AIP | Review of Scientific Instruments

Field-programmable gate array based arbitrary signal generator and oscilloscope for use in slow light and storage of light experiments

Stanko N. Nikolić, Viktor Batić, Bratimir Panić, and Branislav M. Jelenković

Citation: Rev. Sci. Instrum. **84**, 063108 (2013); doi: 10.1063/1.4811147
View online: http://dx.doi.org/10.1063/1.4811147
View Table of Contents: http://rsi.aip.org/resource/1/RSINAK/v84/i6
Published by the AIP Publishing LLC.

Field-programmable gate array based arbitrary signal generator and oscilloscope for use in slow light and storage of light experiments

Stanko N. Nikolić,[1,a)] Viktor Batić,[2] Bratimir Panić,[1] and Branislav M. Jelenković[1]
[1]*Institute of Physics, University of Belgrade, Pregrevica 118, 11080 Belgrade, Serbia*
[2]*Custom Electronics Shack, Belgrade, Serbia*

(Received 10 April 2013; accepted 30 May 2013; published online 19 June 2013)

We present a field-programmable gate array (FPGA) based device that simultaneously generates two arbitrary analog voltage signals with the maximum sample rate of 1.25 MHz and acquires two analog voltage signals with the maximum sample rate of 2.5 MHz. All signals are synchronized with internal FPGA clock. The personal computer application developed for controlling and communicating with FPGA chip provides the shaping of the output signals by mathematical expressions and real-time monitoring of the input signals. The main advantages of FPGA based digital-to-analog and analog-to-digital cards are high speed, rapid reconfigurability, friendly user interface, and low cost. We use this module in slow light and storage of light experiments performed in Rb buffer gas cell.
 [http://dx.doi.org/10.1063/1.4811147]

I. INTRODUCTION

In many physics experiments, it is of utmost importance to quickly and accurately generate arbitrary pulses of laser light and measure changes in their shapes and amplitudes upon transmission through different media. Media with electromagnetically induced transparency (EIT) are of special interest since EIT allows for a fine control of the pulse propagation. Electromagnetically induced transparency[1] is a phenomenon characterized by a narrow transmission resonance of a signal light field resonant to a particular atomic transition. EIT can develop in different media, and therefore its origin can be quite different. While EIT in photonic crystals[2] and metamaterials,[3] is a purely classical phenomena, EIT in hot and cold atomic vapors is a quantum coherent effect.[4] The latter type of EIT is a manifestation of coherent population trapping (CPT),[5] or the "dark" state. This refers to a superposition of the two lower levels, coupled by the control and the signal laser fields in Raman resonance to the same upper level. This configuration is known as Λ atomic scheme, leading to the formation of the quantum state not coupled by the signal laser light. Atoms are quickly pumped into the "dark" state, and transmission peak is generated. EIT thus renders the otherwise opaque medium transparent. If the two lower levels are Zeeman sublevels of a given atomic state, one can use different polarization components of a single laser beam for the signal and the control fields. Such EIT is called Zeeman EIT.

Within a narrow spectral band of an EIT resonance, reduced light absorption is accompanied by extremely high dispersion of the index of refraction, $dn/d\omega$. This in turn leads to slow light phenomenon[6] which is a synonym for reduced group velocity of light pulses given by the following expression:

$$v_g = \frac{c}{n + \omega(dn/d\omega)}. \tag{1}$$

Due to strong control over absorption and dispersion of the laser light, the light pulses slowed to only 17 m/s are obtained in media with coherent EIT.[7] In addition, in the strong coherent interaction of light and atomic states, and by smoothly turning the control field off, photonic excitation can be adiabatically mapped into a purely atomic excitation. This atomic excitation can reversibly be restored back to an optical excitation when the control field is turned back on, thus completing the process known as storage of light.[8,9] The storage memory efficiency ξ is defined[10] as probability of retrieving an incoming single photon after storage in an ensemble with EIT

$$\xi = \frac{\int_{\tau}^{\tau+T} |E_{out}(t)|^2 dt}{\int_{-T}^{0} |E_{in}(t)|^2 dt}, \tag{2}$$

where E, T, and τ correspond to electric field, pulse duration, and storage time, respectively. The maximal storage time is determined by the decoherence rate.

Applications of slow light and storage of light effects in the fields of quantum memory, quantum information, and quantum repeaters[4,11–13] require that fractional pulse delay, defined as a ratio of the pulse delay to its duration, should be $\gtrsim$1. It is necessary to experiment with different waveforms of the control and signal fields in order to exceed this limit.[10,14–19] Several time dependent processes in atomic EIT in buffer gas cells are known to affect the width of EIT, and therefore delay of a light pulse. It was shown[20,21] that diffusion of the coherently prepared atoms out from the laser beam and, after multiple collisions with a buffer gas atoms, coming back into the laser field, narrows the profile of EIT resonance into sub-Lorentzian. This can be beneficial for longer pulse delay and storage efficiency if the atomic coherence evolution time, between the two separated excitation pulses of the Ramsey method,[22] is precisely timed. Enhancing the slow

a)Electronic mail: stankon@ipb.ac.rs

probe signal through its interference with transient pulses (precursors)[23] requires precise timing of multiple pulses, and frequent changes of the time sequence. Implementation of quantum protocols with series of pulses for the control of light pulses propagation through several separate EIT media, also needs complex time protocol. We investigate the influence of Ramsey effect on slow light and stored light properties in Rb buffer gas cell.

Main advantages of the FPGA based electronics for use in such applications are low cost and high speed operation enabled by massive parallel architecture. Laser locking circuit[24] performed by the FPGA showed nice benefits of alternative locking schemes. FPGA based servo controller with lock-in amplifier,[25] reconfigurable scanning probe/optical microscope,[26] and fast measurement and control system for the superconducting cavity of a flash free electron laser[27] have been also shown in the past.

We demonstrate below that the application of the FPGA based device linked with analog electronic circuit and a personal computer (PC) application with a graphical user interface (GUI), brings useful and quick control over many parameters in slow light and storage of light experiments. Pulse waveforms and their number in the sequence of pulses, precise timing between pulse formation and detection, and optimization of the control laser intensity can be quickly and conveniently controlled with FPGA. The role of FPGA is to synchronize all time events in the device and to simultaneously send data to the two digital-to-analog converters (DAC) and receive the data from the two analog-to-digital converters (ADC). In addition, FPGA takes care of storing all input and output data samples in the external memory. Usefulness of FPGA is demonstrated through measurements of dependencies of the group velocity of light pulses and of the storage efficiencies on the shape, intensity, and duration of both the signal and the control laser fields. This is performed by efficient control of the laser intensity and polarization and by numerical analysis of the input signals from the photodetectors. Components on the analog electronic circuit provide power supply as well as offsetting and amplifying of the input and the output voltage signals. PC application enables the user to make arbitrary output signals and to monitor and record the input signals in the graphical form in real-time with additional slow light analysis. Since FPGA and PC application communicate and exchange the data over time through universal serial bus (USB) connection, we can say that arbitrary signal generator and simple form of digital oscilloscope are realized in a single device. FPGA immediately responds to all software parameters that can be easily changed using the mouse and the keyboard. The speed of generating and acquiring voltage samples using DAC and ADC chips is independent of the speed of the attached computer.

