UNIVERSITY OF BRISTOL

University of Bristol - Explore Bristol Research
General rights

🔓 **Open Access**

A Precise High Count-Rate FPGA Based Multi-Channel Coincidence Counting System for Quantum Photonics Applications

Ekin Arabul
Stefano Paesani
Scott Tancock
John Rarity
Naim Dahnoun

A Precise High Count-Rate FPGA Based Multi-Channel Coincidence Counting System for Quantum Photonics Applications

Ekin Arabul [ID]**, Stefano Paesani** [ID]**, Scott Tancock, John Rarity, and Naim Dahnoun** [ID]

Quantum Engineering Technology Labs, H. H. Wills Physics Laboratory and Department of Electrical and Electronic Engineering, University of Bristol, Bristol BS8 1TH, U.K.

Abstract: Coincidence counters play the role of gating events from the background noise in almost every quantum photonics setup. Precise multi-channel and high count-rate co-incidence counting tools have become desirable due to an increase in the complexity of quantum photonics experiments. However, timing analyzers are struggling to meet these needs. We are proposing a Field Programmable Gate Array (FPGA) based coincidence counting system which provides 8 operational channels with 8.9 ps root mean square (RMS) resolution (with a bin width of 7.7 ps) and a count-rate of 320 million counts per second (MCPS) (with 40 MCPS per channel). We have successfully tested our design in different quantum photonics scenarios such as the detection of two-photon interference and pseudo-photon-number resolving detection of a coherent state of light where it has shown its capability of working beyond the saturation of detectors. Also, we have introduced a Dual Data Rate Registration TDC, which improved the linearity of the time tagging operation by using both clock edges without increasing the dead-time or using the space excessively. 1.2 LSB in max DNL error, 1.8 LSB in max INL error, 10 ps in FWHM and 3.2 ps in RMS resolution improvements were achieved.

Index Terms: Time-to-digital converters, coincidence counting, timing measurement, field programmable gate array (FPGA), quantum computing, quantum information, superconducting nanowire single-photon detector.

1. Introduction

Precise time measurement and correlation are important concepts in many engineering and science application. The coincidence counters can be an example of one of these correlation tools which are used in almost every quantum photonics applications. These applications include photonic quantum simulators [1], quantum communication experiments [2] and boson sampling machines [3]. Their purpose in a photonic setup is measuring the correlation between simultaneous trigger events generated by single-photon detectors (SPD). The real-time processing of such events is of increasing importance as quantum photonics experiments increase in complexity, where the total number of events becomes too large to be stored and post-processed [4]. Moreover, coincidence counting allows experiments to filter out the background noise (uncorrelated photons
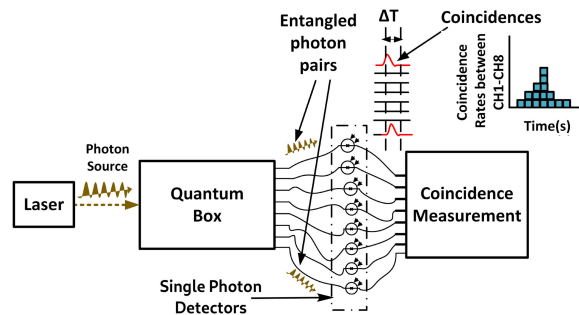
Fig. 1. Example usage of a coincidence measurement in a quantum application.

and dark counts) from the entangled photons generated and processed in quantum experiments. An illustration of coincidence counting in a quantum setup can be seen in Fig. 1.

To implement a precise and flexible coincidence counter, Time-to-Digital Converters (TDCs) can be used. TDCs are used for generating timestamps for events occurring in the time domain. Traditionally, TDCs are used for measuring the time difference between the START and STOP events. However, generally in Field Programmable Gate Array (FPGA) based implementations, the STOP signal is replaced by the system's clock edge, and inputs' positions are measured relative to the clock edge [5]. For fine time measurement, asynchronous logic circuits such as carry chains are used to subdivide the clock period into smaller bins. In quantum photonics applications, TDCs are employed for digitising the time of arrival of a photon [6].

The phenomenon of coincidence is defined as two or more events occurring within a time interval simultaneously. This time interval is known as the coincidence window. In quantum photonics tasks, these coincidences can occur across multiple channels and with different combinations [7], [8]. The coincidence counters are employed to measure these patterns and count them. The coincidence counter uses these patterns internally as addresses, with the associated memory locations incremented when the patterns are detected.

Developments in single photon detectors promise to achieve significantly higher count rates [9] in quantum photonics applications. However, timing measurement tools are struggling to keep up with the data rates of the detectors, especially in high precision and multi-channel instruments. Popular commercial coincidence counter tools available include ID Quantique's ID900 which can count with 4 channels at 25 million counts per second (MCPS) per channel (with a total of 100 MCPS) while the root mean square (RMS) resolution is at 8 ps [10] and PicoQuant's HydraHarp400 [11] with a software-based coincidence counter. The maximum count rate achieved by this product per channel was at 12.5 M counts per second and 96 MCPS across 8 channels.

In the literature, some attempts were previously made at implementing multi-channel coincidence counters. This includes a FPGA (Altera Cyclone III) coincidence counter based on a TDC implemented to use for with ToF PET cameras where they achieved a total count rate of 72 MCPS and the resolution was at 69.8 ps [12]. This implementation used a separate 12 channel TDC board to achieve the coincidence counting, which hindered the count rate of the operation. Another plausible coincidence instrument in the literature is a very large scale 32-channel FPGA (Xilinx Spartan6 LX16) coincidence counter implementation for boson sampling [13]. Their method achieved 390 ps resolution by using an octa-phase TDC. The scale of the implementation was impressive, but, due to the count rate of the operation was limited by the software acquisition and histogramming with 0.5 MCPS. In addition, [14] demonstrates another very large scale 64 channel TDC based coincidence counter for animal PET scanners, which was implemented on a Xilinx Virtex 2 FPGA. This implementation used an external Analogue-to-Digital Converter (ADC) board for the time generation and achieved 0.7 ns resolution with a 32 MCPS data rate.

The majority of the TDC architectures in the current literature use technologies such as FPGAs, Applications Specific Integrated Circuits (ASICs) and analogue circuits. These methods are

preferable over synchronous Integrated Circuits (ICs) such as Central Processing Units (CPUs), Graphics Processing Units (GPUs) and Digital Signal Processors (DSPs) because their precision is not limited by the system's clock. However, due to the high dead-times of analogue circuits such as Time-to-Analogue Converters (TACs) [5], FPGA and ASIC technologies are more desirable for a high count rate timing instrument due to their capabilities of accommodating fast asynchronous and concurrent logic designs, which are proven to measure timings with more precision than system clock speeds [5]. Typically, ASICs are prohibitively expensive compared to FPGAs as the break-even point between ASICs and FPGAs is around 100 thousand units per manufacturing run [15]. Also, FPGAs provide more flexibility in applications due to their reprogrammability in the field, and so our choice of technology in this research was the FPGA.

