Hybrid Spatial and Temporal Computing Histogramer in Soft Processor Core of a FPGA Device

Enrico Ronconi, *Graduate Student Member, IEEE*, Fabio Garzetti, *Member, IEEE*, Nicola Lusardi, *Graduate Student Member, IEEE*, Andrea Costa, *Graduate Student Member, IEEE*, and Angelo Geraci, *Senior Member, IEEE*

***Abstract*— Multi-channel data management is crucial in a world where big data processing is extensively used in research and business. Histogramming is a common technique employed to detect, analyze, and store enormous volumes of data in real-time, making it useful for industrial applications in fields such as biology, chemistry, medical imaging, and spectroscopy. Due for them programming simplicity and low-cost large amount of memory, general-purpose temporal computing processors are commonly used, but they lack the ability to perform parallel computation at high-performance. Field-Programmable Gate Array (FPGA) is a powerful parallel computing solution proposed by both the scientific and industrial worlds, but it is equipped with little memory for these applications. Thus, a hybrid spatial/temporal computing histogram generator has been proposed, which uses a low-area multi-channel histogramming engine in programmable logic which is expanded thanks to an external Double Data Rate Synchronous Dynamic Random Access Memory (DDR) driven by a MicroBlaze Soft Processor Core. The proposed system has been validated on a Xilinx 28-nm 7-Series Artix-7 XC7A100T FPGA hosted on a Nexys4 Evaluation Board. Thanks to this hybrid solution, up to 128 channels can handle in a low-end FPGA occupies 207 LUTs and 325 flip-flops per channel plus a total 630 kb of total BRAM shared between all channels; a power consumption of 10.1 mW per channel is measured.**

***Index Terms*— Histograms, field-programmable gate array (FPGA), MicroBlaze, real-time systems, soft processor core.**

I. Introduction

HISTOGRAMS are frequently the most economical way to compress data for handling it in large quantities [1], [2], [3]. It is common knowledge that histograms, for instance, keep track of the frequency of events by storing the data in a user-defined memory that, roughly speaking, can be dimensioned regardless of the number of events. Today, there is an increasing requirement to manage massive data, which might even be spread over very broad value dynamics [4]. This is true for many applications that may offer significant benefits from being able to obtain a more condensed version of the observed occurrence. In addition, many applications, including weather forecasting [5] and traffic prediction [6], to mention a few, have moved away from deterministic models in favor of stochastic ones [7], [8]. The required information can be obtained and applied in a number of methods, such as by computing statistical moments, such as mean-value, variance, and higher moments [9]. The normalized version of a histogram is actually the closest representation of the random process that created the sequence from a stochastic perspective [10]. Histograms play a significant role in the analysis and processing of data in this scenario due to their ability to simplify the extraction of statistics from the underlying data stream.

The simplicity of histograms also makes them useful in a wide variety of measurement disciplines, from industry to research. Even when considering some end-user applications, such as image and video processing for computer vision [11] and automotive [12] purposes, these applications heavily rely on histogramming methods. The large amount of data and, more importantly, the high data rates are, therefore, the common denominators of all histogram applications. It is crucial to be able to handle these rates with real-time processing in order to prevent the creation of bottlenecks and to ensure the necessary rising performance [13]. Consider computer vision, which uses the gray-scale image's histogram and variance for in-the-moment image recognition [14], [15]. Histograms are frequently used in time-based experiments in the context of scientific study; for instance, they form the foundation of nuclear physics investigations. In this scenario the histogram tool performs energy spectra, measures gamma-photons arrival time distribution in Time-of-Flight Positron Emission Tomography (TOF-PET) [16], Time-Resolved Spectroscopy [17], Time-Correlated Single Photon Counting (TCSPC) [18], Time-of-Flight (TOF) Rangefinder such as Laser Rangefinders [19] (LR), Time-of-Flight Mass Spectrometry [1] (TOF-MS), and so on.

Every quantized data in the digital world is represented by an N-bit wide word, which can be easily translated into the analog world by multiplying it by the so-called Least Significant Bit (*LSB*). In these terminology, the analog values that can be quantized have a Full-Scale Range (FSR) that

Manuscript received 19 October 2023; revised 7 February 2024; accepted 16 May 2024. This article was recommended by Associate Editor C. H. Chang. *(Corresponding author: Nicola Lusardi.)*

The authors are with the Department of Electronics, Information and Bioengineering (DEIB), Politecnico di Milano, 20133 Milan, Italy (e-mail: enrico.ronconi@polimi.it; nicola.lusardi@polimi.it).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSI.2024.3404034.

Digital Object Identifier 10.1109/TCSI.2024.3404034

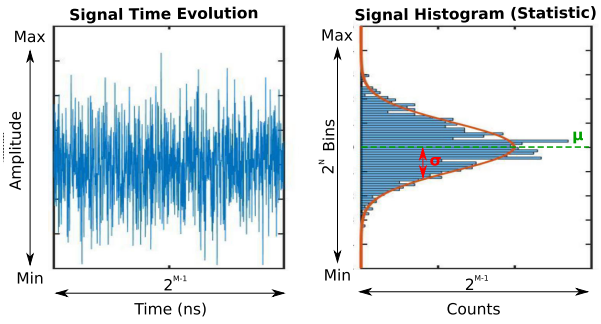


Fig. 1. The left-hand side displays the temporal evolution of the signal of interest, while the right-hand side shows its corresponding histogram (a.k.a., statistics) where the mean value (μ) and standard deviation (σ) are highlighted.

goes up to $2^N \cdot LSB$. If we were to create a histogram in this case, the FSR of the data series should be split into LSB wide classes, sometimes known as bins. On the other hand, a histogram with $\frac{FSR}{LSB} = 2^N$ bins can be successfully stored in a digital memory made up of 2^N cells. Each cell in the histogram's digital memory is identified by the data width M, which specifies the maximum number of counts ($Count_{max} = 2^M - 1$) that a single bin can hold before being saturated. The M parameter establishes the histogram's correctness in that the estimation of the statistical process p is more accurate the more samples there are in the histogram. Figure 1 shows a representation of the histogramming of a generic signal.

The most commonly employed approach for creating a histogram involves the utilization of temporal computing processors, including Central Processing Units (CPUs) or Graphics Processing Units (GPUs) [20], where the histograms are stored on high-density external Random Access Memory (RAM) modules, such as the Double Data Rate Synchronous Dynamic Random Access Memory (DDR SDRAM or simply DDR). This method offers the advantage of algorithmic simplicity, although it is constrained by the limited number of threads, tens, (i.e., histograms) that can operate simultaneously. Modern applications, however, take advantage of multi-channel solutions, where enormous (ranging from tens to hundreds) processing cores, in this case histogramming engines, operate independently. Spatial computing architectures based on Field Programmable Gate Array (FPGA) are essential in these settings [21]. However, this strategy incurs a memory cost, as the use of internal Block RAM (BRAM) is constrained by its physical size, which is limited by the technology of Static RAM (SRAM) [22]. Employing external Dynamic RAM (DRAM) with an FPGA approach can lead to an explosion in system complexity, due to the difficulty of managing access and refresh operations of the DRAM. In this paper, we propose a multi-channel, low-area occupancy hybrid spatial/temporal computing solution, which uses a low-area multi-channel histogramming engine in programmable logic which is expanded thanks to an external DDR driven by a MicroBlaze Soft Processor Core [23]. By doing so, the MicroBlaze will handle the read/write operations of the DDR, reducing the system complexity as compared to when the FPGA itself manages direct DDR access.

The innovation of the proposed structure lies in its ability to implement histograms with 2^8 bins and a depth M of 32 (equivalent to 4294967296 counts per bin), in a multichannel mode of up to 128 (i.e., 128×2^8 total bins), while maintaining an extremely low usage of BRAM (totaling 630 kb). This implies an allocation of less than 5 kb of BRAM for each channel, as the storage effort is shifted to more compact and scalable external DDR solutions, leveraging MicroBlaze to simplify interfacing with DDR and the histogramming process. Although it combines basic ideas, in its simplicity, it efficiently implements a fast multichannel histogramming solution at 95 Msps, considering a maximum clock of 150 MHz (thus suitable for real-world applications), and simultaneously maintains compactness with less than 204 LUTs, 318 FFs, and 5 kb for BRAM per channel. Furthermore, a total power consumption of 1.29 W is observed, indicating a mere 10.1 mW per channel.

The paper is structured as follows: Section II introduces the trend toward parallel computing solutions and the state-of-the-art in the area of high-performance histogram computation; Section III discusses the specifics and features of the suggested architecture. The experimental validation in TOF-PET setup is described in Section IV and them are performed on a Xilinx 28-nm Artix-7 100T FPGA (i.e., XC7A100TCSG324-1) [24] and a 16 MiB Micron RAM (M45W8MW16) hosted in a Digilent Nexys4 Evaluation Board (EVB) [25].

II. State-of-the-Art and Trend of Implementation Strategy

The shift from general-purpose processors like CPUs [26] and GPUs [27] to FPGA architectures for histogram creation is prompted by the demand for efficient parallel computing in multi-channel applications across various research fields [28], [30]. Traditional temporal computing becomes inefficient, especially in scenarios like metrology with increasing parallel input channels. Researchers are now focusing on lower-level approaches, leveraging specialized FPGA architectures for tailored processing solutions, exploiting full-speed data source interfaces [31], [32].

