeed the USB 3.0 adapter is plugged into a PCIe 1.0 slot which features a maximum speed of 2.5 Gbit/s. It has to be mentioned that the purpose of this test is mainly oriented to check the functionality of the C# software and the USB 3.0 setup rather than enhance its transfer rate, indeed no optimization has been done yet, neither on hardware or on software. It is worth noting also that the throughput could be improved by almost 10% by just employing the C++ programming language instead of the C# due to the application and driver-level overheads.

Chapter 4

Data Management Board

This chapter provides a comprehensive description of the 1024-channel TCSPC Data Management Board. Starting from the whole system requirements, mechanical and electrical issues will be discussed in details. A brief introduction will be given for the most relevant components, along with the reason why they have been chosen among others. The chapter highlights also how the communication among six different boards is managed through one USB port. Finally, the Power Delivery Network will be shown, including a short digression about the ferrite bead.

4.1 Board Overview

The Data Management Board is the core subject of this thesis. It is implemented on a $120 \times 120 \text{ mm}^2$, 8-layer PCB board on which 2 cut-out have been made in order to accommodate the SFP+ optical transceiver and the cold fingers. Its dimensions are set by the TCSPC boards. As matter of fact, we need at least $64 + 64 + 32 = 160$ signals (e.g. 32 differential signaling *start*, 32 differential signaling *spad address*, 32 single-ended *TAC ready*) between the Detection Head and each of the two TCSPC boards. Unfortunately, connectors featuring such high-pin count cannot be found on the market. This issue has been resolved by placing two 120-position Samtec connectors [70] in parallel (figure 4.1), with the drawback of an overall larger system dimensions. Figure 4.2 shows the block diagram of the Data Management Board. Heavy processing duties are accomplished by the FPGA which gathers data from 4 different boards and

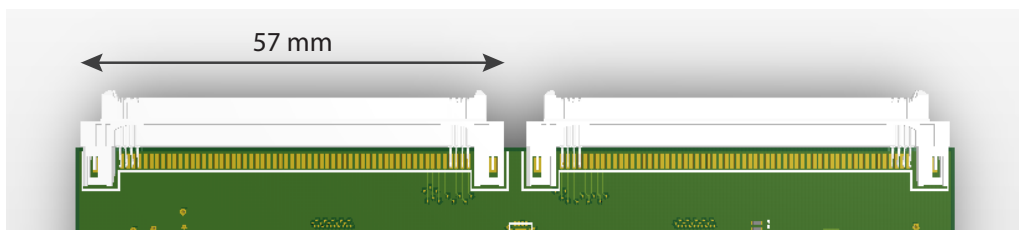


Figure 4.1: Two 120-position Samtec connectors are placed in parallel on each 32-channel TCSPC board.

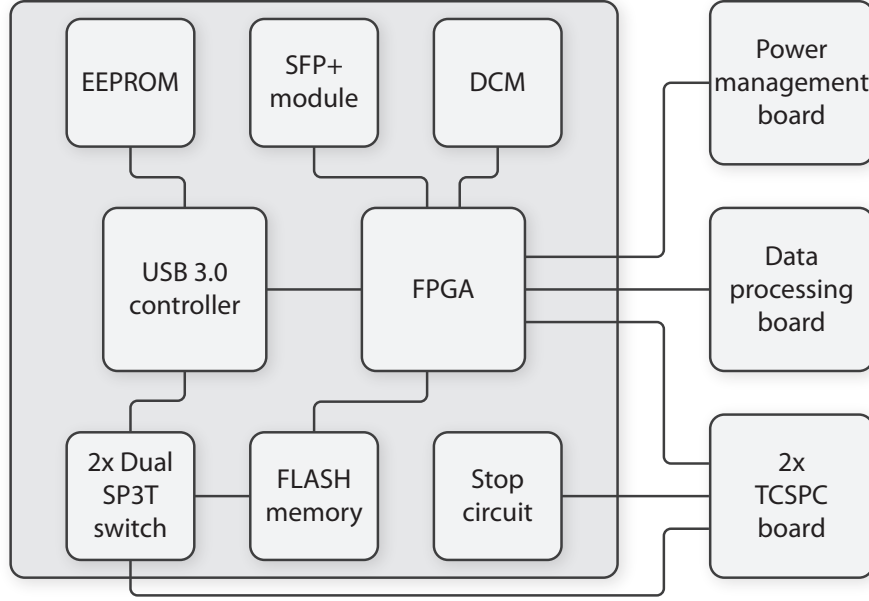


Figure 4.2: Block diagram of Data Management Board. The FPGA is in charge of buffering the gathered data toward the USB 3.0 controller and the SFP+ optical transceiver.

pass it either to the SFP+ module or to the USB 3.0 controller, accordingly to the user choice. In conjunction, control tasks are given to the FX3, such as temperature monitoring and power-on sequencing. Moreover, a smart SPI bus network has been developed to get direct access to the flash memories allowing a non-invasive handling in case of FPGAs firmware update.

Eventually, meticulous care and attention were given to the *stop* signal conditioning stage, indeed it is one of the most critical circuit of the whole TCSPC system in terms of jitter noise.

4.2 Stop Signal Conditioning Stage

As mentioned in section 1.6, the time interval between the arrival instant (*start*) of a photon and the reference pulse (*stop*) from the laser source is measured by means of the Time-to-Amplitude Converter. Early TCSPC instruments developed in our research group had the *stop* signal directly conditioned on the same board of the TAC. Considering that in the 1024-channel system there are two TCSPC boards and also that the Data Management Board is meant to be the only communication path to the external world, it is straightforward to place the *stop* conditioning stage on this very board.

Typically, laser reference pulses are single-ended (NIM standard) signals, hence in order to minimize electronic crosstalk and electromagnetic interference along the transmission line, it is essential to convert the *stop* pulse into differential signaling as soon as possible. A high performance nickel-plated SMA connector [71] is being employed to receive the *stop* trigger from outside and to route it promptly to the conditioning

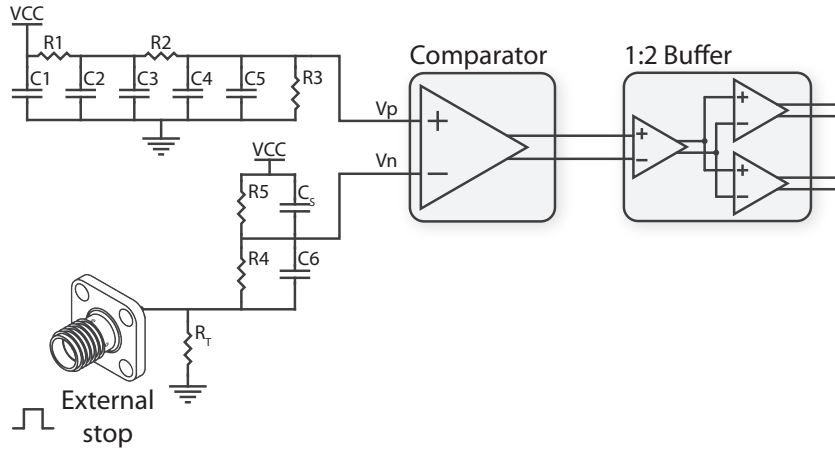


Figure 4.3: Stop conditioning stage: the external single-ended *stop* signal is converted into a pair of differential signaling pulses, one for each 32-channel TCSPC boards.

VCC	R1	R2	R3	R4	R5	R _T	C1	C2	C3	C4	C5	C6	C _S
2.5 V	3 k Ω	390 Ω	1.3 k Ω	1.5 k Ω	3.3 k Ω	50 Ω	10 μ F	1 μ F	0.1 μ F	0.1 μ F	10 nF	2.2 pF	1 pF

Table 4.1: Values of the compensated voltage divider components.

stage shown in figure 4.3. The NIM logic levels swing from 0 V to -0.7 V, hence a proper shift through a compensated voltage divider has to be made in order to match the comparator hysteresis thresholds. In particular, the Analog Devices ADCMP606 [72] comparator was chosen to deliver the differential signaling *stop* triggers. Its input dynamic ranges from -0.5 V to VCC+0.2 V, thus no particular threshold is requested on the positive input pin. However, we should consider the following remarks before proceeding to design the voltage divider [51]:

- The NIM maximum swing is reduced due to the passive partition
- The threshold crossing delay influences the timing performances of the whole instrument

The voltage on the positive pin is set to be $V_p=0.7$ V using a suitably filtered voltage divider; this value was chosen accordingly to the trade-off between maximum achievable swing on V_n ($\Delta V_n=0.481$ V) and relatively balanced voltage divider ($R_4=1.5$ k Ω , $R_5=3.3$ k Ω). Considering the component values shown in table 4.1, we obtain $V_n=0.78$ V when $V_{in}=0$ V and $V_n=0.3$ V when $V_{in}=-0.7$ V. The resulting *noise margin* is $\Delta V=80$ mV, a good compromise between immunity to supply ripples and timing performance. As mentioned before, compensation is needed to minimize parasitic effects on the V_p node. For example, the ADCMP606 shows a $C_S=1$ pF stray capacitance on both its input pins. One way to deal with this is to add a capacitance across R_4 , keeping in mind that it must be inversely proportional to the resistance value, e.g. $R_5/R_4 = C_6/C_S$. Once triggered, the comparator output is fed to a 1:2 buffer which delivers the

stop signal toward both the 32-channel TCSPC boards. In particular it is the SY58606U [73], manufactured by Micrel, which features a fully differential CML buffer optimized to provide two identical output copies with less than 15 ps of skew and only 0.146 ps rms of phase jitter. To conclude, the whole signal conditioning stage is supplied with 2.5 V by linear regulators.

4.3 FPGA

The FPGA is the main processor of the Data Management Board. It is in charge of gathering raw data from four different boards and hand it over to the communication links in a structured manner. Consequently, demanding performances are required, in particular:

- 10GBASE-SR compatible transceivers
- Adequate number of I/O resources to communicate with four boards and the USB 3.0 controller
- Suitable amount of logic resources to implement 10GBASE IP cores

The current FPGA market leaders and long-time industry rivals are Xilinx and Altera; nevertheless, taking into account that I was already familiar with the first manufacturer FPGAs and their development environment, and the fact that their products are almost equivalent, the choice was easy.

The Xilinx *LogiCORE IP 10-Gigabit Ethernet PCS/PMA* [74] is the IP core employed to implement the physical layer of the OSI model for the 10GBASE-SR Ethernet link. It supports Kintex Ultrascale devices, Zynq-7000 All Programmable SoCs, Virtex-7, and Kintex-7 devices containing GTX and GTH transceivers. However, it can be synthesized only on -2 speed grade devices. Each serial transceiver is a combined transmitter and receiver; in particular, the GTX operates up to 12.5 Gbit/s and the GTH achieves 13.1 Gbit/s by employing a combination of ring oscillators and LC tank. Discarding SoCs, Ultrascale and Virtex-7 devices for cost and resource wasting reasons, we are left only with the Kintex-7 family FPGAs.

The 7 series FPGAs offer both high-performance (HP) and high-range (HR) I/O banks. The HP I/O banks are designed to meet the performance requirements of high-speed memory and other chip-to-chip interfaces with voltages up to 1.8 V. The HR I/O banks are designed to support a wider range of I/O standards with voltages up to 3.3V. The maximum available user I/O goes to two particular devices: the XC7K325T and the XC7K410T, they differ from each other only for the number of Logic Cells, CLBs, DSP Slices and Block RAM Blocks. Estimation of the overall required resources was made on the 10GBASE IP cores *resource utilization*, which led me to choose the XC7K325T. Moreover, within the commercial grade solutions there are two available

	Part number	Kintex-7 XC7K325T-2FFG676C
	CLB Slices	50,950
	Logic Cells	326,080
	Maximum Distributed RAM (kbit)	400
	Block RAM/FIFO w/ ECC (36 kbiteach)	445
	Total Block RAM (kbit)	16,020
	CMTs (1 MMCM + 1 PLL)	10
	Maximum User I/O	400
	GTX Transceivers (12.5 Gbit/s Max Rate)	8
	Dimensions (mm)	27 x 27

Table 4.2: Xilinx Kintex-7 XC7K325T-2FFG676C FPGA feature summary table.

Board	32-channel TCSPC	Data Processing	Power Management
Parallel data bus	32-bit	32-bit	10-bit
GTX	1	5	-
Differential Clock	2	1	1
GPIO	6	6	10

Table 4.3: Connections among the FPGA and the four daughter boards. Each GTX transceiver is a combined transmitter and receiver.

packages: FFG and FBG, both utilize flip chip technology [75]. The main difference is the manufacturing cost, indeed the FBG is cheaper with respect to the FFG since it is built with fewer production steps but also features lower data rate and thermal performances. Finally, the Kintex-7 XC7K325T-2FFG676C device was chosen for the Data Management Board. The FPGA main characteristics are reported in table 4.2.

Coming to the actual positioning and orientation of the device on the Data Management Board, I considered at first the arrangement of the four slave boards and the connectors (e.g. USB, SMA etc.) in conjunction with the FPGA pinout and the presence of two 30 x 15 mm² and 16 x 11 mm² board cut-outs, which led to an almost forced choice (figure 4.4). Table 4.3 shows the connections that the FPGA carries toward each of the four daughter board. The GTX routed toward the TCSPC board is meant to be a substitute of a 32-bit parallel communication bus, indeed it is supposed to implement the Xilinx *Aurora* protocol [76] for a high-speed serial link. On the contrary, 5 GTX

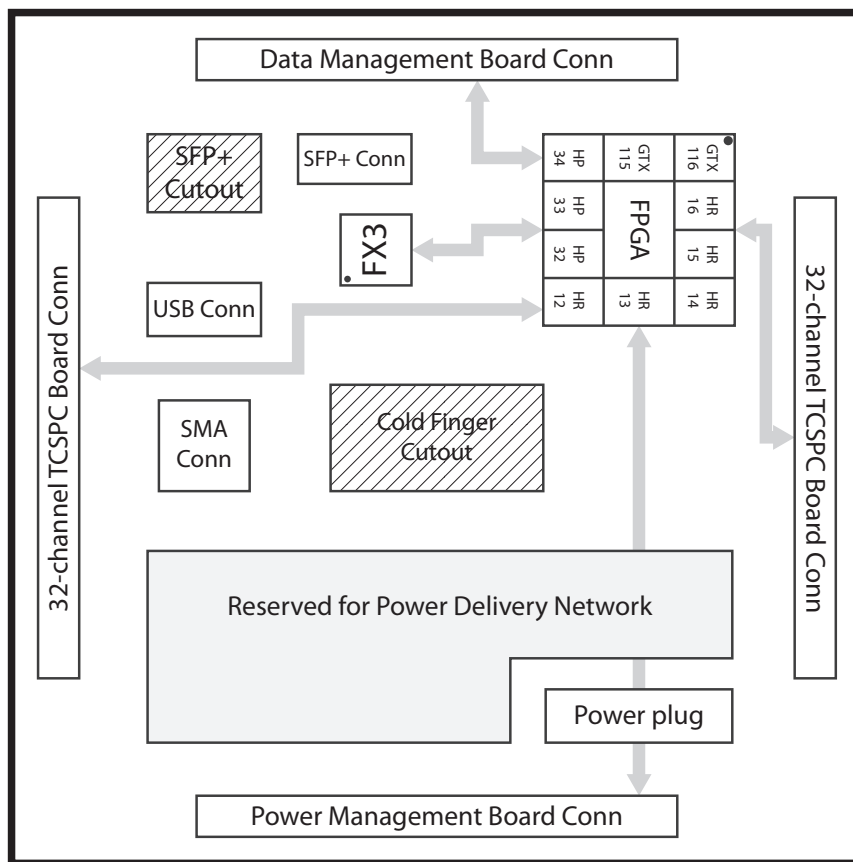


Figure 4.4: Arrangement of the components on the Data Management Board.

transceivers are dedicated to the Data Processing Board which purpose is to build up a PCI x4 channel for fast data exchange between either an on-board processor or a SSD. The communication between the FPGA and the FX3 USB controller, as previously mentioned in section 3.3.1, is implemented on a 32-bit slave FIFO [69] interface synchronous to a 100 MHz clock. Thanks to this parallel data bus, the FX3 allows the FPGA to directly access to its internal buffers making the data exchange very simple. At last, one of the main reasons for the choice of a Kintex-7 FPGA featuring GTX transceivers was because of the SFP+ optical module. Indeed it requires a high performance driver able to deliver serialized data at a line rate of 10.3125 Gbit/s.

