FPGA Utilized to Control the Coincidence Logic of Quantum Entangled Pairs (QEP)

Mohammed Munther A. Majeed

Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia

muhammed8888@gmail.com

Zuriati Ahmad Zukarnain

Faculty of Computer Science and Information Technology, Universiti Putra Malaysia, 43400 UPM, Serdang, Selangor, Malaysia

drzuriati@gmail.com

Abstract- Control of quantum signals is very robust design technique and very important in quantum signal processing (QSP). It is not easy in experimentation platforms. Field Programmable Gate Array (FPGA) is using to control very various range of devices in quantum fields through PCs. FPGA needs to run modules of components to communicate and interfacing with the PC, through decode perform commands to direct control of digital hardware's. If programmer has a real-time control of the FPGA via USB, it can be possible to evaluate design parameters changes in real-time, without reprogramming the FPGA. That makes the proposed design platforms easier for researchers. This paper discusses experiment of control quantum signals by FPGA to control coincidence logic for Quantum Entangled Pair (QEP), that able to measure polarization correlations relationships between photons of the QEP. This FPGA helps to determine detection events at different detectors which can be attributed to a single photon pair. Also to determine the correlation time between two different beams frequencies is extremely accurate according to the delay between the beams, which is very short.

Keywords; Quantum Signal Processing (QSP); Field-Programmable Gate Array (FPGA); Quantum Entangled Pair (QEP)

I. INTRODUCTION

Currently, major FPGA manufacturers have implemented different herring mechanisms to ensure that the details of the design cannot be meaningfully intercepted en-route to the device or accessed once on the device. This helps protect against interception and cloning. However, it does not address over deployment. With the current methodologies, there is no way to limit the number of deployments once at the programming site. In this paper, we present the details of control signals quantum applications by digitally controlled quantum signals in laboratory setup used for research to control two polarization correlations photons of a quantum entangled pair (QEP) in two laser frequencies. Therefore in section II we provide the quantum signal processing (QSP) experiment design, with QSP Algorithm. Section III sets the experiment design illustrating the FPGA Performance in QSP which we apply our design for producing correlate entangled pair. As well illustrating our architecture in more detail, and how we can utilize an FPGA situated in system. In Section IV we present QSP experiment results and discussion. In this section we provide results as well discusses the advantages of the design and analyzing performance overhead of our design. Finally, in Section V, we present our conclusions alongside a discussion of future work which is attractive starting point for further researches.

II. QUANTUM SIGNAL PROCESSING (QSP) DESIGN

A. Topology of Our QSP Experiment

Our quantum signal process (QSP) experiment can explain the design by these chosen and support concept points:

1) Free Space Optical (FSO) communications channel is chosen as a quantum transmission medium, because free space is homogenous and isotropic and hence the polarization states of the photons are unaffected during transmission between corresponding parties, whether in forward or backward direction.

2) A field programmable gate array (FPGA) is used to control the quantum signal process.

3) FPGA sets the timing and synchronizes between laser pump pulses and open/close of detectors gates.

4) The polarization of the laser pump pulse is set by a Glan-Thomson prism.

5) The axis of a specially cut non-linear birefringent crystal is set manually.

6) The crystal, due to its non-linear optical characteristics, splits the pump photon, and a pair of longer wavelengths entangled photons is generated by a spontaneous parametric down-conversion (SPDC) process [4].

7) The axes of two sets of half-wave and quarter-wave plates are set manually or by stepper motors controlled by the

FPGA, to process the entangled photons in two polarization states as required by the quantum key distribution (QKD) [2], [3].

8) The FPGA sets random time intervals for the pump pulses by a built-in random time interval (RTI), thus that the entangled pairs arrive at the receiver end at random intervals, only known to the sending end. The arrival times are used to synchronize the gating times of the detectors.

9) The FPGA sets the synchronization times for the detectors in correspondence with the same random time intervals. Any eavesdropping attack by an adversary will alter the timing and/or the polarization state such that one or both entangled pairs of photons will be missing, which will be detected immediately as a matter of routine. In such as event, the process is stopped and an investigation is carried out.