In this regard, the FPGA domain [33] offers the opportunity to keep the system simpler and the processing to be done closer to the data generation, in real-time. In fact, the possibility of directly building a histogramming IP-Core into the hardware and replicating it to exploit maximum parallelism, as needed by the application, offers the chance to keep the system simple [21]. The device being used, specifically the quantity of resources available, has a general relationship to the parallelization limit.

Furthermore, it is essential to be able to provide at least basic processing in real-time, immediately following the acquisition chain, in an environment where there are likely some detectors producing signals that need to be captured and processed [34]. If this is the case, FPGAs enable the construction of "cores" for data processing alongside Data Acquisition (DAQ), creating a potent combination that, in many cases, ensures there are no bottlenecks caused by data transfer off chip, a stage that is necessary when leaving the processing to an external device like a GPU. Sometimes, it may also be

TABLE I
SYNOPTIC VIEW OF SOME APPLICATIONS WITH CORRESPONDENT DATA RATES ANDHOST PROCESSING UNITS

Reference	Application	Data rate	Processing Unit
[13]	TOF rangefinder	1.6 Gb/s	FPGA
[36]	TOF rangefinder	1.92 Gb/s	FPGA
[20]	TCSPC	10 Gb/s	GPU
[37]	TOF-PET	2.5 Gb/s	FPGA
[38]	Computer vision	800 Mb/s	FPGA
[39]	Image processing	2.24 Gb/s	FPGA
[14]	Image recognition	3.2 Gb/s	FPGA
[40]	Image processing	400 Mb/s	FPGA
[15]	Image processing	2.96 Gb/s	FPGA
[21]	TME	3.58 Gb/s	FPGA

necessary for the DAQ and data processing units to have a real-time, low-latency bidirectional communication in order for the system to be employed in a feedback configuration, which uses the processed data to change acquisition parameters [35]. This is simple to accomplish if the DAQ section, which is often built on FPGAs, is supported by a separate processing unit on the same chip.

Table I offers a concise comparison of various histogram computation methodologies in different fields. The top section focuses on time-domain experiments ([13], [20], [21], [36], [37]), while the bottom section addresses computer vision and image processing applications ([14], [15], [38], [39], [40]). FPGAs emerge as the dominant technology, tailored to specific application areas, emphasizing reprogrammability, multi-channel capabilities, low latency, high throughput, and adaptability. The availability of effective hardware primitives like BRAM and Digital Signal Processor (DSP) for FPGAs [41] has significantly improved real-time histogram computation, making FPGAs a key technology in this domain. Additionally, FPGAs are often employed as a preliminary stage for hardware design verification before final implementation as ASICs [42].

III. SPATIAL/TEMPORAL COMPUTING HYBRID SOLUTION

The main features of a histogram can be extracted by using FPGA-based solutions described in Section I. The first two that may be determined are the maximum number of bins 2^N, also known as the number of values on the abscissa, and the maximum number of counts $2^M - 1$, also known as the values on the ordinate, for each bin. As the reader will quickly realize, it is imperative in this context that a memory, such as BRAM in Xilinx FPGAs, be available and have a minimum storage capacity of $(M - bit/word) \times (2^N - word)$. Also, the chosen bin must have an appropriate increment mechanism, such as an adder or a DSP. Other fundamental figures of merit that are determined by the pipeline that the memory introduces and the increment mechanism include clock cycles of latency L, maximum rate R and $\overline{R}$ without and with losses (Msps), and system clock F_{CLK} (MHz). Last but not least, another factor to take into account is the entire area occupancy. A bigger M necessitates the use of wider increment mechanisms, which are characterized by slower propagation delays. The same idea applies to bigger N, which requires using larger memories and address methods, both of which have slower propagation delays. A pipeline technique is necessary in this case to speed up the system at the expense of a higher area occupancy. Reducing the maximum input rate in respect to the clock frequency [40], could be one way to minimize this trend. In contrast, the technique offered in [15] memorizes histograms using flip-flops rather than traditional BRAM, enabling high-frequency operation without the need for a pipeline and saving space. Similarly, CPU/GPU-based architectures have the same figures-of-merit, with the exception of area occupancy which is replaced by number of core/thread involved (C) and the number of cycles required to perform the accumulation (CY). Latency and maximum rate are strongly depended to the architecture; instead, in contrast to FPGA, big flexibility is intrinsically present concerning the storage capacity.

When it comes to low latency, high throughput, flexibility, and compatibility with multi-channel systems, FPGA solutions are definitely preferable to temporal-computing ones, although they have a strong limitation in terms of storage due to the low density of BRAM. Unlike CPU/GPU architectures designed to manage relatively simple high-density DRAM memories. For this reason, we have decided to present a hybrid architecture that combines the advantages of both. In this sense, we expand the BRAM of a classic histogram implemented in FPGA technology with an external DRAM. To lighten the hardware required to manage such memory, the complex control logic necessary for DDR read and write operations was handled using a MicroBlazeThis gave us the opportunity to integrate the MicroBlaze software programmability with the simple DDR memory interface provided by the Memory Interface Generator (MIG) IP-Core. In this way, the work is greatly eased by using the extensive software libraries that are already built into the MicroBlaze. In doing so, it is possible to use the advantages of both the firmware and software techniques, combining them to produce a flexible, scalable, high-performance solution for the already noted need of controlling massive multichannel systems. Paragraph III-A and Paragraph III-B present the details of the proposed hybrid architecture (considering the maximum clock frequencies allowed without timing errors) and detail about performance and area occupied, while Paragraph III-C will illustrate the trade-offs compared to the classical FPGA-based solution and future developments.

A. Hybrid Architecture Overview

Figure 2 depicts the proposed architecture's conceptual plan organize using IP-Cores. All the interconnection between IPs are performed using the Advanced eXtensible Interface 4 (AXI4) standard [43]. The Memory Manage Engine (MME) [44] is the IP-Core that manage the link between FPGA and Personal Computer (PC) by means of an 2 Mbps RS-232 protocol and it will not be described in this paper [45].

The module named "Histogram Wrapper" is the heart of the system and is responsible for implementing the multi-channel, up to $H = 128$ (where H is the number of channels), low-area, especially BRAM, histogramming mechanism inside the FPGA at 256 bins (i.e., $N = 8$). Each channel consists of an

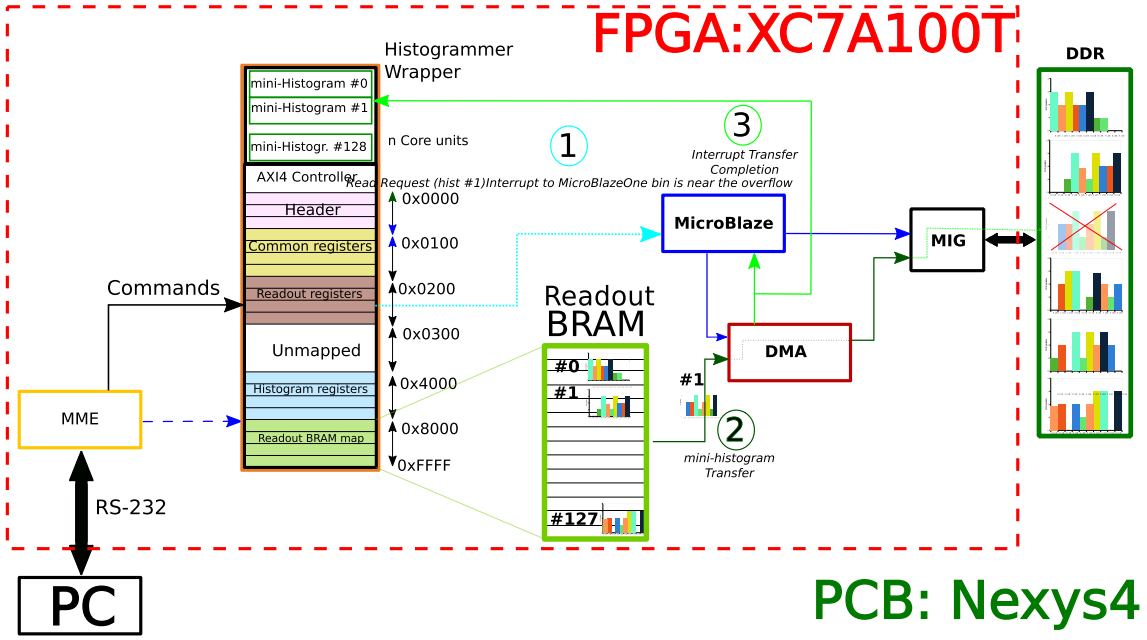


Fig. 2. Proposed hybrid architecture.

input line, control logic, an accumulator, a BRAM memory area, and configuration registers. The histogramming mechanism is therefore a synchronous logic module identical to the one described in [21] and is driven by a 150 MHz FPGA clock (F_{CLK}^{FPGA}). It is a pipeline that allows working at the FPGA's clock frequency, and at each occurrence of the input signal, a "+1" is added to the corresponding BRAM memory cell address through the accumulator. In this way, each channel of the histogram is able to support an accumulation rate equal to the FPGA's clock frequency (i.e., $R = F_{CLK}^{FPGA} = 150\,Msps$) with a latency L of 2. Unlike what is described in [21], in order to achieve up to 128 channels in a low-end FPGA, the BRAM does not need to store the entire histogram but only a limited number of occurrences called "mini-histograms". In fact, the entire histogram will finally be allocated in DDR. Therefore, the BRAM performs the task of a small hardware cache. Thus, each individual channel can accumulate a maximum number of counts equal to $2^{16} - 1$ (i.e., $M = 16$). In this scenario, each "mini-histogram" occupies only 4096 bits of memory (i.e., $2^N \times M = 2^8 \times 16 = 4096$).