In the final stage of preparing this paper, we noticed the work of Bowler *et al.*[28] in which authors developed FPGA based arbitrary waveform generator for controlling the trapped ions. However, there are differences between their and our device. We also created a custom signal generator but through embedded mathematical expressions in GUI. We additionally developed real time monitoring of the input signals together with slow light analysis.

II. FPGA OPERATION

The FPGA board we use is commercial Digilent Nexys-2 1200k board that houses a Spartan 3E-1200 FPGA chip, together with additional input and output headers. The FPGA master clock runs at 50 MHz. However, the reduced 25 MHz clock is used in our digital design.

In the slow light experiment, we produce a sequence of custom shaped light pulses by applying voltage pulses to the Pockels cell. Typically, the duration of the light pulses is in the range from 10 μs to several ms. This imposes the lower limit to the DAC time resolution to $\sim$1 μs. On the other hand, since the typical delay of a slow light pulse is $\Delta t_{delay} \geq 5$ μs, the ADC time resolution less than 1 μs is acceptable in our measurements. We introduce the signal called *ActionType* in our FPGA program to indicate the real time commands for the FPGA chip.

The first task is to program the FPGA so that the two DAC chips continuously generate periodic signals of the same duration T_{DAC}. Each DAC signal is determined by the total number of samples $N_{DAC} \leq 2000$, the array of samples and the sampling time $\Delta t_{DAC} \geq 800$ ns. The FPGA thus needs 4000 or less 12-bit numbers for both converters and the usage of 16 MB of external memory (RAM) organized in 16-bits units is necessary.[29] At the beginning, both DAC arrays generated by the PC application, are transferred to the RAM (*ActionType* = 1 or 2). Upon receiving the samples, continuous generation of the output signals starts (*ActionType* = 3).

The second FPGA task is to receive the data from the ADC chips with a sampling time Δt_{ADC} over the time interval $[T_a, T_b] \subseteq [0, T_{DAC}]$ (T_a and T_b are integer multiplies of Δt_{DAC}), and store the values in the RAM with the total number of samples $N_{ADC} \leq 10\,000$. FPGA needs at least 4 clock cycles (period of 80 ns) to perform a single read/write procedure over chosen RAM unit. Therefore, the key requirement is to organize the four RAM procedures:

- reading a data sample for DAC1
- reading a data sample for DAC2
- writing a data sample from ADC1
- writing a data sample from ADC2,

so they never overlap in time. This limits the ADC sampling time to $\Delta t_{ADC}^{min} = 4 \times 80$ ns $= 320$ ns. The following algorithm takes care about these requirements. After the start of a digital-to-analog conversion, at the time $n \cdot \Delta t_{ADC}$ ($n \in \mathbb{N}$), FPGA takes 24 bits of data from both ADC chips. The readout data from the ADC1 are written to RAM, and after four clock cycles the same procedure is applied for the ADC2 value. When both writing procedures are completed, FPGA can read new DAC samples from RAM. The third step is to ensure that no readings from ADC chips will occur while the readings of the next output samples are in progress. In this way, after completion of one DAC period, N_{ADC} input samples are stored in the memory and FPGA has to transfer them to the computer. While transfer is in progress, no ADC reading is performed. However, DAC chips work continuously as needed in the experiment. Once the data are transferred to PC, the FPGA starts receiving and writing the ADC bits again at the same memory locations. The FPGA algorithm

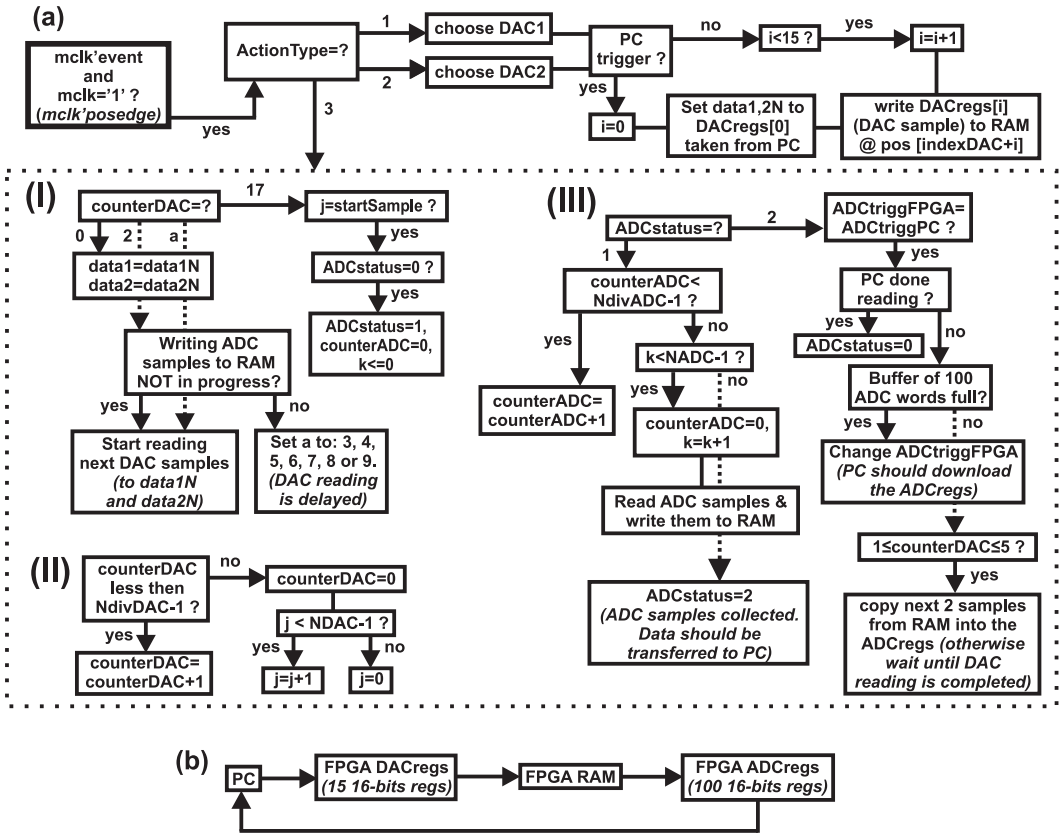

FIG. 1. (a) The simplified FPGA program algorithm and (b) the block diagram showing how the data between the PC and the FPGA board are exchanged. The explanation of all FPGA signals and registers are provided in the supplementary material.[31] Note that the blocks (I), (II), and (III) in (a) are all run in parallel by the FPGA chip.

for handling required procedures is schematically given in Fig. 1(a).