Meticulous care was also dedicated to the design of the FPGA power supply: due to the demanding operating conditions ($1\text{ V} \pm 3\%$ the most critical), properly designed filters were inserted for each supply regulator in conjunction with positioning strategy to avoid electrical crosstalk between the linear regulators and the switched ones. The XC7K325T has 10 usable I/O banks, each consisting of 50 I/Os and sorted into 7 HR banks and 3 HP banks. According to the supply voltage level, each I/O block can perform various signaling standard, including HDL design primitives such as delayers and serdes. In pursuance of lowering the power consumption of the Data Management Board, both High Performance and High Range I/O banks are supplied with a single 1.8 V regulator.

With regard to the configuration options, the FPGA either automatically loads itself with configuration data from a non-volatile flash memory (e.g. *Master SPI configuration mode*), or the configuration data can be downloaded from a host computer through a cable to the JTAG port of the FPGA. This booting configuration allows maximum flexibility. In particular, the Micron N25Q128A [77] 128 Mbit serial NOR flash memory has been chosen to store the FPGA bitstream. Finally, a digital clock manager has been dedicated to the FPGA, in particular it is the Silicon Labs Si5338 [78]. This I²C programmable quad clock generator is employed to deliver 3 differential clocks to the FPGA, one for the core and the remaining two as reference for the GTX transceivers.

4.4 USB 3.0 Controller

The Cypress EZ-USB FX3 has been already introduced in section 3.3.1, although for sake of completeness, it is going to be described in more details. This controller integrates the USB 3.0 and USB 2.0 physical layer (PHY) along with a 32-bit, 200 MHz ARM926EJ-S microprocessor for powerful data processing. In order to provide high-bandwidth access to USB 3.0, FX3 contains a hardware unit called GPIF II that delivers glueless data transfer from GPIF II itself to the USB interface. It features also interfaces to connect to serial peripherals such as SPI, I²C, UART, and I²S, even though only the first two are being actually used. The GPIF II is also employed to interface with the Kintex-7 FPGA, through a 32-bit slave FIFO [69] path. More about this interface will

be discussed in section 5.1. The FX3 can load images from various sources, according to the combination of $PMODE[2:0]$ pins. The boot configuration chosen for the Data Management Board is $PMODE[F1F]$, which corresponds to loading the firmware from a non-volatile 1 Mbit EEPROM memory attached to the I²C, but in case of on failure, USB boot is also enabled.

A 19.2 MHz crystal oscillator is used to clock the controller core, in conjunction with a 32 kHz watchdog timer employed to detect and recover from malfunctions or standby operations. Besides its constrained supply voltage for the logic core (1.2 V) and the clock bank (3.3 V), the remaining FX3 power domains are left to the user application. As for the FPGA, also the FX3 I/Os are chosen to be supplied with a single 1.8 V regulator. Countermeasures against overvoltages and electrostatic discharges were also considered, indeed an external overvoltage protection device (NCP360 [79], manufactured by ON Semiconductor) is placed along the FX3's VBUS pin. Although the FX3 has a built-in ESD protection on the USB 3.0 pins against ± 2.2 kV human body model, I decided to include additional protection by using a low capacitance external ESD device to safely absorb strikes up to ± 15 kV air gap discharge, in particular the SP3010 [80], manufactured by Littelfuse.

Concerning the layout design, decoupling capacitors have been placed as close to the power pins as possible, ensuring that system noise does not propagate into the device through power supply. Indeed improper decoupling can lead to jittery signaling, which results in higher CRC error rate and more transfer retries. All the 32 lines on the GPIF II interface have been length matched within 1 μm (well beyond the recommended 12.7 mm limit) and terminated in series with 22 Ω resistors; these safety measures are employed to minimize timing skew, overshoot and ringing effects on the transmission lines. This is achieved by reducing the source voltage by approximately 50% close to the FPGA. Indeed, when the signal reaches the far end of the transmission line, the high impedance of the receiver causes a reflection which approximately doubles the signal back to its original amplitude. When the reflection returns to the series terminating resistor, the potential across the resistor drops to zero which prevents any more current from entering the transmission line. From the perspective of the receiver, this gives a perfect 100% logic transition without any overshoot or ringing. On the immediate plane underneath the AC coupling capacitors along the SuperSpeed differential signaling transmission lines, I made a cut-out in the shape of the capacitors themselves. In the same way I did for the SuperSpeed related pins on the receptacle side. These design tips avoid extra capacitance on the lines because of the connector pins or capacitor pads. Additionally, in order to treat the USB 3.0 lines in the best fashion way possible, e.g. avoiding stubs and via along the paths, I decided to use the vertical type-A connector as shown in figure 4.5. Indeed, this solution allowed me to route the lines entirely over a solid ground plane while matching the differential pair trace lengths within 5 μm .

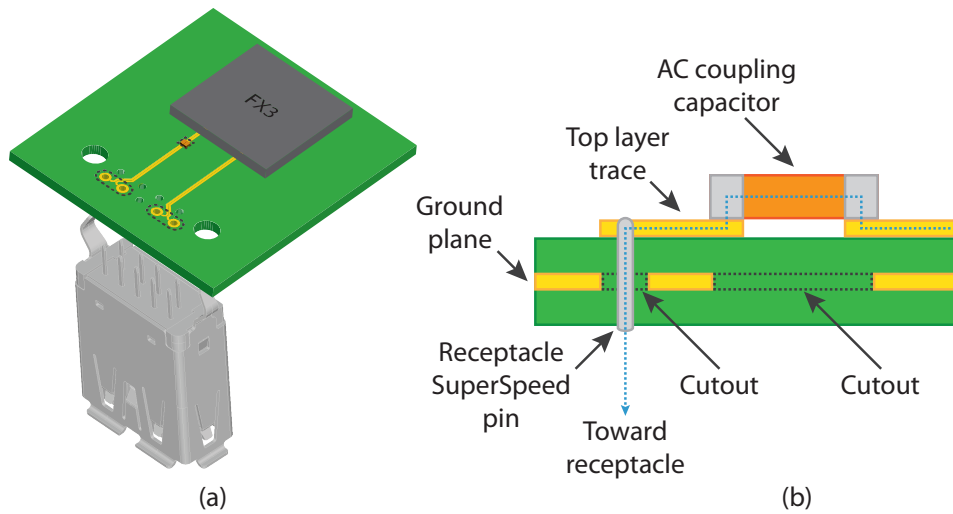


Figure 4.5: USB signals connected on the opposite side of the standard type-A USB receptacle. (a) component arrangement showing the SS TX/RX differential traces and plane cut-outs highlighted with dashed lines; (b) PCB cross-section view: the USB 3.0 type-A through-hole pin acts as a part of the signal trace, thus eliminating the possibility of a stub on the signal line.

4.4.1 SPI master interface

The EZ-USB FX3 supports an SPI master interface on the serial peripherals port and its maximum operation frequency is 33 MHz. This interface is being used to program the flash memory of the FPGA on this very board and the flash memory of the FPGA on each of the 32-channel TCSPC board.

The goal that led to this solution was the need to update FPGA firmwares from the outside, without opening the whole instrument. One possible solution was the Kintex-7 *Master SPI configuration mode* [81], which enables the use of industry-standard SPI flash devices for bitstream storage, indeed the FPGA supports a direct connection to the memory for reading the bitstream image. In conjunction, the Xilinx *iMPACT* programming software provides the ability to program a SPI serial flash through the JTAG interface using an indirect method, e.g. employing the FPGA itself as a medium. However, supposing to implement this solution, it would mean to provide the system with an extra JTAG connector and the three FPGAs suitably connected in daisy-chain, not to mention that the end-user should be supplied with a JTAG programmer. Therefore, discarding this option for the increased hardware overhead, we are left with the FX3 SPI master interface. The implemented architecture is shown in figure 4.6. The four SPI signals (e.g. MISO, SSN, MOSI, and CLK) are split into two pairs and forwarded toward two dual single-pole triple-throw switches (STG3856 [82], manufactured by STMicroelectronics). By properly addressing the target memory through a dedicated custom software, it is possible to program each flash with different bitstreams. Certainly, to avoid any conflict over the bus between FPGAs and FX3, the latter takes advantage of

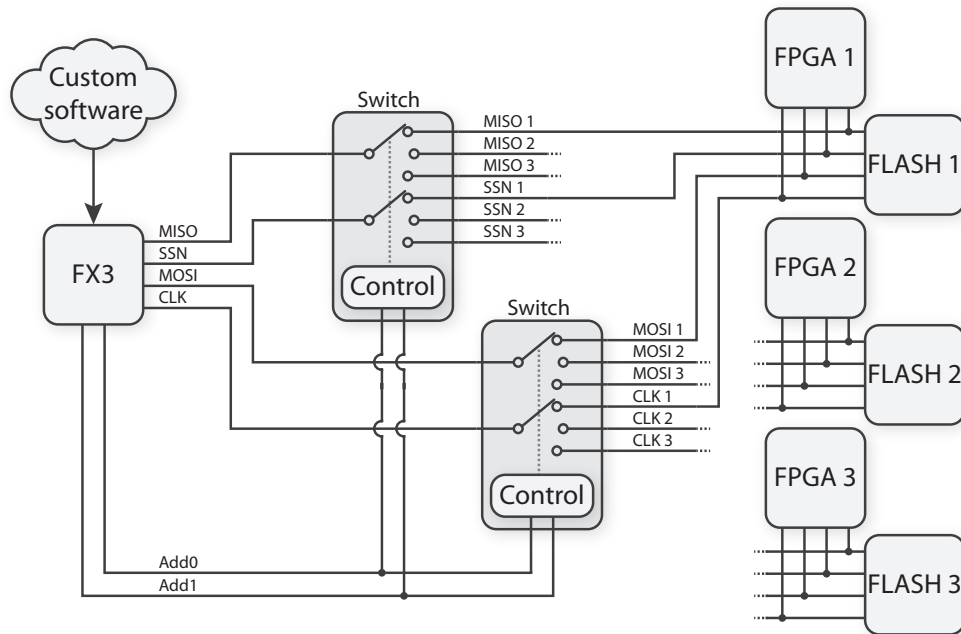


Figure 4.6: SPI bus architecture: this solution allows to address one flash memory per time, using the already embedded SPI interface of the FX3 controller. *INIT_B* and *PROGRAM_B* configuration lines are not depicted.

the FPGA *INIT_B* and *PROGRAM_B* configuration pins. The active-low *INIT_B* pin is driven low when the FPGA is in a initialization state; upon completion, this pin is released and pulled-up unless an external driver held it low to stall the power-on configuration sequence at the end of the initialization process. On the contrary, when *PROGRAM_B* is pulsed low, the FPGA current configuration is cleared and a new configuration sequence is initiated. When firmware update is needed, at system power-on the FX3 keeps the *INIT_B* low to gain command of the SPI bus for the time necessary to transfer the bitstream from PC to the flash memory; instead, when the instrument is already initialized, the FX3 pulses the *PROGRAM_B* low to restart the FPGA configuration process with an updated firmware. In order to let this architecture work properly, it is essential the high impedance state of the two switches when the FX3 is not using the SPI bus to avoid any crosstalk.

4.4.2 I²C serial communication

The FX3 I²C interface is capable of performing only as bus master. The operating frequencies supported are 400 kHz and 1 MHz at 1.8 V supply voltage on the corresponding I/O bank, however not all the slave devices on the Data Management Board are able to work at this signaling voltage. For this reason, a voltage-level translator has been added along the I²C bus. It is the Texas Instruments PCA9306 [83] dual bidirectional I²C voltage translator, which is employed to duplicate the FX3 signals from 1.8 V into 3.3 V. Figure 4.7 shows the FX3 I²C bus and the attached slave devices. On system

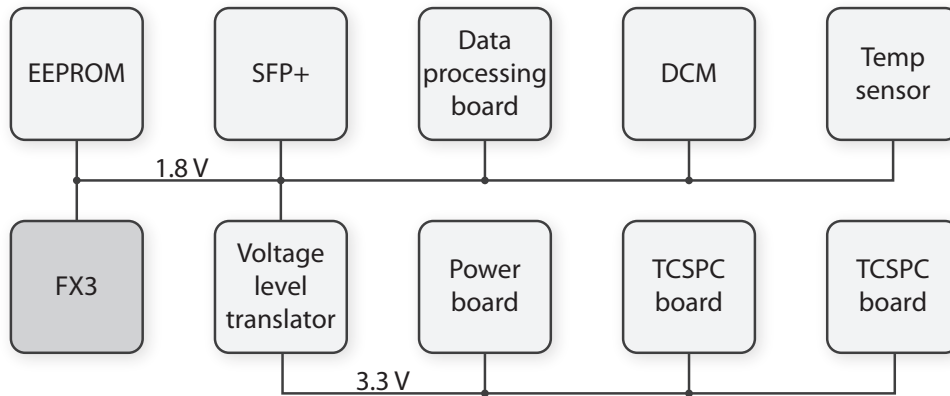


Figure 4.7: FX3 I²C bus and its slave devices split into 1.8 V and 3.3 V resources.

start up, the FX3 loads its firmware from the EEPROM, soon after it programs the DCM to deliver clocks to the FPGA core and GTX transceivers. Eventually, upon user request, the controller queries attached slaves for temperature monitoring (LM73) and operating status (SFP+ module).

