Designing a real-time feedback loop for single atom trapping and cooling

Gustaf Downs*
(Dated: August 22, 2012)

In order to scale up experiments on quantum few-body systems of single atoms in optical dipole traps, it is necessary to be able to verify that all trap sites have been successfully loaded. This involves being able to detect and determine the location of single atoms on timescales that are both practical and compatible with the trap lifetime. The following design employed existing computer-hardware in Professor Cindy Regal's lab, but required establishing new pathways of communication between devices. The result is a real-time feedback communication system for verifying and communicating the presence of up to 21 atoms in an arbitrary array that downloads and processes images taken with an Andor iXon EM+ CCD camera and outputs TTL signals depending on whether it detects atoms at user-defined pixels in the images. The time it takes from image download to TTL output is < 3 ms.

I. APPLICATIONS OF SINGLE ATOM-TRAPPING AND COOLING

Ultra-cold atoms are used in a wide variety of research experiments at the frontier of contemporary physics. Atoms serve as a useful tool for developing new technologies due to our comprehensive grasp of how to control both atoms' internal and external quantum states with lasers. Atoms can be tuned to either ignore the presence of their nearest neighbors or interact quite strongly, allowing for the potential to simulate many condensed matter systems using optical lattices of atoms. Atoms also satisfy all of the necessary requirements to serve as qubits—the quantum computing analog of a classical computing bit. For these reasons and many more, ultra-cold atoms are at the forefront of research in the next generation of atomic clocks, precision measurement past the quantum limit, quantum computing, and quantum simulation.

For much of this research, it is useful to be able to work with small numbers of neutral atoms. In these applications, there are essentially two approaches to working with ultra-cold atoms: a top-down approach where a large ensemble of atoms is cooled into a Bose-Einstein condensate (BEC) and then manipulated from there or a bottom-up approach where single atoms are first isolated and then cooled in optical dipole traps before being manipulated into new formations. Both approaches have their pros and cons, making both approaches useful for some applications. For experiments where it is sufficient to make a single measurement over a large ensemble of atoms in a static optical lattice (ie. quantum simulation), the top-down approach may be more efficient; for experiments involving the ability to manipulate single atoms' positions (ie. quantum computing), the bottom-up approach is necessary.

For the bottom up approach to be efficient, single atoms must be able to be loaded into a series of dipole traps efficiently and reliably. However, current techniques for loading single atoms into optical dipole traps have a maximum trapping efficiency of around 85%, making a brute force approach to probabilistically loading multiple atoms in the same loading cycle incredibly inefficient. It is necessary to have some way of verifying the number of atoms successfully trapped on a given loading cycle, as well as their positions, so that those atoms' traps can be made deeper to prevent escape during the next loading cycle. This verification system can either be done in hardware, using photodetectors and semiconductor diodes, or in software. Hardware circuitry has the benefit of being able to detect atoms almost instantaneously, but doesn't allow for reconfiguration for different runs with arbitrary numbers of atoms or arbitrary positions very easily. Software allows for more customizable experimental runs but it can be challenging to download data from a detector and process it on a time-scale that isn't prohibitive to the rest of the experiment.

The following paper discusses the real-time feedback loop that was implemented into Regal Lab's single-atom trapping and cooling experiment at JILA, University of Colorado-Boulder during summer 2012. First, a brief overview of the current experiment is given. Next, the relevant components of the signal chain are discussed, along with the proposed protocol for communicating information regarding the presence of atoms in an image. To conclude, the process of implementation and optimization is explained, along with final results.