The exchange of the data between the PC and the FPGA is indirectly realized through FPGA registers as shown in Fig. 1(b). On the FPGA side, the communication is realized through slightly amended VHDL design (RegRdWr) of enhanced parallel port (EPP) protocol.[30] This module serves as the interface between the FPGA and the on-board USB controller. It has access to the FPGA registers that store the values of operating parameters which can be set up and changed by the user. We modify the original source code to extend the numbers of single byte registers to maximum value of 256. The organization of these registers is as follows: 23 bytes are related to the FPGA operating parameters for its program control, 30 bytes (15 words) are used for the multiple transfer of the two DACs data samples into the FPGA RAM of the Nexys-2 board, while other 200 bytes (100 words) are used as an intermediary memory buffer when both ADC data are transferred from the external FPGA memory to the PC. Three registers are spare.

III. ANALOG ELECTRONIC CIRCUIT

For generating analog voltage signals, Digilent PmodDA2 module containing two National Semiconductor DAC121S101 digital-to-analog converters were used. The PmodDA2 is connected to the eighth 6-pin peripheral connector of the Nexys-2 board. Both DAC chips share the same input clock pin enabling the user to generate two synchronous but independent output voltage signals in the range from 0 to 3.28 V. The DAC input is a 12-bit number between 0 and 4095. The data bits are transferred by the serial digital interface. The output voltages are driven to the amplifying and offsetting stages and, depending on the state of the DAC jumpers, may be in the following ranges: (0, 3.28) V, (0, 11) V, and (−7, 11) V. Both output impedances are 50 Ω. Analog-to-digital conversion of the input voltage signals is performed with the two Analog Devices AD9224 chips. Both input impedances are 1 MΩ. Input voltages may be in the ranges (−4.5, 0) V, (−11.5, 0) V, and (−7, +7) V depending on the ADC jumpers which define the amplification and the offset of the input signals. ADC 12-bits output data are transferred to the FPGA by parallel digital interface. The maximum sample rate is 40 MHz although we used effectively 2.5 MHz for each conversion, due to external memory issue as discussed in Sec. II. The ADC chips are connected to other seven 6-pin peripheral connectors of the FPGA board. The general schematic of the analog electronic circuit is presented in Fig. 2 (full schematic is provided in the supplementary material[31]). The photograph of the device is shown in Fig. 3 together with marks of the main elements.

IV. USER INTERFACE

The role of the PC application is to communicate with Digilent board in order to send data to and receive data from

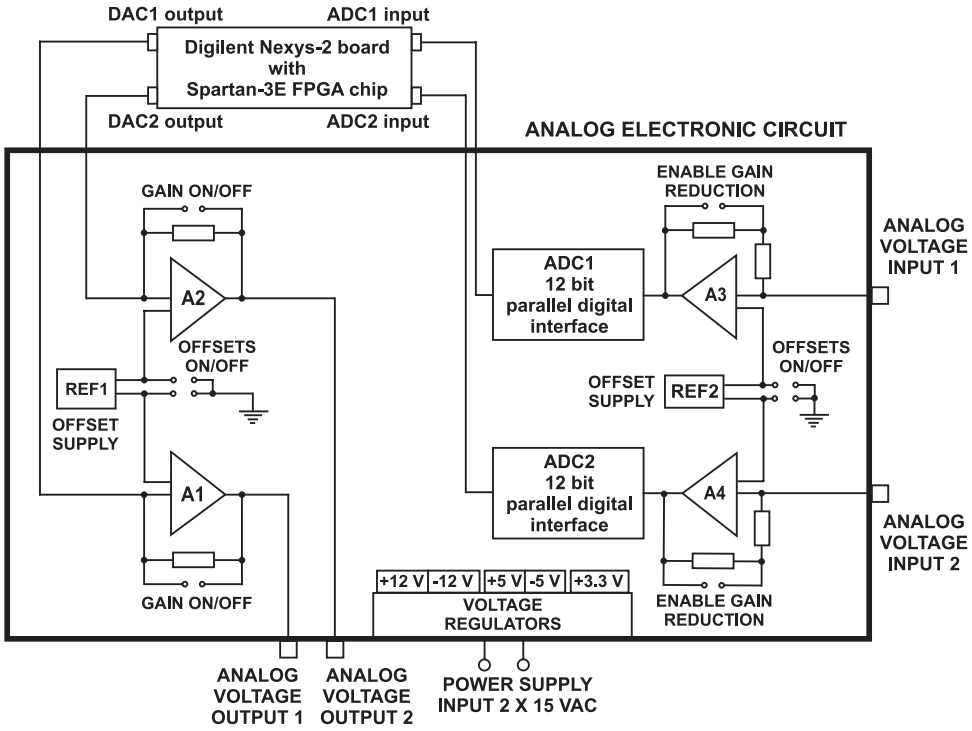

FIG. 2. The simplified scheme of the analog electronic circuit attached to the FPGA board. The circuit contains the power supply unit, the voltage regulators, the ADC chips, and the amplifying and offsetting stages for both input and output voltages.

the FPGA chip. Also, the program allows the user to set up the arbitrary output signals by using mathematical expressions, and to monitor and analyze the input signals in real-time. The PC application developed in Borland C++ Builder 6 consists of the main menu and three tabsheets.

To establish the PC side of the communication between the user interface program and the FPGA registers, the DPCUTIL library provided by Digilent is used. This library contains C++ functions that enable sending and receiving bytes to and from the registers on the FPGA. The FPGA board power supply, transfer of programmable bit files from the PC, and full communication between the PC and the FPGA registers are realized through a single USB port.

