10-2009

Glenn Research Center Quantum Communicator Receiver Design and Development

Murad Hizlan
Cleveland State University, hizlan@cxvax.csuohio.edu

John D. Lekki
National Aeronautics and Space Administration

Binh V. Nguyen
National Aeronautics and Space Administration

Original Citation

Hizlan, M.,Lekki, J.D., & Nguyen, B.V. (2009). Glenn Research Center Quantum Communicator Receiver Design and Development.
Optical Engineering **48**, 105003.

Repository Citation

Hizlan, Murad; Lekki, John D.; and Nguyen, Binh V., "Glenn Research Center Quantum Communicator Receiver Design and Development" (2009).
Electrical Engineering & Computer Science Faculty Publications. 1.
https://engagedscholarship.csuohio.edu/enece_facpub/1

Glenn Research Center quantum communicator receiver design and development

Murad Hizlan
Cleveland State University
Electrical and Computer Engineering
    Department
2121 Euclid Avenue
Cleveland, Ohio 44115
E-mail: m.hizlan@csuohio.edu

John D. Lekki
National Aeronautics and Space Administration
Glenn Research Center
Optical Instrumentation and NDE Branch
MS 77-1
21000 Brookpark Road
Brook Park, Ohio 44135

Binh V. Nguyen
National Aeronautics and Space Administration
Glenn Research Center
Digital Communications Branch
MS 54-8
21000 Brookpark Road
Brook Park, Ohio 44135

Abstract. We investigate, design, and develop a prototype real-time synchronous receiver for the second-generation quantum communicator recently developed at the National Aeronautics and Space Administration (NASA) Glenn Research Center. This communication system exploits the temporal coincidences between simultaneously fired low-power laser sources to communicate at power levels several orders of magnitude less than what is currently achievable through classical means, with the ultimate goal of creating ultra-low-power microsize optical communications and sensing devices. The proposed receiver uses a unique adaptation of the early-late gate method for symbol synchronization and a newly identified 31-bit synchronization word for frame synchronization. This receiver, implemented in a field-programmable gate array (FPGA), also provides a number of significant additional features over the existing non-real-time experimental receiver, such as real-time bit error rate (BER) statistics collection and display, and recovery and display of embedded textual information. It also exhibits an indefinite run time and statistics collection. © 2009 Society of Photo-Optical Instrumentation Engineers. [DOI: 10.1117/1.3241989]

Subject terms: optical communications; receiver design; synchronization; quantum optics; photonics.

1 Introduction

In this paper, we investigate, design, and develop a prototype synchronous real-time receiver for the second-generation quantum communicator recently developed at the National Aeronautics and Space Administration (NASA) Glenn Research Center (GRC). This communication system utilizes the temporal coincidences between simultaneously fired ultra-low-power laser sources to communicate at power levels several orders of magnitude less than what is currently achievable through classical means.[1] The ultimate goal is to create ultra-low-power microsize optical communications and sensing devices for use in microrobots and microspacecraft in support of future NASA exploration missions. The remainder of this paper is organized as follows. Section 2 introduces two generations of the quantum communicator concept, and presents the second generation in detail. Section 3 describes the current experimental setup, and introduces the proposed receiver. In Secs. 4 and 5, we develop and analyze symbol and frame synchronization techniques. In Secs. 6–8, we present bit-error rate (BER) statistics collection and textual information display approaches, and the field-programmable gate array (FPGA) implementation. Conclusions are presented in Sec. 9.

2 GRC Quantum Communicator

The quantum communicator concept is based on defining the data signal in terms of temporal coincidences of photons or light pulses received on two independent channels. The data is encoded through the simultaneous release of photons or simultaneous firing of pulsed lasers on two or more collinear beams (channels) propagating through free space. Detection is performed by comparing the number of coincidences per bit from a coincidence detector to a threshold level for a two-channel system, or by comparing the numbers from two coincidence detectors for a multichannel system. The number of accidental coincidences due to the presence of ambient light (noise) is much smaller than that of signal coincidences, affording reliable communications under very low signal-to-noise ratios[1–3] (SNRs).

Illustrated in Fig. 1 is a two-channel system concept. The data sequence 1 0 1 1 is encoded by three signal photons for each 1, simultaneously fired on channels A and B. Photons from the background radiation are also received independently on the two channels. The coincidence detector will recover signal coincidences and accidental coincidences. Detection is performed by comparing the coincidence counts per bit to a threshold level.

The first-generation quantum communicator implemented by the Quantum Communications and Sensing Team at NASA GRC utilized the quantum entanglement property of phase-entangled photons to generate simultaneously released photons.[4–6]

The second generation employs up to four channels of simultaneously fired pulsed laser sources for significantly more efficient coincident signal generation.[2] Current ex-
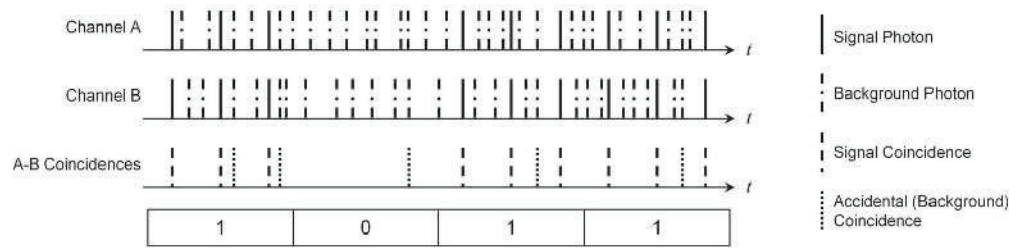
Fig. 1 Two-channel quantum communication concept.

perimental implementation allows for a data rate of 100 kbits/s with 10 laser pulses per bit. The coincidence detector window size is adjustable around 10 ns.

