1-18-2011

Note: Scalable Multiphoton Coincidence-counting Electronics

David Branning
Trinity College, david.branning@trincoll.edu

Sarthak Khanal
Trinity College

Young Ho Shin
Trinity College

Brandon A. Clary
Trinity College, brandon.clary@trincoll.edu

Mark Beck
Whitman College

Note: Scalable multiphoton coincidence-counting electronics

D. Branning,[1] S. Khanal,[1] Y. H. Shin,[1] B. Clary,[1] and M. Beck[2]
[1]*Department of Physics, Trinity College, 300 Summit St., Hartford, Connecticut 06106, USA*
[2]*Department of Physics, Whitman College, Walla Walla, Washington 99362, USA*

(Received 30 September 2010; accepted 14 November 2010; published online 18 January 2011)

We present a multichannel coincidence-counting module for use in quantum optics experiments. The circuit takes up to four transistor–transistor logic pulse inputs and counts either twofold, threefold, or fourfold coincidences, within a user-selected coincidence-time window as short as 12 ns. The module can accurately count eight sets of multichannel coincidences, for input rates of up to 84 MHz. Due to their low cost and small size, multiple modules can easily be combined to count arbitrary *M*-order coincidences among *N* inputs. © *2011 American Institute of Physics*. [doi:10.1063/1.3524571]

Coincidence counting is the simultaneous detection of two or more particles at different detectors. While this technique is widely used in experimental physics, it plays an especially important role in quantum optics. The coincidence counting of photons is an essential tool for exploring and/or exploiting the nonclassical features of correlated light sources. Many such experiments require only sets of twofold coincidence measurements, while for others, it is necessary to count multiphoton coincidences among many detectors.[1–3]

Historically, the most common method of coincidence counting has used time-to-amplitude converters (TACs), with each TAC adding the capability to count one more pair of photons in coincidence. Multiphoton or multichannel coincidence counting quickly becomes cumbersome and expensive this way, and the maximum coincidence-counting rate is limited by the conversion time required for each start/stop event, typically ∼1 μs. In recent years, several solutions to these problems have evolved for particular applications, including quantum information processing,[4–6] fluorescence measurements,[7,8] x-ray microscopy,[9] and physics education.[10,11]

Here, we present the details of a new multichannel coincidence-counting module (CCM) that can be built for less than $600 with off-the-shelf integrated circuit components. Starting with up to four transistor–transistor logic (TTL) signals as inputs, the CCM can register combinations of arbitrary twofold, threefold, or fourfold coincidences (or singles counts), with a coincidence window as short as 12 ns. Eight onboard registers, programmed into a field programmable gate array (FPGA), count the user-defined coincidences for time intervals of between 20 μs and 1 s. The count data are transferred to a personal computer over a universal serial bus (USB) interface, where the counts are collected, integrated, displayed, and stored to disk via freely available software.[12]

In order to improve the coincidence-time resolution, each of the detector signals first enters a pulse-shaping circuit that reduces its width from the 20–50 ns pulse width typically obtained from commercial single photon counting modules (SPCMs).[11] Toggle switches are used to select the width of the shaped pulses of all four inputs, or to bypass the pulse-shaping circuit, leaving the pulse widths unchanged. The shaped pulses have selectable durations of 7.5, 9.0, or 11.5 ns (±0.5 ns, measured full width at half maximum).

The basic coincidence-determination method of the CCM is shown in Fig. 1. The shaped signals A, B, C, and D are sent to OR gates, and then to the inputs of a four-way AND gate. The output of the AND gate is true if and only if all four inputs are simultaneously true—that is, if the four detector pulses arrive at the gate at the same time.

The OR gates allow the user to define arbitrary subsets of the four detector signals to be counted in coincidence. The second input of each OR gate is held high or low, as selected by the user with a switch. When the switch for any particular input is high, that input is effectively removed from the coincidence logic. Any inputs with their corresponding switches held low, however, must still arrive simultaneously in order for the AND gate's output to be true. In this fashion, the output of the AND gate can determine any combination of two-, three-, or fourfold coincidences between the four inputs, or simply deliver the single-channel input rate of any one input (by excluding the other three). There are eight four-input AND gates, and the output of each is sent to the input of a counter, which is implemented on the FPGA. Each counter regularly delivers its recorded number of counts to a personal computer (PC) over a USB interface, and then resets to continue counting.