At the start of the program, the user connects to the FPGA using the first option from the main menu. The next step is to indicate the hardware settings of both DAC and ADC jumpers states. The application is then capable of sending correct numbers to the DAC chips in order to generate desired outputs and to convert the ADC numbers to voltage values at the inputs. After completion of these initial tasks, the user can specify the number of samples for the DAC and the ADC chips. The user is required to enter the period T_{DAC} of both output

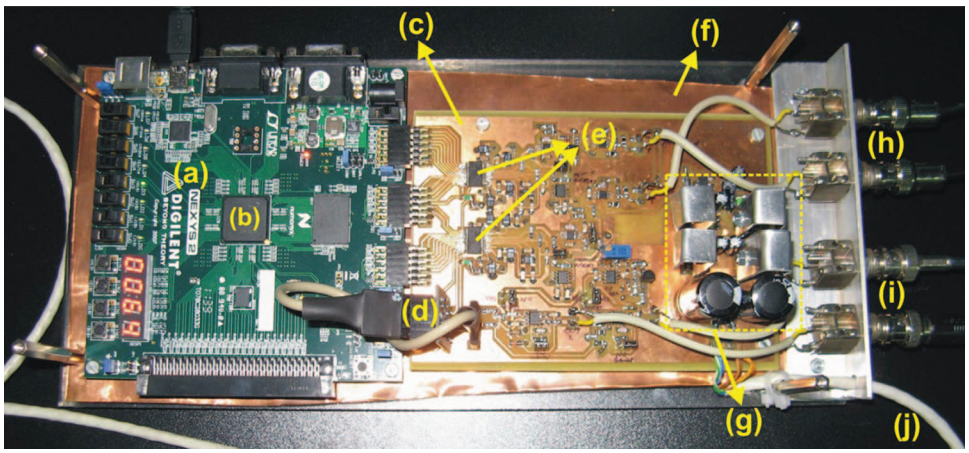

FIG. 3. The device photograph: (a) Digilent Nexys-2 board, (b) Spartan 3E-1200 FPGA chip, (c) printed circuit board, (d) two DAC chips, (e) two ADC chips, (f) the copper plate for grounding the board, (g) voltage regulators from 15 V AC down to +3.3 V, ±5 V, and ±12 V DC, (h) BNC connectors for the two analog voltage inputs, (i) BNC connectors for the two analog voltage outputs, and (j) 15 V AC power supply of the board.

signals which can be divided into several shorter time intervals T_1, T_2, …, T_n ($n \leq 10$). The idea is to shape voltage signals in each time interval (typical time sequence in our slow light experiment consists of voltage pulse during T_1, and zero voltage value during T_2, with the overall duration $T_{DAC} = T_1 + T_2$). Further, the user have to specify whether the time T_{DAC} is fixed so that flexible time parameter $T_n = T_{DAC} - T_1 - \ldots - T_{n-1}$ is changed accordingly. Otherwise, the overall signal duration is $T_1 + T_2 + \ldots + T_n$. Besides these time parameters, the user may specify other parameters relevant for the experiment. We developed the algorithm that calculates the real value of a multiline string expression. Therefore, user can enter mathematical expressions in the two edit controls, for waveform shaping of both output signals. Formulae can include predefined parameters, independent time variable t, and several of the most frequent mathematical functions. After this step, the user needs to enter numerical values of all parameters. The time interval T_{DAC} is then divided into N_{DAC} time points $t_i = i \cdot T_{DAC}/N_{DAC}$ at which output voltages $V_{DAC1}(t_i)$ and $V_{DAC2}(t_i)$ are calculated after successful compilation of both expressions. Corresponding unsigned 16-bit numbers as inputs to the DAC chips are displayed in the first tabsheet table and then sent to the FPGA. Each parameter value can be quickly changed using keyboard. The software immediately calculates new output voltages and resends the data to the FPGA. There are also options to save and load configuration files containing formulae and parameters' names and values. DAC chips controls are placed on the first tabsheet of the program.

We use the polling operation to gather ADC data from the FPGA external memory. Program checks the FPGA status every 5 ms and if the memory buffer is fully loaded with the input data, the FPGA transfers packets of 100 words into its registers. From the registers data are sent to the PC. This procedure is being repeated until all the data are transferred to the computer. Real-time monitoring of the input signals is performed by using integrated graphic functions in Borland C++ Builder. In this way, a simple digital oscilloscope is realized. Upon receiving new data, the program clears, piece-by-piece, the old graphs in both channels and draws plots with the new data. At any time, user can monitor raw or averaged signals, and change the voltage scales and offsets. These settings are easily performed by the use of mouse and keyboard. Frequently in the experiments, only fractions of the input signals are of interest. Therefore, we enable the user to indicate time interval $[T_a, T_b]$ over which the ADC chips will sample the data either by using track bars or by typing formulae for T_a and T_b. Since the useful time interval is effectively reduced in this way, higher time resolution and increased acquisition speed are achieved. Controls for the ADC chips are placed on the second tabsheet of the program.

We also added the option of dynamic slow light analysis. When the user double-clicks on the scope (or presses F2), program calculates time and voltage coordinates at the given point, time delay of the slow pulse, and corresponding group velocity. These results are recorded for any set of parameters' values and are written in the grid on the third tabsheet of the program. The user can also export all input and slow light data into textual files.

V. SOURCE CODE

The VHDL source code for the FPGA chip and C++ source code for the user interface application are provided in the supplementary material.[31] We used Xilinx Webpack ISE 14.1 and Borland C++ Builder 6 compilers. The compiled WINDOWS executable and FPGA bit files are also provided.

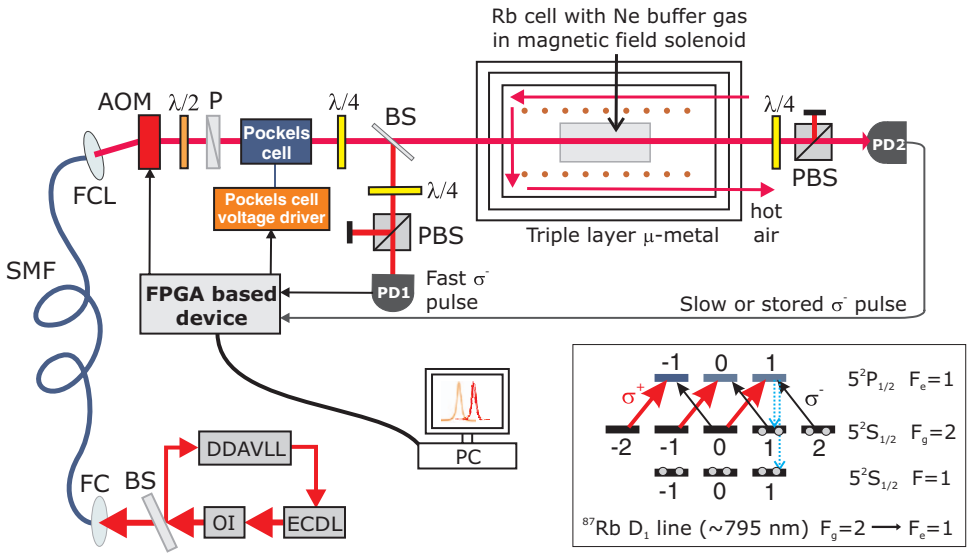

FIG. 4. Experimental setup and the atomic transition used in the experiment. ECDL: external cavity diode laser; OI: optical isolator; DDAVLL: Doppler-free dichroic atomic vapor laser lock; BS: beam splitter; FC: fiber coupler; SMF: single-mode fiber; FCL: fiber collimator; AOM: acousto-optic modulator; P: polarizer; PBS: polarizing beamsplitter; PD: large area photodetector. Output voltage signals from FPGA device control the AOM and the Pockels cell, while signals from the two photodetectors are input signals to the FPGA device.