Data encoding with the four-channel system is accomplished by simultaneous firing of laser pulses on channels A and B for each 1, and simultaneous firing on channels C and D for each 0, as illustrated in Fig. 2. A comparison of the number of A to B coincidences (hereafter referred to as channel 1) and the number of C to D coincidences (hereafter referred to as channel 2) produces the detected data symbol. For the three-channel system, channel A may fire continuously while channel B fires for each 1 and channel C fires for each 0. Detection is accomplished by comparing A to B coincidences with A to C coincidences.

3 Experimental Setup

The current experimental setup allows for repetitive transmission of a prestored sequence of 16,384 bits. The receiver simply stores the coincidence times from the two coincidence detectors (channels 1 and 2) for a total of 65,536 data points. This corresponds to about 5000 to 20,000 data bits, depending on the amount of background radiation and signal loss. Analysis is performed by postprocessing of the stored coincidence times, which involves the estimation of symbol boundaries, detection of symbols and BER computation.[2]

The proposed experimental setup includes the same transmitter as before. The receiver, however, is a real-time synchronous receiver with significant additional capabilities. The proposed receiver, to be implemented on an FPGA, performs symbol synchronization (data clock recovery), data detection, frame synchronization, on-the-fly BER statistics collection and display, and recovery and display of embedded textual information. It also enjoys an indefinite run time and statistics collection.

4 Symbol Synchronization

For symbol synchronization, we chose to adapt an early-late gate type of closed-loop synchronizer to the quantum communicator. Early-late gate synchronizers (Ref. 7, pp. 628–631) are commonly used for rf communications and can be adapted in a straightforward manner for use in classical optical communications. However, their use for a discrete event-counting type of an optical communication system is a first in our application.

The early-late gate concept is a simple one. As shown in Fig. 3, the receiver computes an early integral (from 0 to T-d) and a late integral (from d to T) of the demodulated signal within the estimated T-s symbol window (gate). If the estimate is correct, both early and late integrals produce the same result in the absence of noise. If the estimate is incorrect, either the early (for advanced timing) or the late
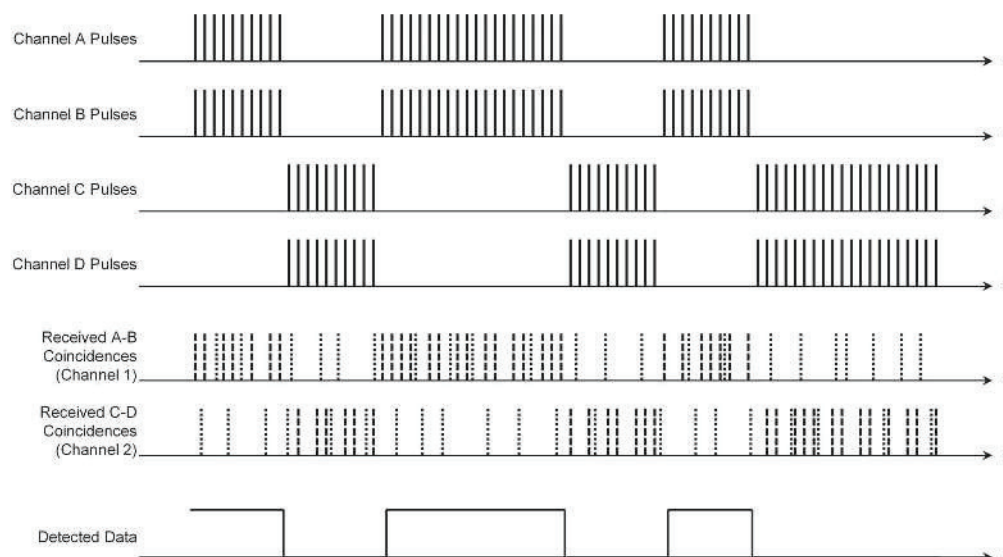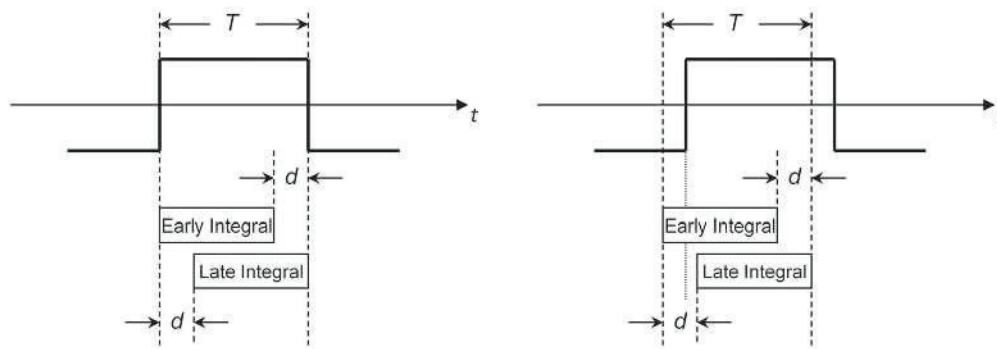

Fig. 2 Second-generation GRC quantum communicator concept.

$$feedback\ error\ =\ |\ early\ integral\ |\ -\ |\ late\ integral\ |$$

Fig. 3 Early-late gate synchronizer.