II. CURRENT EXPERIMENTAL PROCEDURE

The goal of the existing experimental procedure is to trap a single Rubidium-87 atom in an optical dipole trap and consistently cool it to the vibrational ground state of its confining potential. This is accomplished in two experimental stages. The first stage's sole purpose is to load an atom into the dipole trap, in which an atom is successfully loaded only 50% of the time. After this stage is complete, an image is taken to see if an atom was successfully loaded into the trap. This image is later used to go back and post-select only the data points that were taken on runs where an atom was successfully loaded. The second stage is different depending on what the goal

* gdowns@mit.edu

of the experimental run is, but typically involves using resolved-sideband Raman cooling to try to cool the atom down to its vibrational ground state. The following section is a non-technical description of these two stages.

II.1. Loading phase

Although it is unnecessary to start this experiment with a BEC, it is also impossible to capture a single room-temperature atom, so first a cloud of atoms are cooled and collected in a magneto-optical trap (MOT). A Rubidium kicker maintains a constant vapor pressure of Rubidium in a glass cell, which is held at high vacuum. This constant Rubidium background pressure ensures that there are atoms available to fill the MOT that is created in the center of the glass cell. The MOT serves to cool the atoms, as well as increase the density of the atomic vapor in the cell—on the order of 10^6 atoms in a spherical cloud with a diameter of $\sim$ 1mm.

From there, the magnetic field at the site of the cloud is turned to zero and the ensemble is cooled by polarization gradient cooling (PG or sisyphus cooling), using the same laser beams used to generate the MOT. After PG cooling, the atoms in the cloud are on the order of 1 μK.

The next step is to turn on the optical dipole trap beam. This beam is first blown up to a radius of about 2 cm before being focused down to a waist of $\sim$.76 microns through a high numerical aperture lens (NA = 0.65). This beam serves to generate a spatially dependent light shift in the Rubidium atoms that causes them to be attracted to the small region of high electric field intensity at the focal point of this beam. While near the focal point of this beam, atoms are actively PG cooled into the trap or else they would simply pass right through. During the amount of time that the PG cooling remains active, roughly 10 to 20 atoms are collected in the dipole trap.

In order to go from this number of atoms in the dipole trap to the desired number—one—a method is used that maps starting numbers of atoms to either 1 or 0 final number of atoms. Essentially, light is shone onto the atoms which causes them to form pseudo-molecular pairs. When this light is turned off, all of the energy that was stored in the molecule to bind the two atoms together gets converted to kinetic energy, kicking both atoms out of the trap. However, if there is an odd number of atoms to begin with, there will be one atom that is left not in a molecular pair. When the light is turned off, this atom remains in the trap. In this way, the "kill-pairs" sequence maps even numbers of initial atoms to 0 atoms and odd numbers of initial atoms to 1 atom.

While in the procedure described, there is only a 50% chance of successfully trapping an atom, there are other techniques of accomplishing this same task that instead of reducing 2→0 reduces 2→1 atom. This method is quoted as having a loading efficiency of 85%, as alluded to in Section I.

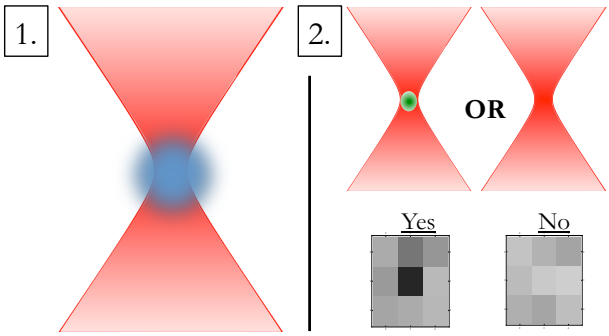

FIG. 1. (1.) After the dipole trap is turned on, a small number of atoms is trapped at the focal point of the laser, represented by the small blue cloud. (2.) After the "kill-pairs" sequence is performed, the dipole trap is either left with a single atom or no atoms, as indicated by the photoelectron count of the pixel representing the atom. Shown at the bottom of the figure are negatives of two images taken by the Andor camera, corresponding to the presence of an atom and the absence of an atom, respectively. Image adapted from [2].