Although TDCs have been used for coincidence counting for a while, the main problem that hinders the data rate of the coincidence counting is the data transfer between the TDC and coincidence counter since data cannot be processed as it is generated. Our proposed implementation integrates the coincidence counting and time tag generation into the same FPGA fabric, and this mitigates the use of any data transfer protocol between the TDC and coincidence counter. Hence, the instrument provides a precision of 8.9 ps RMS with a count rate of 320 MCPS. The proposed system achieves the highest count rate available of any 8-channel precise timing instrument at the moment. Unlike our previous scheme explained in [16], we are avoiding usage of any first-in first-out (FIFO) structures, buffers or data serialisers to achieve it. This theoretically provides the best possible real-time operation and count-rate since there are no tags stored during the process and coincidences are measured as the events are detected.

In addition to the changes in the coincidence counting logic, we have also implemented a dual-data rate registration scheme to improve the linearity of the TDC. This method focuses on using both the clock edges to quantise the events and improve the linearity by averaging the two time measurements. There are a few implementations such as double registration and wave union launchers [17] that use a similar method to improve the linearity of the TDC through averaging. However, their common problem is extending the measurement range, and this introduces an additional dead-time. With this method, since both the clock edges are used for this operation, no additional dead-time is introduced, and thus, the count rate is not affected by the linearisation.

In this paper, our proposed precise high count-rate TDC based coincidence counter architecture will be discussed in later sections. In Section 2, the system architecture and implementation details will be discussed, in Section 3, the results obtained from both quantum photonics experiments and tests are detailed, and in Section 4 the conclusion and future work can be found.

2. System Architecture

The proposed system architecture was implemented on a Spartan6 LX150 FPGA located on an Opal Kelly XEM6310. This board mounts a breakout board designed at the University of Bristol which generates a low jitter clock and provides 8 channels input and output ports. The proposed coincidence counting and TDC techniques are applicable to all FPGA platforms, and therefore the usage of the Spartan 6 architecture in this research does not have any significance other than describing the implementation.

The first part of the design is the TDC module where a delay line is used for generating the thermometer codes for the input trigger, a priority encoder for converting thermometer codes to 9 bit codes, a tag selector for averaging the codes generated with both the clock edges and a calibration block which calibrates codes to mitigate the effects of the non-linearity.

The second part of the design is the coincidence counter logic which consists of a multi-tag correlator, coincidence detector and coincidence counter. The multi-tag correlator finds the time difference between time tags, the coincidence detector forms the coincidence address and the coincidence counter counts them using a RAM block.

The final part of the system is the output logic and the software interface which was used for transferring the coincidence counting results to the PC. The output logic is based on the USB 3.0 connection which used a Cypress FX3 chip located on an Opal Kelly XEM6310 board. A FIFO
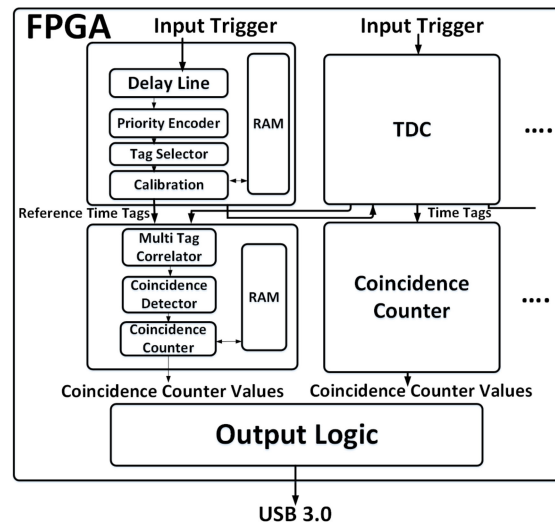
Fig. 2. Overview of the architecture.

located inside the FPGA uses a 101 MHz USB clock for this operation. Also, a software interface utilising an Opal Kelly API was implemented for data acquisition and data decoding for the USB 3.0. This software can adjust the digital delays affecting each channel and the coincidence window sizes. These delays and windows sizes can be set to multiples of 7.7 ps since it is the bin width. Also, the size of the tags in this architecture is 28-bits which implies each channel's adjustable digital delays and window sizes can be set between 7.7 ps to 2.06 ms. The FPGA logic utilisation of our system was recorded as follows: 11,636 Slices (50%), 24,926 Registers (13%), 3,329 Look-up-tables (LUTs) (36%) and 85 I/Os (100%).

The modified parts of our system since our last publication [16] will be discussed in this section. The architecture overview can be seen in Fig. 2.

2.1 Dual Data Rate Registration Time-To-Digital Converter

The TDC was implemented by using 512-bins of carry chain primitives located inside the Spartan6 LX150 FPGA, which are utilised for arithmetic operations. The system clock was set to run at 125 MHz which was generated by the Digital Clock Manager (DCM) IP of Xilinx from a 20 MHz low jitter clock on the breakout board. The system clock incremented the 18-bit coarse counter value every rising edge of the clock and the dead-time of the delay line was 8 ns, due to the clock speed.

Our original TDC scheme used a carry chain based tapped delay line which measures the trigger signal's relative position with respect to the clock's rising edge. With the occurrence of the clock's rising edge, the state of the delay line was captured with the D type flip flops which were connected to every bin of the delay line. These flip flops captured the thermometer code that represents the propagation of the trigger until the clock edge and this thermometer code were passed to the priority encoder to be converted from 511 bit code into 9 bit fine tag.

In addition to our previous paper, we have developed a new TDC scheme which aims to improve the resolution and the linearity of the TDC for precise measurements without altering the carry chain that is used or introducing any further dead-time. This method is named as the dual data rate registration TDC, which uses both of the clock edges to record the state of the 511 bin carry chain without extending the carry chain. This method was inspired by the double registration TDC [17]. However, it takes a more time-efficient approach by simply using both clock edges.