For simplicity, all "mini-histograms" share the same address space of 2^{16} addresses with all BRAMs allocated between addresses 0×8000 and 0xFFFF (i.e., up to 128 "mini-histograms" with 256 bins, $H \cdot 2^N$). Instead, addresses between 0×0000 and $0 \times 7FFF$ are occupied by configuration registers necessary to properly set histogram parameters such as acceptable maximum and minimum values, refresh rate, etc. "Mini-histograms" stored in the BRAM and when one bin is near to the overflow it is transferred to the so-called $H \cdot 4096$ bit-wide "Readout BRAM" (with dimension equal to $H \cdot 2^N \times M$ bits) clocked at F_{CLK}^{FPGA} of 150 MHz, another BRAM shared between the "Histogram Wrapper" controller and the MicroBlaze, where the latter interfaces via Direct Memory Access (DMA) [46], [47], [48]. Each "mini-histogram" is accompanied by an ID (from 0 to 127) that uniquely identifies the channel of the "Histogram Wrapper" that produced it.

The MicroBlaze, clocked at F_{CLK}^{MB} of 130 MHz, after a proper interrupt signal reads the Readout BRAM through DMA, appropriately identifies "mini-histograms", and controls the MIG to store them in DDR. Therefore, if the histogram has an ID that is not present in DDR, a memory area is allocated in DDR where it is saved. Conversely, if the ID is already present, the content of the newly arrived "mini-histogram" is added to the existing histogram in DDR. Thanks to this mechanism, the MicroBlaze extend the maximum number of counts per bin (i.e., $2^M - 1$) from 16 to 32 without using BRAM. This costs for the MicroBlaze, that is programmed in bare-metal, 10 clock cycles (i.e., $CY = 10$) for one bin at 130 MHz. In this way, in the worst case of operation (i.e., all histograms collect events at the same rate), the total bandwidth of 130 MHz (i.e., F_{CLK}^{MB}) is shared among all bins of all histograms (i.e.,$H \cdot 2^N = H \cdot 256$), limiting the maximum measurement rate without losses to $50.8/H\,ksps$ (i.e., $R = F_{CLK}^{MB}/(CY \cdot H \cdot 2^N) = 130/(10 \cdot H \cdot 256) = 50.8/H\,ksps$);

$$R = F_{CLK}^{MB} \cdot \frac{C}{CY} \frac{1}{2^N \cdot H} \tag{1}$$

However, many applications allow for burst processing, especially in cases where the data to be analyzed is already stored and the analysis is triggered by the user (e.g., video analysis). Alternatively, even in continuous acquisition systems, the loss of samples may be inconsequential for analytical purposes and simply result in a proportional increase in the acquisition time, as observed in TCSPC systems and generally in TOF experiments based on statistical concepts. In such cases, it is meaningful to consider the average histogramming rate based on the maximum acquisition rate with losses ($\overline{R}$) rather than the maximum rate without losses (R). In the proposed FPGA-side (i.e., "mini-histogram") system,

the worst-case scenario for histogram saturation occurs after acquiring $2^M - 1$ (i.e., $2^{16} - 1 = 65535$) samples in the same bin simultaneously across all H histograms in parallel. This, as reported in (2) (where M is the width of the BRAM), corresponds to an FPGA accumulation time of 437 μs independent from H (i.e., $T_{FPGA} = (2^M - 1)/F_{CLK}^{FPGA} = 65535/150\,MHz = 437\,\mu s$).

$$T_{FPGA} = \frac{2^M - 1}{F_{CLK}^{FPGA}} \tag{2}$$

After this accumulation time, the acquired data in the BRAM needs to be transferred to the MicroBlaze processor via DMA, extended to 32 bits, and integrated into DDR. From an acquisition time perspective, as previously discussed, this transfer takes 10 clock cycles (CY) at 130 MHz (F_{CLK}^{MB}) per bin per core (C), which amounts to 2.52 ms, this time, as reported in (3), it depends on N and H (i.e., $T_{MB} = (CY/F_{CLK}^{MB}/C) \cdot 2^N \cdot H = (10/130\,MHz/1) \cdot 2^8 \cdot H = 76.9\,ns \cdot 256 \cdot H = 19.7\,\mu s \cdot H = 2.52\,ms$).

$$T_{MB} = \frac{CY}{C} \cdot \frac{2^N \cdot H}{F_{CLK}^{MB}} \tag{3}$$

Thus, by tolerating a dead-time of 2.52 ms (i.e., T_{MB}), we can acquire for a maximum duration of 437 μs (i.e., T_{FPGA}) at a rate of 150 MHz (i.e., F_{CLK}^{FPGA}), resulting, as shown in (4), in an average rate of 95 Msps considering 128 histogram (i.e.,$\overline{R} = F_{CLK}^{FPGA} \cdot T_{FPGA}/(T_{FPGA} + T_{MB}) = f_{CLK}^{FPGA}/(1 + T_{MB}/T_{FPGA}) = 150/(1 + 0.00451 \cdot H)\,Msps$).

$$\overline{R} = F_{CLK}^{FPGA} \cdot \frac{1}{1 + \frac{F_{CLK}^{FPGA}}{F_{CLK}^{MB}} \cdot \frac{CY}{C} \cdot \frac{2^N \cdot H}{2^M - 1}} \tag{4}$$

With regard to total latency, the time required for writing to "Readout BRAM" (L_{RB}), the latency of the DMA (L_{DMA}), and the latency of the MicroBlaze (L_{MB}) to access the DDR must be added to the latency of 2 clock pulses at 150 MHz (i.e., F_{CLK}^{FPGA}) of the "Histogram Wrapper" (i.e., $L_{HW} = 2/F_{CLK}^{FPGA} = 2/150\,MHz = 13.3\,ns$). Concerning the BRAM, Equation (5), each "mini-histogram" requires 2^N clock cycles to be memorized, so 1.71 μs per channel are requested (i.e., $L_{RB} = 2^N \cdot H/F_{CLK}^{FPGA} = (2^8/150\,MHz) \cdot H = (1.71\,\mu s) \cdot H$).

$$L_{RB} = \frac{2^N \cdot H}{F_{CLK}^{FPGA}} \tag{5}$$

The latency of the DMA is been measured as 77 μs per channel (i.e., $L_{DMA} = (77\,\mu s) \cdot H$). Referring to the latency of the MicroBlaze, considering that the integration process costs 10 clock cycles at 130 MHz per bin per core we can derive (6) and thus estimating a value of 19.7 μs per channel (i.e., $L_{MB} = (CY \cdot 2^N \cdot H/F_{CLK}^{MB}/C) = (10 \cdot 2^8/130MHz/1) \cdot H = (19.7\,\mu s) \cdot H$).

$$L_{MB} = \frac{CY}{C} \cdot \frac{2^N \cdot H}{F_{CLK}^{FPGA}} \tag{6}$$

In this way, the total latency, as reported in (7), is the sum of theses contributions (i.e., $L = L_{HW} + L_{RB} + L_{DMA} + L_{MB} = 13.3\,ns + (1.71\,\mu s + 77\,\mu s + 19.7\,\mu s) \cdot H \cong 98.4\,\mu s \cdot H$) that means 98.4 μs per channels dominated by the DMA and MicroBlaze.

$$L = \frac{2}{F_{CLK}^{FPGA}} + \frac{2^N \cdot H}{F_{CLK}^{FPGA}} + (77\mu s) \cdot H + \frac{2^N \cdot H}{F_{CLK}^{MB}} \cdot \frac{CY}{C} \tag{7}$$

B. Performance and Figures-of-Merits

1) Area Occupancy: Table II and Fig. 3 show the area occupancy offered by the proposed solution considering M equal to 16 and 32 in BRAM and DDR respectively as a function of the number H of histograms implemented and the number of bin 2^N differentiating the programmable logic (i.e., FPGA and BRAM) and in temporal computing (i.e., DDR and MicroBlaze) sections. For a more straightforward comparison among different architectures and technological nodes, the area occupancy is expressed in terms of number of Lookup Tables (LUTs), Flip-Flops (FFs) and kilobits of BRAM and DDR both for the entire system (ToT) and each individual channel (per CH). Indeed, the number of LUTs, FFs, and kilobits of BRAM can be considered cross-cutting parameters across the technological node inside FPGAs and therefore taken as reference parameters. So, we can derive the average area occupied in FPGA for each single channel as function of the number of bins 2^N that correspond to 207 LUTs, 325 FFs, and 630 kb of BRAM shared between all channels for $N = 8$, 215 LUTs, 339 FFs, and 2166 kb of BRAM (shared between all channels) for $N = 10$, and 210 LUTs, 353 FFs, and 8310 kb of BRAM (shared between all channels) for $N = 12$.

Observing Tab. II and Fig. 3, it is evident that, thanks to the proposed hybrid architecture, the FPGA resources (LUTs, FFs, and BRAM) employed in a single channel decrease with the number of parallel histograms H. This is especially true for the most sensitive resource, the BRAM. In fact, all the memory efforts are shifted to the external DRAM; thus, the DDR usage increases with the number of histograms H. This translates into a significant relaxation of the FPGA's area occupation requirements, as all the storage effort is shifted to the external DDR, which is much more compact and has considerably lower power consumption than the FPGA.