The optical anisotropy of the medium will cause one component to travel faster than the other, which causes a phase change between them, and hence the polarization in the general case will be elliptical. A probe pulse can be sent through the channel to measure the amount of anisotropies, by determining the orientation of the major axis and the ellipticity. Common ellipsometric techniques can be employed to counter this effect, by manually setting or by programming the FPGA to introduce opposing elliptically polarized light, such that the returning pulse will be linearly polarized in the intended orientation.

B. QSP Algorithm

Step 1: Start (Sender station);
Step 2: Run experiment by RTI sub-program in FPGA (Sender station);
Step 3: experiment prepares quantum states by pump specific leaser ray to crystal (Sender station);
Step 4: Send two quantum states to the detectors (Sender station);
Step 5: check and compares the synchronization of entangled pair, if are same, go to step 4, repeat according to count set, if OK How many times? Comparison goes to Step 8 (Receiver station);
Step 6: if is a not entangled pair, go to Step 7 (Receiver station);
Step 7: Stop the process and investigate;
Step 8: Enable another step of processing.
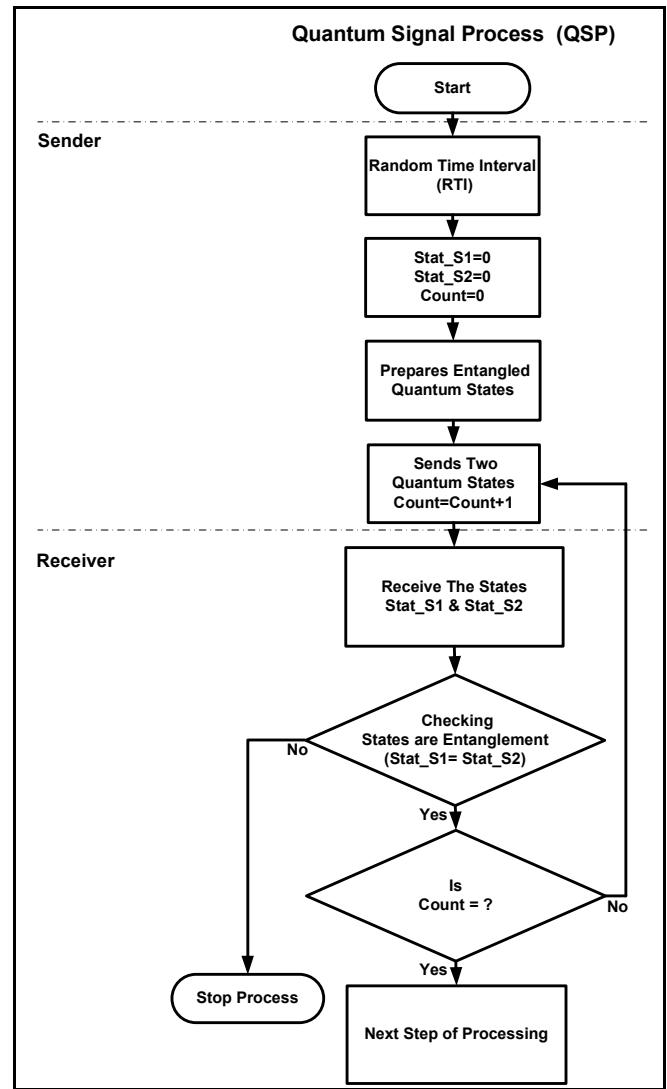Fig. 1 illustrates the flow chart diagram of the QSP.

Fig.1 Shown the flow chart diagram of the quantum signal process (QSP).

III. EXPERIMENT DESIGN

A free space optical (FSO) quantum channel is important, because this field is both isotropic and homogeneous. The polarization may change due to optical activity in an anisotropic medium.