At this point, an image is taken of the atom site to determine whether there is still an atom after the kill-pairs procedure. The presence of an atom is indicated by a higher photoelectron count on one of the CCD camera's pixels, due to increased levels of scattered light from the lasers used to PG cool the atom. However, regardless of whether there is an atom or not, the experiment still proceeds with the rest of the experimental sequence. This seems counter-intuitive and inefficient—and it is. If there was some way to verify that the atom had been loaded or not, then the experiment could determine whether it should proceed with the Raman cooling sequence or back into the loading cycle. For single atom trapping, this increase in efficiency would be on the order of 18% to 22%, depending on the length of the experimental procedure to be executed.

II.2. Resolved side-band Raman cooling

The next sequence of the experiment is made up of much more than just the resolved-sideband Raman cooling sequence depending on what the experimenter is looking to observer, but typically most experimental runs involve some form of resolved-sideband Raman cooling. For simplicity, this second half of the experiment will be referred to as the "cooling" cycle from now on.

Resolved-sideband Raman cooling is a laser cooling technique used for cooling trapped atoms down to the vibrational ground state of their confining potential—the lower limit on the motional energy of trapped particles. For the purposes of this section, we will limit ourselves to discussing sideband Raman cooling of atoms in a one-dimensional harmonic oscillator trap. Thus, when describing the energy state of a constituent atom, it is necessary to specify its internal electronic state $|j\rangle$ ($|g\rangle$ and

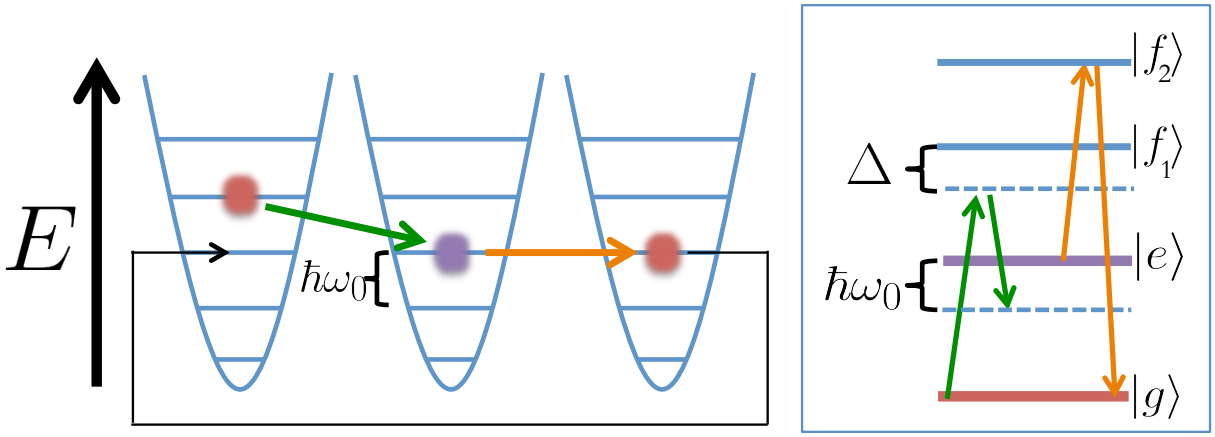

FIG. 2. On the left is a diagram illustrating the external vibrational state of the atom at the beginning and end of each transition in the Raman cooling process, while the right side of the figure. The atom starts in its internal ground state (as indicated by the fact that it is red) in some high vibrational state. The first transition (in green) is a two-photon Raman transition that puts the atom in a lower vibrational state but in some excited internal state (purple). The reason that a two-photon stokes transition is used is that if a single-photon transition is used, the photon wouldn't have enough momentum to cause the atom to change vibrational states. The first photon in the transition is detuned from some higher excited internal state, $|f_1\rangle$, by Δ to couple the atom to some virtual state without directly transitioning the atom the atom to the excited state. The next photon is used to couple the virtual state to the excited internal state $|e\rangle$, but again it is detuned from the actual state by an amount $\hbar\omega_0$, so that $\hbar\omega_0$ of the atom's kinetic energy is consumed by the transition, effectively reducing the atom's vibrational energy level by one. The next transition optically pumps the atom back to its original internal state (orange), through some other higher electronic state, $|f_2\rangle$, while maintaining its new, lower vibrational energy.