(for retarded timing) integral will produce a result smaller in absolute value. The difference of the absolute values of the early and late integrals produces the feedback signal, which is used to adjust the timing. Of course, if there is no level transition in the demodulated signal, the feedback error will be zero, again in the absence of any noise. Illustrated at the right of Fig. 3 is an advanced timing estimate, which would produce a negative feedback error that in turn would retard the timing.

Adaptation of this idea to the quantum communicator involves counting coincidences rather than performing integrals. The performance of the synchronizer is limited by the time resolution of the received signal in this case. Specifically, since we have 10 discrete pulses per data bit, the synchronizer cannot resolve timing differences of less than $0.1T$. This corresponds to a possible timing offset of up to $0.05T$ (in either direction) even when the feedback error is zero, in the absence of any noise.

We seek to implement the proposed symbol synchronizer in an FPGA running a system clock rate of 100 MHz. This corresponds to 1000 system clock pulses per symbol period (for a 100-kbits/s data rate of the current experimental setup). The operation of the synchronizer is illustrated in Fig. 4. The early integral corresponds to a count, over the first half of the symbol period (500 clock pulses nominally), of the difference between channel 1 coincidences and channel 2 coincidences, while the late integral corresponds to the same count over the second half of the symbol period. Here, d is taken to be $T/2$. The feedback error,
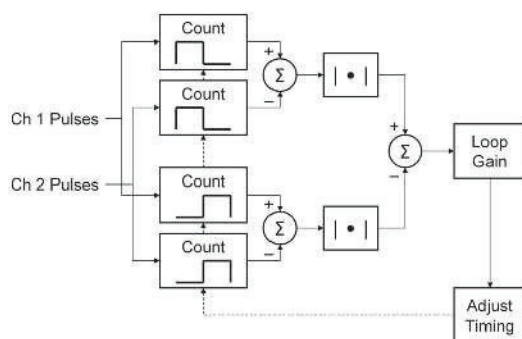
defined as before, is multiplied by the integer loop gain and the result is subtracted from 500 to obtain the duration of the half symbol period for the next symbol. The net result is either retardation or advancement of the symbol timing estimate, in terms of the number of clock pulses per half symbol period.

We simulated the performance of the symbol synchronizer using Mathcad™ with both simulated and experimentally collected data of varying background noise and signal loss levels. Figure 5 plots the performance of the proposed synchronizer in terms of the absolute value of the timing offset normalized by the symbol period for approximately 11,000 bits of experimentally collected data. The root-mean-square (rms) value of the timing offset (rms jitter) for this example was calculated to be approximately $0.04T$, which agrees well with the one-sided timing resolution of $0.05T$.

In addition to timing jitter, another crucial measure of performance for the symbol synchronizer is the synchronization acquisition time. Based on all the experimental data collected, worst-case acquisition time with a sequence of alternating 1's and 0's, and starting with a $0.5T$ offset is about 60 bits for a feedback loop gain of $0.008T$ (8 clock pulses). This value of the feedback gain provides the best compromise between a fast acquisition time and a low timing jitter. Figure 6 shows the symbol timing acquisition with an initial timing offset of $0.4T$ for a relatively noisy and lossy sequence of alternating 1's and 0's, while Fig. 7 shows the midstream timing estimate and observed channel 1 coincidences for the same data set.

Note that in addition to proportional feedback, integral feedback was also studied and found to be useful only when there is more than 0.1% offset between the transmitter and receiver master clocks. However, only proportional feedback was considered for the proposed implementation.

5 Frame Synchronization

Frame synchronization is essential for identifying the beginning of valid data within a received data packet. After symbol synchronization takes place, a unique frame synchronization word is transmitted to signal the beginning of valid data.

We chose to use a 128-bit preamble for symbol and frame synchronization in the proposed experimental setup.