The polarization states are realized by the angular settings of two half-wave and quarter-wave plates. The pairs of photons are generated by a non-linear crystal where polarized light is split into two polarized components of different wavelengths, by a process known as spontaneous parametric down-conversion (SPDC). It can be described as a process of pumping photons into pairs of correlated photons with lower frequencies. Fig. 2 illustrates how the ordinary and extraordinary down converted photons are generated from a pump laser source. The electric field vector $\vec{E}$ and the displacement vector $\vec{D}$ of the extraordinary polarization, although linearly related, are non-parallel.

In this case, the state of the down converted field, calculated in the first order of perturbation has the following form [1]:

$$|\Psi\rangle = |vac\rangle + \int d\vec{k_1} d\vec{k_2} \Psi(\vec{k_1}, \vec{k_2}) |V\rangle_1 |V\rangle_2 \quad (1)$$

where $\vec{k_1}, \vec{k_2}$ are wave phase vectors of the down converted photons and $\Psi(\vec{k_1}, \vec{k_2})$ is the photon amplitude characterizing the angular frequency spectrum of SPDC radiation.

In the stationary case, the photon frequencies satisfy the condition $\omega_1 + \omega_2 = \omega_p$ as required by the law of conservation of energy.

The frequencies of the photon pair are determined by the practical setup and the angular orientation of the crystal aided by narrow-band filters and pinholes. The polarization states of the two photons are described by a state vector $|\psi\rangle = |V_1\rangle + |V_2\rangle$, where V stands for vertical and the indices 1 and 2 correspond to different wavelengths, which have special propagation directions. An arbitrary two photon polarization state can be expressed as two qubits:

$$|\Psi\rangle = c_1|1\rangle + c_2|2\rangle + c_3|3\rangle + c_4|4\rangle \quad (2)$$

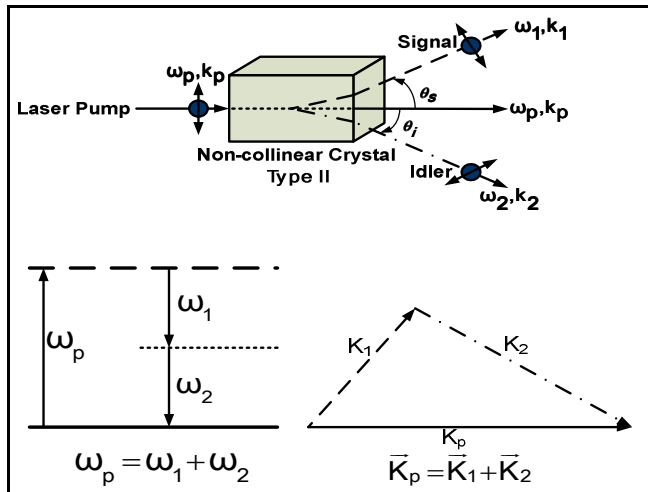$$|\Psi\rangle = c_1|H_1 H_2\rangle + c_2|H_1 V_2\rangle + c_3|V_1 H_2\rangle + c_4|V_1 V_2\rangle$$

Fig. 2 Shown the illustration of type-II SPDC.

where $c_j = |c_j| e^{i\phi_j}$ (j=1,…,4) and $|c_j|$ are complex probability amplitudes. One of the two terms in (2) either $c_2|H_1 V_2\rangle$ or $c_3|V_1 H_2\rangle$ can be canceled, because of their similar polarization states, although they are of different wavelengths. Hence, the states polarizations identified by [1]:

$$|z+\rangle = |H\rangle, |z-\rangle = |V\rangle, |x+\rangle = |D\rangle, \text{ and}$$

$$|x-\rangle = |A\rangle, |y+\rangle = |R\rangle, |y-\rangle = |L\rangle \quad (3)$$

Fig. 3 shows our QSP experiment setup. The two beams are combined on the dichroic beam splitter (DBS)

transmitting 880nm and reflecting 750nm and sent to Sender station. A set of quarter and half wave plates of appropriate wavelengths are inserted in each arm of the setup to realize the polarization transformations.

The experiment setup consists of a one or two specially half wave (545mkm) cut quartz plates. Such plate acts as half wave plate on both frequencies (750nm and 880 nm). To perform $\hat{X}$ or $\hat{Y}$ transformation only one plate is needed, oriented at 90° or 0° correspondingly.