$|e\rangle$, as indicated by the colors red and purple in Figure 2) as well as its external vibrational state $|n\rangle$, with $n \geq 0$. The full state is then denoted by $|\Psi\rangle = |j\rangle \otimes |n\rangle = |j, n\rangle$.

The name "resolved-sideband" refers to the fact that this technique is intended for cooling atoms in tightly bound traps, where the characteristic vibrational frequency of the potential, ω_0 (found in $V(x) = \frac{1}{2}m\omega_0^2 x^2$), is much larger than the optical pumping rate of the transition used to cool the atoms. This imposed experimental condition manifests itself in the absorption and emission spectra of the trapped atoms by having well-resolved peaks, spaced evenly by ω_0. It is relatively easy to create traps of this strength for ions, but for neutral atoms this requires traps with oscillation frequencies on the upper end of what is possible for optical lattices

Resolved sideband Raman cooling is designed to slowly cool atoms from higher vibrational states down to the ground state of their confining potential. As previously mentioned, we impose the condition that the vibrational energy splittings be much larger than the recoil energy from the various optical transitions used in the cooling process. This is known as the Lamb-Dicke regime, and is mathematically expressed by the Lamb-Dicke parameter η:

$$\eta = \sqrt{\frac{E_R}{\hbar\omega_0}} = \sqrt{k^2} \times \sqrt{\frac{\hbar}{2m\omega_0}} = |k| \cdot x_0 \quad , \quad \eta \ll 1 \quad (1)$$

where x_0 is the characteristic length of the harmonic oscillator trap [1]. Essentially, this allows for very discrete transitions between vibrational energy levels, with small but finite probability of transitioning into a state with higher or lower quantum number n. These discrete transitions manifest themselves in their resonance absorption and emission spectrum as small, narrow peaks spaced by ω_0. If the linewidths of these transitions are too large, the probability of transitioning into other vibrational states increases which disrupts the ability to ensure certain transitions.

By imposing this condition, $\eta \ll 1$, the transitions utilized in resolved-sideband Raman cooling have a linestrength of nearly unity—that is, the probability of any other transition occurring is basically zero. Additionally, the stronger this condition is, the darker the ground state will be, which is one of the crucial features of resolved-sideband Raman cooling [1].

Resolved-sideband Raman cooling is comprised of two stages: the first stage is made up of a two-photon stimulated Raman transition that puts the atom into a lower vibrational state of a higher electronic energy level by first passing through some virtual state. The second stage consists of a two-photon spontaneous anti-stokes Raman transition, which optically pumps the atom back to its electronic ground state while it most likely remains in the lower vibrational state. By repeating this process many times, it is possible for the atom to achieve its vibrational ground state while simultaneously being in its electronic ground state. See Figure 2 for a more detailed illustration.

Once in its vibrational ground state, there is no lower vibrational state of the higher electronic energy level to transition to. This means that the first Raman transition

doesn't couple the vibrational ground state to any other state, leaving the atom where it is. This condition on the ground state satisfies the operational definition of a dark state.

III. REAL-TIME FEEDBACK LOOP

In order to install this real-time feedback loop, it was necessary to both install new hardware as well as establish new lines of communication between existing hardware. The following sections outlines the previously existing hardware and signal chain pathways, as well as justify the need for all new hardware and communication channels.