The switches connected to the OR gates that determine which coincidences are counted are latching pushbuttons, with an embedded orange (590 nm) light-emitting diode (LED). When a switch is depressed, the center pole is connected to ground and the LED is lit, indicating that the corresponding input is included in the four-way AND logic. The switches are arranged in a 4 × 8 grid.[13] The four rows correspond to the four inputs, and the eight columns correspond to the eight counters. In this way, the user can very easily set (and observe) which coincidences are being registered by which counter.

In addition to being sent to the FPGA, the output of each four-way AND gate is also connected to a line driver and a BNC output, providing TTL output pulses which can be monitored externally. By using these output pulses as the inputs to additional CCMs, coincidences among an arbitrarily high number of inputs can be monitored.

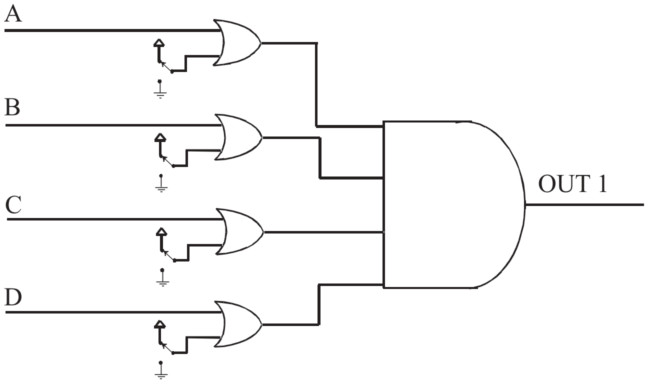

FIG. 1. Four-way AND gate with OR gates on each input. For each input (A, B, C, D), a switch connects one of the OR inputs to 0 or 5 V so that the input is either included (0 V) or excluded (5 V) from the logic at the AND gate.

A block diagram is shown in Fig. 2 (a full schematic of the entire circuit is available online).[13] In addition to the coincidence logic, a TTL clock signal is provided at a BNC output by dividing the FPGA's 50 MHz oscillator down to a user-selectable rate from 10^7 to 1 Hz in decades.

The coincidence determination (OR and AND gates) is implemented using F-series 5V TTL logic. The FPGA and USB capabilities are provided by an 80-pin MORPH-IC module from Future Technology Devices International (FTDI), which contains an Altera Acex 1 K FPGA and a USB interface with FTDI's FT2232D first-in-first-out (FIFO) buffer. The FPGA is configured by flashing a compiled VHDL program onto it. The program creates eight independent counting registers from cells in the FPGA, with 16 bits in each channel register, respectively. The number stored in each counting register is incremented on the leading edge of each TTL pulse from the four-input AND gate. After a user defined counting time (20 μs to 1 s) has elapsed, the value in each counting register is copied to a storage register, and the counting registers are reset to zero. While the counting registers begin incrementing again, the storage register values are written into the FIFO buffer. After a predefined number of storage values are written to the buffer, they are transferred in a block to an array in the computer random access memory (RAM) via USB. The sets of count values in this array are then integrated for a user defined time interval, displayed on the computer monitor, and/or stored to hard disk. All of these tasks, as well as the automatic loading of the VHDL program onto the FPGA, are accomplished by freely-available LABVIEW software.[12]

The transfer of the counting register values to the storage registers occupies one cycle of the FPGA's 50 MHz master oscillator; during this 0.2 μs time interval, the counting registers cannot be incremented, and are therefore "blind" to the arrival of any new TTL pulses. One such "blind cycle" will occur after each counting time bin has elapsed; thus, for an elapsed time T, the true duration of active data acquisition time is $T_{\text{active}} = T[1 - R/(50\,\text{MHz})]$, where R is the (user-selected) rate of data acquisition. The available values range from $R = 1$ Hz to 50 kHz.