In this experiment should use a deterministic measurement scheme that consists of two detectors, joined by a coincidence scheme, to control the quality of states preparation and transformations. A coincident counting unit, and a set arbitrary wave plate in every beam, and a polarizer (Glan-Thompson prism), are also employed. Receiver should perform projective measurement in the same basis that was used to create initial quantum state. To do this should rotate quartz plates at the same angles that were used in preparation part.

The experimental realization is set laser ray for non-collinear degeneration. The entangled pairs are sent to the receiver station via FSO link. A (β-BaB$_2$O$_4$) 10mm beta-barium-borate (BBO) non-linear optical crystal as shown in Fig. 3 is cut for non-collinear degeneration. It is pumped by a 50mW laser source operating at 405nm. The crystal axis is oriented horizontally so that the resulting photons are vertically polarized. The Glan-Thompson prism (H) ensures the horizontal polarization of the pump. The wavelengths of the down converted photons are $\lambda_1 = 750$nm and $\lambda_2 = 880$nm. The various polarization states of the photons can be manipulated by two sets of (quarter and half) wave plates.

This experiment starts from first step which is the RTI, followed by the execution steps of the algorithm. Concurrently, the main VHDL program controls all sub-programs which execute the steps of the QSP.
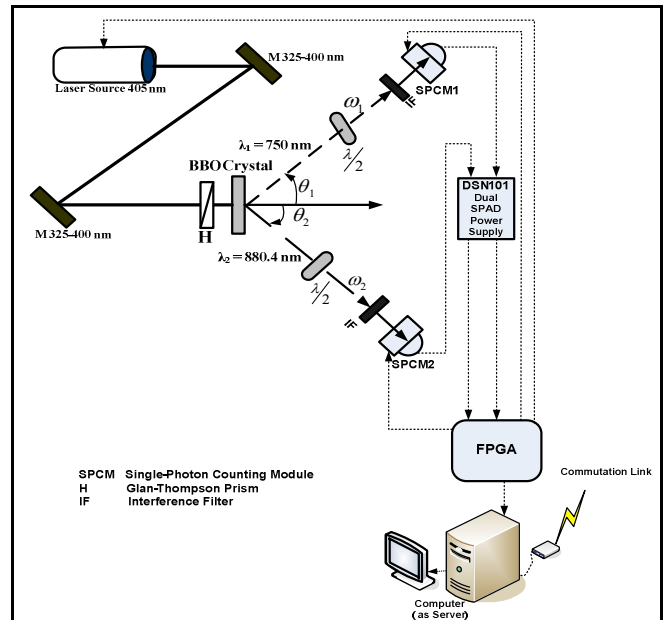

Fig. 3 Shown the QSP experiment setup

These random generate pulses of the laser source pump the crystal to generate entangled pairs of photons. At receiver's end, receiver measures deterministically the transformed two qubit states using the setup. The various polarization states of the photons can be manipulated by two sets of (quarter and half) wave plates. The results can be obtained from implementation of the FPGA controlled coincidence measurement, which is in two parts. First, the preparation of the input signals for the FPGA by transforming the photon click signals from the SPCM modules to electronic pulses. Second, the processing of the pulses in the FPGA by sampling the detector signals and comparing them with the laser pulses, thereby creating detector click patterns. Then the signals are transferred to the computer via USB to display the results.

A. FPGA Performance in QSP Experiment

From an overall point of view, the experiment was designed from model of the quantum electronic circuits (QEC) are illustrated in Fig 4.