4.5 SFP+ Daughter Board

As already mentioned in section 3.2.3, the SFP+ module serves only as optical-to-electrical converter, and the other way around. Due to its connector form-factor, e.g. right angle mounting only, it resulted into a quite challenging positioning plan. Indeed, laying the optical module parallel to the Data Management Board, besides the area occupation issue (15 x 49 mm²), it would require a cutout on one of the side board. A more elegant solution was then conceived: by employing a LC to LC fiber optic adapter, the Ethernet link receptacle could be moved close the SMA and USB 3.0 connectors as depicted in figure 4.8. Nevertheless, taking into account the misalignment that the adapter could introduce along the optical path and presence of a twisted fiber inside the instrument, also this solution was at last discarded. For these reasons, it has been decided to engineer a suitable solution in order to put the optical module orthogonally to the main board. Discarding the option of moving it on one of the four side board to keep the Data Management Board as independent as possible, a small daughter board was considered to be the best answer. By employing two mating 20-pin connectors [84] [85] manufactured by Samtec, a 48 x 45 mm² 2-layer PCB board was developed. This solution gives also the possibility to disconnect the optical module when Ethernet is not employed as transfer link, lowering the overall power consumption and cost, besides the fact that it saves extra space on the main board. Figure 4.9 and figure 4.10 show the SFP+ daughter board and the 3D rendering of the final Data Management Board structure respectively. From figure 4.9 you can notice that the SFP+ cage is sticking quite out from the board edge. Indeed, two out of eleven ground

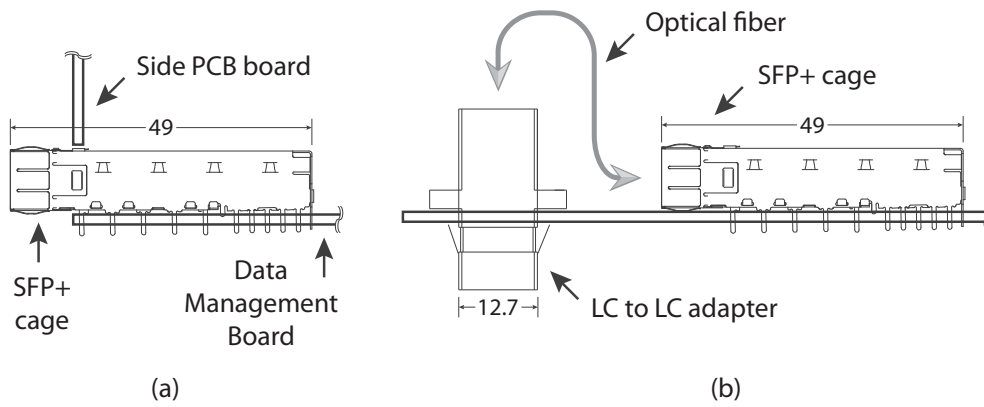


Figure 4.8: Section view of the Data management board. (a) The SFP+ assembly is meant to be installed on the PCB board edge: by doing so, the 1024-channel TCSPC system would have an inconvenient arrangement of connectors. (b) By employing a LC to LC fiber optic adapter, the Ethernet channel plug is placed on the same face of the other connectors. All measurements are in mm unless otherwise indicated.

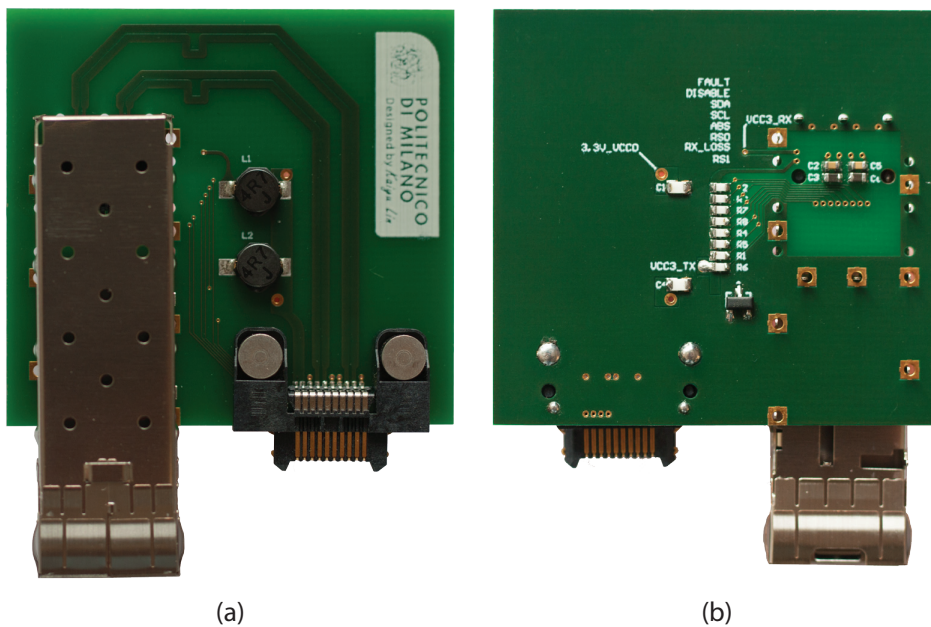


Figure 4.9: SFP+ daughter board. (a) Top view. (b) Bottom view.

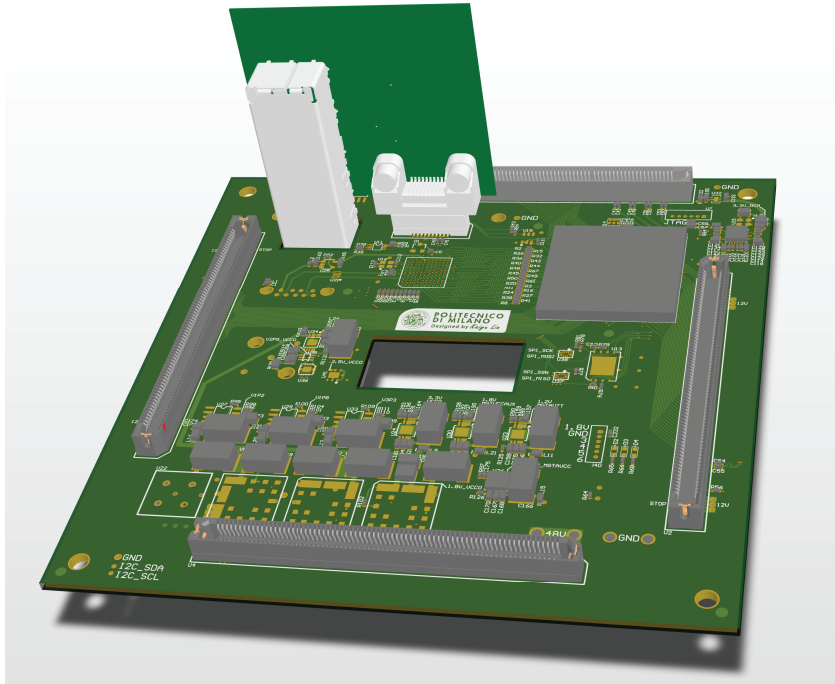


Figure 4.10: 3D rendered image of the final assembly of the Data Management Board.

pins have been intentionally left unconnected in order to meet the Data Management Board connectors height requirement explained in section 4.7. A single 3.3 V supply voltage is delivered from the main board to the SFP+ V_{ccR} and V_{ccT} pins, e.g. receiver and transmitter power supplies. Additionally, suitable filter stages are added: inductors with DC resistance of less than $1\ \Omega$ are used in order to maintain the required voltage at the SFP+ input pin within $3.3\ \text{V} \pm 4.8\%$.

4.6 Power Delivery Network

A complete scheme of the Power Delivery Network is shown in figure 4.11. The whole 1024-channel TCSPC system is supposed to be supplied with a single external AC adapter at 48 V and maximum 5.85 A (GS280A48-C4P [86], manufactured by Mean Well). These 48 V are directly conveyed toward the Power Management Board that deliver back a lower 12 V shared among the Data Processing Board, the two 32-channel TCSPC boards and the Data Management Board.

On this board a compact EMI filter, in particular the Murata BNX016 [11], is placed on the very beginning of the network before any load. It ensures a minimum of 40 dB insertion loss, in the range of frequencies between 100 kHz and 1 GHz. Since no load requires supply higher than 3.3 V, I decided to reduce the 12 V down to 5 V. This intermediate step-down conversion improves the overall efficiency, indeed it lowers both distribution losses and dropout voltage across linear regulators. The estimated maximum power consumption of the Data Management Board is less than 12 W, which

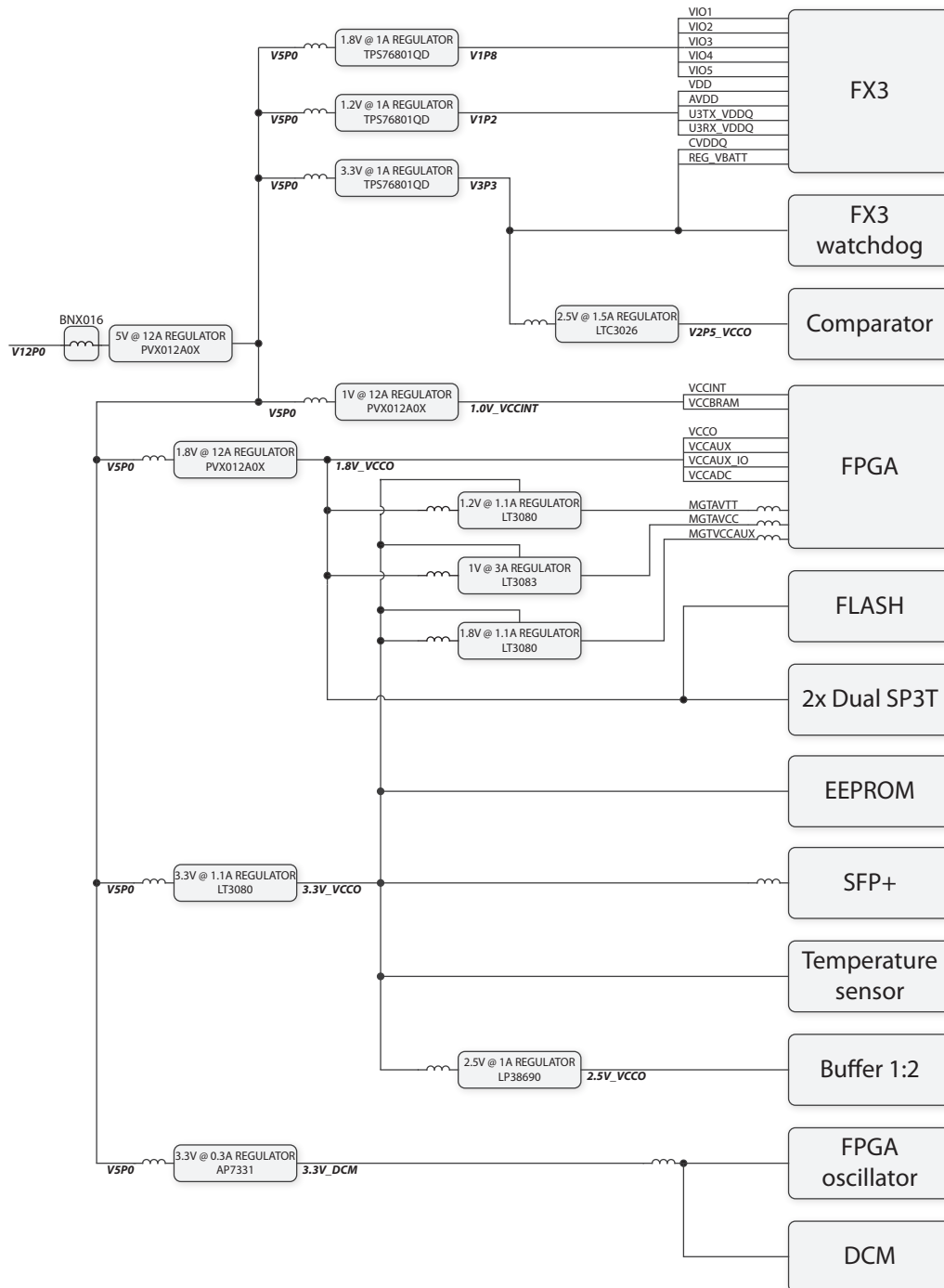


Figure 4.11: Power Delivery Network of the Data Management Board. A single 12 V domain is supplied from the Power Management Board, then filtered through the BNX016 [11] hence down-regulated into 5 V using a compact buck converter module. The inductor sign stands for a noise filtering stage designed with ferrite bead.

means that the previously mentioned regulator must be able to deliver at least 2.4 A at 5 V. A DC-DC buck converter was chosen to supply the entire board thanks to its higher efficiency compared to linear regulators. The complete design of a feedback network for a buck converter is not an easy job, indeed many issues need to be taken into account such as stray effects in the layout, decoupling of input ground reference from the output one, current loops etc. In order to overcome all these problems, I decided to make use of compact power modules manufactured by General Electric. They are a family of non-isolated DC-DC board-mounted units that feature efficiency up to 94% with a tunable analog loop. In particular, the PVX012A0X [87] was chosen to down-convert the 12 V into 5 V. This surface mounting 12.2 x 12.2 mm² module is able to supply output currents up to 12 A with only 10 mV load regulation bounce over a full swing of load current. These characteristics match perfectly the demanding FPGA core supply requirements, hence two more PVX012A0X have been used to deliver the *1.0_VCCINT* and the *1.8_VCCO* as shown in figure 4.11. However, the GTX transceivers power supply specifications are even more critical with respect to the load regulation capabilities of the PVX012A0X. The Kintex-7 datasheet reports a maximum of 10 mV peak-to-peak noise on the *MGTAVTT* and *MGTAVCC* banks. For this reason, a further stage of linear regulators have been dedicated to the GTX transceivers. They are the LT3080 [88] and the LT3083 [89], manufactured by Linear Technology, which features 40 μ V rms output noise and less than 1 mV load regulation. In conjunction, properly designed ferrite filters are placed close to the FPGA supply pins. With reference to the FX3 controller, it is powered by three identical linear regulators (TPS76801QD [90] manufactured by Texas Instruments) that deliver respectively 1.2 V, 1.8 V and 3.3 V. It is also worth speaking of the 2.5 V supply voltage for the *stop* signal conditioning stage. Concerning the comparator [72], it can operate from a single 2.5 V to 5.5 V positive supply whereas the 1:2 buffer [73] from 2.5 V up to 3.3 V. From experimental results, it turned out that the optimum timing performance is reached when this stage operates at the minimum supply voltage, e.g. with 2.5 V; low noise linear regulators are dedicated to these two devices to further improve timing characteristics.