Regarding inferring the DSP for operations in the 7-Series Xilinx FPGA (i.e., DSP48), the decision has been left to the compiler, which did not find it advantageous to infer them. This is most likely because DSP48s become beneficial when words are long (e.g., >32 bits) and comparable to the DSP48's own width (i.e., 48 bits), and in this case, it is not applicable.

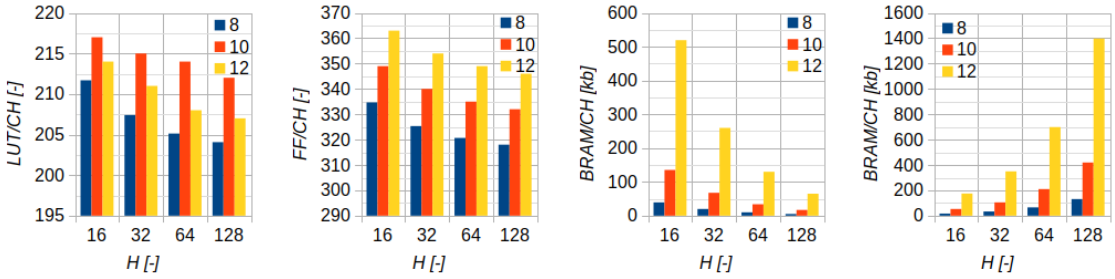
2) Rates and Latency: Table III and Fig. 4 show the figures of merit (i.e., R, $\overline{R}$, and L), reported in (1), (4), and (7), offered by the proposed solution considering M equal to 16 and 32 in BRAM and DDR, respectively, with clocks of 150 MHz and 130 MHz for the FPGA and MicroBlaze (i.e., $F_{CLK}^{FPGA} = 150\,MHz$, $F_{CLK}^{MB} = 130\,MHz$), as a function of the number H of implemented histograms and the number of bins 2^N.

For a more straightforward comparison among different architectures and technological nodes, it has been decided to reference and document, in Tab. III, the maximum acquisition rate with losses $\overline{R}$ expressed as a percentage (i.e., α) of

TABLE II

AREA UTILIZATION OFFERED BY THE PROPOSED SOLUTION CONSIDERING M EQUAL TO 16 AND 32 IN BRAM AND DDR RESPECTIVELY; AS BRAM WE INTENDTHE TOTAL BRAM USED BY THE COMPLETE PROJECT

H	N	$H \cdot 2^N$	$LUTs$		FFs		BRAM $[kb]$		DDR $[kb]$	
			Tot	per CH	Tot	per CH	Tot	per CH	Tot	per CH
16	8	2^{12}	3387	212	5355	335	630	39	131.1	16
32	8	2^{13}	6637	208	10411	326	630	20	262.1	33
64	8	2^{14}	13128	206	20523	321	630	10	524.3	66
128	8	2^{15}	26120	204	40707	318	630	5	1049	131
16	10	2^{14}	3472	217	5584	349	2166	135	524.4	52
32	10	2^{15}	6880	215	10880	340	2166	68	1,049	105
64	10	2^{16}	13696	214	21440	335	2166	34	2,097	210
128	10	2^{17}	27136	212	42496	332	2166	17	4,194	419
16	12	2^{16}	3424	214	5808	363	8310	519	2,097	175
32	12	2^{17}	6752	211	11328	354	8310	260	4,194	350
64	12	2^{18}	13312	208	22336	349	8310	130	8,389	699
128	12	2^{19}	26496	207	44288	346	8310	65	16,778	1398



(a) LUTs per CH. (b) FFs per CH. (c) BRAM per CH. (d) DDR per CH.

Fig. 3. Area Occupancy offered by the proposed solution (Tab. II) as function of the number of channels H with $N = 8$ (blue), $N = 10$ (orange), and $N = 12$ (yellow).

the maximum clock frequency possible in the FPGA (i.e., $F_{CLK}^{MAX} = 680\,MHz$). Therefore, $\alpha = \overline{R}/F_{CLK}^{MAX}$.

Moreover, from Tab. III, we can observe that the maximum rate with losses$\overline{R}$ represents a significant percentage, 50% or more, of the maximum clock frequency in the system (i.e., 150 MHz) when the number of bins used is less than2^{16}, highlighting the effectiveness of the system. On the contrary, the processing latency has been found to be entirely non-optimized.

3) Power Consumption: Power analysis was conducted for the proposed solution, considering M equal to 16 and 32 in BRAM and DDR, respectively. The clocks for the FPGA and MicroBlaze were set at 150 MHz and 130 MHz (i.e.,$F_{CLK}^{FPGA} = 150\,MHz$, $F_{CLK}^{MB} = 130\,MHz$), as a function of the number H of implemented histograms and the number of bins 2^N. The results are reported in Tab. IV and Fig. 5.

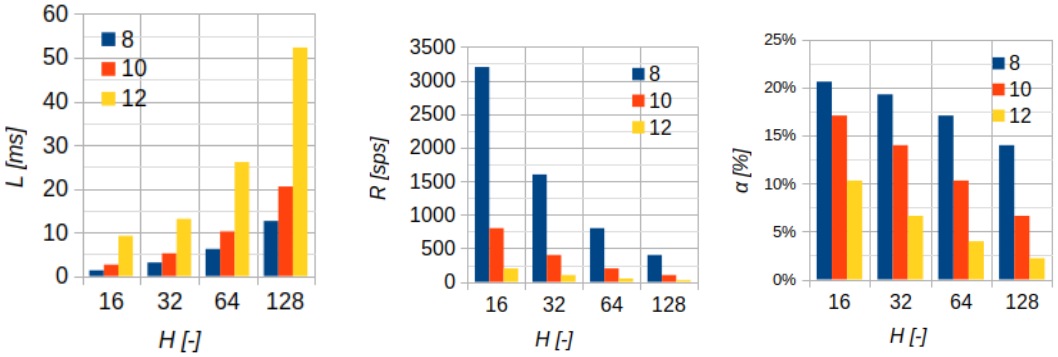
Referring to the functional blocks depicted in Fig. 2 (MIG, MicroBlaze, DMA, and "Histogram Wrapper"), Tab. IV shows that, unlike the "Histogram Wrapper", all functional blocks exhibit dynamic power independent of H and N (i.e., MIG 642 mW, MicroBlaze 115 mW, and DMA 24 mW). In contrast, "Histogram Wrapper" has dynamic power proportional to the number of channels H, equal to $H \times 4.0\,mW$ for $N = 8$, $H \times 4.5\,mW$ for $N = 10$, and $H \times 5\,mW$ for $N = 12$. We can observe the advantage of the following architecture also from the perspective of power dissipation; despite a significant power overhead of 781 mW due to MicroBlaze, MIG, and DMA, the utilization of multiple channels in parallel is expected to result in a very low consumption per individual channel, reaching only a few milliwatts in the configuration with 128 channels.

4) Comparison: In Table V, it is possible to compare the area occupation and figures of merits as function of H and the number of bins 2^N of the proposed hybrid solution with the classic programmable logic architecture proposed in [21] and taken as a reference and used for the design of the "Histogram Wrapper". The trade-off between BRAM occupation and maximum measurement rate without loss (i.e., R) is evident. However, the maximum acquisition rate with losses ($\overline{R}$) of both solution is comparable, hundreds of Msps, if the total number of bins $H \cdot 2^N$ stay below 2^{15} (i.e.; $N = 8$ and $H = 128$, $N = 10$ and $H = 32$). It is important to underline that [21] for $N = 10$ and $N = 12$ no implementations with $M = 32$ are available but only with $M = 20$ and $M = 16$. The comparison of the two implementations reported in Table V, being on systems implemented in the same technological node (i.e., 28-nm Xilinx 7-Series) and within the same FPGA family (i.e., Artix7), does not require the use of α introduced in Paragraph III-B.2. Moreover, considering the 28-nm Xilinx technological node and the Artix-7 XC7A100T (126,800 FFs, 63,400 LUTs, and 1.188 kb BRAM) as target, the proposed hybrid solution, from an area occupancy point of view, is advantageous compared to the classical one presented in [21] for $H > 32$. Furthermore, if $H > 66$ only the proposed

TABLE III

RATES AND LATENCY OFFERED BY THE PROPOSED SOLUTION CONSIDERING M EQUAL TO 16 AND 32 IN BRAM AND DDR RESPECTIVELY WITH CLOCKS OF 150MHZ AND 130 MHZ FOR THE FPGA AND MICROBLAZE