VI. PERFORMANCE IN THE SLOW LIGHT AND THE STORAGE OF LIGHT EXPERIMENTS

The setup for slow light and storage of light experiments is shown in Fig. 4. Our home-built external cavity diode laser uses semiconductor laser diode (Eagleyard Photonics EYP-RWE-0790-04000-0750-SOT01-0000). The diode is powered by a low noise current supply and temperature controller (Thorlabs ITC102). After passing through the optical isolator (Isowave I-80-T4-L), the laser beam is frequency locked to the hyperfine $F_g = 2 \rightarrow F_e = 1$ transition of the D_1 line in ^{87}Rb by using the Doppler-free dichroic atomic vapor laser lock (DDAVLL) method.[32,33] The laser beam is introduced by the fiber coupler (Newport F-916, MV-10X, FPH-CA4) to a single mode fiber (Oz optics SMJ-3A3A-852-5/125-3-1). After the fiber, the laser beam passes through AOM (Isomet 1206C) and the diffraction maximum of -1 order is used in the experiment. AOM is used as a fast optical switch, to turn the laser beam on and off. The driver of the AOM (Isomet 530-C) has modulation pin which enables the amplitude control of the RF wave passing through the crystal and hence the control of the selected diffraction maximum power. The output voltage from the FPGA device controls the power of the first AOM diffraction maximum: $V_{DAC1} = 0$ V gives $P_{laser} = 0$ and $V_{DAC1} = 3.28$ V gives maximal P_{laser} value. Linear laser beam polarization is adjusted with polarizer (Thorlabs GTH5-B). We use Pockels cell with temperature stabilized x-cut MgO:$LiNbO_3$ crystal (produced by Castech) as the birefringent element to rotate the polarization of the linearly polarized laser light. Pockels cell and $\lambda/4$ plate (Thorlabs WPQ05M-780) are used to set the polarization of the laser beam before entering the Rb cell. With no voltage applied to the Pockels cell, a pure σ^+ polarization of the laser beam is obtained behind the Pockels cell and the $\lambda/4$ plate. By applying a voltage U to the $LiNbO_3$ crystal, the phase difference Γ between the two orthogonal components of a linearly polarized laser beam is induced due to Pockels effect.[34] Using the Jones vectors representation of the polarization states, it can be easily derived that the relative power of the σ^- pulse is

$$P_{\sigma^-}/P_{laser} = \sin^2\frac{\Gamma}{2} = \sin^2\frac{C\cdot U}{2}. \tag{3}$$

The parameter $C = 7.6 \times 10^{-3}$ V^{-1} from the above equation is the calibration constant of the crystal. If one wants to obtain a weak and time dependent σ^- polarization component of a laser beam $\eta(t) = P_{\sigma^-}(t)/P_{laser}$ ($\eta \lesssim 30$ %), voltages up to 150 V should be applied to the crystal. We use the home-made high-speed high-voltage amplifier, with amplification of $G = 22.1$. Output analog voltage V_{DAC2} up to 7 V from the FPGA device serves as input to the amplifier, and the amplified signal is applied to the Pockels cell. According to Eq. (3), the output voltage from the FPGA device is

$$V_{DAC2}(t) = \frac{2}{C\cdot G}\arcsin\sqrt{\eta(t)}. \tag{4}$$

In our experiments, we produce the pulses of σ^- polarized light with the Gaussian profile. This pulse propagates together with the strong σ^+ component of the laser beam through the Rb cell containing a natural abundance of Rb isotopes and 30 Torr of Ne buffer gas. The cell is 8 cm long and has a 25 mm diameter. It is placed in a plastic box and heated to $\sim$82 °C by hot air circulating around the cell. The Rb cell is placed in the solenoid for producing the longitudinal magnetic field and is shielded by the triple layer μ-metal which reduces the stray magnetic fields below 10 nT. Because of the presence of the strong σ^+ field, the resonant σ^- pulse can freely propagate through the otherwise opaque medium due to EIT effect, but with a substantially reduced group velocity. In order to extract only σ^- polarization from the laser beam, we use detection system consisting of a $\lambda/4$ plate followed by a PBS (Thorlabs PBS102) and a home-made photodetector with the large area photodiode (IHTM Belgrade FD80N). The two detection systems are placed before and after the Rb cell so the fast and the slow σ^- polarization pulses, previously split from a single laser beam with a beam splitter (Thorlabs EBS2), can be measured. We use the FPGA device to gather signals from the two photodetectors. These input data are written to the external FPGA memory for the purpose of real-time monitoring and slow light analysis.

We perform the slow light measurements in the two configurations. In the first one, the power of AOM diffraction maximum is kept constant over time, while the Pockels cell generates single Gaussian-shaped σ^- polarization pulse for one DAC period. In the second configuration, we investigate the Ramsey effect on slow light propagation, i.e., the influence of repeated interaction of σ^- pulses with coherently prepared atoms in the dark state on the fractional time delay of the σ^- pulse. We first prepare Rb atoms into the dark state with a weak σ^- pulse of constant intensity (Π-shaped) over the time T_1. Then, we turn the laser beam off by the AOM for the time T_2 to enable a free time evolution of the dark state. Next, we synchronously turn the laser beam back on and generate a weak Gaussian-shaped σ^- pulse of duration T_3 with relative peak power η. We keep pure σ^+ polarization of a laser beam for a long time T_4 to "reset" and optically pump the atoms back in the ground state. Output signals from the FPGA device, in both configurations, control both the AOM and the Pockels cell and have the same overall duration $T_{DAC} = T_1 + T_2 + T_3 + T_4$. According to Eq. (4), output voltages are tailored by following mathematical expressions edited in the PC application:

$$\begin{aligned}
V_{DAC1}(t) = {} & 3.28\cdot\mathrm{sgn}(t)\cdot\mathrm{sgn1}(T_1 - t)\\
& + \mathrm{Ctrl1}\cdot\mathrm{sgn}(t - T_1)\cdot\mathrm{sgn1}(T_1 + T_2 - t)\\
& + 3.28\cdot\mathrm{sgn}(t - T_1 - T_2)\cdot\mathrm{sgn1}(T_1 + T_2 + T_3 - t)\\
& + 3.28\cdot\mathrm{sgn}(t - T_1 - T_2 - T_3)\\
& \cdot\mathrm{sgn1}(T_1 + T_2 + T_3 + T_4 - t),\\
V_{DAC2}(t) = {} & \mathrm{Ctrl2}\cdot\tfrac{2}{C\cdot G}\arcsin\sqrt{\eta}\cdot\mathrm{sgn}(t)\cdot\mathrm{sgn1}(T_1 - t)\\
& + 0.00\cdot\mathrm{sgn}(t - T_1)\cdot\mathrm{sgn1}(T_1 + T_2 - t)\\
& + \tfrac{2}{C\cdot G}\arcsin\sqrt{\eta}e^{-35\cdot(t - T_1 - T_2 - T_3/2)^2/T_3^2}\\
& \cdot\mathrm{sgn}(t - T_1 - T_2)\cdot\mathrm{sgn1}(T_1 + T_2 + T_3 - t)\\
& + 0.00\cdot\mathrm{sgn}(t - T_1 - T_2 - T_3)\\
& \cdot\mathrm{sgn1}(T_1 + T_2 + T_3 + T_4 - t),
\end{aligned} \tag{5}$$

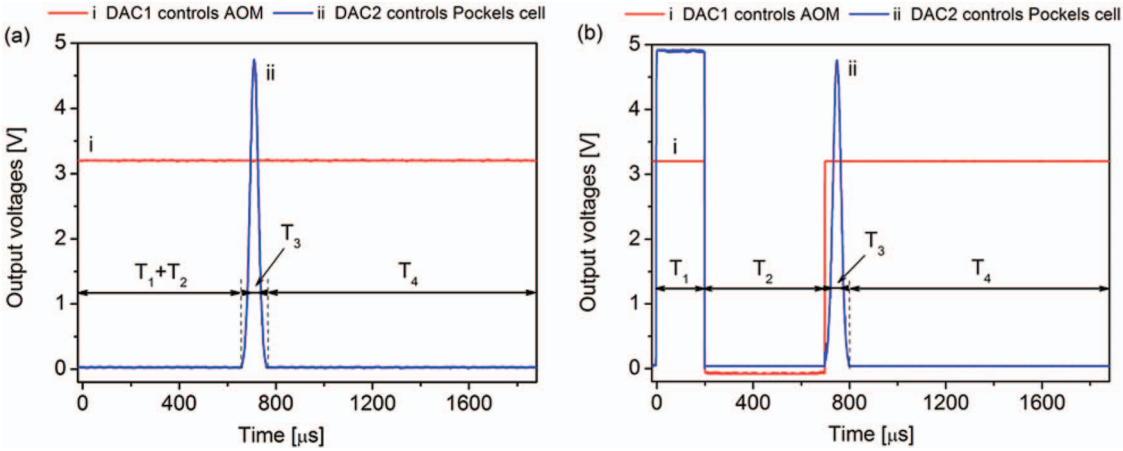

FIG. 5. Two output DAC signals from the FPGA device that control the AOM and the Pockels cell in the slow light experiment: (a) single Gaussian σ^- light pulse of duration T_3 with AOM turned on all the time, and (b) preparation Π-shaped σ^- light pulse of duration T_1 followed by the AOM turned off during time T_2 and Gaussian σ^- light pulse of duration T_3.

where sgn(t) and sgn1(t) denote signum functions (sgn(t) = 1 for $t \geq 0$ and sgn(t) = 0 otherwise, while sgn1(t) = 1 for $t > 0$ and sgn1(t) = 0 otherwise) and serve as mathematical analogs to if-statements in programming practice. The above expressions use control parameters Ctrl1 and Ctrl2 thus covering both experimental configurations: the first one is obtained for Ctrl1 = 3.28 and Ctrl2 = 0.00, and the second one for Ctrl1 = 0.00 and Ctrl2 = 1.00. Output signals from the FPGA device which control AOM and the Pockels cell are shown in Fig. 5. The longitudinal magnetic field is ∼1 μT in both configurations. Signals from the two photodetectors measuring fast (reference) and slow σ^- light pulses are inputs to the FPGA device and are shown in Fig. 6. In the first configuration, the light pulse is delayed for $\Delta t = 12.5$ μs and the group velocity is $v_g = 6.4$ km/s with a large absorption within the cell. In the second configuration, the time delay is shorter ($\Delta t = 8.24$ μs and $v_g = 9.7$ km/s) but the transparency is substantially enhanced.

In the storage of light experiment, we have two similar configurations like in the experiment with the slow light. In the first configuration, a single Gaussian-shaped σ^- polarization pulse of duration $T_3 + T_4$ is generated and the AOM is smoothly turned off at a peak of the pulse to ensure that the most of the energy of slowed pulse is within the cell. The laser beam is kept off for the storage time $\tau = T_4 + T_5$ after which the light is turned back on. In the second configuration, the preparation Π-shaped σ^- pulse of length T_1 is applied to the Pockels cell, after which the AOM turns the laser off for the time T_2. After the time $T_1 + T_2$, both output signals have the same waveforms as in the first configuration. Output voltages are tailored by the following mathematical expressions:

$$\begin{aligned} V_{DAC1}(t) = {} & 3.28 \cdot \mathrm{sgn}(t) \cdot \mathrm{sgn1}(T_1 - t) \\ & + \mathrm{Ctrl1} \cdot \mathrm{sgn}(t - T_1) \cdot \mathrm{sgn1}(T_1 + T_2 - t) \\ & + 3.28 \cdot \mathrm{sgn}(t - T_1 - T_2) \cdot \mathrm{sgn1}(T_1 + T_2 + T_3 - t) \\ & + 0.00 \cdot \mathrm{sgn}(t - T_1 - T_2 - T_3) \\ & \cdot \mathrm{sgn1}(T_1 + T_2 + T_3 + T_4 + T_5 - t) \\ & + 3.28 \cdot \mathrm{sgn}(t - T_1 - T_2 - T_3 - T_4 - T_5) \\ & \cdot \mathrm{sgn1}(T_1 + T_2 + T_3 + T_4 + T_5 + T_6 - t), \\ V_{DAC2}(t) = {} & \mathrm{Ctrl2} \cdot \tfrac{2}{C \cdot G} \arcsin \sqrt{\eta} \cdot \mathrm{sgn}(t) \cdot \mathrm{sgn1}(T_1 - t) \\ & + 0.00 \cdot \mathrm{sgn}(t - T_1) \cdot \mathrm{sgn1}(T_1 + T_2 - t) \end{aligned} \tag{6}$$