III.1. Experimental signal chain

Regal Lab's single atom trapping and cooling experiment involves the following components: a CCD camera for imaging the atom, lasers and other table-top hardware for creating the necessary experimental conditions, a glass cell where the Rubidium atoms are trapped, and two computers—one for controlling all experimental timing and procedures and one for downloading the images taken by the CCD camera (referred to as "experiment computer" and "Andor computer", respectively, from now on). A simple diagram of the experimental signal chain is illustrated in Figure 3.

Essentially, the experiment computer controls every process in the experiment. It runs Visual Basic code adapted from the original BEC experiments performed in Eric Cornell and Carl Wiemann's lab. Visual Basic was used back then due to its superior timing capabilities. As seen in Figure 3, in the initial experimental set-up the experiment computer only communicated with the lasers and magnetic fields to tell them when to turn on, as well as what power/current they should run at, and the Andor CCD camera to tell it when to take a picture. Image acquisition needs to be controlled by the experiment computer so that the timing of the images is synched with the rest of the experiment.

However, as this computer is responsible for determining which stage of the experiment is run next, it is necessary for this computer to be able to accept and respond to inputs from the Andor computer telling it information about the atoms currently in the traps. Additionally, in order to make sure that the Andor computer and the experiment computer are synced, it is also necessary that the experiment computer be able to communicate when it is starting certain stages to the Andor computer, and for the Andor computer to be able to read these signals. New hardware was necessary to be able to send and receive signals from the Andor computer.

In order to analyze the images in real time and write digital outputs in a timely fashion, the Andor Solis software provided with the iXON EM+ camera could no longer be used, and as a result new software was needed to be able to both control the camera's acquisition parameters and process the images after they were captured. LabVIEW was selected as the programming environment of choice due to the fact that there already existed many LabVIEW functions for controlling and communicating with the Andor camera.

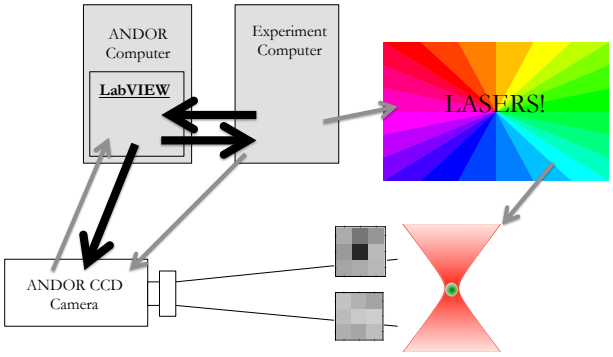

FIG. 3. All major components of the experiment are indicated in the above diagram, with previously established communication pathways indicated by gray arrows. The new communication pathways necessary to perform this feedback process are indicated in black. Not shown: the NI-PCI card that allowed for the communication pathway between the two computers.

III.2. Pulse Communication Scheme

Before jumping into installing new hardware and writing new code, it was first necessary to develop a plan for how the two computers would communicate with each other. After realizing that there are certain processes in the data processing that take substantial time, but aren't directly involved in determining the presence of an atom–things like reshaping arrays and saving them to files—I wanted to be able to perform these operations during a time when the Andor computer would otherwise be stalling, like during the loading cycle and the cooling cycle. In order for the Andor computer to know that these cycles were starting and that it was ok to now perform these processes, I designed the pulse communication scheme with this in mind. An illustration of the pulse communication scheme is shown in Figure 4.

The first stage, shaded in red, corresponds to the amount of time that the experiment is in the loading cycle. At the start of this stage, the experiment computer sends out a pulse to tell the Andor computer that it is starting the loading cycle. This lets the Andor computer know that it is ok for it to perform its peripheral, time-intensive data-processing operations while it would otherwise be sitting idle for ∼300 ms. After the Andor computer has read this "start loading cycle" pulse and performed its extra processes, it starts waiting for the pulse that corresponds to the camera trigger.