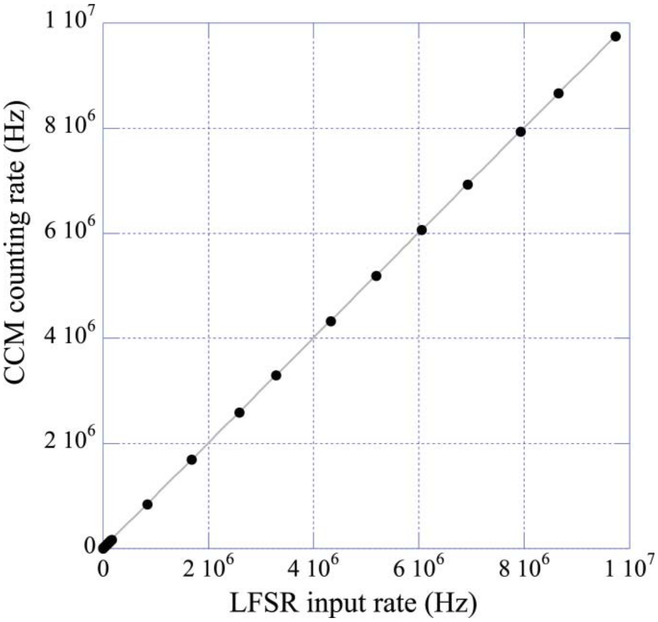

FIG. 3. (Color online) Mean single-channel counting rate in the CCM vs mean input pulse rate from an LFSR, acquired during 10-s intervals. The pseudorandom input pulses were counted independently with external 50-MHz counters. A least-squares fit (solid line) of the form $y = mx$ yielded $m = 1.002 \pm 0.002$. Similar results were observed in the other input channels.

The CCM was tested with a TTL pulse generator and was able to count coincidences at frequencies of up to 37 MHz without losses. To achieve this, the pulse generator was phase-locked to the FPGA clock with the 10 MHz clock output, and a phase offset was added to prevent input pulses from coinciding with the blind cycles. Above 37 MHz, the blind cycles could not be avoided, and exactly R counts per second were missing from the totals. Above 74 MHz, exactly $2R$ counts per second were missing. The total remained stable up to 84 MHz; above this input rate, the coincidences fluctuated and ultimately fell to zero at 147 MHz, as successive pulses overlapped within the rise/fall times of the AND gates.

To test the scalability with multiple modules, the phase-locked pulses from the generator were fanned out to eight

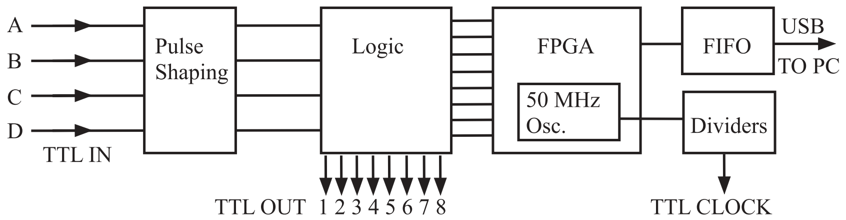

FIG. 2. Block diagram of the CCM architecture. Each input has a selectable impedance of 50 Ω or 1 kΩ. The input pulses are shortened and then fanned out to form the inputs to eight copies of the circuit in Fig. 1. The eight output channels are sent to BNC outputs, and also to the counting registers on the FPGA.

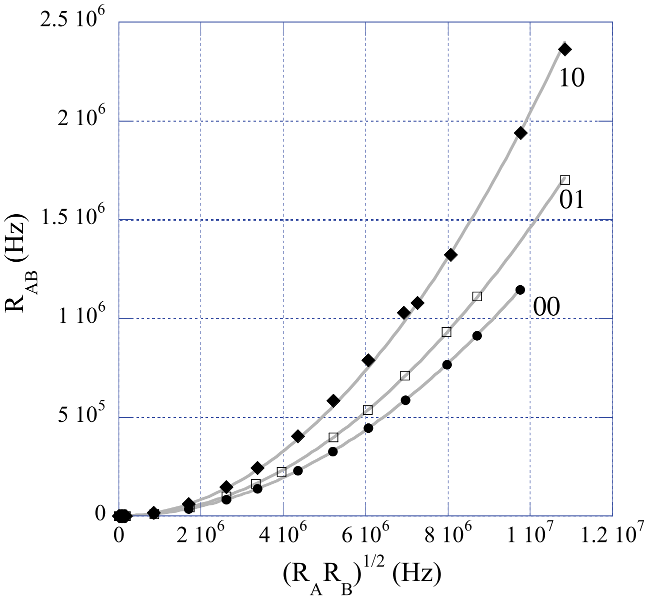

FIG. 4. (Color online) Coincidence rates R_{AB} in the CCM for pseudorandom input rates R_{A} and R_{B} on channels A and B, as a function of $x = \sqrt{R_{\mathrm{A}} R_{\mathrm{B}}}$ for various pulse width settings. A least-squares fit (solid line) to the parabola $y = \tau_c x^2$ yielded coincidence times $\tau_c = 12.033 \pm 0.006$, 14.56 ± 0.02, and 20.38 ± 0.09 ns for settings 00, 01, and 10. Similar results were observed for coincidences among the other input channels.

copies and delivered to the inputs of two CCMs. The four-way coincidence output from each CCM was fed to an input of a third CCM, which then registered up to eightfold coincidence counts at a rate of up to 30 MHz (the limit of the fan-out), without losses.