Fig. 4 Early-late gate synchronizer implementation.

| Normalized Timing Offset |
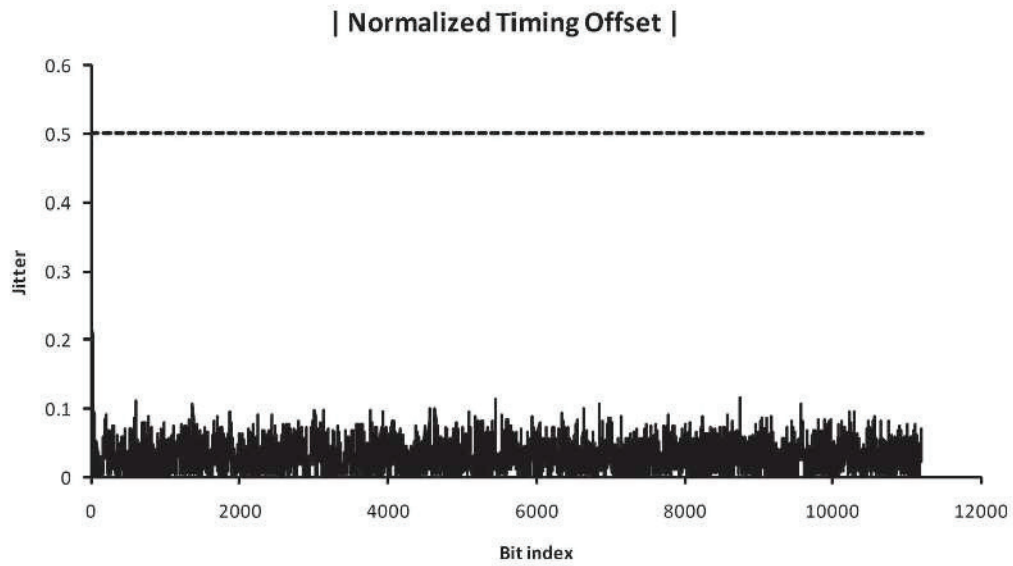
Fig. 5 Experimental data: timing jitter.

| Normalized Timing Offset |
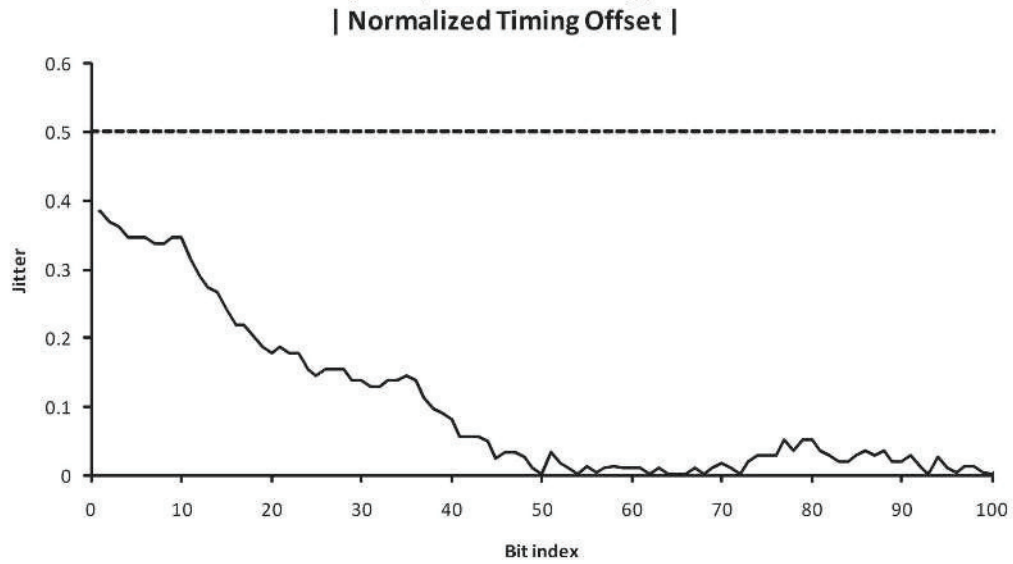
Fig. 6 Experimental data: acquisition with $0.4\,T$ initial timing offset.

Channel 1 Coincidences and Estimated Boundaries
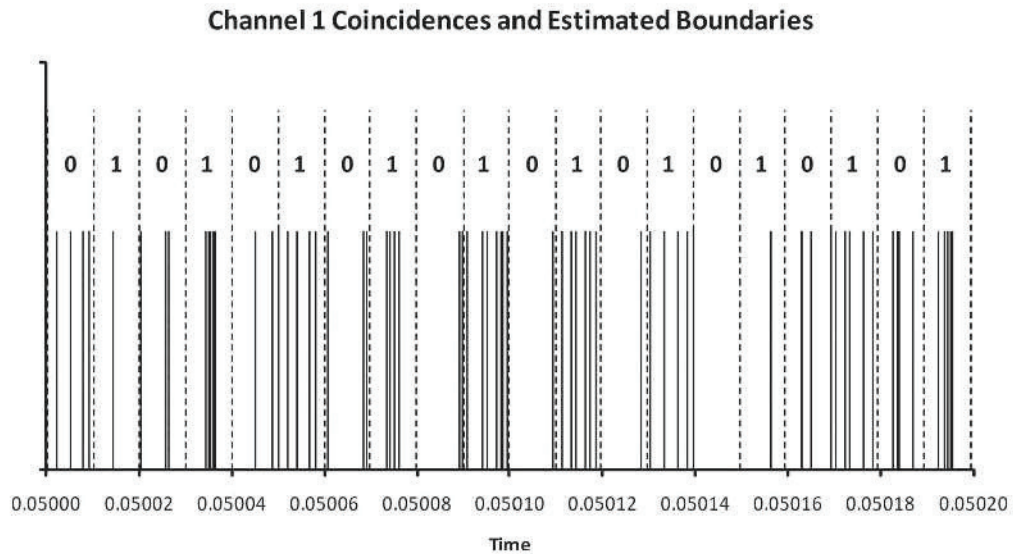

Fig. 7 Experimental data: synchronization with a noisy and lossy signal.
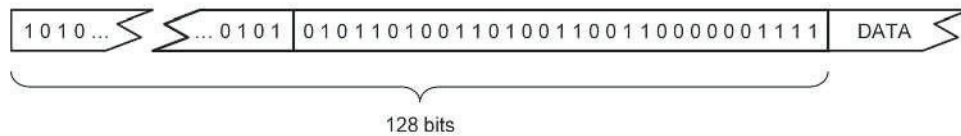
Fig. 8 Preamble for symbol and frame synchronization.