In order to implement such a scheme, two sets of D type Single Data Rate (SDR) Flip-Flops are used where one captures the thermometer code with the rising edge of the clock whilst the other works on the falling edge of the clock. The D Type Single Data Rate (SDR) Flip-Flop operating at
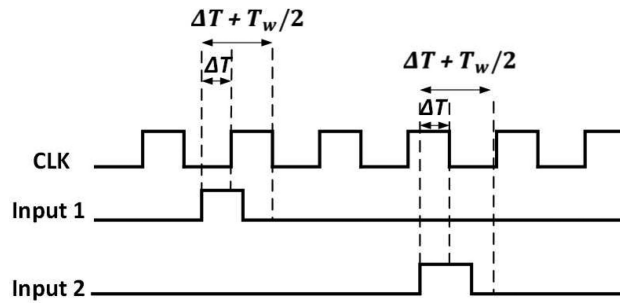
Fig. 3. Example of dual clock edge quantisation for inputs.

the falling edge captures the thermometer code with a signal which is 180° out of phase from the system clock. The code captured at the falling edge becomes available to use in the next clock cycle, since the system operates at the rising edge of the system clock. Once the thermometer codes are generated, the codes and the triggers are passed to the priority encoder to generate 9 bit codes for both 512 bit thermometer codes. To be able to use these generated codes, the clock region where the trigger signal occurred must be identified. This can be done by checking the value of the fine tag. If the input trigger falls into the clock low region (the region before the rising edge), the code generated with the rising edge should be less than the one generated at the falling edge. However, when the input trigger falls within the clock high region (the region before the falling edge), the falling edge's code is observed as smaller than the rising edge's. This is due to the fact that the trigger signal which is closer to the edge produces smaller fine tag ($\triangle$T) whilst the other one has an additional a half clock period ($T_{clk}/2$) to quantise ($\triangle$T + $T_{clk}/2$). In Fig. 3, the explanation of these cases can be seen.

After the correct detection of the clock region, the fine codes from both clock edges are averaged. The averaging operation essentially adds an extra bit to the fine tags and totals them. However, due to the different clock regions some corrections also need to be applied. When the trigger is detected in the clock high region, since the system operates with the rising edge of the clock, the code that is generated by the falling edge comes from the previous cycle, and the current cycle's code is generated by the rising edge that is being used. On the other hand, when the trigger is in the clock low region, the codes generated for both clock edges in the most recent cycle are being used. Another correction is due to the half clock period time difference between clock edges. Therefore, when the trigger is in the clock high region, 256 is subtracted from the average and if the trigger is in the clock low region, 256 is added to the average of the codes. The final code generated after the averaging has 10-bit code. This code is input to the calibration block for the next step of the operation. The additional bit has a smoothing effect by increasing the LSB resolution. For calibration, the implementation described in [18] was used. The last recorded bin that could be reached in the delay line during the 8 ns clock period was recorded as 405 with the new TDC design, which corresponds to bin 810 for the 10-bit code. Therefore, the average resolution of the delay line is $8\,ns/810 = 9.8\,ps$.

The least significant bit (LSB) of the code can be expressed as below:

$$LSB = \frac{T_{Clock}}{2^N} \tag{1}$$

where T_{Clock} is the clock period and N is the number of bits reserved for fine quantisation. When the 8 ns clock period and the 10 bit are substituted into the formula given above, the LSB of the TDC code can be calculated as below:

$$LSB = \frac{8 \times 10^{-9}\,s}{2^{10}} = 7.7\,ps \tag{2}$$

The explanatory diagram for the tag generation scheme can be seen in Fig. 4.
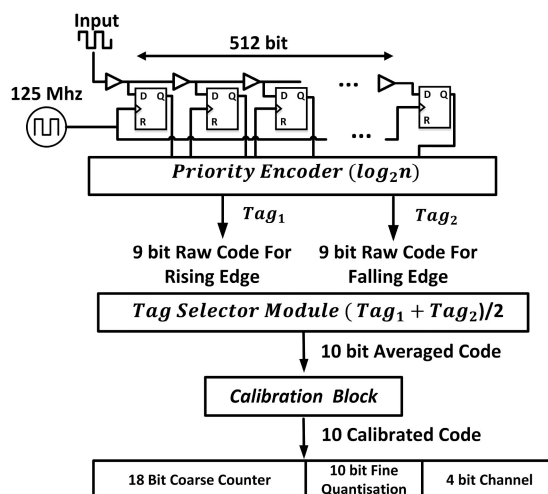
Fig. 4. The tag generation scheme.

It should be noted, that until the calibration is over, the system is kept in a soft-reset state which prevents tags from leaving the calibration block. Once the calibration is completed the soft reset is lifted. Except the additional bit added to the code, the calibration block is the same as the one we published in [18] and each channel has its separate calibration blocks. Overall, the proposed TDC implementation demonstrates a unique technique which uses two tags, one on each clock edge, in conjunction with the code density calibration and clock region corrections to achieve high resolution and precision in an FPGA.

2.2 Independent Coincidence Detectors

The tag correlation, which was described in [18], is followed by the coincidence detection logic. Coincidence detection is a process where the delta times are checked for being within the coincidence windows. The bits of the coincidence patterns are set by checking each tag with 8 independent coincidence detectors and, thus, each coincidence between the channels can be checked from the channel's perspective without being limited by the data serialisation. Each generated coincidence pattern is used as the address of the counter, which corresponds to the index of the coincidence counting block.

The coincidence algorithm forms coincidence patterns based on generated tags falling within a coincidence window between two trigger signals of the reference channels. Thus, the trigger on the reference channel acts as the reset for the coincidence pattern. The coincidence pattern is formed based on comparing the tags with the coincidence window, and when a tag is detected within the window, the bit corresponding to its channel is denoted by a logical one in the coincidence pattern, otherwise logical zero. The coincidence detector aims to find the largest coincidence fold possible and leaves the sub-fold finding to the post-processing step. The equation of coincidence pattern formation can be seen in equation 4.

$$V(n) = \begin{cases} 0 & \text{if } \Delta T_n \geq T_w \\ 1 & \text{if } \Delta T_n < T_w \end{cases} \tag{3}$$

2.3 Independent Coincidence Counting Blocks

Each independent coincidence counting block has its RAM block which holds the counter values for each channel. Each channel has a 511 deep and 32 wide BRAM, which accommodates the counter for all possible coincidence patterns. When a coincidence pattern is detected, the coincidence
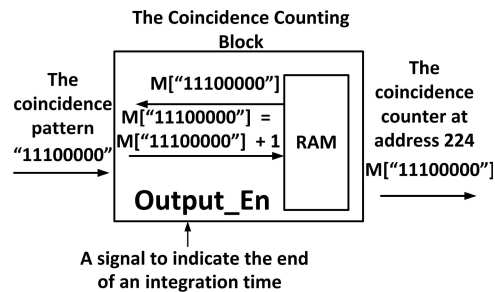
Fig. 5. Illustration of a coincidence counting block.

pattern is used as the index of the RAM block, and the counter located in the address is read and incremented by one.