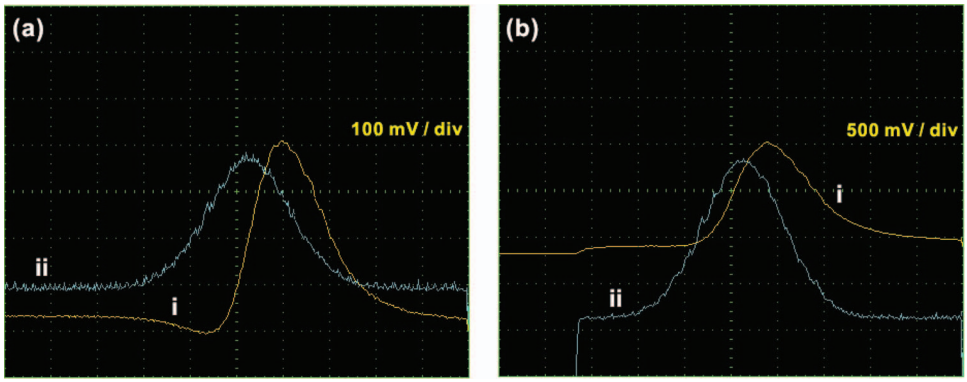

FIG. 6. Screenshots of real-time monitoring of the two photodetectors' signals that measure slow (i) and fast (ii) σ^- light pulses: (a) no preparation pulse with the AOM turned on all the time, (b) preparation Π-shaped σ^- light pulse followed by the AOM turned off and Gaussian-shaped σ^- pulse. Parameters' values: $T_{DAC} = 2$ ms, $T_1 = 200$ μs, $T_2 = 500$ μs, $T_3 = 100$ μs, $\eta = 0.15$, and $P_{laser} = 1.3$ mW. Note the different voltage scalings for curves (i) in (a) and (b).

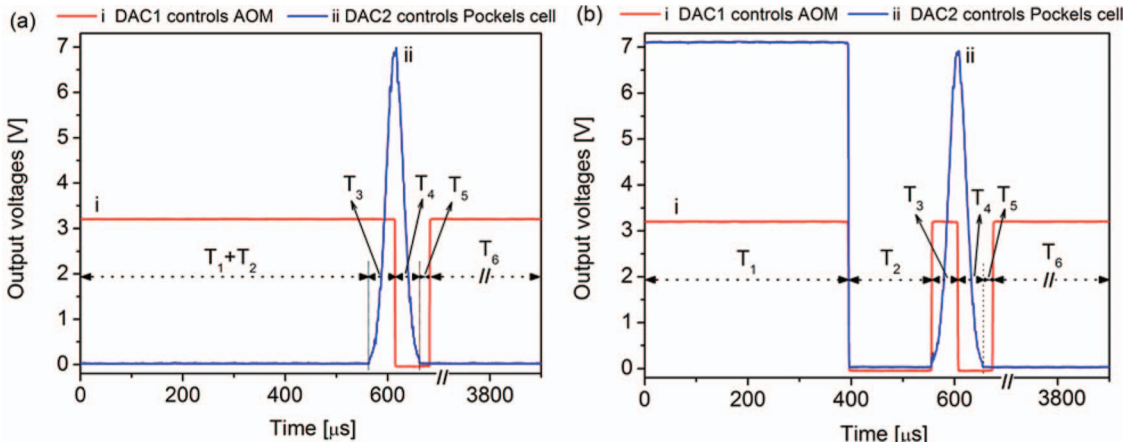

FIG. 7. Two output DAC signals from the FPGA device that control the AOM and the Pockels cell in the storage of light experiment: (a) single Gaussian-shaped σ^- light pulse of duration $T_3 + T_4$ with AOM turned off for the storage time $T_4 + T_5$, and (b) preparation Π-shaped σ^- light pulse of duration T_1 followed by the AOM turned off during the time T_2 prior to the storage of light signals' conditioning.

$$+ \frac{2}{C \cdot G} \arcsin \sqrt{\eta} e^{-35 \cdot (t - T_1 - T_2 - \frac{T_3 + T_4}{2})^2 / (T_3 + T_4)^2}$$

$$\cdot \operatorname{sgn}(t - T_1 - T_2) \cdot \operatorname{sgn1}(T_1 + T_2 + T_3 + T_4 - t)$$

$$+ 0.00 \cdot \operatorname{sgn}(t - T_1 - T_2 - T_3 - T_4)$$

$$\cdot \operatorname{sgn1}(T_1 + T_2 + T_3 + T_4 + T_5 + T_6 - t).$$

Note that in both Eqs. (5) and (6) some addends are multiplied with zero to indicate the time intervals over which the zero voltages are required in the experiment. Output signals from the FPGA device which control the AOM and the Pockels cell are shown in Fig. 7. The longitudinal magnetic field is $\sim 1\ \mu$T in both configurations. Signals from the two photodetectors measuring fast and stored σ^- light pulses are shown in Fig. 8. Light storage efficiencies obtained in two experimental configurations are $\xi_1 = 14.5\%$ and $\xi_2 = 20\%$. Enhanced storage efficiency of a σ^- polarization pulse in the second configuration is due to higher transparency of resonant laser light, i.e., combined effects of preparation pulse and free evolution of the dark state in Rb atoms. This technique, known as Ramsey technique, improves the storage of light efficiency and will be investigated in more detail in our forthcoming work.

VII. CONCLUSION

We described the FPGA based device, consisting of an arbitrary signal generator and a simple digital oscilloscope. This device synchronously generates two analog voltage signals and acquires two analog voltage signals. Analog electronic circuit connected to the FPGA provides the desired offsetting and amplifying of both input and output voltages. All input and output data are written in the external memory of the FPGA board. The PC application which communicates and exchanges the data with the FPGA, enables the user to model waveforms of output signals through mathematical expressions including predefined parameters, time variable, and most frequent built-in mathematical functions. Quick and easy change of any parameter or formula is immediately sent to the FPGA. Simultaneously, the FPGA records the data from the inputs and right after completion, sends them to the PC application, which presents the data in a graphical form. This real-time monitoring may be further adjusted by the user, so the raw or averaged data over different time and voltage intervals could be monitored and analyzed. The detailed schematic of the analog electronic circuit, as well as VHDL source code for the FPGA chip and Borland C++ Builder code for a graphical user interface are available in the supplementary material.[31] We demonstrate effectiveness of the FPGA device