The Data Management Board is made on a custom stack-up 8-layer 1.6 mm PCB. Of the overall available layers for routing, I have dedicated 4 to signals, 4 to power planes. In particular, 2 solid ground planes were placed just beneath the top and the bottom layers, whereas the two internal planes were entirely used to implement the Power Delivery Network. The reason why I chose a custom solution arises from the 32-channel TCPSC board, indeed the former has to deal with reduced trace width and via diameter in order to get to the TAC pads. Making both boards with the same technology would result in an overall cheaper production costs. The custom stackup is reported in table 4.4 with the target single-ended and differential signaling impedance.

Layer No.	Description	Layer Name	Dielectric Thickness	Copper Thickness	Trace Clearance	Trace Width	Z _o [Ω]	Z _{diff} [Ω]
	Soldermask		12.7					
1	Signal	Top		35	115	140	67.56	100.01
	Prepreg		160					
2	Plane	GND		35				
	Core		250					
3	Signal	Inner 3		35	140	115	55.83	87.9
	Prepreg		240					
4	Plane	Vcc		35				
	Core		250					
5	Plane	Vcc		35				
	Prepreg		240					
6	Signal	Inner 6		35	140	115	55.83	87.9
	Core		250					
7	Plane	GND		35				
	Prepreg		160					
8	Signal	Bottom		35	115	140	67.56	100.01
	Soldermask		12.7					

Table 4.4: The Data Management Board custom stackup and the resulting trace impedance. All measurements are in μm unless otherwise indicated.

4.6.1 Ferrite Bead Filter Design

In such a complex Power Delivery Network and demanding load devices, it is good practice to impose countermeasures against noise propagation and crosstalk among different power domains. The easiest way to reduce voltage ripples are bypass capacitors, indeed if properly placed close to power supply pins, can dampen AC components and compensate sudden voltage droop caused by large current transients. To further enhance the filtering effects, a ferrite bead could be employed in conjunction with bypass capacitors to isolate noise spreading.

Ferrite beads are inductors constructed using one of the many available ferrite materials and show up typically a bell-shape frequency behavior known also as ZRX curve. This component can be modelled with four elements: R_{DC} , L_{BEAD} , R_{AC} and C_S [91] as shown in figure 4.12. At low frequencies, the inductive element dominates the impedance, whereas at medium frequencies the ferrite bead appears resistive. At high frequencies instead, the capacitance element dominates but usually the bead is not employed for this characteristic. Leaving all the theoretical analysis behind (for further readings refer to *PDN Application of Ferrite Beads* by Steve Weir), I will explain to you how to make a nice ferrite filter through a practical example design applied to a critical power node of the Data Management Board. Figure 4.13 shows a typical filter configuration that makes use of a ferrite bead. Before any calculation, it is necessary to establish the following requirements for the filter:

- Load maximum input voltage ripple ($\Delta V|_{MAX}$)
- Load maximum current ($I|_{MAX}$)

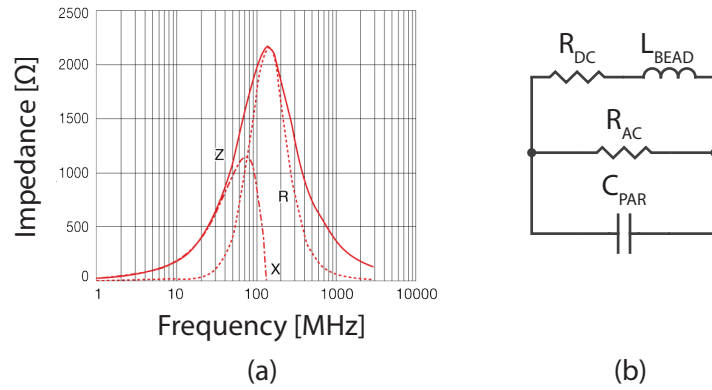


Figure 4.12: (a) Typical ZRX curve of a ferrite bead: the contiguous red line (Z) is the overall impedance behavior versus frequency. (b) First order approximation model of a ferrite bead.

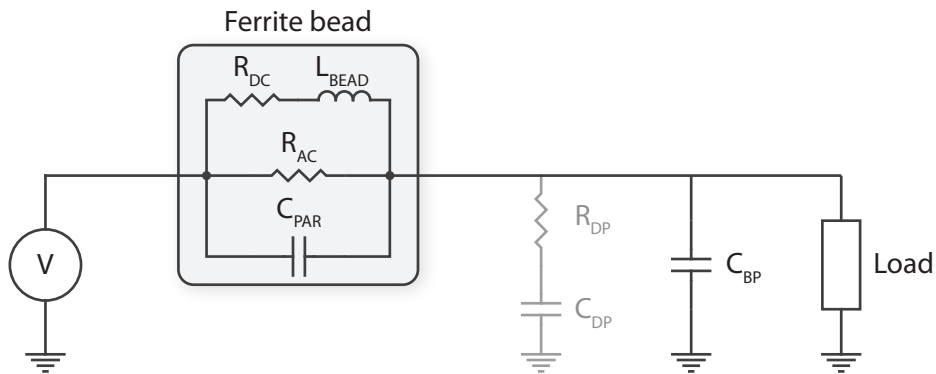


Figure 4.13: Typical filter configuration: ferrite bead in conjunction with a bypass capacitor. R_{DP} and C_{DP} make up a compensation stage to reduce resonance issues.

- Attenuation frequency range (BW)
- Attenuation magnitude (A)

Once known the specifications, we can proceed with the following ten steps:

1. Determine the load side impedance versus frequency (Z_{22})
2. Determine the maximum bead DC resistance (R_{DC})
3. Determine the maximum allowable bypass network equivalent series inductance (ESL)
4. Determine the Z_{BEAD} to meet the attenuation magnitude requirement
5. Determine the stop band high frequency limit (F_{CO})
6. Determine the L_{BEAD}
7. Look for a commercially available ferrite bead with the former specifications
8. Determine the value of the bypass capacitor
9. Determine the value of the compensation capacitor (when necessary)
10. Simulation of the filter

As example, let's review the design of the filter that comes before the FX3 1.8 V power supply regulator (TPS76801QD) shown in figure 4.11. This power network node is critical because it shared among many other regulators, both switched and linear, hence in order to isolate the FX3 from the remaining loads and the other way around, it is a good choice to put a ferrite bead filter in this very position.

The minimum allowed voltage is 1.7 V [68] on the FX3 I/O banks, which result into a maximum ripple of $\Delta V|_{MAX} = 100$ mV. The estimated overall power consumption is 360 mW whereas the regulator dissipates 570 mW at 25°C. This translates into a peak current flowing through the ferrite bead of $I = (360 \text{ mW} + 570 \text{ mW}) / 5 \text{ V} = 186$ mA; let's take a safe margin and approximate it with $I|_{MAX} = 200$ mA. The maximum working frequency of these I/O pins is 100 MHz, by setting the attenuation range from 20 MHz up to 200 MHz the resulting filtering effect should be quite effective; the lower limit is chosen by taking a margin factor 5 below the switching frequency, instead the upper limit is calculated using the following equation [92]:

$$F_{knee} = \frac{0.5}{T_{rise}} \quad (4.1)$$

where F_{knee} is the frequency below which most energy in digital pulses concentrates, whereas T_{rise} is the signal rise time. The FX3 T_{rise} is 3 ns as reported in its datasheet, hence the resulting F_{knee} equals 166 MHz which has been overestimated into 200 MHz.

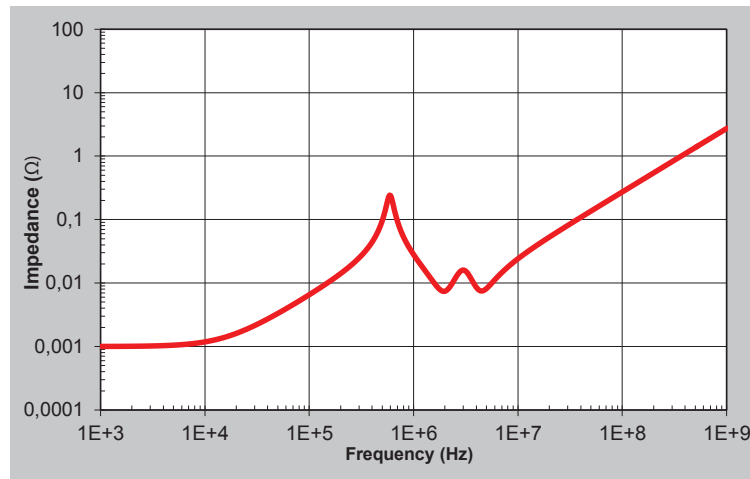


Figure 4.14: Load side impedance versus frequency with one $4.7 \mu\text{F}$ bypass capacitor. The resulting Z_{22} red line stays below 1Ω until 400 MHz.

At last, let's force the minimum attenuation magnitude to $A = 40 \text{ dB}$. Just for sake of simplicity, the requirements are here reported:

- $\Delta V|_{\text{MAX}} = 100 \text{ mV}$
- $I|_{\text{MAX}} = 200 \text{ mA}$
- $\text{BW} = 20 \text{ MHz} \div 200 \text{ MHz}$
- $A = 40\text{dB}$

We can now carry on with the design steps:

1. The maximum load side impedance in the range of frequencies of interest is estimated through simulation. I used the Altera Power Delivery Network tool [93] shown in figure 4.14. The red line stands for the Z_{22} impedance when one $4.7 \mu\text{F}$ capacitor is employed to bypass the input node as suggested by the regulator datasheet. As you can see, the resulting impedance stays always below 1Ω in the BW range, hence by taking a little margin we can set $Z_{22} = 3 \Omega$
2. R_{DC} is the maximum series resistance which corresponds also to the maximum allowable voltage drop across the ferrite bead: $R_{\text{DC}} = \Delta V|_{\text{MAX}} / I|_{\text{MAX}} = 0.5 \Omega$
3. The maximum bypass network equivalent series inductance is calculated using this formula:

$$2 \cdot \pi \cdot 100\text{MHz} \cdot \text{ESL} \leq Z_{22} \quad (4.2)$$

which results to be $\text{ESL} \leq 4.7 \text{ nH}$. Note that in this calculation, the stray inductive effects due to the layout are neglected leading to a conservative estimation.

4. To get a 40 dB attenuation, the ferrite bead must feature a big impedance at the switching frequencies in order to dissipate noise into heat. This parameter is calculated with the following formula:

$$Z_{\text{BEAD}} \geq Z_{22} \cdot A = 300\Omega \quad (4.3)$$

5. The filter cut-off frequency F_{CO} is found using this equation:

$$F_{\text{CO}} = F_{\text{low}} \cdot 10^{-A/40} = 20\text{MHz} \cdot 10^{-40/40} = 2\text{MHz} \quad (4.4)$$

6. L_{BEAD} must satisfy the following condition [91]:

$$L_{\text{BEAD}} \geq \frac{Z_{22} \cdot 0.71 \cdot}{2 \cdot \pi \cdot F_{\text{CO}}} = 169.5\text{nH} \quad (4.5)$$

7. The Würth Elektronik 742792035 [94] ferrite bead has excellent characteristics that accomplish the former requirements:

- $L_{\text{BEAD}} = 0.448 \mu\text{H}$
- $Z_{\text{BEAD}} = 300 \Omega \text{ } 100 \text{ MHz}$
- $R_{\text{DC}} = 0.3 \Omega$
- $I_{\text{MAX}} = 300 \text{ mA}$

8. The bypass capacitor (C_{BP}) of the filter must satisfy the most critical of the following two conditions:

$$F_{\text{CO}} \leq \frac{1}{2 \cdot \pi \cdot \sqrt{L_{\text{BEAD}} \cdot C_{\text{BP}}}} \quad (4.6)$$

$$\sqrt{L_{\text{BEAD}}/C_{\text{BP}}} \leq Z_{22} \quad (4.7)$$

from equation 4.6, $C_{\text{BP}} \geq 7.9 \text{ nF}$, instead from equation 4.7, $C_{\text{BP}} \geq 88.9 \text{ nF}$. Let's choose $C_{\text{BP}} = 100 \text{ nF}$ and verify through Altera tool that the Z_{22} still lays below 3Ω .

9. This filter architecture presents a resonance frequency at $\Omega = 1/\sqrt{L \cdot C}$; trying to find out the quality factor in order to evaluate if damping compensation is necessary could be a tricky job. Unless you have a restricted area for your layout, it is good to apply the compensation anyway. Typically, a *5x dominant pole* [91] compensation works just fine. It consists of a bulk capacitor C_{DP} in series with a resistor R_{DP} that introduces a pole well below the design cut-off frequency. Their values are here calculated:

$$C_{\text{DP}} = 5 \cdot (C_{\text{BP}} + 4.7\mu\text{F}) = 24\mu\text{F} \quad (4.8)$$

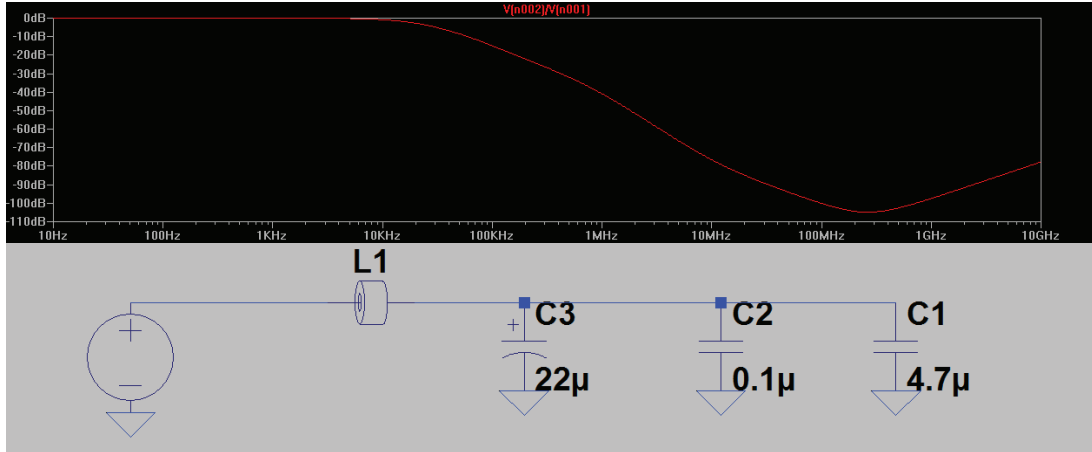


Figure 4.15: LTspice simulation results of the design ferrite bead filter. A commercially available value of $C_{DP} = 22 \mu\text{F}$ is chosen instead of the $24 \mu\text{F}$, but this slight modification does not influence the filter performance.

$$R_{DP} \geq 1.3 \cdot \sqrt{\frac{L_{\text{BEAD}}}{2 \cdot C_{DP}}} = 50\text{m}\Omega \quad (4.9)$$

Since R_{DP} is such a small value, we can think of employing a tantalum capacitor with a proper ESR.