Having independent coincidence counting blocks makes the coincidence counting independent from other channels, and the data serialisation is no longer required, unlike our previous work in [16]. This essentially makes the data rate of our coincidence counting operation independent from the channel number, and thus the coincidence counting scheme becomes scalable. This scalability differentiates the presented scheme from the other mentioned examples in the literature. The coincidence counting operation is run for the operation time which is set to 60 ms. When the operation time is finished, the results are sent out, and the contents of the BRAM are cleared. A block diagram for the coincidence counting block can be seen in Fig. 5, where M represents the RAM block's memory and the index of the M represents the coincidence address.

3. Results and Discussion

Our coincidence counting system was tested in two different aspects of the design. The first aspect of our results was based on the performance of the TDC in terms of linearity and precision. The second part was the performance of the coincidence counting operations. The coincidence counting system was also tested in quantum photonics experiments. The results obtained in this research will be discussed in this section.

3.1 The Timing Jitter Measurement

The timing jitter measurement is a test which is concluded to determine the precision of the TDC. From the results, the RMS resolution and Full Width Half Maximum (FWHM) of the measurements can be calculated. In this test, two TDC channels were fed by the same input signals where the delay between them was the routing delays. From the measured peak's standard deviation, the RMS resolution could be calculated.

For this test, two 3.12 MHz signals were generated by the Xilinx DCM from the 100 MHz Low Jitter Clock Oscillator located on the Opal Kelly Board and roughly 20000 tags were generated for each test. As a result, a histogram of the time difference was plotted. The result of this measurement can be seen in Fig. 6. The RMS resolution was measured as 8.9 ps, while the standard deviation was 12.6 ps and the FWHM was 29.6 ps. It should be noted that there is a significant improvement in the RMS resolution and the Full-Width Half Maximum (FWHM) after the linearisation method. In Fig. 6, a is the timing jitter plot for the original TDC before the linearisation, b is the timing jitter measurement after the double registration linearisation method. In Fig. 6(b) almost 3 ps improvement in RMS resolution and 10 ps in FWHM can be seen. It should be noted since both devices had different routing delays between channels; the delay offsets between them were slightly different. In Fig. 6(b), the routing delay was observed roughly 200 ps more than the one in Fig. 6(a).
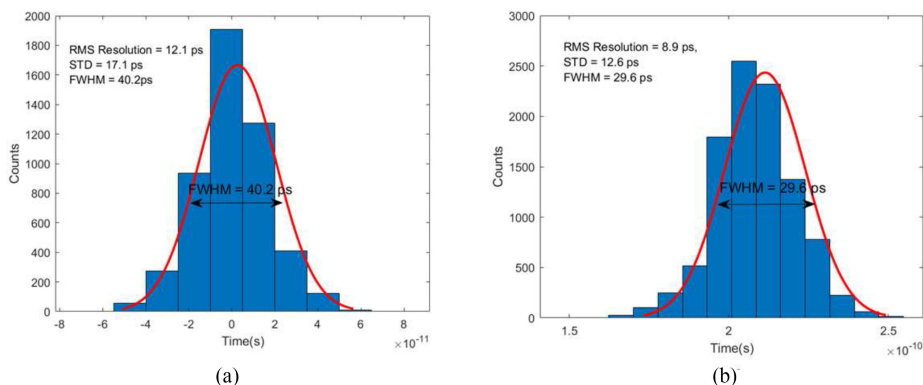
Fig. 6. The timing jitter measurements, where a is before the linearisation and b is after the linearisation.

3.2 TDC Linearity

Another measure of the TDC performance is the linearity of the input against the output. Ideally, every bin in the delay line should have an equal number of hits due to the equal sized delay elements. However, this is never achieved because of the non-linearity issues caused by many factors mentioned before. DNL and INL is used for measuring the linearity of the TDC. As we have previously discussed in [18], the linearity of a TDC is typically affected by the inconsistent routing delays between FPGA slices, temperature fluctuations and power fluctuations. The non-linearity appears as very large bins or missed bins. DNL error is a parameter which defines how much each bin deviates from the ideal step (1.249 LSB), whilst INL error defines how much each code deviates from the ideal transfer function ($y = 1.249x$). As it was discussed in [19], the code density calibration can be used for calibrating the INL of a TDC based on the DNL affecting each bin. In this implementation, we apply the Dual Data Rate Registration to improve the DNL of the code before the calibration. The improved max DNL of the code was observed as 2.9 LSB, while it was 4.1 LSB before the linearisation. The DNL improvement is also reflected on empty bin numbers in the code. The number of empty bins measured before the calibration was 208, and after the calibration, this number increased to 222 due to the stretch of the transfer function from 405 to 511 code representations. However, after the Dual Data Rate Registration was applied, the total number of empty bins was measured as 93 bins out of 511 bins. The INL was also measured as a part of this experiment. The maximum INL error observed before the calibration was −31.8 LSB. After the code density calibration, the maximum INL was 10.64 LSB, and with the calibration and linearisation combined, the maximum INL error was measured as 8.8 LSB. Thus, around 1.8 LSB max INL error improvement was observed.

The results obtained from linearity tests can be seen from Fig. 7, where Fig. 7(a) is the INL comparison between non-calibrated, calibrated and linearised codes, and Fig. 7(b) is the DNL comparison between linearised and non-linearised codes. It should be noted that the linearised code also uses the code density calibration method whilst the calibrated code only uses code density calibration.