As shown in Fig. 8, the preamble consists of a 97-bit sequence of alternating 1's and 0's, starting and ending with a 1, for symbol synchronization, and a 31-bit unique word for frame synchronization. Based on worst-case symbol timing acquisition simulations, 97 bits of alternating 1's and 0's should be more than sufficient for symbol timing acquisition.

The two crucial measures of performance for frame synchronization are detection probability and false alarm probability. Due to channel errors, one cannot expect a complete match between the reference frame synchronization word stored at the receiver and the sequence received from the channel. Therefore, the receiver should declare a "match," or detection, with less than a complete match between the received sequence and the reference word. Clearly, one objective is to maximize the probability of such a match, or the detection probability. Again, due to channel errors, it is also possible that the receiver may declare a "match" before the actual synchronization word is transmitted. Another objective is, then, to minimize the probability of such false detections, or the false alarm probability. These two objectives are conflicting goals since relaxing the detection criteria increases both probabilities, while tightening them decreases both. On the other hand, reducing the channel error rate, increasing the synchronization word size, and optimizing the synchronization word support both objectives.

Qualities of a good synchronization word are a low autocorrelation and a low cross-correlation with the preceding sequence for minimization of the false alarm probability. Such sequences have been studied for decades (see, e.g.,

Refs. 8–10), and low-autocorrelation sequences exhibiting low cross-correlations with preceding all 0's, preceding all 1's, and preceding random bits are known. There exists, however, no prior study of sequences exhibiting low cross-correlations with a preceding sequence of alternating 1's and 0's. We therefore performed an exhaustive search of all sequences of up to 32 bits in length with minimum autocorrelation and cross-correlation with alternating 1's and 0's. Our search was limited to a maximum length of 32 bits due to the exponentially increasing run times and bit-level manipulation within a (32-bit) integer to increase coding efficiency.

We identified a 31-bit sequence, 0 0 0 0 0 0 0 1 0 0 1 1 1 0 1 0 0 0 1 1 1 1 0 0 1 1 0 1 0 1 1 0, that gave reasonable performance. Setting the false alarm probability to be approximately equal to the miss probability (1 − detection probability), we found that 23 matches out of 31 provides the best compromise between detection and false alarm probabilities. Figure 9 plots the probability of a match for the identified sequence (where a match is defined as at least 23 out of 31 bits matching) as a function of position before the arrival of the actual synchronization word for a worst-case channel error probability of $p=0.1$. The detection probability is shown at position 0, and false alarm probabilities are shown at negative positions.

For the design channel error probability of $p=0.1$, the detection probability is $1-2.6 \times 10^{-3}$ and the overall false alarm probability is less than 4.1×10^{-3}. Both probabilities improve rapidly with decreasing channel error rates, with the detection probability reaching $1-1.7 \times 10^{-11}$ and the
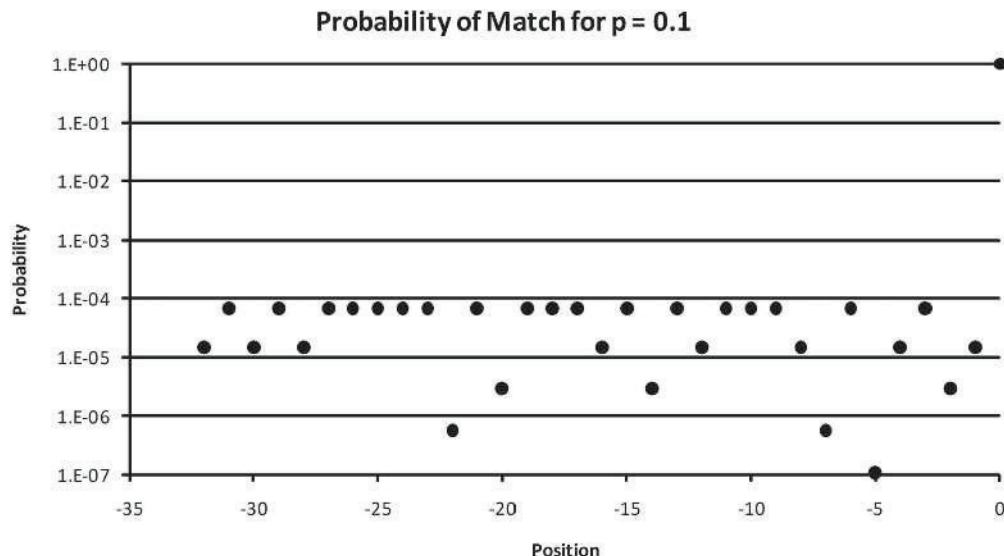