H	N	$H \cdot 2^N$	$L\,[s]$ HW	RB	DMA	MB	Tot	FPGA	$R\,[sps]$ MicroBlaze	Tot	$\overline{R}$ $[sps]$	α [%]
16	8	2^{12}	$13\,n$	$18.72\,\mu$	$1.2\,m$	$0.3\,m$	$1.3\,m$	$150\,M$	$3.2\,k$	$3.2\,k$	$140\,M$	20.6
32	8	2^{13}	$13\,n$	$37.44\,\mu$	$2.5\,m$	$0.6\,m$	$3.1\,m$	$150\,M$	$1.6\,k$	$1.6\,k$	$131\,M$	19.3
64	8	2^{14}	$13\,n$	$74.88\,\mu$	$4.9\,m$	$1.3\,m$	$6.2\,m$	$150\,M$	$0.8\,k$	$0.8\,k$	$116\,M$	17.1
128	8	2^{15}	$13\,n$	$149.8\,\mu$	$9.9\,m$	$2.5\,m$	$12.6\,m$	$150\,M$	$0.4\,k$	$0.4\,k$	$95\,M$	14.0
16	10	2^{14}	$13\,n$	$74.88\,\mu$	$1.2\,m$	$1.3\,m$	$2.6\,m$	$150\,M$	$0.8\,k$	$0.8\,k$	$116\,M$	17.1
32	10	2^{15}	$13\,n$	$149.8\,\mu$	$2.5\,m$	$2.5\,m$	$5.2\,m$	$150\,M$	$0.4\,k$	$0.4\,k$	$95\,M$	14.0
64	10	2^{16}	$13\,n$	$0.3\,m$	$4.9\,m$	$5.0\,m$	$10.2\,m$	$150\,M$	$0.2\,k$	$0.2\,k$	$70\,M$	10.3
128	10	2^{17}	$13\,n$	$0.6\,m$	$9.9\,m$	$10\,m$	$20.5\,m$	$150\,M$	$0.1\,k$	$0.1\,k$	$45\,M$	6.62
16	12	2^{16}	$13\,n$	$0.3\,m$	$1.2\,m$	$5.0\,m$	$9.2\,m$	$150\,M$	$0.2\,k$	$0.2\,k$	$70\,M$	10.3
32	12	2^{17}	$13\,n$	$0.6\,m$	$2.5\,m$	$10\,m$	$13.1\,m$	$150\,M$	$0.1\,k$	$0.1\,k$	$45\,M$	6.62
64	12	2^{18}	$13\,n$	$1.2\,m$	$4.9\,m$	$20\,m$	$26.1\,m$	$150\,M$	50	50	$27\,M$	3.97
128	12	2^{19}	$13\,n$	$2.4\,m$	$9.9\,m$	$40\,m$	$52.3\,m$	$150\,M$	25	25	$15\,M$	2.21



(a) Latency L.

(b) Rate without losses R.

(c) Maximum acquisition rate with losses $\overline{R}$ expressed as a percentage (i.e., α)

Fig. 4. Rates and Latency offered by the proposed solution (Tab. Table III) as function of the number of channels H with $N = 8$ (blue), $N = 10$ (orange), and $N = 12$ (yellow).

TABLE IV

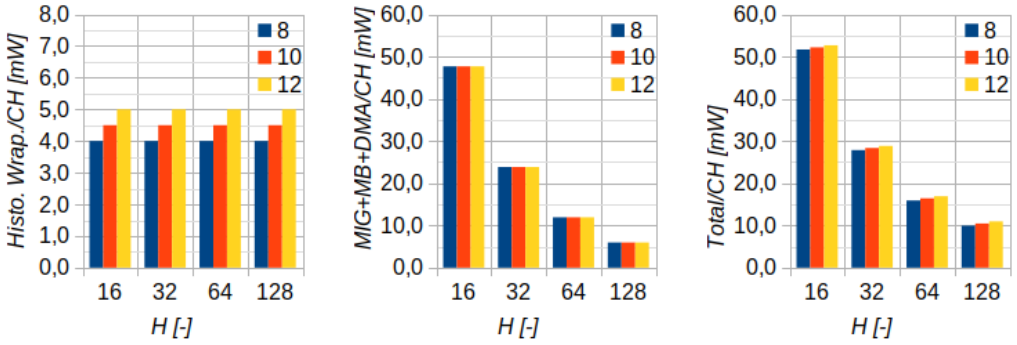
DYNAMIC POWER CONSUMPTION IN MW OFFERED BY THE PROPOSED SOLUTION CONSIDERING M EQUAL TO 16 AND 32 IN BRAM AND DDR RESPECTIVELY WITH CLOCKS OF150 MHZ AND 130 MHZ FOR THE FPGA AND MICROBLAZE

H	N	$H \cdot 2^N$	"Histogram Wrapper" per CH	"Histogram Wrapper" ToT	MIG per CH	MIG ToT	Micro Blaze per CH	Micro Blaze ToT	DMA per CH	DMA ToT	System per CH	System ToT
16	8	2^{12}	4.0	64	40.1	642	7.19	115	1.50	24	52.8	845
32	8	2^{13}		128	20.1		3.59		0.75		28.4	909
64	8	2^{14}		256	10.0		1.80		0.36		16.2	1037
128	8	2^{15}		512	5.02		0.90		0.19		10.1	1293
16	10	2^{14}	4.5	72	40.1	642	7.19	115	1.50	24	53.3	853
32	10	2^{15}		114	20.1		3.59		0.75		28.9	925
64	10	2^{16}		288	10.0		1.80		0.36		16.7	1069
128	10	2^{17}		576	5.02		0.90		0.19		10.6	1357
16	12	2^{16}	5	80	40.1	642	7.19	115	1.50	24	53.8	861
32	12	2^{17}		160	20.1		3.59		0.75		29.4	941
64	12	2^{18}		320	10.0		1.80		0.36		17.2	1101
128	12	2^{19}		640	5.02		0.90		0.19		11.1	1421

hybrid solution is feasible because the classical one saturates the BRAM.

Instead, in Table VI it is possible to compare the proposed solution used for the experimental validation in Section IV with the state-of-the-art exposed in Section II. Considering the use of different technological nodes, the figures of merit regarding FPGA area occupancy (number of LUTs, FFs, and kilobytes of BRAM) are to be referred to a single channel H. As for the rate, it is also reported through the coefficient α introduced in Paragraph III-B.2.

Concerning the area occupancy, it is noticeable that the proposed hybrid solution consistently exhibits one of the



(a) Histogram Wrapper per CH. (b) MIG, MicroBlaze (MB), DMA per CH. (c) Total per CH.

Fig. 5. Power dissipation offered by the proposed solution (Tab. IV) as function of the number of channels H with $N = 8$ (blue), $N = 10$ (orange), and $N = 12$ (yellow).

TABLE V

PERFORMANCE AND OCCUPANCY OFFERED BY THE PROPOSED HYBRID SOLUTIONIN COMPARISON WITH CLASSIC PROGRAMMABLE LOGIC ARCHITECTURE PROPOSED IN [21] AS FUNCTION OF THE NUMBER OF CHANNELS H AND THE NUMBER OF BIN 2^N. IT IS IMPORTANT TO UNDERLINE THAT [21] FOR $N = 10$ AND $N = 12$ NO IMPLEMENTATIONS WITH $M = 32$ ARE AVAILABLE

Ref	N	M	LUTs	FFs	BRAM	L	R	$\overline{R}$
[21]	8	32	$320 \cdot H$	$459 \cdot H$	$18\,kb \cdot H$	$9.7ns\,\forall H$	$206\ Msps\,\forall H$	$206\ Msps\,\forall H$
This	8	32	$207 \cdot H$	$325 \cdot H$	$630\,kb\,\forall H$	$98.4\,\mu s \cdot H$	$50.8/H\ ksps$	$150/(1 + 0.00451 \cdot H)\ Msps$
[21]	10	20	$242 \cdot H$	$332 \cdot H$	$36\,kb \cdot H$	$9.7ns\,\forall H$	$206\ Msps\,\forall H$	$206\ Msps\,\forall H$
This	10	32	$215 \cdot H$	$339 \cdot H$	$2166\,kb\,\forall H$	$162.5\,\mu s \cdot H$	$12.8/H\ ksps$	$150/(1 + 0.01803 \cdot H)\ Msps$
[21]	12	16	$214 \cdot H$	$298 \cdot H$	$72\,kb \cdot H$	$9.5ns\,\forall H$	$210\ Msps\,\forall H$	$210\ Msps\,\forall H$
This	12	32	$210 \cdot H$	$353 \cdot H$	$8310\,kb\,\forall H$	$575\,\mu s \cdot H$	$3.2/H\ ksps$	$150/(1 + 0.07212 \cdot H)\ Msps$

lowest utilizations of LUTs and FFs per channel (208-202 LUTs and 318-335 FFs), alongside [40] (218 LUTs, 213 FFs), [49] (976 LUTs, 359 FFs). However, it keeps the kilobytes of BRAM low (<39 kb) compared to [40] (90 kb) and [49] (594 kb). Moreover, those who offer zero utilization of BRAM compensate for this with an incredible utilization of LUTs and FFs [14], [15], [38], [39]. With regard to the rate, despite the significant area savings, it is still possible to maintain a coefficient α in the range of 14% to 20.6%, comparable to other conventional (i.e., full-parallel) solutions.

C. Trade-Offs and Future Developments

Referring to the maximum rate without lossless R (1), we can see that it depends on the MicroBlaze execution speed (i.e., $F_{CLK}^{MB} \cdot \frac{C}{CY}$) and is inversely proportional to the total number of bins (i.e., $2^N \cdot H$) due to the bottleneck caused by the intrinsic sequential execution of temporal computing. As a result, with the same total number of bins, the only way to increase R is by using fast processors (increasing F_{CLK}^{MB}, reducing CY) and multi-core architectures (increasing C). If we put this figure in relationship with respect to classical FPGA-based solution, like [21] where maximum rate without loss rate correspond to F_{CLK}^{FPGA}, it is evident the gain given by multi-core capability (C) and the need to execute the storage of the histogram in DDR (CY) as fast as possible.

The situation becomes much more promising when considering the average rate $\overline{R}$ (4). In addition to the processor speed, the maximum number of counts possible in the BRAM (i.e., $2^M - 1$) helps alleviate the bottleneck of the total number of bins (i.e., $2^N \cdot H$), providing the designer with an additional tuning factor (2). Naturally, an increase in M results in an increment of $\overline{R}$ and a corresponding increase in area occupancy.

L (7) is significantly worsened and strongly limited by the DMA, thus, in a first approximation, independent of the design parameters.