After the loading cycle is complete, the experiment

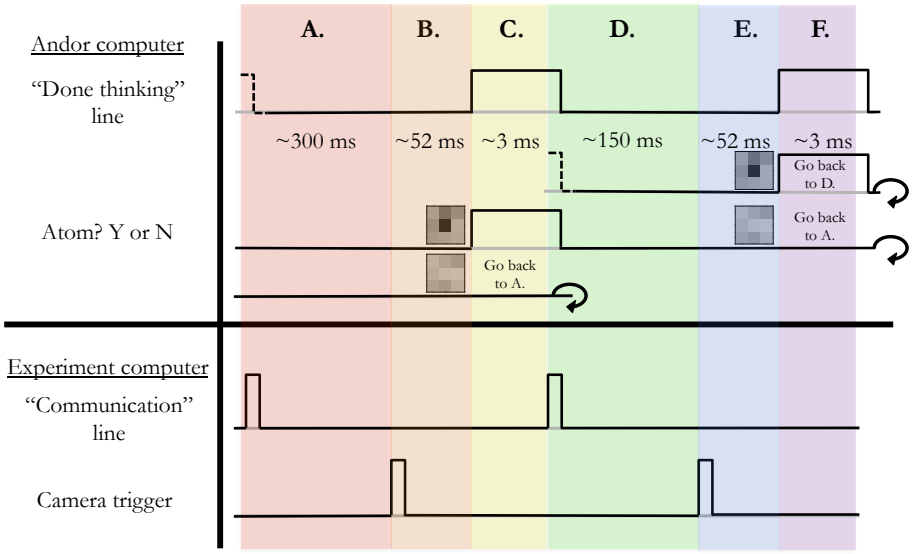

FIG. 4. The following figure illustrates the pulse communication protocol used in the feedback loop. Each experimental run can be thought of as six smaller stages (colored red through purple), three corresponding to each loading and cooling cycle. The horizontal pulse trains correspond to the four DO lines used in the experiment—two from each computer. The first line from the Andor computer ("done thinking") waits until all of the other DO lines are written before letting the experiment computer know that it is ok to read the other DO lines. The other DO lines referred to are the lines that represent each of the atom trap sites, and are set to be high (yes) or low (no) corresponding to whether or not an atom is detected in that location. The other two DO lines are from the experiment computer and serve to communicate with the Andor computer when it is starting a new experimental cycle ("communication line") and when it is taking a picture ("camera trigger"). The y-axis corresponds to the voltage on each line, while the x-axis corresponds to time (not to scale).

computer sends out a TTL pulse to tell the camera to take a picture, verifying whether or not an atom was successfully loaded into the trap. This TTL pulse is T'd off and read by the Andor computer to let it know that it will soon be able to download the image from the camera. After sending out these pulses, the experiment computer starts waiting for the Andor computer's "done thinking" line to go high.

The Andor computer waits for the camera to acquire the image before downloading the most recently acquired image in a one dimensional array from the camera's circular buffer. Once the image has been downloaded then, it is a simple matter of indexing the array at the sites of the atoms and comparing the photo-electron count values of those entries to the user-specified "photo-electron threshold". Before running the program, the user specifies the location of the atoms in their new cropped and binned coordinate system. These coordinates are converted to a corresponding pixel index value in the one dimensional array, and this pixel value is then stored in a one dimensional array of size equal to the number of atoms to be checked.

In this way, once the "image" (the one dimensional array of photo-electron count values) has been downloaded to LabVIEW, it is a simple matter of running a for-loop to check the photo-electron values of these pixel indices, compare it to the threshold value, and then store the resulting "true" or "false" value in a one-dimensional Boolean array. This all corresponds to the second stage, shaded in orange in Figure 4.

In stage three, this one dimensional boolean array is then written simultaneously to n digital out lines, where n is the number of elements in the array. After these lines have been successfully written, the Andor "done thinking" line is told to go high, letting the experiment computer know that it can now read the various Andor "Yes or No" lines.