3.3 The Benchmark Coincidence Counting Performance

To benchmark, the coincidence counter, a test of the coincidence rate as a function of the delay was conducted. In this test, two identical test signals were generated by using a signal generator (B&K Precision BK4054B), a splitter and a delay box (Ortec DB463 Delay Box). The splitter splits the signal into the two channels of the delay box, and the identical signals with a fixed delay between each other were generated. Both signals were running at 1 MHz which implies each channel generated 60000 tags since the operation time of the system was set to 60 ms. The fixed
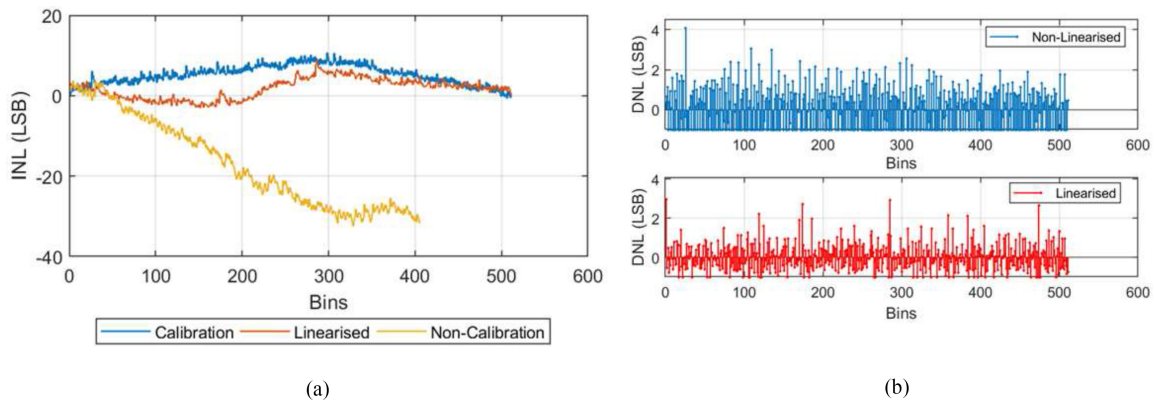
Fig. 7. (a) The INL before and after the Dual Data Rate Registration, (b) The DNL before and after the Dual Data Rate Registration linearisation.
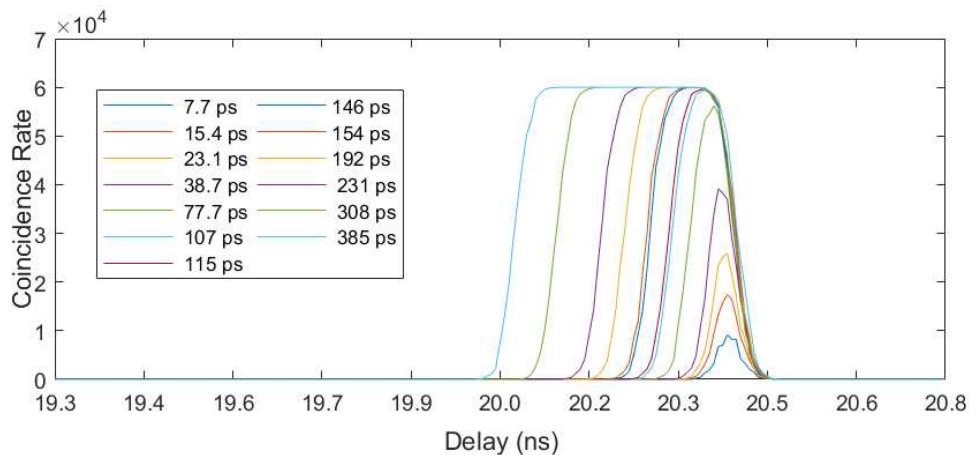

Fig. 8. The coincidence rate as a function of the delay with window size 380 ps - 7.7 ps.

delay was set around 20 ns but, due to routing and cable lengths, it was observed to be slightly more than 20 ns.

As a result of this test, the system was proved to be working since the highest coincidence rates were observed at the same offset delay of 20.4 ns with different windows sizes. However, 107 ps was the last coincidence window where all the excepted coincidences were captured. Hence, 107 ps was the point where the last saturation of coincidence rates was observed. In Fig. 8, the coincidence rates for window sizes between 7.7 and 380 ps can be seen.

The second delay test was conducted to show the coincidence counter's functionality with large window sizes. As can be seen from Fig. 9, the coincidence rates are all saturated at 60000, and the width of the pulses is roughly equal to the set coincidence window sizes which shows that our system also performs as expected with larger coincidence windows.

3.4 Coincidence Counting in Quantum Photonics Applications

To demonstrate the suitability of the counting logic module for multi-photon experiments, we implemented it in the detection scheme of quantum linear optical experiments. In the experiments, superconducting nanowire single-photon detectors (SNSPDs) are used, since they are very high-performance detectors, and generate very clear detection signals. In particular, they produce very
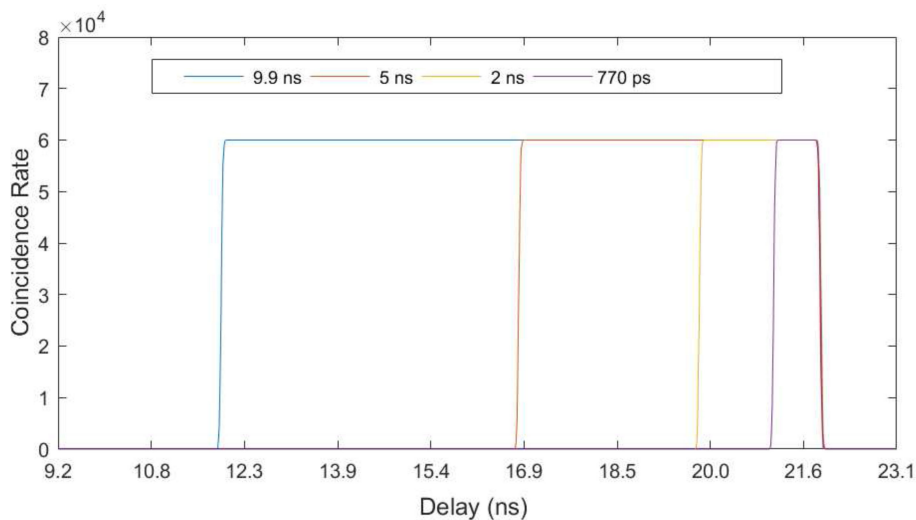
Fig. 9. The coincidence rate as a function of the delay with window size 770 ps - 9.9 ns.

low-jitter electronic signals, which are relevant in our tests, given the high time precision of the electronics. The high efficiency and low dead-time allowed us to reach very high count rates in our experiments, which was important to test the counting rate of the module. More generally, they are the best commercially available single photons detectors, and thus, they are the most used detectors in multi-photon quantum optics experiments. As a first benchmark, we used it for the detection of quantum interference occurring in the reversed Hong-Ou-Mandel (rev-HOM) effect [20], [21]. In this experiment, the quantum photonics principles of two photon-interference, on-chip photon generation and manipulation were utilised. As represented in Fig. 10(a), the rev-HOM effect is a two-photon interference effect given by the time-reversal of the standard Hong-Ou-Mandel effect [22]. Here, a bunched photon pair is prepared in a superposition of two-photon Fock states $(|20\rangle + |02\rangle)/\sqrt{2}$, and then an optical phase ϕ is inserted on one of the modes before injecting them into a balanced 50 : 50 beam-splitter. A pulsed laser with 2 nm bandwidth, 50 MHz repetition rate and 1 mW average power were used for this experiment. These values were chosen in order to achieve 2500 Hz coincidence rate at the peak of the fringes in Fig. 10(a), corresponding to a power of a 0.6 fW. Thus, 0.6 fW/2500 CPS = 0.24 aJ per coincidence was obtained. This is a 200 dB attenuation from the 1 mW/50 MHz = 20 nJ per laser pulse. Two single-photon detectors and the counting logic unit are finally required to measure the quantum state emerging from the interferometer. In our experiment, the photon-pair generation and the interferometer were integrated on a single silicon-photonic chip using standard silicon quantum photonics components [8], [23]. Two spontaneous four-wave mixing waveguide sources were coherently pumped to generate the superposition state, and a thermal phase shifter and a 2×2 multi-mode interferometer were used to implement the phase shift and the beam-splitter, respectively.