10. I used the Linear Technology LTspice tool to simulate the whole filter design. Figure 4.15 shows the insertion loss versus frequency diagram. As you can see from the plot, the required attenuation of 40 dB is indeed achieved starting from 1 MHz, featuring an even greater stop band frequency range and without any resonance peak thanks to the compensation stage.

From the obtained results, you can easily infer how powerful is a properly designed ferrite bead filter. The same design steps have been applied to the remaining critical nodes of the Data Management Board power network.

4.7 Mechanical considerations

During the design process of the Data Management Board, besides mere electronic considerations, also mechanical issues were examined. Being the only interface to the outside world, all connectors (USB 3.0, SMA, SFP+, DC supply) had to be mounted accordingly on the same face of the board, keeping in mind that their height must be compliant with the instrument box as shown in figure 4.16. Moreover, the Data Management Board acts as the main mechanical support for the side boards: four high performance 150-pin Samtec connectors [95] have been chosen to ensure stability to the whole instrument.

Concerning thermal management of the 1024-channel TCSPC module, the heat generated by the detection head must be carried out from the instrument. The bulky

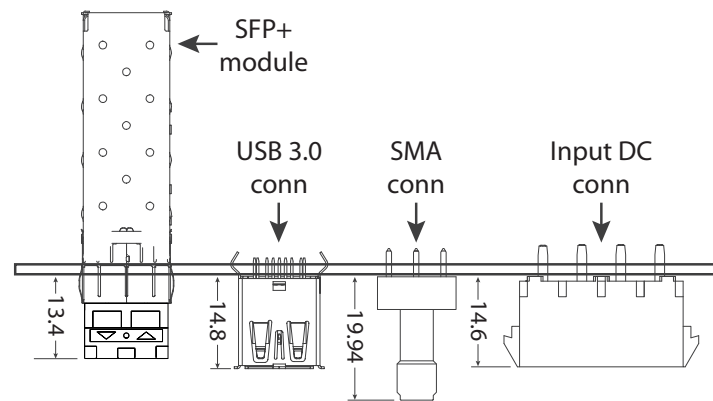


Figure 4.16: Section view of the Data management board: the connectors were carefully chosen to exhibit almost the same height. All measurements are in μm unless otherwise indicated.

solution of an active heat sink was discarded in favor of a more efficient liquid cooling system. For this reason a $15 \times 30 \text{ mm}^2$ cutout has been made in the middle of Data Management Board to host two $3/8$ " cold fingers.

Chapter 5

Firmware and Software

The purpose of this chapter is to describe the most relevant firmware and software elements that work behind the system. In particular, I will explain the State Machine implemented in VHDL that communicates with the FX3 GPIF II interface. At last, I will outline briefly the C# software under development used to manage both the SuperSpeed USB 3.0 and the 10GBASE-X communication link, along with control features.

5.1 Slave Fifo Interface: VHDL State Machine

As already introduced in section 4.4, the GPIF II interface is employed to communicate with the Kintex-7 through a 32-bit synchronous slave FIFO [69] link. Allowing the FPGA to directly access the internal memory of the FX3 for data read/write operations, makes this interface an excellent choice for high throughput applications such as the Time-Tag. Figure 5.1 shows the interface signals:

- *SLCS#*: this is the active-low chip select signal
- *PKTEND#*: this strobe is employed to transfer a short packet or a zero length packet
- *FLAGA/FLAGB*: strobes that signal the availability of an FX3 memory socket
- *A[1:0]*: 2-bit address bus for socket selection
- *D[31:0]*: 32-bit data bus
- *SLWR#*: this is the active-low write strobe asserted to perform write operation
- *SLRD#*: this is the active-low read strobe asserted to perform read operation
- *SLOE#*: this is the active-low output enable signal that must be asserted to allow the FX3 to drive the D[31:0] data bus for read operation
- *PCLK*: this is the interface clock

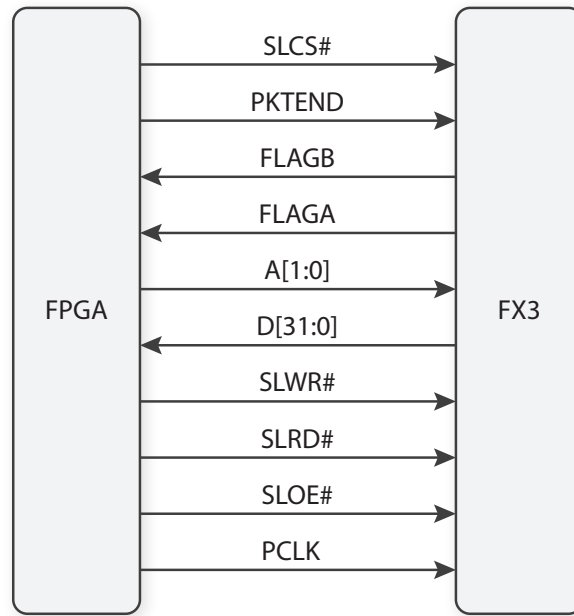


Figure 5.1: Synchronous Slave FIFO interface diagram implemented between the Cypress EZ-USB FX3 and the Xilinx Kintex-7 FPGA.

In order to properly understand how these signals allow a glueless data transfer from the FPGA toward the FX3 RAM, I will briefly explain the concepts of Socket, DMA descriptor, DMA buffer, and finally GPIF thread. Figure 5.2 illustrates the relationship among these entities with a transfer out example. A socket is a point of connection between a peripheral hardware block such as the GPIF II and the FX3 RAM. It includes a set of registers, which point to the active Direct Memory Access (DMA) descriptor.

A socket that write data into the FX3 RAM is called a *producer* socket. A socket that reads data is called a *consumer* socket. The DMA descriptor instead holds information about the address and size of a DMA buffers, known also as DMA channel, which is just a section of the available RAM; the maximum buffer size is 64 kB. A GPIF II thread is a dedicated data path that connects the external data pins to a socket. The EZ-USB FX3 provides up to four physical hardware threads for data exchange over the GPIF II, addressed with the $A[1:0]$ signals. The buffer empty, full, partially empty, or partially full states are signaled with flags that can be associated with dedicated thread or current thread.

The FX3 controller comes with a software development environment, in particular the *GPIF II designer* allows to implement GPIF II related firmware by creating a header file in ANSI C language. Finally, this header file is included in the ultimate firmware using *Eclipse* platform. In this very case, I used the *Streaming IN* sub-component of the Slave FIFO firmware that performs a continuous one-direction transfer from FPGA to the host PC. The main characteristics of this firmware are:

- Enumerates the Cypress VID/PID to 0x04B4/0x00F1

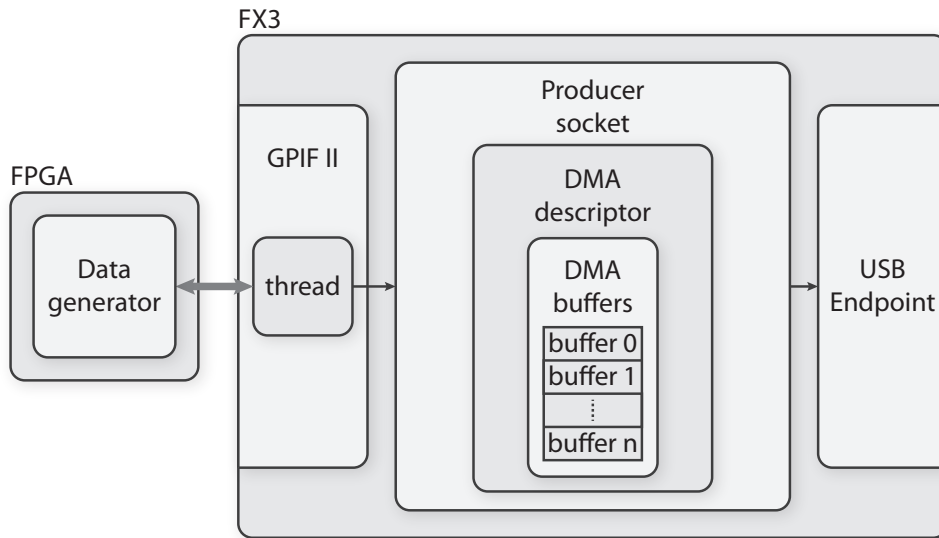


Figure 5.2: Interaction between FPGA and FX3 internal entities while performing a write operation.

- Configures two flags:
 - FLAGA: Full flag dedicated to thread0
 - FLAGB: Partial flag with watermark value 6, dedicated to thread0
- Sets up 8 buffers of 16 kB each for *Streaming IN* transfer

Note that the watermark value asserts the partial flag when the number of 32-bit words that may be written after clock edge at which the flag is sampled low is equal to *watermark* - 4. A partial flag may only be used when to end a transfer, hence if a counting mechanism is implemented on the FPGA in order to write always an amount of data that do not exceed the size of the buffers, this flag can be avoided. Concerning the State Machine implementation, figure 5.3 shows the logic block diagram. The Stream IN operation can be divided into four stages. The entry point is the *Idle phase*, in which the FPGA sets the control signals shown in figure 5.1 as follows:

- PKTEND# = 1
- SLOE# = 1
- SLRD# = 1
- SLCS# = 0
- SLWR# = 1
- A[1:0] = 00

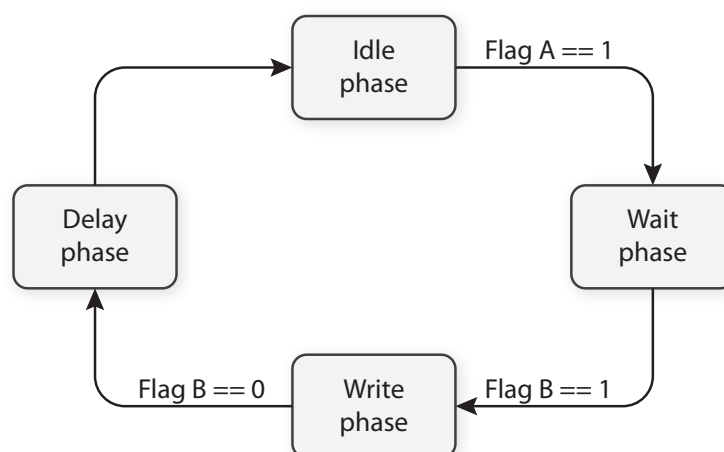


Figure 5.3: FPGA State Machine for Stream IN operation.

Whenever Flag A equals 1, the State Machine enters the *Wait phase* and holds up for Flag B to be asserted. When this happens, e.g. *Write phase*, it means that the FX3 has an available empty buffer hence the master FPGA performs a write operation by setting the control signals as follows:

- $\text{PKTEND}\# = 1$
- $\text{SLOE}\# = 1$
- $\text{SLRD}\# = 1$
- $\text{SLCS}\# = 0$
- $\text{SLWR}\# = 0$
- $\text{A}[1:0] = 00$

When the Flag B is sampled low, the FPGA changes $\text{SLWR}\#$ to high and the State Machine enters into the last condition, e.g. *Delay phase*. This phase awaits for one clock cycle before moving to the initial *Idle phase*. The delay is necessary in order to attain to the timing specification of the FX3.

5.2 1000BASE-X: VHDL implementation

The hardware setup built to test the Gigabit Ethernet has been already introduced in chapter 3. The goal of this section is to further explain the logic core implemented in the FPGA. In particular, two Xilinx IP cores have been exploited to manage this communication link: the *Tri-Mode Ethernet Media Access Controller* [63] (TEMAC) and the *Ethernet 1000BASE-X PCS/PMA or SGMII* [64]. Figure 5.4 shows a block diagram concerning the FPGA internal architecture. The whole design is synchronous

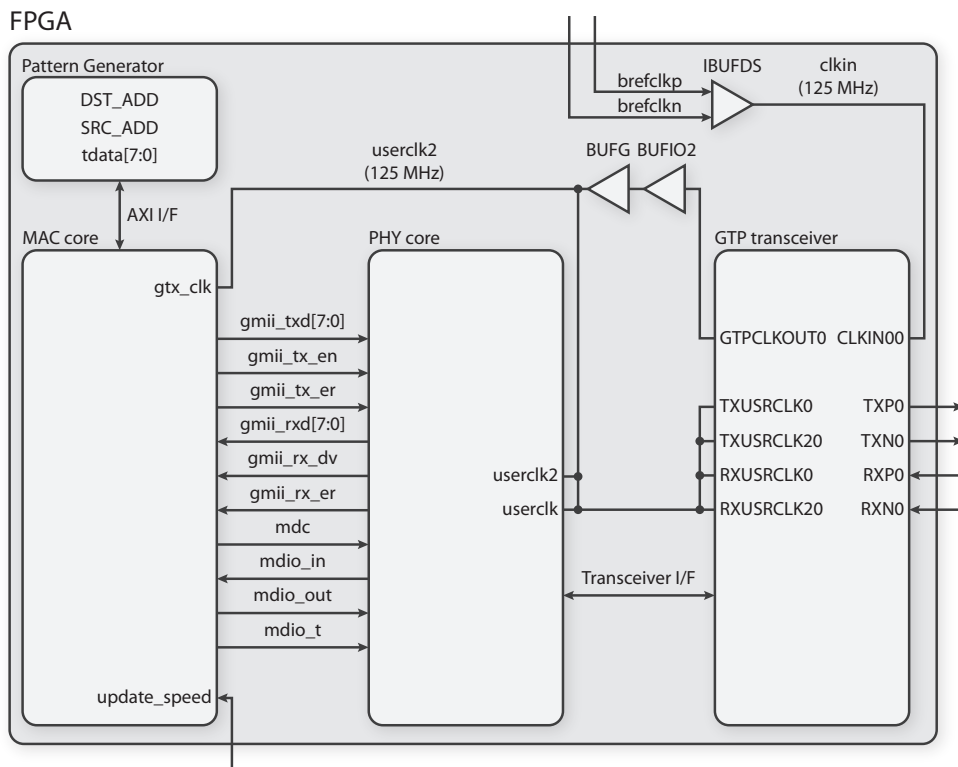


Figure 5.4: Implementation of the 1000BASE-X core: an external 125 MHz clock is fed to the GTP transceiver block that forward out a copy for the all the remaining logic. The Pattern Generator is in charge of hard-coding the MAC addresses and the Ethernet frames. The interface between the two main cores is based on the Gigabit Media Independent Interface (GMII) along with a Management Data I/O control interface.

to a single clock domain at 125 MHz. This clock is buffered from the outside through a *IBUFDS* primitive, fed to the GTP transceiver block and finally delivered to each of the sub-component by a cascading of *BUFIO2* + *BUFG* primitives. The Pattern Generator block packs known data into Ethernet frames and transfer them toward the MAC block through the Advanced eXtensible Interface (AXI). Information exchange between the MAC core and the PHY core is implemented on the Gigabit Media Independent Interface (GMII), in conjunction with an auxiliary Management Data I/O interface (MDIO) to allow the MAC core to access the embedded configuration and status registers of the PHY core. Finally, parallel data (*gmii_txd[7:0]*) is serialized and sent toward the SFP+ optical module at 1.25 Gbit/s lane rate.