Fig. 9 Probability of a match before the arrival of the synchronization word.
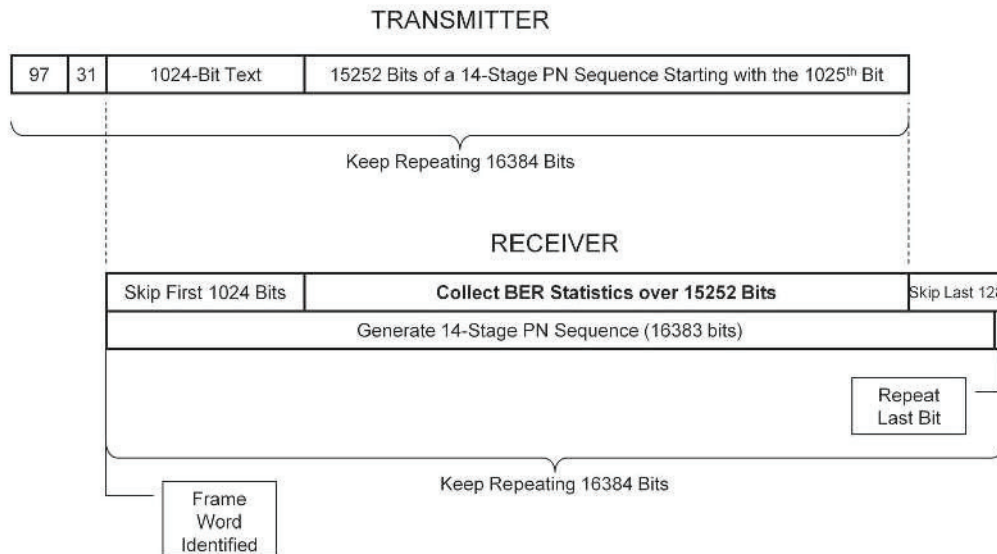
TRANSMITTER

97	31	1024-Bit Text	15252 Bits of a 14-Stage PN Sequence Starting with the 1025th Bit

Keep Repeating 16384 Bits

RECEIVER

Skip First 1024 Bits	**Collect BER Statistics over 15252 Bits**	Skip Last 128

Generate 14-Stage PN Sequence (16383 bits)

Repeat Last Bit

Keep Repeating 16384 Bits

Frame Word Identified

Fig. 10 BER statistics collection illustration.

false alarm probability reaching under 5.1×10^{-11} at a conservative channel error probability of $p = 0.01$.

6 BER Statistics Collection

One convenient feature of the proposed experimental setup for the quantum communicator is the real-time BER statistics collection and display. As stated before, the current transmitter is capable of continuously transmitting pre-stored sequences of length up to 16,384 bits. We chose to store a pseudonoise (PN) sequence in the transmitter memory and synchronously generate the same sequence at the receiver for error statistics collection.

As shown in Fig. 10, the first 128 bits of the 16,384-bit sequence stored at the transmitter are reserved for symbol and frame synchronization. The next 1024 bits are reserved for textual information, which is discussed in the next section. The remaining 15,252 bits are obtained from a particular PN sequence. At the receiver side, once the frame synchronization word is detected, the same PN sequence is generated repeatedly using a 14-stage linear feedback shift register PN generator. The last bit of the resulting 16,383-bit sequence is repeated once to bring the total number of bits per cycle to 16,384. BER statistics collection starts 1024 bits after the detection of the frame synchronization word to skip the textual information. The last 128 bits are also skipped in each cycle to ignore the repetitive transmission of the synchronization bits. BER statistics collection therefore takes place over the remaining 15,252 bits for each cycle.

Bits in error are identified by comparing the detected bit

TRANSMITTER

97	31	**1024-Bit Text**	15252 Bits of a 14-Stage PN Sequence Starting with the 1025th Bit

Keep Repeating 16384 Bits

RECEIVER

Write into RAM

Display at a Rate of 3 Characters Per Second
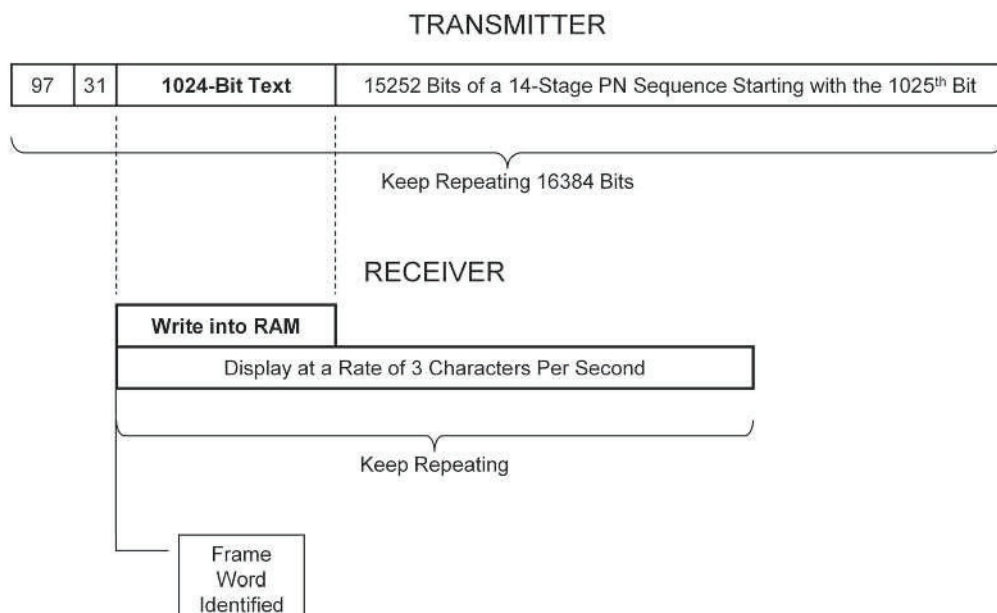
Keep Repeating