After the interference, photons were coupled off-chip into optical fibres and sent to the detection scheme. In the measurement, two superconducting nanowire single-photon detectors (SNSPDs) [24] were employed (PhotonSpot) with approximately 85% efficiency, 100 Hz dark counts, 50 ns dead-time and 70 ps internal jitter. After the amplification, the electronic signal emitted from each detector upon photon detection was given by an exponential decay shape with a peak of approximately 0.8 V (see inset of Fig. 10(a)). The electronic signals from the two SNSPDs were sent into two input channels of the counting logic unit, set to a threshold of 0.1 V, which then counted the simultaneous events within a coincidence window of 2 ns. To observe the rev-HOM quantum interference, we continuously monitored the coincidence events rate while scanning the phase between 0 and 2π. The results are shown in Fig. 10(b). The measured rev-HOM quantum
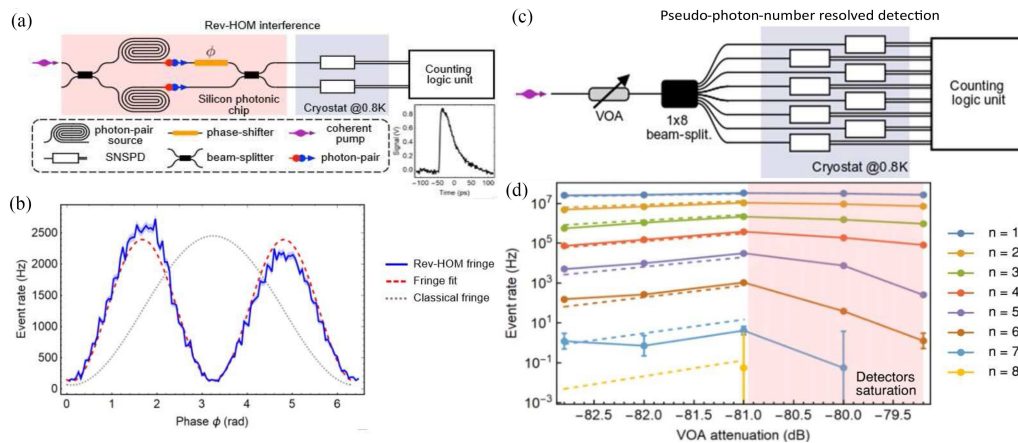
Fig. 10. Quantum photonics experimental benchmarks of the counting system. (a) Schematic of the rev-HOM quantum interference experiment. An integrated circuit on a silicon chip is used to generate a photon-pair in a quantum superposition and perform the quantum interference, while the measurement is performed off-chip using SNSPDs and the counting system. Inset: electronic signal emitted by the detectors upon photon detection. (b) Results of the rev-HOM experiment, the solid blue line represents the measured coincidence rates, which form an interference fringe with a 90% visibility as measured by the fitted curve (red dashed line). The black dashed line is the classical fringe obtained with classical light rather than single photons. (c) Schematic of the pseudo-photon-number resolving measurement on a coherent state. (d) Results of the pseudo-photon-number resolving experiment for different amplitudes of the weak-coherent state. Marks with different colours refer to data for different n-fold coincidence events, up to $n = 8$ photons. Dashed lines are obtained by fitting the data to the expected rates from a coherent state, assuming uniform channel and detector efficiencies. The shaded area represents the region where the SNSPDs start to saturate. Error bars are at one standard deviation confidence intervals, calculated from Poissonian counting statistics (not shown when smaller than markers).

interference fringe, which presents the typical doubled frequency when scanning ϕ with respect to the classical fringe [20], is observed with visibility of 90%, consistent with previous measurements on the same device.

We further tested the capability of the device to support high-count rates for multi-photon experiments by using it to perform pseudo-photon-number resolving detection of a coherent state of light with up to eight-photon resolution. In this experiment, the quantum photonics principle of photon counting from coherent states was used, which is one of the basic principles of quantum information processing [25]. This operation was implemented by injecting weak-coherent pulses into a 1×8 fibre beam-splitter, which probabilistically splits the photons of the coherent state of light into the 8 output channels measured via 8 SNSPDs (see Fig. 10(c)). The laser specifications of the second experiment were the same as the first experiment. Correlation measurements between the detection events from the 8 SNSPDs were performed using the 8 input channels of the counting logic unit. The detector delays were calibrated with respect to the channel 1 by adding delays to other channels and sweeping results within -10 ns to 10 ns range for measuring the 2-fold coincidences between channel 1 and the others. This can be seen in Fig. 11. A variable optical attenuator (VOA) was used to tune the amplitude of the coherent state, which allowed us to measure the output photon distributions for various amplitudes. The results are shown in Fig. 10(d). The counting logic unit can support more than 50 MHz singles event rate over all channels and was used to measure up to 8-photon coincidences. The measured count rates are limited only by the saturation level of the SNSPDs, which start to saturate at a few MHz counts, rather than the capability of the counting electronics. This demonstrates that the processing capability of the counting logic unit surpasses the detection rate capacity of the single-photon detectors, showing its suitability for many-photon experiments where high-count rates over many channels are required [4].
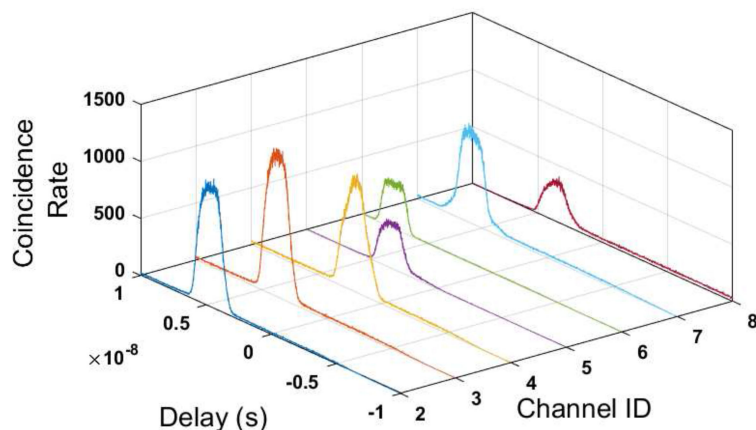
Fig. 11. The detectors' delay calibration with respect to the reference channel (channel 1).
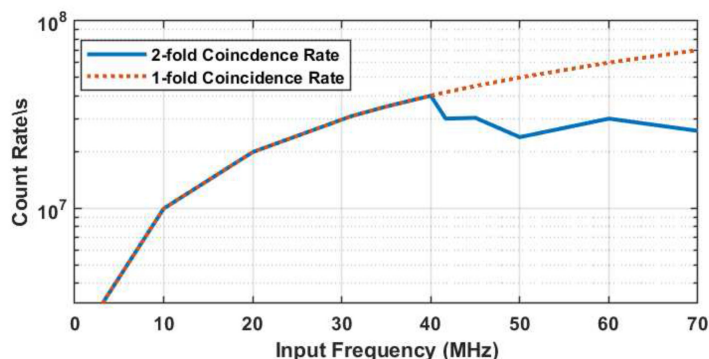