5.3 GUI C# Software

As for the previous TCSPC systems developed in our research group, a custom software is needed to manage the communication links. Concerning the Data Management Board, this translates into a more challenging development since the 10GBASE-X does not come with a ready-to-use API nor DLL libraries. To deal with this issue, I intend to investigate the Microsoft *Socket Class* of .NET framework. In order to keep the user interface similar to be previous one and work in a friendly coding environment, I chose the C# programming language. So far, few logic blocks of the final software have been already developed and tested for the SuperSpeed USB link, such as acquiring and saving data from the FPGA through the USB controller, manage flash memories read/write operations (section 4.4.1), and I²C bus transactions. These functional blocks make use of the DLL library provided by Cypress Semiconductor. Yet much work has to be done, indeed all these features must be packed into one simple firmware that interacts with the GUI application. Figure 5.5 shows one possible flowchart of the software currently under investigation. Once the FX3 has loaded its firmware and verified that all the FPGAs are initialized, it goes into the *IDLE* state. From this point on, many operation can be performed accordingly to the user requests through the GUI: read/write I²C slave registers, read/write one of the three FPGA flash memories, or start a data transfer from the TCSPC boards. If the USB 3.0 channel is selected to download data, the controller must reconfigure itself with the *Slave FIFO* firmware. Indeed, this step is necessary since much of the FX3 internal resources have to be reallocated. Instead, when the Ethernet link is chosen the controller is free to manage further instruction by eventually returning into *IDLE* state.

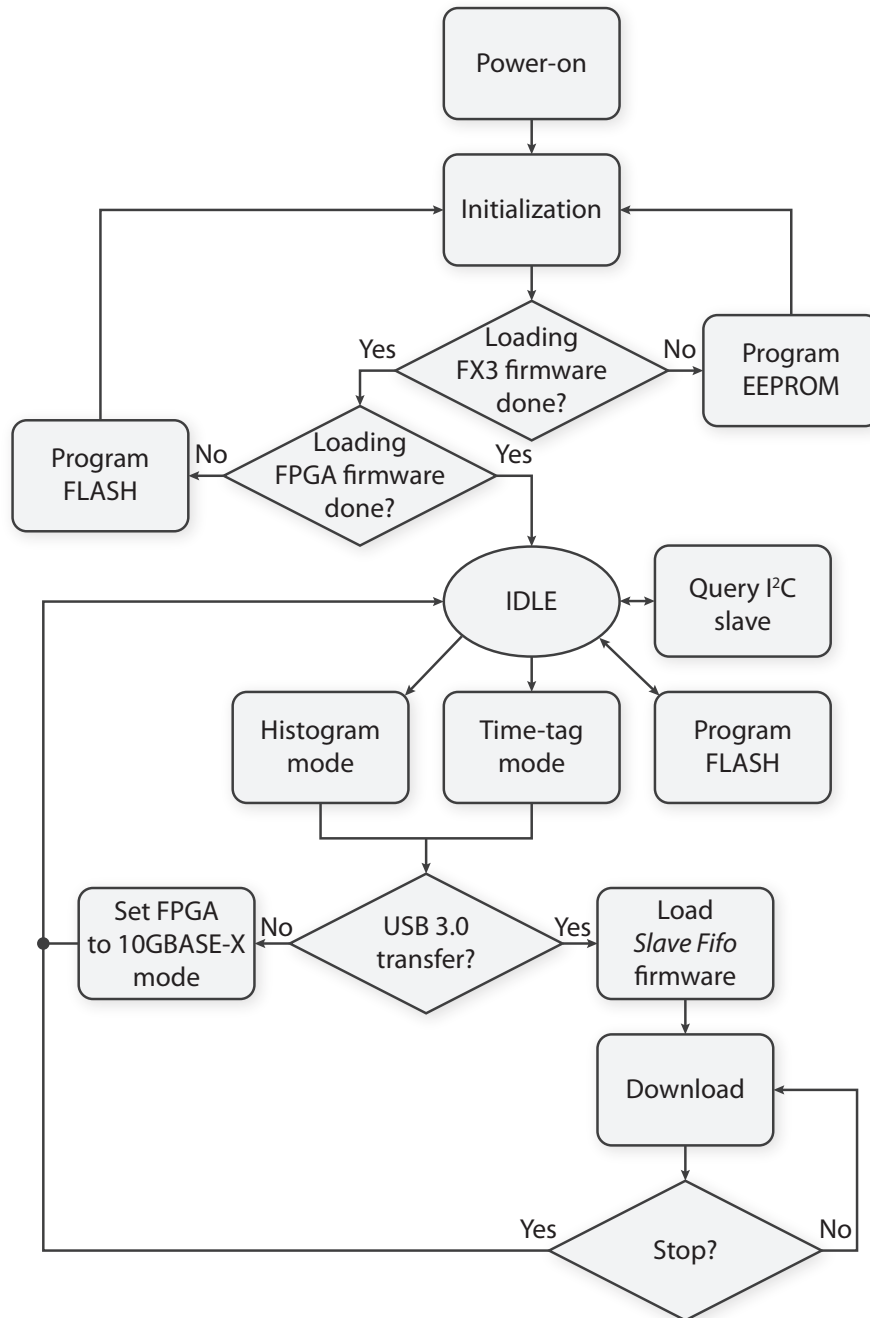


Figure 5.5: Scratch of the software flowchart that I intend to develop. The *IDLE* state is the key point: when the FX3 is in this condition, it awaits for user requests. Note that when the user choose to transfer data through the Ethernet channel, the FX3 is free to accept further instructions by returning in *IDLE* state.

Chapter 6

Experimental Results

On the behalf of the successful feasibility tests of both the SuperSpeed USB 3.0 link and the 1000BASE-X link, the Data Management Board has been finally assembled. This chapter is dedicated to show the achieved result in terms of throughput for USB 3.0 communication channel, in particular it will focus on the optimization of the FX3 firmware. Furthermore, the work being done on the Ethernet 10GBASE-X will be discussed along with some encountered design issues.

6.1 SuperSpeed USB 3.0 results

In order to supply power to the Data Management Board without an external AC-DC adapter, a small 2-layer PCB has been purposely designed. It mounts the same connector as the one employed by the side boards, hence it acts like the Power Management Board by buffering the 12 V from a laboratory power supply.

The very first test I made before any firmware level optimizations, was to check if any improvement has been introduced by just placing the FPGA and the FX3 next to each other on the same board. This experiment, made on the same PC and software as the ones mentioned in section 3.3.2, is aimed to get the achievable throughput of the Data Management Board over the USB 3.0 channel. Reasonably, we expect to get almost the same bitrate of the previous setup test as shown in figure 6.1. Even though the experiment worked out smoothly, the peak speed is still far from the achievable 400 MB/s reported in the FX3 datasheet. Considering that the throughput is heavily dependent on the host PC controller and operating system, I tried to maximize the USB 3.0 channel bandwidth by exploiting the FX3 transfer parameters and reallocating its internal hardware resources.

The following test is supposed to show the bare maximum attainable bitrate, avoiding any overhead introduced by processing data in FIFO or exchanging off-chip data, e.g. the FPGA is not employed. To further enhance performance, the FX3 instruction cache is enabled and the data cache is disabled. The SuperSpeed endpoints support a maximum data packet payload size of 1 kB and burst sizes from 1 to 16. By data burst-

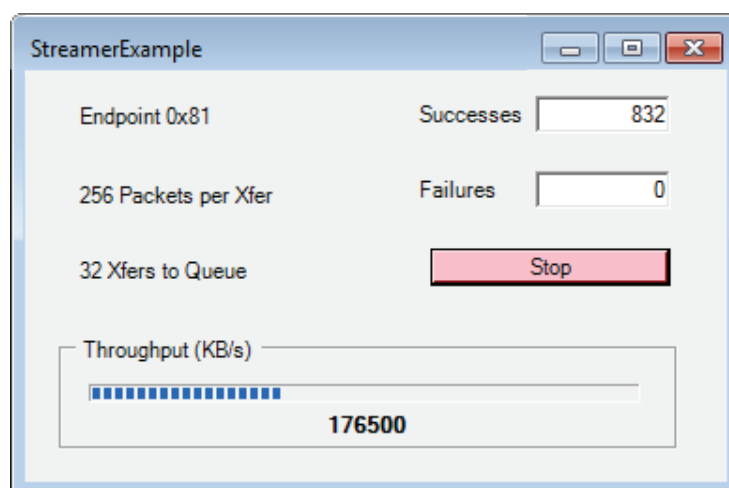


Figure 6.1: Snapshot of the custom software I developed to receive and store data from the Cypress FX3. The displayed throughput is comparable with the one obtained during feasibility tests.

ing, we allow a certain number of packets to be transferred over an endpoint without requiring a handshake in between packets. The maximum achievable throughput for bulk transfer mode will be approximately around 4 Gbit/s, indeed 20% of the theoretical bandwidth is reserved for handshakings and protocol-level overheads. By going through different combination of burst length, buffer size and number of buffers per DMA channel, the optimum solution appears to be 16, 48, and 2 respectively. For sake of clarity, a DMA buffer is a section of RAM used for intermediate storage of data transferred through the FX3 device endpoints, and they are instantiated in the RAM by the FX3 firmware. In particular, the previous mentioned configuration allocates 2 buffers of 48 x 1024 B each in RAM, whereas the FX3 performs 16 sequential transfers of 1 kB packets toward the PC. Using two large DMA buffers (e.g. 48 kB) that can include multiple burst of data improves the performance. The average time for a 16 kB data transfer at 443900 kB/s is 36 μ s, whereas the API functions (e.g. *CyU3PDmaChannelGetBuffer* and *CyU3PDmaChannelCommitBuffer*) employed to switch in between buffers and the endpoint costs about 40 μ s as reported in the datasheet. It is clear that the bigger the buffer size, the more negligible is the firmware processing time for one buffer. Moreover, in order to prove that the bandwidth is directly dependent on the host software and hardware configuration, figure 6.2 shows a comparison between the Cypress C++ Streamer application run on two different machines, instead table 6.1 lists the corresponding PC components.

Once again, it should be pointed out that these transfer bandwidths are obtained just to show up the achievable performance, indeed no useful information was being delivered from the FPGA. The actual FIFO interface between the Kintex-7 and the FX3 is currently under development, in conjunction with the optimization parameters

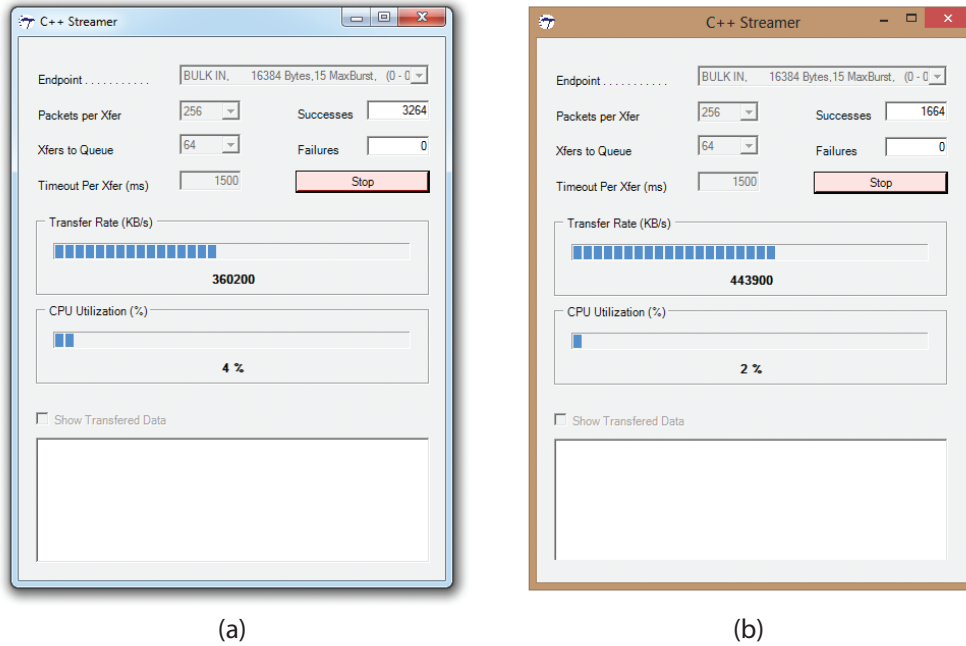


Figure 6.2: Screenshot of the Cypress C++ Streamer software that illustrates the real-time throughput by selecting 256 *Packets per Xfer* and 64 *Xfers to Queue*. (a) software run on Windows 7 64-bit featuring ASMedia Host Controller. (b) software run on Windows 8 64-bit featuring Intel USB 3.0 eXtensible Host Controller.

USB 3.0 Host Controller	Throughput	PC Information	OS
(a) ASMedia Host Controller	360200 kB/s	Intel(R) Xeon(R) CPU E21230 16 GB RAM	Win 7 64-bit
(b) Intel USB 3.0 eXtensible Host Controller	443900 kB/s	Intel(R) Core(TM) CPU i7-3537U 4 GB RAM	Win 8 64-bit

Table 6.1: PC components employed to test the transfer speed. Intel's host controller outperform Asus one by about 20%.

shown above. As preliminary conjecture, the best obtainable result would be limited to 400 MB/s: the maximum interface clock frequency is set to 100 MHz that, together with the 32-bit parallel data path, can move at most 3.2 Gbit per second.