Moreover, performance in terms of maximum rate without loss and latency can be further improved using System-on-Chip like Xilinx 28-nm Zynq-7000 or 18-nm Zynq Ultrasclale+ where an high-speed (i.e., from 667 MHz to 1.2 GHz for Zynq and from 1.2 GHz to 1.5 GHz for Zynq Ultrasclale+ as F_{CLK}^{MB}) and multi-core (i.e., dual and quad) ARM processor (i.e., ARM-Cortex-A9 for Zynq and ARM-Cortex-A53 or ARM-Cortex-A72 for Zynq Ultrasclale+) can replace the MicroBlaze. In this way, the term $F_{CLK}^{MB} \cdot C$ can be increased by a factor between 10 (dual core at 667 MHz) up to 46 (quad core at 1.5 GHz) proportional to the number of cores and the clock frequency. This means that, referring to the 128-channel implementation, if a Zynq or a Zynq Ultrasclale+ are used the maximum rate without losses (R) is speed up to $4\,ksps$ and $18.4\,ksps$ respectively (instead of the proposed $0.4\,ksps$ proposed with MicroBlaze), while the average rate ($\overline{R}$) saturates to F_{CLK}^{FPGA}.

IV. ARCHITECTURE FOCUS AND EXPERIMENTAL VALIDATION

The simple solution in Section III with $N = 8$ and $H = 128$ has been experimentally validated using time

TABLE VI
COMPARISON OF THE PROPOSED SOLUTION WITH RESPECT TO STATE-OF-THE-ARTIN TABLE I

Ref	FPGA Model	F_{CLK}^{MAX} $[MHz]$	H	N	M	LUTs per CH	FF per CH	BRAM $[kb]$ per CH	L $[s]$	F_{CLK} $[MHz]$	R $[sps]$	$\overline{R}$ $[sps]$	α [%]
[38]	Altera Cyclone IV EP4CE22	433	1	8	32	3400	6800	0	$60\,n$	100	$100\,M$	$100\,M$	23.1
[39]	unspecified	N.A.	1	8	22	15280	30560	0	-	280	$280\,M$	$280\,M$	-
[14]	Xilinx Zynq XC7020	800	1	8	8	11850	9594	0	-	100	$100\,M$	$100\,M$	12.5
[40]	Xilinx Artix-7 XC7A100T	680	1	8	14	218	213	90	-	200	$50\,M$	$50\,M$	7.35
[15]	Xilinx Zynq XC7Z030	800	1	16	8	3865	4903	0	-	370	$370\,M$	$370\,M$	46.3
[49]	Xilinx Virtex II Pro	400	1	16	8	976	359	594	-	85	$85\,M$	$85\,M$	21.3
[50]	unspecified	N.A.	1	16	8	1265	1862	512	$0.38\,m$	121	$121\,M$	$121\,M$	-
[21]	Xilinx Artix-7 XC7A35T	680	1	8	32	320	459	18	$9.7\,n$	206	$206\,M$	$206\,M$	30.3
This	Xilinx Artix-7 XC7A100T	680	16	8	32	212	335	39	$1.6\,m$	150	$3.2\,k$	$140\,M$	20.6
This	Xilinx Artix-7 XC7A100T	680	32	8	32	208	326	20	$3.1\,m$	150	$1.6\,k$	$131\,M$	19.3
This	Xilinx Artix-7 XC7A100T	680	64	8	32	206	321	10	$6.3\,m$	150	$0.8\,k$	$116\,M$	17.1
This	Xilinx Artix-7 XC7A100T	680	128	8	32	204	318	5	$12.6\,m$	150	$0.4\,k$	$95\,M$	14.0

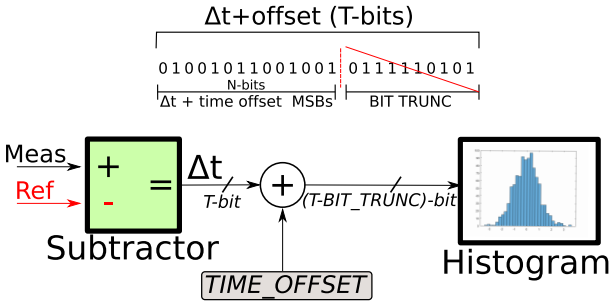


Fig. 6. Proposed histogram is used to visually represent the N bits of Δt that are stored.

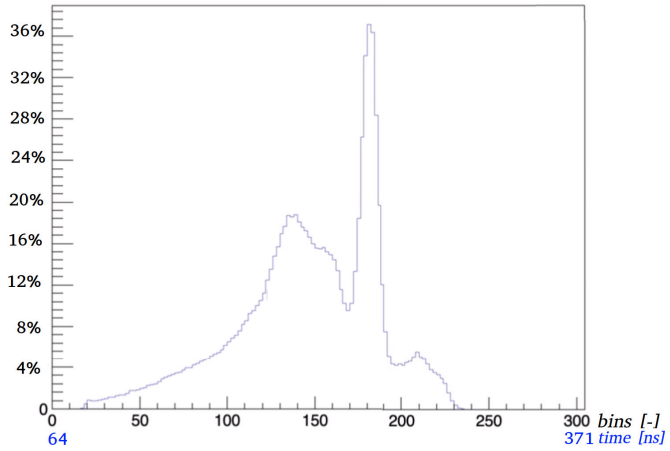


Fig. 7. Single channel histogram at 2^8 bins of the ToT with $FSR \in [64\,ns; 307.2\,ns]$ and $BW = 1.2\,ns$.

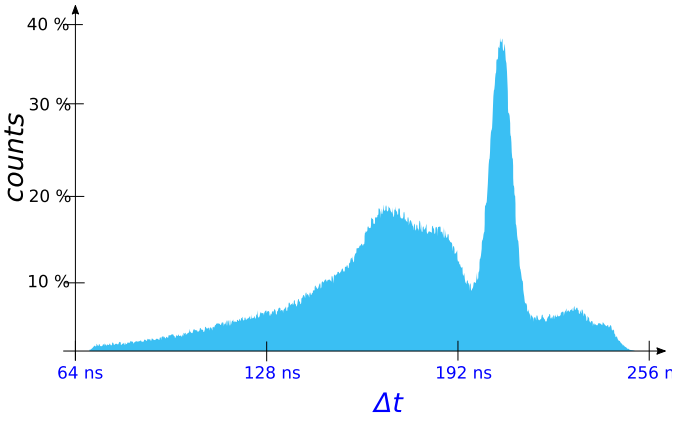


Fig. 8. Multi channel histogram at 128×2^8 bins of the ToT with $FSR \in [64\,ns; 256\,ns]$ and $BW = 4.69\,ps$.

measurements via Time-to-Digital Converter (TDC) IP-Core provided by TEDIEL S.r.l. [51], [52], [53] as a case study.

The TDC is a digital system that converts the time difference between two events (i.e., START and STOP) into a T-bit wide digital code called Timestamp (i.e., Δt). The TDC considered has a Timestamp of 32-bit wide and characterized by an LSB is 36.6 fs. In this sense, each single channel of the TDC directly generates the data to be histogrammed by a single histogram channel of the proposed hybrid system. As was already indicated, the ability to create histograms of timestamps is very valuable in applications like TCSPC, TOF-PET, optical spectroscopy, and many others [54], [55], [56], [57].

Furthermore, to better adjust the histogram Bin Width (BW) and Full-Scale Range (FSR), as depicted in Figure 6, two registers, for each "mini-histogrammer", called $TIME\_OFFSET$ and BIT_TRUNC have been introduced to accommodate the size of Δt, which is T- bits wide (i.e., 32), to the 2^N bins (i.e., 256) provided by the histogram. This way, each channel of the histogram offers a BW equal to $2^{BIT\_TRUNC} \times LSB$ and an FSR ranging from $TIME\_OFFSET \times LSB$ up to $IME\_OFFSET \times LSB + 2^N \times BW$. Thus, only the bits from BIT_TRUNC to $N + BIT\_TRUNC$ of the timestamp offset in time (i.e., $\Delta t + TIME\_OFFSET \times LSB$) are histogrammed, rather than all T-bits of Δt.

Following, we present acquisitions from a TOF-PET setup, in which a Time-over-Threshold (ToT) over a SiPM spanning from 70 ns up to 252 ns is read out by the aforementioned TDC and histogrammed using the proposed solution. Thanks to the DDR aproch we can achive up to $2^{32} - 1$ counts per each single bin.

In Figure 7, a single histogram consisting of 256 bins from one of the 128 available channels is observable. In this configuration, the FSR of the histogram has been set within the range of 64 ns to 371.2 ns with a bin width (BW) of 1.2 ns. Specifically, considering an LSB of 36.6 fs, BIT_TRUNC has been set to 15, while $TIME\_OFFSET$ is fixed at 1747626.

Furthermore, it is possible to utilize all 128×2^8 bins available across the $H = 128$ histograms to obtain a single histogram with a more resolved bin width by appropriately programming each$TIME_OFFSET$ so that the FSR of each channel results contiguous to one another. In this regard, Figure 8 is histogramming the same information as Figure 7 using an FSR from 64 ns to 256 ns, but here, by operating all 128 histograms in parallel on the same measurement, it is possible to achieve a bin width of 4.69 ps. In this context, all 128 histograms have the same BIT_TRUNC equal to 7, while $TIME\_OFFSET$ scales linearly from channel to channel of the histogramming module, where the $h-th$ channel (with $h \in [0; 128-1]$) has a $TIME\_OFFSET$ equal to $1747626 + 2^8 \times h$.