Frame Word Identified

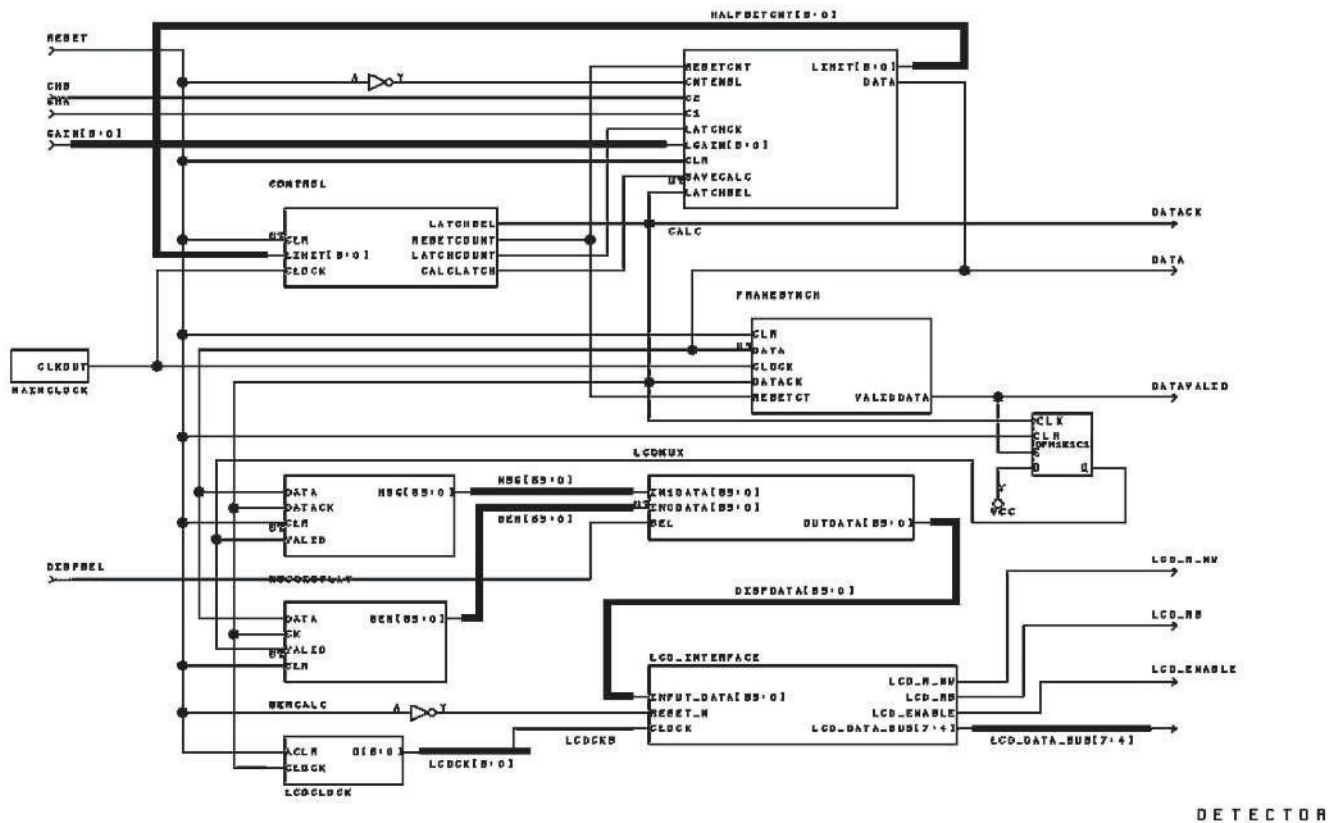**Fig. 11** Textual information display illustration.

Fig. 12 Top-level diagram of the receiver implementation.

with the locally generated replica at the receiver. We chose to make use of an on-board 8-digit LCD display on the Actel Fusion™ Starter Kit[11] for BER statistics display. The total number bits in error and the total number of bits received are displayed in turn every 200,000 bits (approximately every 2 s). BER statistics collection can go on "indefinitely," or at least until 10^8 errors are collected.

7 Textual Information Display

An additional, "nice-to-have" feature of the experimental setup we propose for the quantum communicator is the ability to decode and display textual information embedded in the transmitted data. For this purpose, the first 1024 bits of "valid" data after the frame synchronization word are reserved for 128 characters of text, as shown in Fig. 11.