Fig. 12. Coincidence rates (singles and 2-fold) vs input frequency.

In addition, to investigate the maximum count rate of the system, the highest 2-fold coincidence rate that can be detected per channel was investigated. This measurement aimed to find the last measurable count rate, where the measurements were still accurate. In order to do this, two identical signals were generated with the Xilinx DCM module varying from 3.12 MHz to 70 MHz and split into the two channels. Since the signals were the same, the 2-fold coincidence rates should have been equal to the singles rate for a channel. The window was set to 2 ns, which was large enough to compensate for any routing delays. As was mentioned in [18], typically the clock period is the dead time of the TDC core, which was 8 ns in our case. However, the coincidence counter core requires 3 clock cycles to accumulate a coincidence into a BRAM, resulting in a maximum count rate of 41.67 MCPS overall for the system.

As 3 clock cycles imply 24 ns is the dead-time of the system. However, due to cycle-to-cycle jitter in the source repetition rate and system clock, we begin to observe a break-down after 40 MHz. This is because the system requires 24 ns (T_{clk} * 3 = 8 ns * 3) $\pm$ clock jitter to accumulate the coincidence, but the source is repeating at 25 ns $\pm$ source jitter. If the most negative source jitter + the most positive clock jitter exceeds 1 ns (e.g: clock jitter = +200 ps and source jitter = −1 ns, overall 1.2 ns), then a coincidence will be lost. Thus, at 41.67 MHz, around 30% of coincidences, at 45 MHz, 35% of coincidences and at 50 MHz, 50% of coincidences are dropped. Between 41.67 MHz source repetition rate (24 ns = 3 clock cycles) and 62.5 MHz (16 ns = 2 clock cycles), we expect to see the coincidence detection efficiency drop from 100% to 50% (at 2 cycles/tag, every other tag will be dropped since there will be no instances of 3 clock cycles between two tags). This expected behaviour was observed with the measurements taken after 45 MHz, and

TABLE 1

Performance Parameters of the Proposed Coincidence Counting System and Existing TDC Based Coincidence Counters

Performance Parameters	This Work	IDQuantique ID900[10]	PicoQuant Hydra-harp400[11]	[12]	[13]	[26]
Total Count Rate	320 MCPS	100 MCPS / 400 MCPS	100 MCPS	72 MCPS	0.5 MCPS	32 MCPS
Resolution	7.7 ps	13 ps / 100 ps	1 ps	68.3 ps	390 ps	0.7 ns
Single-Shot Precision	8.9 ps	8 ps	8.5 ps	N/A	N/A	0.7 ns
Slices	11,636 (50%)	N/A	N/A	66%	2257 (99%)	N/A
Registers	24926 (13%)	N/A	N/A	94%	6051 (33%)	N/A
I/O	85 (100%)	N/A	N/A	89%	153 (65%)	N/A
Channel No:	8	4	8	12	32	64
DNL	2.9 LSB	N/A	2% peak	N/A	1 LSB	N/A
INL	8.8 LSB	N/A	N/A	N/A	N/A	N/A

the count rates were observed to be half of the expected theoretical rates until 60 MHz (i.e. 50 MHz = 25 MCPS, 60 MHz = 30 MCPS etc.). However, at 70 MHz (14.2 ns) since it is less than 2 clock cycles (16 ns), only 1 count could be detected for every 24 ns (i.e. 70 MHz = 23.3 MCPS). These results can be seen in Fig. 12, where the red dashed line is the measured single rates (theoretical coincidence rate), and the blue line is the measured 2-fold coincidence rates.

4. Conclusion

In conclusion, we have developed an FPGA based coincidence counting system for quantum photonic applications which differs from other timing instruments by providing a very high count-rate of 320 MCPS with 8 operational channels and providing high precision of 8.9 ps RMS resolution. This system potentially provides a solution to the need for a coincidence counting instruments with high precision and high-count rates over many channels to account for the recent developments in the area of quantum photonics.

The fast count rate of the system is provided by the complete integration of coincidence counting and TDC modules into the same FPGA fabric in a channel independent and concurrent manner. Thus, no FIFO or buffering was required in the design. The TDC itself provided 8.9 ps RMS resolution with an FWHM of approximately 29.6 ps while the minimum bin width was 7.7 ps. Also, the TDC had a Dual Data Rate Registration scheme which improves the linearity through averaging by using both clock edges to capture the state of the delay line. Hence, no further dead-time or the additional delay line logic was added.

The system has been tested with some quantum photonics experiments to prove its functionality. These tests were the detection of quantum interference by measuring the rev-HOM Dip Effect and the pseudo-photon-number resolving detection of a coherent state of light with eight-photon resolution. These tests showed that our system has some potential of being used in quantum applications where a high count rate is required over many channels. Also, the highest count rate that can be input on a single channel without compromising the accuracy was measured as 40 MCPS which corresponded to 320 MCPS for 8 channels. A comparison of the performance parameters of the proposed design with other important TDC-based coincidence counting systems can be seen in Table 1.