6.2 10GBASE-X Development Status

The VHDL synthesis of the Ethernet 10GBASE-X is the most critical one in terms of design effort. The preliminary test I am currently working on is aimed to implement the 1000BASE-X core as done during the feasibility assessments. By doing so, I will be sure that the hardware mounted on the Data Management Board is working properly. The Kintex-7 requires different clock distribution architecture (figure 6.3) with respect to one implemented in the Spartan-6, hence few modifications need to be done on the

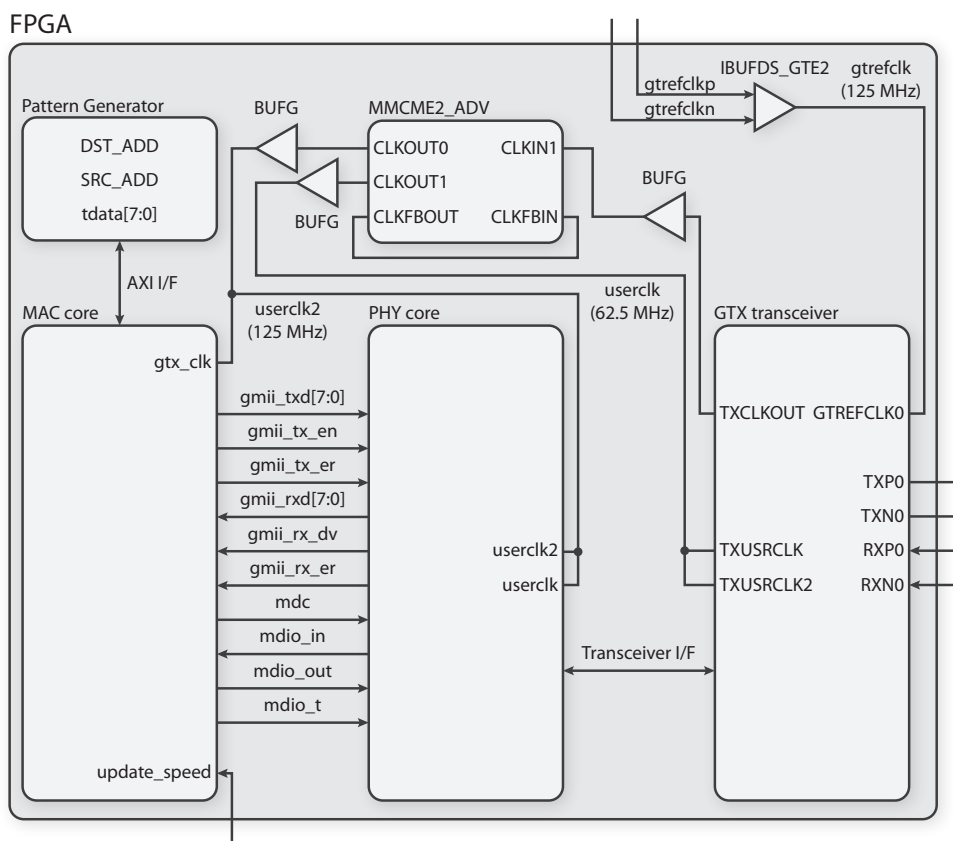


Figure 6.3: Implementation of the 1000BASE-X core: an external 125 MHz clock is fed to the GTX transceiver block that forward out a 62.5 MHz clock. A *MMCME2_ADV* primitive is employed to output two high quality global clocks for the remaining logic.

previous wrapper of the two cores along with proper timing constraints. Concerning the 10GBASE-X core, it is necessary to employ two new IP cores. In particular, they are the Xilinx *LogiCORE IP 10-Gigabit Ethernet PCS/PMA v4.1* [74] core and the LGPL licensed *Ethernet 10GE MAC* core [96]. With respect to the 1000BASE-X, the MAC (or client) side of the PHY core has now a full-duplex 64-bit parallel data path plus 8 control signals implementing an XGMII interface, all synchronous to a 156.25 MHz clock. In addition, an auxiliary MDIO interface could also be instantiated for management of the 10-Gigabit Ethernet PCS/PMA core. The actual implementation of this communication protocol is still under investigation. At last, as ultimate release, I would like to merge into one firmware the code blocks that separately concerns the USB 3.0 and Ethernet communication channels.

Conclusions

Nowadays, the TCSPC technique is widely employed among photoluminescence applications. While the demand of higher performance instruments increases continuously, modern technology grants for either high number of parallel channels or good timing characteristics. In this scenario, the work I have done during this Master's thesis is aimed to make valuable contributions in order to overcome this trade-off.

The TCSPC instrument currently under development in our research group is intended to feature 1024 SPAD detectors, 64 parallel time-to-amplitude conversion channels, and performance comparable with the best equipments both cited in literature and commercially available. With such high number of pixels, the resulting demand of transfer capabilities grows accordingly. In particular, the Data Management Board I have engineered during this thesis (figure 6.4) is supposed to downstream data from two twin 32-channel TCSPC boards and deliver it toward a PC over SuperSpeed USB 3.0 and Ethernet 10GBASE-X links. Compared to the Hi-Speed USB 2.0 employed in our previous systems, we gained an enhanced throughput performance up to 13 times higher with the USB 3.0 and almost 34 times higher with the Ethernet port. This accomplishment is remarkably meaningful, indeed only with these transfer rates, the 1024-channel TCSPC instrument would be able to properly operate both in histogram mode and in time-tag mode. Moreover, by exploiting two independent parallel transmission channels, it is possible to manage data download and control functions simultaneously. In particular, the assessment tests I made on the FX3 device during this work is proving more useful than ever, indeed most of the systems under development in our research laboratories are actually upgrading to the USB 3.0 interface. Unfortunately, I was not able to include in this thesis any experimental results concerning the 10GBASE-X communication channel because it turned out to require more work than expected. Nevertheless, all the time-consuming tests accomplished throughout this year were unavoidable in order to get to this very point. The presented board, despite of being target designed for the TCSPC instrument, could be easily adapted to different applications that require high performance transfer capabilities.

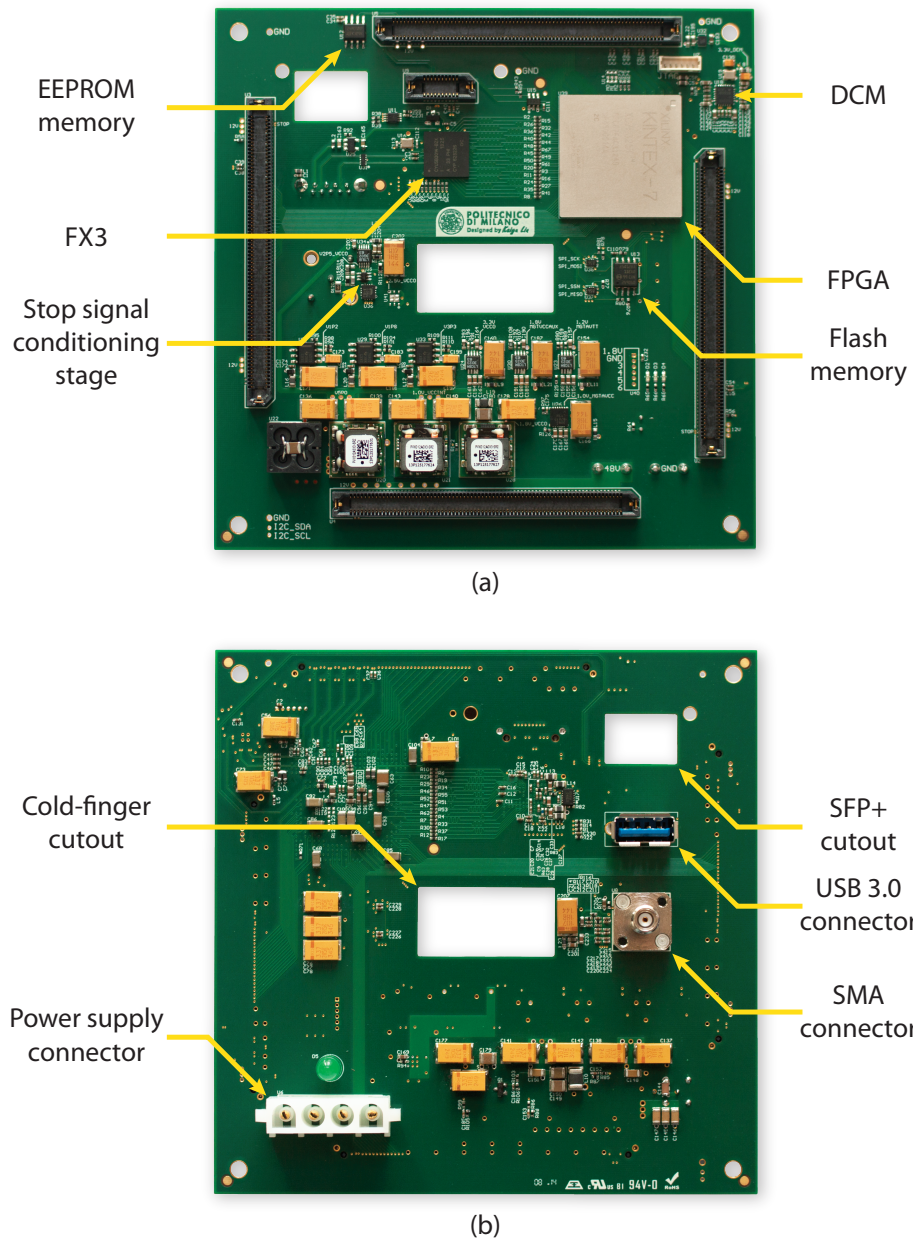


Figure 6.4: The Data Management Board and its main hardware component. (a) TOP layer. (b) BOTTOM layer.

Bibliography

- [1] B. Wolfgang. *The bh TCSPC Handbook*. Becker & Hickl GmbH, 2nd Edition, 2006. v, 9, 10
- [2] F. Villa, R. Lussana, D. Tamborini, D. Bronzi, B. Markovic, A. Tosi, F. Zappa, and S. Tisa. Cmos single photon sensor with in-pixel tdc for time-of-flight applications. *Time-to-Digital Converters (NoMe TDC), 2013 IEEE Nordic-Mediterranean Workshop on*, pages 1–6, Oct 2013. v, 11
- [3] Siemon. *QSFP30-01 datasheet*. <http://files.siemon.com/int-download-catalogs-system-catalog/2013-siemon-full-catalog.pdf>. vi, 35
- [4] Mellanox. *MHQH29C-XTR datasheet*. http://www.mellanox.com/related-docs/user_manuals/ConnectX%20_VPI_UserManual.pdf. vi, 35
- [5] One Stop Systems. *OSS-PCIe-CBL-x4-1M datasheet*. <http://www.onestopsystems.com/documents/OSS-PCIe-CBL-x4.pdf>. vi, 36
- [6] One Stop Systems. *OSS-PCIe-HIB35-x4 datasheet*. http://www.onestopsystems.com/documents/OSS-PCIe-HIB35-x4_001.pdf. vi, 36
- [7] Apple. *Thunderbolt cable datasheet*. <http://store.apple.com/it/product/MD861ZM/A/cavo-thunderbolt-apple-2-m-bianco>. vi, 37
- [8] Asus. *ThunderboltEX2 datasheet*. https://www.asus.com/Motherboards/ThunderboltEX_II/#overview. vi, 37
- [9] Intel. *E10GSFPSR datasheet*. <http://www.intel.com/content/dam/doc/product-brief/ethernet-sfp-optics-brief.pdf>, 2011. vii, 43
- [10] Intel. *E10G42BTDA datasheet*. <http://www.intel.com/content/dam/doc/product-brief/ethernet-x520-server-adapters-brief.pdf>, 2011. vii, 43
- [11] Murata Manufacturing Co. *Block Type EMIFIL (LC Combined) Lead Type*. <http://search.murata.co.jp/Ceramy/image/img/PDF/ENG/L0117S0161BNX01.pdf>, 2012. viii, 63, 64

- [12] PicoQuant. *Diffuse Optical Tomography and Imaging*. <http://www.picoquant.com/applications/category/life-science/diffuse-optical-tomography-and-imaging#tabbed-nav=description>. 2, 3
- [13] W. Becker. *Advanced time-correlated single photon counting techniques, 1st Edition*. Springer, 2005. 3
- [14] S. Antonioli. *Development of High Performance Electronics for Time-Correlated Single-Photon Counting Systems*. PhD thesis, Politecnico di Milano, Italy, 2013. 4, 18, 22, 45
- [15] J. R. Lakowicz. *Principles of fluorescence spectroscopy, 3rd Edition*. Springer, 2006. 6
- [16] A. Jablonski. Efficiency of anti-stokes fluorescence in dyes. *Nature Publishing Group*, Jun 1933. 9
- [17] US National Library of Medicine Medical Subject Headings. http://www.nlm.nih.gov/cgi/mesh/2011/MB_cgi?mode=&term=Optical+Tomography. 9
- [18] M. Wahl. *Time-Correlated Single Photon Counting*. http://www.picoquant.com/images/uploads/page/files/7253/technote_tcspc.pdf. 9
- [19] E. Lapointe, J. Pichette, and Y. Berube-Lauziere. A multi-view time-domain non-contact diffuse optical tomography scanner with dual wavelength detection for intrinsic and fluorescence small animal imaging. *Review of Scientific Instruments*, 83(6):063703–063703–14, Jun 2012. 10
- [20] M. Crotti. *Picosecond Resolution Integrated Electronics for Single Photon Detector Arrays*. PhD thesis, Politecnico di Milano, Italy, 2012. 10
- [21] B. Herman. *Fluorescence microscopy, 2nd Edition*. Springer, 1988. 11
- [22] K. Káñnig. Multiphoton microscopy in life sciences. *Journal of Microscopy*, 200(2):83–104, Nov 2000. 11
- [23] J. Pawley. *Handbook of biological confocal microscopy, 2nd Edition*. Plenum Press, 1995. 11
- [24] A. Periasamy. *Methods in Cellular Imaging*. Oxford University Press, 2001. 11
- [25] Becker & Hickl. *SPC-134 datasheet*. <http://www.becker-hickl.de/pdf/dbspc134b-2.pdf>. 12
- [26] Becker & Hickl. *SPC-154 datasheet*. <http://www.becker-hickl.de/pdf/dbspc154-3.pdf>. 12