The CCM was also tested with pulses from a linear feedback shift register (LFSR), which generated a pseudorandom binary TTL output with controllable mean rates of up to 10 MHz. Figure 3 shows the single-channel response of the CCM for all four input channels. The CCM is observed to precisely count the input pulses, all the way up to the maximum output rate of the LFSR.

The coincidence times were measured using two independent LFSRs on pairs of inputs. For randomly arriving pulses with mean rates R_{A} and R_{B} in inputs A and B, the coincidence rate R_{AB} is given by

$$R_{\mathrm{AB}} = \tau_c R_{\mathrm{A}} R_{\mathrm{B}}, \tag{1}$$

where τ_c is the coincidence time, equal to twice the pulse duration τ minus a small amount necessary for sufficient overlap.[14] Single-parameter fits to the data, as shown in Fig. 4, yielded values of $\tau_c = 12.033 \pm 0.006$, 14.56 ± 0.02, and 20.38 ± 0.09 ns for the pulse-shaping toggle-switch positions 00, 01, and 10. The coincidence times were also measured using two SPCMs and scattered light from a laser (which should produce independent random streams of photons at the two detectors), yielding values of $\tau_c = 12.140 \pm 0.007$, 14.133 ± 0.008, and 21.47 ± 0.014 ns via Eq. (1). These coincidence times differ slightly from those measured with the LFSRs, due to differences in the input pulse heights and shapes from the SPCMs. The measured values of τ_c from both methods are consistent with the times that we would expect, given the duration of the output pulses from the pulse-shaping circuit.

In conclusion, for applications where time-tagging of individual photon detections is not needed, our CCM offers some attractive features. It takes four inputs and determines user selectable two-, three-, or fourfold coincidences (or single counts) on eight counting channels. The CCM has a high maximum count rate of 84 MHz, and its coincidence resolution is as low as 12 ns. Furthermore, several CCMs can be cascaded together to count arbitrary *M*-order coincidences among *N* inputs. Because of its small size, low cost, and intuitive user interface, the CCM is also well-suited to undergraduate physics laboratories.

We thank David Ahlgren, Sagar Bhandari, John Bower, Adam Katcher, Larry North, Steve Petkovsek, Wayne Strange, and Jared Zimmerman for help with design, assembly, and testing. This work was supported by NASA through the Connecticut Space Grant College Consortium.

[1] J. Gea-Banacloche, Prog. Opt. **46**, 311 (2004).
[2] P. Kok, W. J. Munro, T. C. Ralph, J. P. Dowling, and G. J. Milburn, Rev. Mod. Phys. **79**, 135 (2007).
[3] V. Scarani, H. Bechmann-Pasquinucci, N. Cerf, M. Dusek, N. Lütkenhaus, and M. Peev, Rev. Mod. Phys. **81**, 1301 (2009).
[4] S. Gaertner, H. Weinfurter, and C. Kurtsiefer, Rev. Sci. Instrum. **76**, 123108 (2005).
[5] http://www2.physics.utoronto.ca/~astummer/pub/mirror/Projects/Archives/Coincidence%20Counter/Coincidence%20Counter.html.
[6] http://www.nist.gov/physlab/div844/grp03/multicoincidence.cfm.
[7] S. Felekyan, R. Kuhnemuth, V. Kudryavtsev, C. Sandhagen, W. Becker, and C. A. M. Siedel, Rev. Sci. Instrum. **76**, 083104 (2005).
[8] M. Wahl, H.-J. Rahn, I. Gregor, R. Erdmann, and J. Enderlein, Rev. Sci. Instrum. **78**, 033106 (2007).
[9] Y. Acremann, V. Chembrolu, J. P. Strachan, T. Tyliszczak, and J. Stohr, Rev. Sci. Instrum. **78**, 014702 (2007).
[10] D. Dehlinger and M. W. Mitchell, Am. J. Phys. **70**, 898 (2002).
[11] D. Branning, S. Bhandari, and M. Beck, Am. J. Phys. **77**, 677 (2009).
[12] Resources for the construction and operation of this CCM, including an assembly guide, an operating manual, and data acquisition software (for LABVIEW or as a standalone executable), may be freely downloaded from our web site: www.trincoll.edu/~dbrannin.
[13] See supplementary material at http://dx.doi.org/10.1063/1.3524571 for the complete ciruit diagram and photographs of the assembled CCM.
[14] C. Eckart and F. R. Shonka, Phys. Rev. **53**, 752 (1938).