In the future, our research will focus on increasing the number of channels and improving the resolution of the TDC. This future work is likely to involve the migration of the system into a newer FPGA chip since the Spartan 6 architecture is slowly becoming obsolete. Moreover, a compression scheme for the RAM blocks used in the coincidence counting block will be researched due to

the exponential increase in the usage of memory whenever a new channel is introduced into the design. Also, optimisation of the coincidence counting module to improve the RAM read and write throughput will be considered to improve the count rate per channel. Finally, the TDC research presented in this work highlighted improvements to the quantisation noise which improved the precision. Therefore, as future work, analysing and improving other noise sources such as clock jitter, temperature and voltage noise should be considered.

Acknowledgment

References

[1] A. Aspuru-Guzik and P. Walther, "Photonic quantum simulators," *Nature Phys.*, vol. 8, no. 4, pp. 285–291, 2012.
[2] S. Pirandola *et al.*, "Advances in quantum cryptography," 2019. [Online]. Available: https://arxiv.org/abs/1906.01645.
[3] S. Aaronson and A. Arkhipov, "The computational complexity of linear optics," in *Proc. 43rd Annu. ACM Symp. Theory Comput.*, 2011, pp. 333–342. [Online]. Available: https://dl.acm.org/citation.cfm?id=1993682
[4] S. Paesani *et al.*, "Generation and sampling of quantum states of light in a silicon chip," *Nature Phys.*, vol. 15, pp. 925–929, 2019.
[5] S. Tancock, E. Arabul, and N. Dahnoun, "A review of new time-to-digital conversion techniques," *IEEE Trans. Instrum. Meas.*, vol. 68, no. 10, pp. 3406–3417, Oct. 2019.
[6] A. O. C. Davis, P. M. Saulnier, M. Karpiski, and B. J. Smith, "Pulsed single-photon spectrometer by frequency-to-time mapping using chirped fiber bragg gratings," *Opt. Express*, vol. 25, no. 11, 2017, Art. no. 12804.
[7] B. T. Gard, K. R. Motes, J. P. Olson, P. P. Rohde, and J. P. Dowling, "An introduction to boson-sampling," 2014. [Online]. Available: https://arxiv.org/abs/1406.6767
[8] J. Wang *et al.*, "Multidimensional quantum entanglement with large-scale integrated optics," *Sci.*, vol. 360, pp. 285–291, 2018. [Online]. Available: http://science.sciencemag.org/content/360/6386/285
[9] P. Eraerds, M. Legre, A. Rochas, H. Zbinden, and N. Gisin, "SiPM for fast photon-counting and multiphoton detection," *Opt. Express*, vol. 15, no. 22, 2007, Art. no. 14539.
[10] I. Quantique, "ID900 time controller," 2017. [Online]. Available: https://www.idquantique.com/single-photon-systems/products/id900-time-controller
[11] PicoQuant, "Home," 2014. [Online]. Available: https://www.picoquant.com/products/category/tcspc-and-time-tagging-modu les/hydraharp-400-multichannel-picosecond-event-timer-tcspc-module
[12] C. Wang *et al.*, "A real time coincidence system for high count-rate TOF or non-TOF PET cameras using hybrid method combining and-logic and time-mark technology," in *Proc. 16th IEEE-NPSS Real Time Conf.*, 2009, pp. 321–325.
[13] W. Li *et al.*, "Time-tagged coincidence counting unit for large-scale photonic quantum computing," *Revi. Sci. Instrum.*, vol. 89, no. 10, 2018, Art. no. 103113.
[14] M. Tetrault *et al.*, "Real-time coincidence detection system for digital high resolution APD-based animal PET scanner," in *Proc. IEEE Nucl. Sci. Symp. Conf. Rec.*, 2005, pp. 2849–2853.
[15] R. Singh, "FPGA vs ASIC: Differences between them and which one to use?" Jul. 2018. [Online]. Available: https://numato.com/blog/differences-between-fpga-and-asics/
[16] E. Arabul, J. Rarity, and N. Dahnoun, "FPGA based fast integrated real-time multi coincidence counter using a time-to-digital converter," in *Proc. 7th Mediterranean Conf. Embedded Comput.*, 2018, pp. 1–4.
[17] J. Wu and Z. Shi, "The 10-ps wave union TDC: Improving FPGA TDC resolution beyond its cell delay," in *Proc. IEEE Nucl. Sci. Symp. Conf. Rec.*, 2008, pp. 3440–3446.
[18] E. Arabul, A. Girach, J. Rarity, and N. Dahnoun, "Precise multi-channel timing analysis system for multi-stop lidar correlation," in *Proc. IEEE Int. Conf. Imag. Syst. Techn.*, 2017, pp. 1–6.
[19] M. W. Fishburn and E. Charbon, "Time-to-digital converters for pet: An examination of metrology aspects," in *Proc. IEEE Nucl. Sci. Symp. Med. Imag. Conf. Rec.*, Oct. 2012, pp. 839–840.
[20] J. W. Silverstone *et al.*, "On-chip quantum interference between silicon photon-pair sources," *Nature Photon.*, vol. 8, no. 2, pp. 104–108, 2014. [Online]. Available: https://www.nature.com/articles/nphoton.2013.339
[21] A. E. Ulanov, I. A. Fedorov, D. Sychev, P. Grangier, and A. Lvovsky, "Loss-tolerant state engineering for quantum-enhanced metrology via the reverse Hong–Ou–Mandel effect," *Nature Commun.*, vol. 7, 2016, Art. no. 11925.
[22] C. K. Hong, Z. Y. Ou, and L. Mandel, "Measurement of subpicosecond time intervals between two photons by interference," *Phys. Rev. Lett.*, vol. 59, pp. 2044–2046, 1987. [Online]. Available: https://link.aps.org/doi/10.1103/PhysRevLett.59.2044
[23] D. Bonneau, J. W. Silverstone, and M. G. Thompson, *Silicon Quantum Photonics*. Berlin, Germany: Springer, 2016, pp. 41–82.
[24] W. H. Pernice *et al.*, "High-speed and high-efficiency travelling wave single-photon detectors embedded in nanophotonic circuits," *Nature Commun.*, vol. 3, no. 1325, pp. 1–10, 2012.
[25] J.-P. Gazeau, "Coherent states in quantum information: An example of experimental manipulations," *J. Phys.: Conf. Ser.*, vol. 213, Jan. 2010, Art. no. 012013.
[26] M. A. Tetrault *et al.*, "Real time coincidence detection system for digital high resolution apd-based animal pet scanner," in *Proc. IEEE Nucl. Sci. Symp. Conf. Rec.*, Oct. 2005, vol. 5, pp. 2849–2853.