The quantum signal processing (QSP) can be verified both by software simulation, and by practical implementation using FPGA. The design, which has been developed using VHDL, has achieved the performance requirements for the whole process to be more practicable.
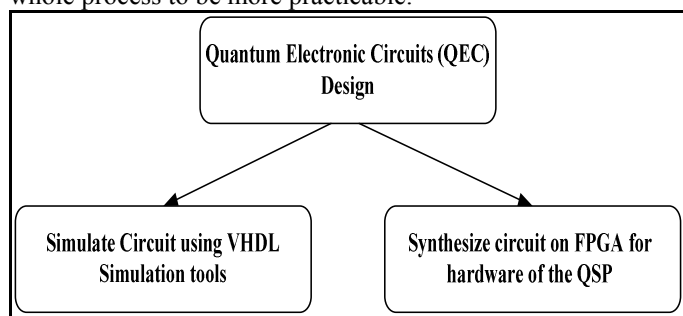

Fig. 4 shown the model of quantum electronic circuits (QEC) using the VHDL for quantum signal processing (QSP).

Quantum signal processing (QSP) experiment design uses two modes: Schematic and Hardware Description Language (HDL) mode. HDL describes the behavior of our system as individual gates. There are several popular hardware description languages such as very high speed hardware description language (VHDL) which we used in this experiment, Verilog and ABEL. The schematic flow mode allows the user to create the experiment design which consists of a top-level schematic file. It can contain underlying schematic diagrams, state machine macros, VHDL macros, LogiBLOX, and CORE Generator modules. Figure 5 shows the HDL flow mode of experiment design steps. This HDL flow mode can contain a VHDL or a schematic top-level design with underlying VHDL schematic modules.

When we made the experiment design by used the Synopsis FPGA Express package which comes with the Foundation Express tools (the software used is called Xilinx ISE® Design Suite). In experiment implementation, these tools fit the design into the FPGA architecture. These tools compile the experiment design files into a configuration file that is optimized in terms of use of logic gates and interconnections in FPGA. The bit stream file is then downloaded from PC into FPGA using a universal serial bus (USB). This design can be verified with functional simulation, in-circuit testing, and timing simulation. The function of simulation can be done after the design entry to verify the proper operation of the circuit.
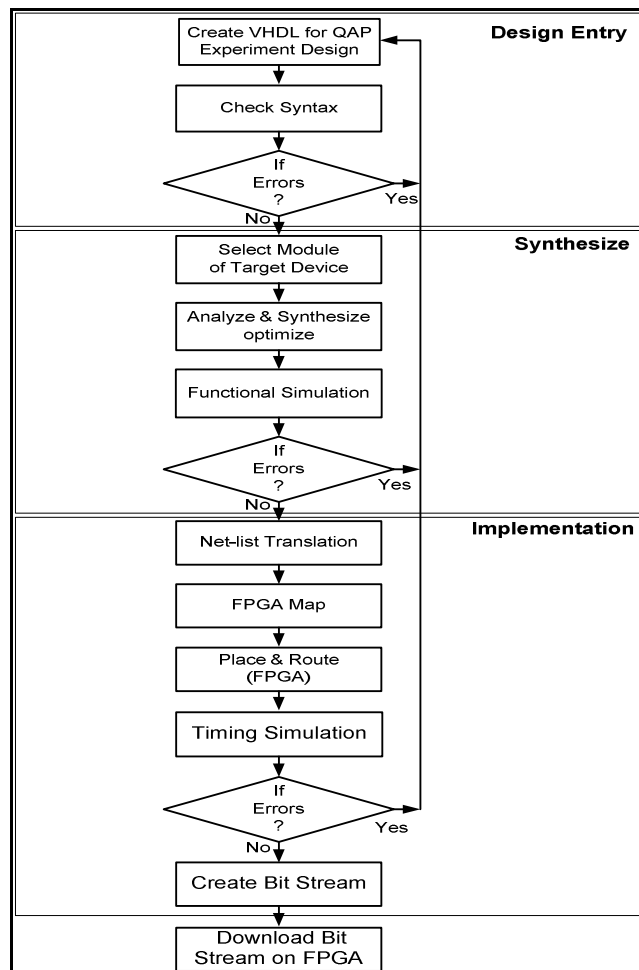