V. Conclusion

This paper presents a multi-channel histogramming mechanism, up to a maximum of 128 channels, based on a hybrid spatial and temporal computing technology. The proposed system with the 256 bins configuration was experimentally validated in a measurement setup for TOF-PET, where dozens/hundreds of histograms are required. The proposed technique enables the implementation of up to 128 histograms, each consisting of 256 bins of 32 bits (8192 bits/histogram), in a low-end Xilinx 28-nm 7-Series FPGA (i.e.; Artix-7 XC7A100T) using minimal hardware resources such as 207 LUTs and 325 FFs for a single histogram and 630 kb of BRAM shared between all channels; a power consumption of 10.1 mW per channel is measured. This is made possible by storing the histogram not only the BRAM (which has a maximum capacity of 1.2 Mb in the target FPGA), but in an external M45W8MW16 DDR (0.134 Gb). Communication between programmable logic and DDR is enabled through a MicroBlaze that works in combination with an MIG for read/write operations to the DDR and a DMA for reading histograms from programmable logic. This temporal-computing approach has greatly simplified the control logic that would be required for direct interfacing between programmable logic and DDR. The bottleneck of this architecture has been found to be latency L, i.e. $\sim (98.4\,\mu s) \cdot H$, and maximum rate without losses R, i.e. $\sim (50.8\,ksps)/H$, where H is the number of histogram implemented. However, if losses of some data are allowed (e.g., TCSPC and TOF-PET), the system can sustain an average rate $\overline{R}$ up to 95 Msps among both the 128 histograms.

The great results (in terms of trade-off between number of channels and RAM) obtained from the first tests suggest that the realized architecture has perspectives for being further developed and investigated for optimization and performance enhancing. Many potential improvements have already been identified, such as the migration of the architecture from bare-metal to a Linux based system, in order to take advantage of all the features a Linux system can provide. Moreover, performance in terms of latency and rate without losses can be further improved using System-on-Chip like Zynq or Zynq Ultrascale+ where an high-speed (i.e., from 667 MHz to 1.2 GHz for Zynq and from 1.2 GHz to 1.5 GHz for Zynq Ultrasclale+) and multi-core (i.e., dual and quad) ARM processor (i.e., ARM-Cortex-A9 for Zynq and ARM-Cortex-A53 or ARM-Cortex-A72 for Zynq Ultrasclale+) can replace the MicroBlaze. In these terms, the bandwidth of 50.8 ksps (obtained with one core clocked at 130 MHz) can be increased by a factor between 10 (dual core at 667 MHz) and up to 46 (quad core at 1.5 GHz), proportional to the number of cores and the clock frequency.

Acknowledgment

The authors would like to thank TEDIEL S.r.l. for materials, the TDC IP-Core, and the support.

References

[1] A. Gundlach-Graham, L. Hendriks, K. Mehrabi, and D. Günther, "Monte Carlo simulation of low-count signals in time-of-flight mass spectrometry and its application to single-particle detection," *Anal. Chem.*, vol. 90, no. 20, pp. 11847–11855, Oct. 2018.

[2] S. Dutta, A. Abhinav, P. Dutta, P. Kumar, and A. Halder, "An efficient image compression algorithm based on histogram based block optimization and arithmetic coding," *Int. J. Comput. Theory Eng.*, vol. 4, no. 6, pp. 954–957, 2012.

[3] R. Kapoor, R. Gupta, L. H. Son, S. Jha, and R. Kumar, "Detection of power quality event using histogram of oriented gradients and support vector machine," *Measurement*, vol. 120, pp. 52–75, May 2018. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0263224118300940

[4] Z. Istvan, L. Woods, and G. Alonso, "Histograms as a side effect of data movement for big data," in *Proc. ACM SIGMOD Int. Conf. Manage. Data*, 2014, pp. 1567–1578.

[5] T. N. Palmer, "Stochastic weather and climate models," *Nature Rev. Phys.*, vol. 1, no. 7, pp. 463–471, May 2019.

[6] Y. Han and S. Ahn, "Stochastic modeling of breakdown at freeway merge bottleneck and traffic control method using connected automated vehicle," *Transp. Res. B, Methodol.*, vol. 107, pp. 146–166, Jan. 2018.

[7] Y. Peng, M. C. Fu, B. Heidergott, and H. Lam, "Maximum likelihood estimation by Monte Carlo simulation: Toward data-driven stochastic modeling," *Operations Res.*, vol. 68, no. 6, pp. 1896–1912, Nov. 2020.

[8] H. Pereira and R. C. Marques, "An analytical review of irrigation efficiency measured using deterministic and stochastic models," *Agricult. Water Manage.*, vol. 184, pp. 28–35, Apr. 2017.

[9] W. Yang, S. Tang, M. Li, Y. Cheng, and Z. Zhou, "Steganalysis of low embedding rates LSB speech based on histogram moments in frequency domain," *Chin. J. Electron.*, vol. 26, no. 6, pp. 1254–1260, Nov. 2017.

[10] R. Lima and R. Sampaio, "Parametric analysis of the statistical model of the stick-slip process," *J. Sound Vibrat.*, vol. 397, pp. 141–151, Jun. 2017.

[11] A. Anand, V. Jha, and L. Sharma, "An improved local binary patterns histograms techniques for face recognition for real time application," *Int. J. Recent Technol. Eng.*, vol. 8, no. 2S7, pp. 524–529, 2019.

[12] T. Surasak, I. Takahiro, C. Cheng, C. Wang, and P. Sheng, "Histogram of oriented gradients for human detection in video," in *Proc. 5th Int. Conf. Bus. Ind. Res. (ICBIR)*, May 2018, pp. 172–176.

[13] F. M. Della Rocca et al., "A 128 × 128 SPAD motion-triggered time-of-flight image sensor with in-pixel histogram and column-parallel vision processor," *IEEE J. Solid-State Circuits*, vol. 55, no. 7, pp. 1762–1775, Jul. 2020.

[14] T. Bonny, T. Rabie, and A. H. A. Hafez, "Multiple histogram-based face recognition with high speed FPGA implementation," *Multimedia Tools Appl.*, vol. 77, no. 18, pp. 24269–24288, Sep. 2018, doi: 10.1007/s11042-018-5647-8.

[15] S. Hazra, S. Ghosh, S. P. Maity, and H. Rahaman, "A new FPGA and programmable SoC based VLSI architecture for histogram generation of grayscale images for image processing applications," *Proc. Comput. Sci.*, vol. 93, pp. 139–145, 2016. https://www.sciencedirect.com/science/article/pii/S1877050916314338

[16] J. J. Hamill, "2D energy histograms for scatter estimation in an SiPM PET scanner," in *Proc. IEEE Nucl. Sci. Symp. Med. Imag. Conf. (NSS/MIC)*, 2019, pp. 1–4.

[17] M. Alayed and M. Deen, "Time-resolved diffuse optical spectroscopy and imaging using solid-state detectors: Characteristics, present status, and research challenges," *Sensors*, vol. 17, no. 9, p. 2115, Sep. 2017.

[18] J. Bouchard et al., "A low-cost time-correlated single photon counting system for multiview time-domain diffuse optical tomography," *IEEE Trans. Instrum. Meas.*, vol. 66, no. 10, pp. 2505–2515, Oct. 2017.

[19] O. M. Mozos, H. Mizutani, H. Jung, R. Kurazume, and T. Hasegawa, "Categorization of indoor places by combining local binary pattern histograms of range and reflectance data from laser range finders," *Adv. Robot.*, vol. 27, no. 18, pp. 1455–1464, Dec. 2013.

[20] A. Margara, P. Peronio, G. Acconcia, G. Cugola, and I. Rech, "High-accuracy and video-rate lifetime extraction from time correlated single photon counting data on a graphical processing unit," *Rev. Sci. Instrum.*, vol. 90, no. 10, Oct. 2019, Art. no. 104709.

[21] A. Costa, N. Corna, F. Garzetti, N. Lusardi, E. Ronconi, and A. Geraci, "High-performance computing of real-time and multichannel histograms: A full FPGA approach," *IEEE Access*, vol. 10, pp. 47524–47540, 2022.

[22] Xilinx. (2019). *7 Series FPGAs Memory Resources*. [Online]. Available: https://www.xilinx.com/support/documentation/user_guides/ug473_7Series_Memory_Resources.pdf

[23] (2019). *MicroBlazeProcessor Reference Guide*. [Online]. Available: https://www.xilinx.com/support/documentation/sw_manuals/xilinx2019_1/ug984-vivado-microblaze-ref.pdf

[24] (2020). *7 Series FPGAs Data Sheet: Overview*. [Online]. Available: https://www.xilinx.com/support/documentation/data_sheets/ds180_7Series_Overview.pdf

[25] Digilent. *Digilent Nexys 4*. Accessed: Sep. 2023. [Online]. Available: https://digilent.com/reference/programmable-logic/nexys-4/start

[26] W. Kehl, F. Tombari, S. Ilic, and N. Navab, "Real-time 3D model tracking in color and depth on a single CPU core," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. (CVPR)*, Jul. 2017, pp. 465–473.

[27] M. Poostchi, K. Palaniappan, D. Li, M. Becchi, F. Bunyak, and G. Seetharaman, "Fast integral histogram computations on GPU for real-time video analytics," 2017, *arXiv:1711.01919*.

[28] E. Wang et al., "Multichannel spatially nonhomogeneous focused vector vortex beams for quantum experiments," *Adv. Opt. Mater.*, vol. 7, no. 8, Apr. 2019, Art. no. 1801415.