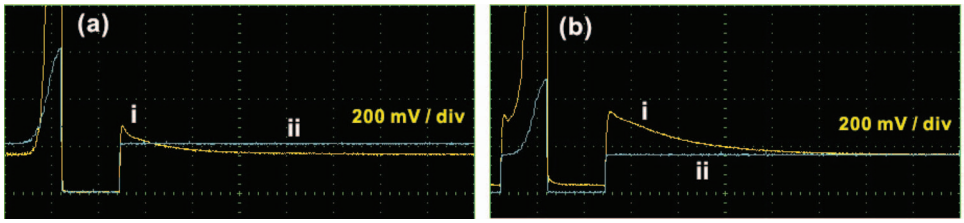

FIG. 8. Screenshots of real-time monitoring of the two photodetectors' signals that measure slow (i) and fast (ii) σ^- light pulses: (a) no preparation pulse with the AOM turned on all the time, (b) preparation Π-shaped σ^- light pulse followed by the AOM turned off and Gaussian σ^- pulse. Parameters' values: $T_{DAC} =$ 4 ms, $T_1 = 400\ \mu$s, $T_2 = 160\ \mu$s, $T_3 = 50\ \mu$s, $T_4 = 50\ \mu$s, $T_5 = 20\ \mu$s, $\eta = 0.30$, and $P_{laser} = 1.7$ mW.

in the slow light and the storage of light experiments. Outputs of the device control the AOM and the Pockels cell. In the first configuration of both experiments, we measure the propagation of a single Gaussian-shaped polarization pulse through the Rb cell. In the second configuration, prior to Gaussian-shaped pulse, we generate Π-shaped preparation pulse followed by the free evolution of the coherently prepared atoms. The signals from the two photodetectors that measure fast and slow light pulses are inputs to the FPGA device. Monitoring of the input signals showed us that enhanced transparency and storage efficiency could be obtained in the second configuration due to repeated interaction of the coherently prepared atoms with the laser light.

ACKNOWLEDGMENTS

The authors gratefully acknowledge N. M. Lučić, M. Minić, and Arne Schwettmann for fruitful discussions. This work was supported by Grant Nos. 45016 and 171038 of the Ministry of Education and Science of the Republic of Serbia and Scopes JRP IZ7370_127942.

[1]K. J. Boller, A. Imamoğlu, and S. E. Harris, Phys. Rev. Lett. **66**, 2593 (1991).
[2]T. F. Krauss, J. Phys. D: Appl. Phys. **40**, 2666 (2007).
[3]H. Xu, Y. Lu, Y. Lee, and B. S. Ham, Opt. Express **18**, 17736 (2010).
[4]M. Fleischhauer, A. Imamoğlu, and J. P. Marangos, Rev. Mod. Phys. **77**, 633 (2005).
[5]E. Arimondo, Prog. Opt. **35**, 257 (1996).
[6]J. B. Khurgin and R. S. Tucker, *Slow Light Science and Applications* (CRC Press, 2009).
[7]L. V. Hau, S. E. Harris, Z. Dutton, and C. H. Behroozi, Nature (London) **397**, 594 (1999).
[8]D. F. Phillips, A. Fleischhauer, A. Mair, R. L. Walsworth, and M. D. Lukin, Phys. Rev. Lett. **86**, 783 (2001).
[9]M. Fleischhauer and M. D. Lukin, Phys. Rev. Lett. **84**, 5094 (2000).
[10]I. Novikova, R. L. Walsworth, and Y. Xiao, Laser Photonics Rev. **6**, 333 (2012).
[11]M. D. Lukin, Rev. Mod. Phys. **75**, 457 (2003).
[12]A. B. Matsko, O. Kocharovskaya, Y. Rostovtsev, G. R. Welch, A. S. Zibrov, and M. O. Scully, Adv. At., Mol., Opt. Phys. **46**, 191 (2001).
[13]L. M. Duan, M. D. Lukin, J. I. Cirak, and P. Zoler, Nature (London) **414**, 413 (2001).
[14]A. V. Gorshkov, A. André, M. Fleischhauer, A. S. Sørensen, and M. D. Lukin, Phys. Rev. Lett. **98**, 123601 (2007).
[15]I. Novikova, A. V. Gorshkov, D. F. Phillips, A. S. Sørensen, M. D. Lukin, and R. L. Walsworth, Phys. Rev. Lett. **98**, 243602 (2007).
[16]A. V. Gorshkov, A. André, M. D. Lukin, and A. S. Sørensen, Phys. Rev. A **76**, 033804 (2007).
[17]A. V. Gorshkov, A. André, M. D. Lukin, and A. S. Sørensen, Phys. Rev. A **76**, 033805 (2007).
[18]A. V. Gorshkov, A. André, M. D. Lukin, and A. S. Sørensen, Phys. Rev. A **76**, 033806 (2007).
[19]A. V. Gorshkov, T. Calarco, M. D. Lukin, and A. S. Sørensen, Phys. Rev. A **77**, 043806 (2008).
[20]Y. Xiao, I. Novikova, D. F. Phillips, and R. L. Walsworth, Phys. Rev. Lett. **96**, 043601 (2006).
[21]Y. Xiao, I. Novikova, D. F. Phillips, and R. L. Walsworth, Opt. Express **16**, 14128 (2008).
[22]N. F. Ramsey, *Molecular Beams* (Oxford University Press, 1956).
[23]H. Jeong and S. Du, Opt. Lett. **35**, 124 (2010).
[24]A. Schwettmann, J. Sedlacek, and J. P. Shaffer, Rev. Sci. Instrum. **82**, 103103 (2011).
[25]G. Yang, J. F. Barry, E. S. Shuman, M. H. Steinecker, and D. DeMille, JINST **7**, P10026 (2012).
[26]D. B. Nowak, A. J. Lawrence, Z. K. Dzegede, J. C. Hiester, C. Kim, and E. J. Sánchez, Rev. Sci. Instrum. **82**, 103701 (2011).
[27]K. T. Pozniak, Meas. Sci. Technol. **18**, 2336 (2007).
[28]R. Bowler, U. Warring, J. W. Britton, B. C. Sawyer, and J. Amini, Rev. Sci. Instrum. **84**, 033108 (2013).
[29]R. E. Haskell and D. M. Hanna, *Digital Design Using Digilent FPGA Boards VHDL/Active-HDL Edition* (LBE Books, Rochester Hills, MI, 2009).
[30]See discussion and material at http://www.makestuff.eu/wordpress/nexys2-nexus-part-1-the-epp-emulator about VHDL design of enhanced parallel port protocol for Digilent Nexys-2 board.
[31]See supplementary material at http://dx.doi.org/10.1063/1.4811147 for the schematic of the analog electronic circuit and complete VHDL and C++ source code of our FPGA based arbitrary signal generator and digital oscilloscope.
[32]G. Wasik, W. Gawlik, J. Zachorowski, and W. Zawadzki, Appl. Phys. B **75**, 613 (2002).
[33]T. Petelski, M. Fattori, G. Lamporesi, J. Stuhler, and G. M. Tino, Eur. Phys. J. D **22**, 279 (2003).
[34]B. E. A. Saleh and M. C. Teich, *Fundamentals of Photonics* (John Wiley & Sons, Inc., Hoboken, NJ, 2007).