Fig. 5 Shown the HDL flow mode of experiment design steps

Fig. 6 shows the implementation and downloading of FPGA logic design steps using VHDL, and the modifications to suit the hardware implementation. A logic synthesizer program is used to transform the VHDL or schematic into a Net-list. That Net-list is just a description of the various logic gates in our design. For that design we use implementation tools to map the logic gates and interconnections into the FPGA. Once the implementation phase is complete, a program extracts the state of the switches in the routing matrices and generates a Bit Stream where the ones and zeroes correspond to open or closed switches. The last step is downloading the Bit-stream into a physical FPGA chip. Upon completing the download, the FPGA will perform the operations specified by our VHDL code.
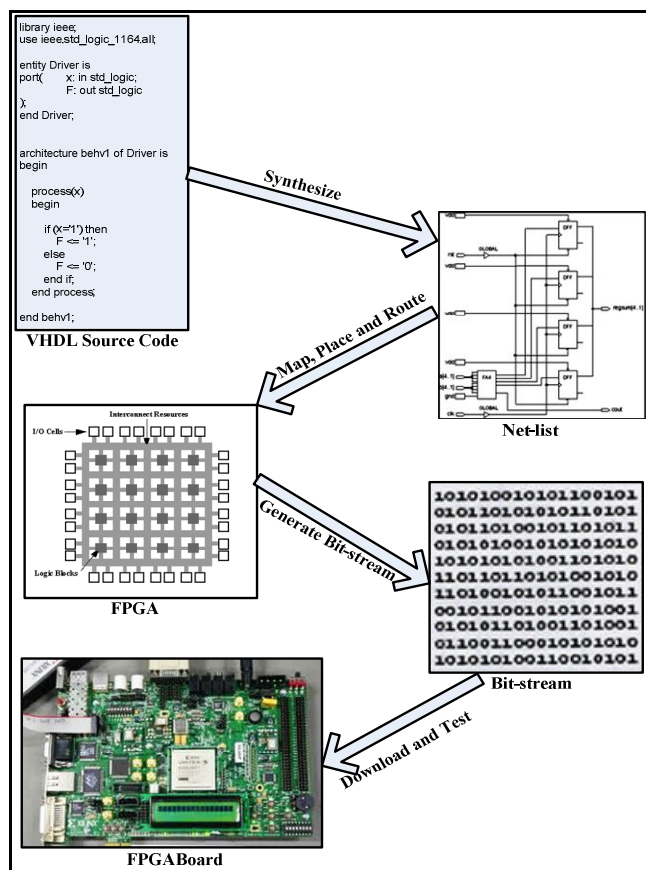
Fig. 6 Shown the implementation and downloading on FPGA logic design steps.

The function of FPGA is controlled coincidence logic in QSP experiment design, and it performance to measure polarization correlations between photons of an entangled pair, one has to determine which detection events at different detectors can be attributed to a single photon pair. The correlation time between two beams is extremely accurate. The delay between these two beams (according to different beams frequencies) is very short.

This process of filtering out pair events from a set of single photon detection event is called FPGA controlled coincidence detection, and is described in this section. This QSP implementation of FPGA controlled coincidence logic can be described by five steps, which are shown in Fig. 7 illustrated FPGA micro modules (every micro module is sub VHDL program) for QSP main project using VHDL language:

1) FPGA prepares input signals for transforming a laser source signal.
2) FPGA prepares input signals for transforming detector gate control according to input signals from the laser source.
3) FPGA prepares input signals for transforming detectors' signals and the master clock derived from the laser frequency.
4) Processing of detector clicks by the FPGA: detector signals and laser frequency create a detector click pattern.

5) Universal serial bus (USB) signal preparation, detector click patterns are transformed using USB data bus to be transferred to the computer.