- [27] PicoQuant. *HydraHarp 300 datasheet*. <http://www.picoquant.com/images/uploads/downloads/picoharp300.pdf>. 12
- [28] PicoQuant. *HydraHarp 300 datasheet*. <http://www.picoquant.com/images/uploads/downloads/hydraharp400.pdf>. 12
- [29] D. Resnati, I. Rech, A. Gallivanoni, and M. Ghioni. Monolithic time to amplitude converter for time correlated single photon counting. *Review of Scientific Instruments*, 80(8):23–24, Aug 2009. 13
- [30] P. KerÅdnen, K. MÅdÅdttÅd, and J. Kostamovaara. Wide-range time-to-digital converter with 1-ps single-shot precision. *Instrumentation and Measurement, IEEE Transactions on*, 60(9):3162–3172, Sept 2011. 13
- [31] J.P. Jansson, V. Koskinen, A. Mantyniemi, and J. Kostamovaara. A multichannel high-precision cmos time-to-digital converter for laser-scanner-based perception systems. *Instrumentation and Measurement, IEEE Transactions on*, 61(9):2581–2590, Sept 2012. 13, 15
- [32] B. Markovic, D Tamborini, F. Villa, S. Tisa, A Tosi, and F. Zappa. 10 ps resolution, 160 ns full scale range and less than 1.5% differential non-linearity time-to-digital converter module for high performance timing measurements. *Review of Scientific Instruments*, 83(7):074703–074703–10, Jul 2012. 13
- [33] Princeton Lightwave. *128ÅÜ32 GmAPD Camera datasheet*. http://www.princetonlightwave.com/images/pli_content/PLI%20128x32%20GmAPD%20Camera%20-%20ProdSum%20Rev%201.2.pdf. 12, 13
- [34] D. Stoppa, F. Borghetti, J. Richardson, R. Walker, L. Grant, R.K. Henderson, M. Gersbach, and E. Charbon. A 32ÅÜ32-pixel array with in-pixel photon counting and arrival time measurement in the analog domain. *ESSCIRC, 2009. ESSCIRC '09. Proceedings of*, pages 204–207, Sept 2009. 13
- [35] C. Veerappan, J. Richardson, R. Walker, Day-Uey Li, M.W. Fishburn, Y. Maruyama, D. Stoppa, F. Borghetti, M. Gersbach, R.K. Henderson, and E. Charbon. A 160ÅÜ128 single-photon image sensor with on-pixel 55ps 10b time-to-digital converter. *Solid-State Circuits Conference Digest of Technical Papers (ISSCC), 2011 IEEE International*, pages 312–314, Feb 2011. 13
- [36] C. Niclass, C. Favi, T. Kluter, M. Gersbach, and E. Charbon. A 128ÅÜ128 single-photon image sensor with column-level 10-bit time-to-digital converter array. *Solid-State Circuits, IEEE Journal of*, 43(12):2977–2989, Dec 2008. 13
- [37] F. Villa, D. Bronzi, S. Bellisai, G. Boso, A. Bahgat Shehata, C. Scarcella, A. Tosi, F. Zappa, S. Tisa, D. Durini, S. Weyers, and W. Brockherde. Spad imagers for

- remote sensing at the single-photon level. *Proc. SPIE*, 8542:85420G–85420G–6, 2012. 13
- [38] S. Cova, M. Ghioni, A. Lacaita, C. Samori, and F. Zappa. Avalanche photodiodes and quenching circuits for single-photon detection. *Applied Optics*, 35(12):1956–1976, Apr 1996. 14
- [39] C. Cammi, A. Gulinatti, I. Rech, F. Panzeri, and M. Ghioni. Spad array module for multi-dimensional photon timing applications. *Journal of Modern Optics*, 59(2):131–139, Jan 2012. 14
- [40] C. Cammi, F. Panzeri, A. Gulinatti, I. Rech, and M. Ghioni. Custom single-photon avalanche diode with integrated front-end for parallel photon timing applications. *Review of Scientific Instruments*, 83(3):033104–033104–8, Mar 2012. 14
- [41] M. Crotti, I. Rech, and M. Ghioni. Four channel, 40 ps resolution, fully integrated time-to-amplitude converter for time-resolved photon counting. *Solid-State Circuits, IEEE Journal of*, 47(3):699–708, Mar 2012. 15
- [42] I. Rech, A. Gulinatti, M. Crotti, C. Cammi, P. Maccagnani, and M. Ghioni. Towards picosecond array detector for single-photon time-resolved multispot parallel analysis. *Journal of Modern Optics*, 58(3–4), Jan 2011. 17
- [43] L. Schuchman. Dither signals and their effect on quantization noise. *Communication Technology, IEEE Transactions on*, 12(4):162–165, Dec 1964. 18
- [44] C. Cottini, E. Gatti, and V. Svelto. A new method for analog to digital conversion. *Nuclear Instruments and methods*, 24:241–242, 1963. 18
- [45] Cypress. *CY7C68013A datasheet*. <http://www.cypress.com/?rID=38801>. 18, 45
- [46] FTDI. *FT2232H datasheet*. http://www.ftdichip.com/Support/Documents/DataSheets/ICs/DS_FT2232H.pdf. 19, 45
- [47] Microchip. *USB2517 datasheet*. <http://ww1.microchip.com/downloads/en/DeviceDoc/2517.pdf>. 21
- [48] S. Antonioli, M. Crotti, A. Cuccato, I. Rech, and M. Ghioni. Time-correlated single-photon counting system based on a monolithic time-to-amplitude converter. *Journal of Modern Optics*, 59(17):1512–1524, Jul 2012. 22
- [49] S. Antonioli, L. Miari, A. Cuccato, M. Crotti, I. Rech, and M. Ghioni. 8-channel acquisition system for time-correlated single-photon counting. *Review of Scientific Instruments*, 84(6), Jun 2013. 22

- [50] S. Antonioli, A. Cuccato, L. Miari, I. Labanca, I. Rech, and M. Ghioni. Ultra-compact 32-channel system for time-correlated single-photon counting measurements. *Proc. SPIE*, 8773:87730D–87730D–11, 2013. 22
- [51] L. Miari. *Progetto e Realizzazione di un Sistema di Acquisizione Compatto a Otto Canali per Misure TCSPC*. MSc thesis, Politecnico di Milano, Italy, 2011. 23, 53
- [52] A. Cuccato. *Development of Electronic Systems for Single-Photon Avalanche Diode Arrays*. PhD thesis, Politecnico di Milano, Italy, 2013. 25, 26, 29
- [53] Xilinx. *XCVU160 datasheet*. http://www.xilinx.com/support/documentation/data_sheets/ds890-ultrascale-overview.pdf. 26
- [54] Cypress. *CY7C1625KV18-333BZX* datasheet. <http://www.cypress.com/?docID=40501>. 27
- [55] Micron. *MT47H256M8EB* datasheet. <https://www.micron.com/parts/dram/ddr2-sdram/mt47h256m8eb-25e-it>. 27
- [56] B. W. Marsden. *Communication network protocols*. Chartwell-Bratt, 2nd Edition, 1986. 31
- [57] D. Comer. *Internetworking with TCP/IP: Principles, protocols, and architecture*. Prentice Hall, 2000. 32, 40
- [58] M. Agrawal. *Business Data Communications*. Wiley, 2011. 32
- [59] R. Blahut. *Algebraic Codes for Data Transmission*. Cambridge University Press, 2004. 33
- [60] Samsung. *SSD 840 PRO 512 GB SATA III*. http://www.samsung.com/it/business-images/resource/case-study/2014/02/Samsung_SSD_840_PRO_Series_Data_Sheet_ITA-0.PDF, 2014. 38
- [61] WG802.1 Higher Layer LAN Protocols Working Group. *802.1Q-2011 - IEEE Standard for Local and metropolitan area networks—Media Access Control (MAC) Bridges and Virtual Bridged Local Area Networks*. <http://standards.ieee.org/findstds/standard/802.1Q-2011.html>, 2011. 40
- [62] Xilinx. *Spartan-6 FPGA Data Sheet: DC and Switching Characteristics*. http://www.xilinx.com/support/documentation/data_sheets/ds162.pdf, 2011. 40
- [63] Xilinx. *LogiCORE IP Tri-Mode Ethernet MAC v8.1*. http://www.xilinx.com/support/documentation/ip_documentation/tri_mode_ethernet_mac/v8_1/pg051-tri-mode-eth-mac.pdf, 2013. 41, 76

- [64] Xilinx. *LogiCORE IP Ethernet 1000BASE-X PCS/PMA or SGMII v14.1*. http://www.xilinx.com/support/documentation/ip_documentation/gig_ethernet_pcs_pma/v14_1/pg047-gig-eth-pcs-pma.pdf, 2013. 41, 76
- [65] IEEE. *IEEE 802.3-2012*. <http://standards.ieee.org/about/get/802/802.3.html>, 2012. 41
- [66] Hewlett-Packard Company, Intel Corporation, Microsoft Corporation, NEC Corporation, ST_NXP Corporation, and Texas Instruments. *Universal Serial Bus 3.0 Specification*. http://www.gaw.ru/pdf/interface/usb/USB%203%200_english.pdf, 2008. 45
- [67] Cypress. *EZ-USB FX3 Development Kit*. <http://www.cypress.com/?docID=41926>, 2013. 47
- [68] Cypress. *EZ-USB FX3*. <http://www.cypress.com/?docID=44322>, 2013. 47, 68
- [69] Cypress. *AN65974 - Designing with the EZ-USB FX3 Slave FIFO Interface*. <http://www.cypress.com/?docID=47020>, 2013. 47, 57, 73
- [70] Samtec. *ERM8-EM series 0,80 mm Edge Rate Rugged High Speed Terminal Strip, Edge Mount*. <https://www.samtec.com/technical-specifications/Default.aspx?SeriesMaster=ERM8-EM>. 51
- [71] TE Connectivity. *PC Board Mount Vertical Jack*. <http://www.te.com/commerce/DocumentDelivery/DDEController?Action=srchrtv&DocNm=6274096&DocType=Customer+Drawing&DocLang=English>. 52
- [72] Analog Devices. *Rail-to-Rail, Very Fast, 2.5 V to 5.5 V, Single-Supply CML Comparators*. http://www.analog.com/static/imported-files/data_sheets/ADCMP606_607.pdf. 53, 65
- [73] Micrel. *4.25Gbps Precision, 1:2 CML Fanout Buffer with Internal Termination and Fail Safe Input*. http://www.micrel.com/_PDF/HBW/sy58606u.pdf. 54, 65
- [74] Xilinx. *LogiCORE IP 10-Gigabit Ethernet PCS/PMA v4.1*. http://www.xilinx.com/support/documentation/ip_documentation/ten_gig_eth_pcs_pma/v4_1/pg068-ten-gig-eth-pcs-pma.pdf, 2013. 54, 84
- [75] Xilinx. *Device Package User Guide*. http://www.xilinx.com/support/documentation/user_guides/ug112.pdf, 2012. 55
- [76] Xilinx. *LogiCORE IP Aurora 8B/10B v10.1*. http://www.xilinx.com/support/documentation/ip_documentation/aurora_8b10b/v10_1/pg046-aurora-8b10b.pdf, 2013. 55

- [77] Micron. *Micron Serial NOR Flash Memory*. http://www.micron.com/~/media/Documents/Products/Data%20Sheet/NOR%20Flash/Serial%20NOR/N25Q/n25q_128mb_1_8v_65nm.pdf, 2012. 57
- [78] Silicon Labs. *I²C-PROGRAMMABLE ANY-FREQUENCY, ANY-OUTPUT QUAD CLOCK GENERATOR*. <http://www.silabs.com/Support%20Documents/TechnicalDocs/Si5338.pdf>, 2013. 57
- [79] ON Semiconductor. *USB Positive Overvoltage Protection Controller with Internal PMOS FET and Status FLAG*. http://www.onsemi.com/pub_link/Collateral/NCP360-D.PDF, 2012. 58
- [80] Littelfuse. *TVS Diode Arrays*. http://www.littelfuse.com/data/en/data_sheets/littelfuse_tvs_diode_array_spa_sp3010.pdf, 2012. 58
- [81] Xilinx. *7 Series FPGAs Configuration*. http://www.xilinx.com/support/documentation/user_guides/ug470_7Series_Config.pdf, 2013. 59
- [82] STMicroelectronics. *Low voltage 1.0 Ω max dual SP3T switch with break-before-make feature*. <http://www.st.com/web/en/resource/technical/document/datasheet/CD00081251.pdf>, 2010. 59
- [83] Texas Instruments. *DUAL BIDIRECTIONAL I²C BUS AND SMBus VOLTAGE-LEVEL TRANSLATOR*. <http://www.ti.com/lit/ds/symlink/pca9306.pdf>, 2004. 60
- [84] Samtec. *ERF8-010-05.0-L-DV-L-K-TR*. <http://www.samtec.com/documents/webfiles/cpdf/ERF8-XXX-XX.X-X-DV-XXXX-XX-MKT.pdf>, 2014. 61
- [85] Samtec. *ERM8-010-01-L-D-RA-L-K-TR*. <http://www.samtec.com/documents/webfiles/cpdf/ERM8-XXX-XX-X-D-RA-XX-FOOTPRINT.pdf>, 2014. 61
- [86] Mean Well. *280W AC-DC Single Output Desktop*. <http://www.meanwell.com/search/gs280/gs280-spec.pdf>, 2013. 63
- [87] General Electric. *12A Analog PicoDLynx: Non-Isolated DC-DC Power Modules*. <http://apps.geindustrial.com/publibrary/checkout/PVX012A0X?TNR=Data%20Sheets|PVX012A0X|generic>, 2013. 65
- [88] Linear Technology. *Adjustable 1.1 A Single Resistor Low Dropout Regulator*. <http://cds.linear.com/docs/en/datasheet/3080fc.pdf>, 2007. 65
- [89] Linear Technology. *Adjustable 3 A Single Resistor Low Dropout Regulator*. <http://cds.linear.com/docs/en/datasheet/3083fa.pdf>, 2011. 65

- [90] Texas Instruments. *FAST TRANSIENT RESPONSE, 1 A LOW-DROPOUT VOLTAGE REGULATORS*. <http://www.ti.com/lit/ds/symlink/tps76801.pdf>, 2006. 65
- [91] S. Weir. *PDN Application of Ferrite Beads*. http://www.ipblox.com/pubs/DesignCon_2011/11-TA3Paper_Weir_color.pdf, 2011. 66, 70
- [92] H. Johnson. *High Speed Digital Design: A Handbook of Black Magic*. Pearson Education, 1993. 68
- [93] Altera. *Power Distribution Network Design Tool*. <http://www.altera.com/technology/signal/power-distribution-network/sgl-pdn.html>. 69
- [94] Würth Elektronik. *Würth Elektronik 742792035 ferrite bead*. <http://katalog.we-online.de/pbs/datasheet/742792035.pdf>, 2011. 70
- [95] Samtec. *0,80 mm Edge Rate Rugged High Speed Socket Strip*. <https://www.samtec.com/technical-specifications/Default.aspx?SeriesMaster=ERF8>. 71
- [96] A. Tanguay and M. Pratik. *Ethernet 10GE MAC*. http://opencores.org/project,xge_mac, 2013. 84