[29] C. L. Ruiz et al., "Multichannel, triaxial, neutron time-of-flight diagnostic for experiments at the z facility," *Phys. Rev. Accel. Beams*, vol. 23, no. 2, Feb. 2020, Art. no. 020401.

[30] T. Ochiai, S. Watanabe, T. Hori, and J. R. Hershey, "Multichannel end-to-end speech recognition," in *Proc. Int. Conf. Mach. Learn.*, 2017, pp. 2632–2641.

[31] W. Fu, B. Wei, X. Li, Q. Wang, and X. Hu, "A low delay transmission method of multi-channel video based on FPGA," *IOP Conf. Ser., Mater. Sci. Eng.*, vol. 322, no. 5, 2018, Art. no. 052032.

[32] D. Shi, W.-S. Gan, J. He, and B. Lam, "Practical implementation of multichannel filtered-x least mean square algorithm based on the multiple-parallel-branch with folding architecture for large-scale active noise control," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 28, no. 4, pp. 940–953, Apr. 2020.

[33] A. A. Khedkar and R. Khade, "High speed FPGA-based data acquisition system," *Microprocessors Microsystems*, vol. 49, pp. 87–94, Mar. 2017. [Online]. Available: https://www.sciencedirect.com/science/article/pii/S0141933116303453

[34] J. Lu et al., "Real-time FPGA-based digital signal processing and correction for a small animal PET," *IEEE Trans. Nucl. Sci.*, vol. 66, no. 7, pp. 1287–1295, Jul. 2019.

[35] A. C. Therrien, R. Herbst, O. Quijano, A. Gatton, and R. Coffee, "Machine learning at the edge for ultra high rate detectors," in *Proc. IEEE Nucl. Sci. Symp. Med. Imag. Conf. (NSS/MIC)*, Oct. 2019, pp. 1–4.

[36] T. Okino et al., "5.2 A 1200×900 6μm 450fps geiger-mode vertical avalanche photodiodes CMOS image sensor for a 250m time-of-flight ranging system using direct-indirect-mixed frame synthesis with configurable-depth-resolution down to 10 cm," in *IEEE Int. Solid-State Circuits Conf. (ISSCC) Dig. Tech. Papers*, Feb. 2020, pp. 96–98.

[37] H.-J. Choe, Y. Choi, D. J. Kwak, and J. Lee, "Prototype time-of-flight PET utilizing capacitive multiplexing readout method," *Nucl. Instrum. Methods Phys. Res. A, Accel. Spectrom. Detect. Assoc. Equip.*, vol. 921, pp. 43–49, Mar. 2019.

[38] L. Maggiani, C. Salvadori, M. Petracca, P. Pagano, and R. Saletti, "Reconfigurable architecture for computing histograms in real-time tailored to FPGA-based smart camera," in *Proc. IEEE 23rd Int. Symp. Ind. Electron. (ISIE)*, Jun. 2014, pp. 1042–1046.

[39] J. O. Cadenas, R. S. Sherratt, P. Huerta, and W.-C. Kao, "Parallel pipelined array architectures for real-time histogram computation in consumer devices," *IEEE Trans. Consum. Electron.*, vol. 57, no. 4, pp. 1460–1464, Nov. 2011.

[40] B. Younis and B. Younis, "Low cost histogram implementation for image processing using FPGA," *IOP Conf. Ser., Mater. Sci. Eng.*, vol. 745, Mar. 2020, Art. no. 012044.

[41] Xilinx. (2011). *7 Series FPGAs Configurable Logic Block*. [Online]. Available: https://www.xilinx.com/support/documentation/user_guides/ug474_7Series_CLB.pdf

[42] J. Cadenas, R. S. Sherratt, P. Huerta, W.-C. Kao, and G. M. Megson, "C-slow retimed parallel histogram architectures for consumer imaging devices," *IEEE Trans. Consum. Electron.*, vol. 59, no. 2, pp. 291–295, May 2013.

[43] ARM. (2011). *AMBA AXI ACE Protocol Specification*. [Online]. Available: http://www.gstitt.ece.ufl.edu/courses/fall15/eel4720_5721/labs/refs/AXI4_specification.pdf

[44] N. Corna et al., "High-performance physical-independent address-based communication interface for FPGA in custom scientific equipment," in *Proc. IEEE Nucl. Sci. Symp. Med. Imag. Conf. (NSS/MIC)*, Oct. 2020, pp. 1–4.

[45] E. Ronconi, N. Corna, A. Costa, F. Garzetti, N. Lusardi, and A. Geraci, "Multi-COBS: A novel algorithm for byte stuffing at high throughput," *IEEE Access*, vol. 10, pp. 78848–78859, 2022.

[46] Xilinx. (2019). *AXI Video Direct Memory Access V6.2*. [Online]. Available: https://www.xilinx.com/support/documentation/ip_documentation/axi_dma/v7_1/pg021_axi_dma.pdf

[47] (2019). *AXI DMA V7.1*. [Online]. Available: https://www.xilinx.com/support/documentation/ip_documentation/axi_dma/v7_1/pg021_axi_dma.pdf

[48] (2018). *AXI Central Direct Memory Access V4.1*. [Online]. Available: https://www.xilinx.com/support/documentation/ip_documentation/axi_cdma/v4_1/pg034-axi-cdma.pdf

[49] A. Shahbahrami, J. Hur, B. Juurlink, and S. Wong, "FPGA implementation of parallel histogram computation," in *Proc. 2nd HIPEAC Workshop Reconfigurable Comput.*, 2008, pp. 63–72.

[50] P. Mondal and S. Banerjee, "A reconfigurable memory-based fast VLSI architecture for computation of the histogram," *IEEE Trans. Consum. Electron.*, vol. 65, no. 2, pp. 128–133, May 2019.

[51] Accessed: Sep. 2023. [Online]. Available: https://tediel.com/

[52] F. Garzetti, N. Corna, N. Lusardi, and A. Geraci, "Time-to-digital converter IP-core for FPGA at state of the art," *IEEE Access*, vol. 9, pp. 85515–85528, 2021.

[53] N. Lusardi, F. Garzetti, A. Costa, E. Ronconi, and A. Geraci, "From multiphase to novel single-phase multichannel shift-clock fast counter time-to-digital converter," *IEEE Trans. Ind. Electron.*, vol. 71, no. 8, pp. 9886–9894, Aug. 2024.

[54] F. Garzetti et al., "Assessment of the bundle SNSPD plus FPGA-based TDC for high-performance time measurements," *IEEE Access*, vol. 10, pp. 127894–127910, 2022.

[55] N. Lusardi, F. Garzetti, N. Corna, S. Salgaro, N. Bachetti, and A. Geraci, "Plug-and-play tunable and high-performance time-to-digital converter as IP-core for Xilinx FPGAs," in *Proc. IEEE Nucl. Sci. Symp. Med. Imag. Conf. (NSS/MIC)*, Oct. 2020, pp. 1–3.

[56] N. Lusardi et al., "High-resolution imager based on time-to-space conversion," *IEEE Trans. Instrum. Meas.*, vol. 71, pp. 1–11, 2022.

[57] N. Lusardi, N. Corna, F. Garzetti, S. Salgaro, and A. Geraci, "Cross-talk issues in time measurements," *IEEE Access*, vol. 9, pp. 129303–129318, 2021.

Enrico Ronconi (Graduate Student Member, IEEE) received the master's degree in electronic engineering from Politecnico di Milano in 2020. His research topic is focused on advanced programmable logic (PL) and software architectures for data processing and transfer in field programmable gate arrays (FPGA) implemented scientific equipment, currently used and developed together with the time-to-digital and digital-to-time converters (TDC and DTC) developed in the same laboratory.

Fabio Garzetti (Member, IEEE) received the Ph.D. degree (cum laude) in 2022. He developed his thesis work from the Digital Electronics Laboratory, Department of Electronics, Information and Bioengineering (DEIB), on a topic regarding innovative solutions for calibration and triggering of asynchronous signals for time-to-digital converters (TDCs) in field programmable gate arrays (FPGA). He is currently a Temporary Researcher with the Digital Electronics Laboratory, DEIB.

Nicola Lusardi (Graduate Student Member, IEEE) received the Ph.D. degree in 2018. He developed his thesis work from the Digital Electronics Laboratory, Department of Electronics, Information and Bioengineering (DEIB), on a topic regarding high-resolution time-to-digital converters (TDCs) in field programmable gate arrays (FPGA). He is currently a Temporary Researcher with DEIB, a Professor of electronics with Politecnico di Milano, and an Associate Member of Italian National Nuclear Physics Institute (INFN).

Andrea Costa (Graduate Student Member, IEEE) was born in Piacenza in 1995. He received the B.Sc. degree in biomedical engineering and the M.Sc. degree in electronics engineering from Politecnico di Milano in 2017 and 2020, respectively. His research topics relate to innovative hardware architectures for data processing in the field of FPGA time-domain devices and FPGA DAQ for high data rate environments.

Angelo Geraci (Senior Member, IEEE) received the Ph.D. degree (cum laude) in electronics from Politecnico di Milano in 1996. He has been an Associate Professor at the Department of Electronics, Information and Bioengineering (DEIB), Politecnico di Milano, since 2004. His research activity is mainly focused on digital electronics based on microcontrollers, and DSP and FPGA devices, specifically in the areas of radiation detection and medical imaging.

Open Access funding provided by 'Politecnico di Milano' within the CRUI CARE Agreement