The FPGA board is a Virtex-5 XC5VLX50 mounted on a ML501 evaluation platform provided by Xilinx®. For the present implementation, an input bus width of two detector signals (denoted as [11:0]) is used. This is sufficient for measuring the photon states of two qubits with two detectors. The names of the signals used follows the description corresponding to the names of VHDL files. These files were compiled using Xilinx® ISE Design Suite 12.2.
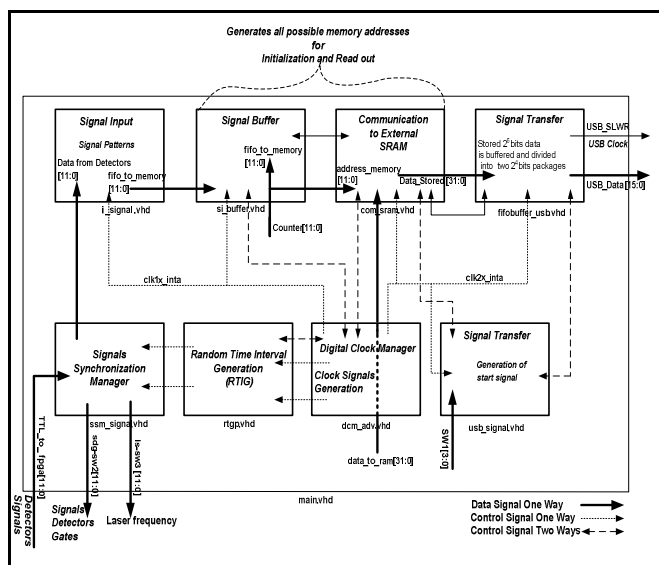

Fig.7 Shown the FPGA micro modules of QSP main project.

IV. QSP EXPERIMENT RESULTS AND DISCUSSION

Fig. 8 shows the QSP experiment waveforms of the FPGA that represent the signal processing steps during the 2^8 times, which equal the nth frame time. After starting the process, the random time intervals, which are based on the internal clock of the FPGA, are generated by executing the relevant part of the algorithm written in VHDL. The random pulses operate the laser source, which pumps the non-linear crystal that produces the quantum entangled pair (QEP) of photons. These entangled photons are sent to the receiver station via free space. The different pulses produced by FPGA control the gates of detectors to open and close to allow authentic light pulses for photon count. The comparison of entangled states starts during the time that the detector gates are all open. If any of the two entangled photons is not missing or has changed polarization, the photon correlation will be valid. The FPGA repeats the sequence two more times for the laser to initiate another entanglement. Otherwise, the FPGA stops the process and the user investigates for eavesdropping. Else, the FPGA sends a pulse to enable another process maybe another entangled process.
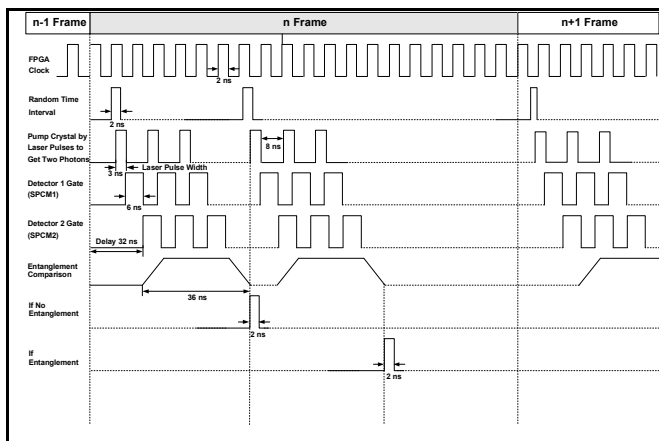
Fig. 8 Shown the QSP wave frame during n time.

A. Advantages of the Design

In discussion based on previous subsections, we can summarize the advantages, which contain over current experiment setup methodologies as follows:

1) Our approach simultaneously addresses hardware on software integrity by limiting of resources.

2) The programmer's knowledge of program structure, coupled with the programmability of the FPGA, provides the application developer with flexibility in terms of the security of the system. The techniques available to any given application range provides limited protection with a minimal impact on performance to a full entanglement solution that provides the highest level of pronounced impact on performance.

3) Our design provides the ability to combine both hardware specific techniques with hardware-optimized implementation based method proposed recently.

4) The selection of the FPGA as our component minimizes additional hardware design. Moreover, the choice of FPGA architecture enables our system to be immediately applicable to current designs with reconfigurable logic on-chip, such as the Xilinx Virtex-5 architecture.