At the receiver, once the synchronization word is detected, the next 1024 bits are written into a RAM. The stored textual information is then decoded and continuously scrolled through the 8-digit on-board LCD display at a rate of 3 characters/s. BER statistics collection and textual information decoding simultaneously takes place at the receiver, and either can be displayed at any time by throwing a switch.

8 FPGA Implementation

Complete receiver design was implemented using Libero™ IDE v7.2 (Ref. 12) for Actel Fusion™ ASF600 FPGA chip.[13] Figure 12 shows the top-level block diagram of the receiver implementation.

9 Conclusions and Future Work

We investigated, designed, and implemented a real-time synchronous receiver for the second-generation quantum communicator being developed at NASA GRC. The receiver uses a unique adaptation of the early-late gate method for symbol synchronization and a newly identified 31-bit word for frame synchronization. The receiver also provides a number of significant additional features such as real-time BER statistics collection and display, and recovery and display of embedded textual information. It also exhibits indefinite run time and statistics collection.

Further studies will concentrate on error control coding techniques for, and improved generations of, the quantum communicator. Our ultimate goal is to create reliable ultra-low-power microsize optical communications and sensing devices.

References

1. J. D. Lekki, Q.-V. Nguyen, T. Bizon, B. Nguyen, and J. Kojima, "An extremely low power quantum optical communication link for autonomous robotic explorers," in *Proc. AIAA Infotech@Aerospace Conf. 2007*, Rohnert Park, CA, May 7–10, 2007, pp. 1925–1933, AIAA, Reston, VA (2007).
2. J. Lekki, Q.-V. Nguyen, T. Bizon, B. Nguyen, J. Kojima, and M. Hizlan, "Extremely low power quantum optical communication link for miniature planetary sensor stations," *J. Aerospace Comput. Inf. Commun.* 5(10), 396–408 (2008).
3. J. Lekki, "Quantum entanglement enables low-power optical communication to orbital ranges," *Opt. Eng.* 47(1), 015003 (2008).
4. M. Hizlan, "Quantum communications system analysis," NFFP 2004 Report, NASA Glenn Research Center, Cleveland, OH (Aug. 2004).
5. M. Hizlan, "GRC quantum communicator experimental analysis," NSFROP 2005 Report, NASA Glenn Research Center, Cleveland,

OH (Aug. 2005).

6. J. D. Lekki, Q.-V. Nguyen, B. Nguyen, and M. Hizlan, "Quantum optical communication for micro robotic explorers," in *Proc. AIAA Infotech@Aerospace Conf. 2005*, AIAA, Reston, VA (2005).

7. B. Sklar, *Digital Communications: Fundamentals and Applications*, Prentice Hall PTR, Upper Saddle River, NJ (2001).

8. U. Somaini, "Binary sequences with good autocorrelation and cross correlation properties," *IEEE Trans. Aerosp. Electron. Syst.* **11**(6), 1226–1231 (1975).

9. M. Grayson and M. Darnell, "Optimum synchronisation preamble design," *Electron. Lett.* **27**(1), 36–38 (1991).

10. Ş. E. Kocabaş and A. Atalar, "Binary sequences with low aperiodic autocorrelation for synchronization purposes," *IEEE Commun. Lett.* **7**(1), 36–38 (2003).

11. The Fusion® Starter Kit, http://www.actel.com/products/hardware/devkits_boards/fusion_starter.aspx (accessed May 5, 2009).

12. Libero, Actel's Flagship FPGA Development Software, http://www.actel.com/products/software/libero/default.aspx, (accessed May 5, 2009).

13. Actel Fusion®, http://www.actel.com/products/fusion/default.aspx (accessed May 5, 2009).

Murad Hizlan received his BS degree from Bogazici University, Istanbul, Turkey, in 1985 and his MSE and PhD degrees from The Johns Hopkins University, Baltimore, Maryland, in 1987 and 1990, all in electrical engineering. In 1991 to 1992, he was first a postdoctoral fellow, then an associate research scientist with the Department of Electrical and Computer Engineering, The Johns Hopkins University. In 1992, he joined the Department of Electrical and Computer Engineering, Cleveland State University, Ohio, where he is currently an associate professor. Dr. Hizlan also spent several summers as a summer faculty fellow at the National Aeronautics and Space Administration (NASA) Glenn Research Center, Cleveland, Ohio, working on various research projects. Hizlan has more than 20 years of research experience in digital communications, in particular robust communications, spread spectrum, multiple access, and coding. He is a member of IEEE, Eta Kappa Nu, and Tau Beta Pi.

John D. Lekki is an optical systems research engineer at the National Aeronautics and Space Administration (NASA) Glenn Research Center. His research interests include quantum optical systems, remote sensing systems, and sensor systems for aircraft. He received his BS degree in electrical engineering from Michigan State University in 1993, his MS degree in physics from Cleveland State University in 2002, and his PhD degree in electrical engineering from Michigan State University in 2008.

Binh V. Nguyen is an electronics engineer at the Digital Communications Branch of the National Aeronautics and Space Administration (NASA) Glenn Research Center, Cleveland, Ohio.