5) We have up to this point in this paper demonstrated the usefulness of our architecture by providing specific sub-program of how our come up can be used in a software and hardware schemes. What remains to be seen, however, is to the extent to which the insertion of the FPGA in the instruction memory hierarchy affects system and performance.

6) Our experiment shows, that for most of our benchmark the inclusion of the FPGA within the instruction stream incurs a performance penalty of not less than 20%, and that this number can be greatly improved upon with the utilization of unreserved reconfigurable resources for architectural optimizations, such as buffering and parallel pipelined programs.

B. Analyzing Performance

Since the majority of the techniques we leverage operate on a single program basic micro module at a time, it makes sense to analyze the effect of the FPGA on instruction memory hierarchy performance at that level of granularity. We begin by considering the replacement penalty of a single micro module of cache directly from a pipelined main memory. With a fixed micro module size and memory bus width (in terms of number of bytes), we can estimate this penalty as the sum of an initial non-sequential memory access time for the first bytes from memory plus the delay for a constant amount of sequential accesses, which would be proportional to the product of the micro module size with the inverse of the memory bus width.

Now, considering a basic micro module of instructions of a certain length in isolation from its surrounding program, we can estimate the number of instruction cache misses as the number of bytes in the basic micro module divided by the number of bytes in the cache micro module , as each instruction in the basic micro module would be fetched sequentially. Consequently the total instruction cache miss delay for a single fetched basic micro module can be approximated as the product of the number of fetches and average fetch delay as described above.

What effect would our software protection architecture have on performance? We identify two dominant factors: the occasional insertion of new instructions into the executable and the placement of the FPGA into the instruction fetch stream. For the first factor, the inserted instructions will only add a small number fetching of the modified basic micro module, since for most cases the number of inserted bytes will be considerably smaller than the size of the micro module itself. For the second factor, we note that the majority of the operations performed in the FPGA can be modelled as an increase in the instruction fetch latency. Assuming a pipelined implementation of whatever translation and validation is performed for a given configuration, we can estimate the delay for our FPGA as that of a similar bandwidth memory device, with a single non-sequential access latency followed by a number of sequential accesses. In the remainder of this section, we explain our experimental approach that demonstrates how these terms are affected by the QSP requirements.

These results clearly demonstrate the flexibility of our approach. With developer can evaluate both a tamper-resistant register encoding system that covers the entire application with a significant performance detriment, to quantum cryptography solution that has a much more measured impact on performance. While the results in QSP experiment results section show that the instructions generally not be the case when considering the larger applications that could be used with our approach. This, combined with the possibility that programmable FPGA resources will not all be allocated for software purposes motivates the configuration of these resources as performance-oriented architectural optimizations.

V. CONCLUSIONS AND FUTURE WORK

In conclusion, we have demonstrated the ability to design and control an quantum entangled pair (QEP) in experiment

for quantum signal processing (QSP). This paper clearly reveals the FPGA performance to control the equipments that able to prepare the entangled states with high quality. The difficulties of quantum states implementation is based on how can be produce two qubits states and make synchronization between them. However, there are principal designs using FPGA schemes that have been presented before this work, but this paper is verified, analyzed and discussed, how implemented quantum signal processing (QSP) with FPGA seem to be an attractive starting point for further research.

VI.   REFERENCES

[1] A. P. Shurupov, S. S. Straupe, S. P. Kulik, M. Gharib and M. R. B. Wahiddin , "Quantum state engineering with two qubirs: Application for deterministic QKD protocol,"A Letter Journal Exploring The Frontiers of Physics, EPL, 2009, 87 10008.

[2] P. W. Shor, and J. Preskill, "Simple proof of security of the BB84 quantum key distribution protocol," Phys. Rev. Lett., 85, 2000, pp. 441-444.

[3] Hoi-Kwong Lo, and H. F. Chau. "Unconditional Security of Quantum Key Distribution over Arbitrarily Long Distances," *Science* Vol. 283 no. 5410, 1999, pp. 2050-2056.

[4] Klyshko D. N. "Photons and Nonlinear Optics," Gordon and Breach, New York, 1988.