At this point, it is up to the experiment computer to read these lines and react accordingly. For a single atom, if an atom is present then the experiment can proceed with the cooling cycle and if there isn't an atom go back into the loading cycle again. For multiple atoms it gets more complicated, as the atoms that were successfully trapped need to be held onto by increasing the power in their trap, which is dependent on the position of the trapped atom. Only once all of the atoms in the array are present does the experiment proceed.

The important thing to realize at this point, by looking at Figure 4 is that the communication scheme for the cooling cycle is identical to the communication scheme used in the loading cycle—no new protocol is necessary. After reading the "Yes or No" lines and seeing that they are all high, the experiment computer enters the "cooling" procedure[1] and indicates this by sending a pulse

[1] Although the name "cooling" cycle is a misnomer in any case,

to the Andor computer. This tells the Andor computer that it can reset its DO lines to false and perform any manipulations on the previously acquired image—things like reshaping, updating the intensity graph on the front panel, and saving the data to text files. This corresponds to stage 4, in green.

Once the cooling cycle is done, an image is acquired to see if the atom is still there, which means that a TTL pulse is sent to both the camera as well as the Andor computer. This tells the Andor computer to get ready to download the image and process it. The image downloading and analysis need not be any different than the protocol used in the loading cycle, so the same procedure is followed, including the feedback "Yes or No". This allows for the possibility of saving whatever atoms are left at the end of the run and recycling them for the next cycle.

Essentially what this means is that the Andor code need only be programmed to perform the first three stages and simply keep track of which runs were loading and which runs were cooling in a register. This eliminates the need for messy if-else structures that introduce potential for the code to get off track.

IV. IMPLEMENTATION

While both the Andor system and the NI-PCI card are already well integrated with LabVIEW, both required a nuanced approach to establishing communication pathways. The following section is most relevant for anybody who plans on using this software in the future.

IV.1. Andor iXon EM+ CCD Camera

Andor provides a software development kit along with its cameras for use in controlling their devices outside of the standard Solis software package. This software development kit currently (SDK) exists for LabVIEW, C, and Matlab. In order to control the camera efficiently and effectively outside of the standard Solis environment, it was necessary to develop a means of controlling the initialization, cooling, parameter upload, acquisition, image download, acquisition abort, and camera shutdown procedures.

Examples were provided with the SDK to illustrate the large majority of these procedures, but many were either inefficiently coded or done in such a way that didn't allow for user input between runs. The work that was done on this end of the project was to determine ways of safely and reliably communicating between the camera and LabVIEW in such a way that the user could efficiently control and specify acquisition parameters between runs, as well as start and stop runs without causing the camera to freeze up.

This was accomplished by first creating a general framework from the examples provided and then slowly beefing up the code to accomplish new tasks. Although the SDK comes with comprehensive technical documentation on each of the VIs and their function, it is still insufficient to tell how each VI is going act in a LabVIEW script. After reading and determining what I felt to be the best way to accomplish a certain task, I'd write up several variations of the code and have to test each by trial and error. This process was long and drawn out due to the limited availability of the Andor computer, which was in use nearly 24 hours a day.

Once a comprehensive and versatile LabVIEW environment had been developed to control the Andor camera, it was necessary to develop a method for quickly downloading the captured images to LabVIEW and analyzing them for atoms. Several different methods were tested to determine the fastest method for downloading the images. In order to determine whether or not an atom is present, the photo-electron count of the pixels of interest are compared to a user defined "threshold" photo-electron count. If the photo-electron count of the pixel is greater than the threshold, the program determines that there is an atom present at that location.

Additional data processing is still needed to be done as well. The data from each image needs to be stored in various arrays in order to make the data processing a quicker and more intuitive process. In the old set-up, the data could only be analyzed at the end of each run, which meant that hour-long runs would often end up being worthless due to a mis-set parameter. In the new set-up, data can be accessed and processed mid-run, allowing for the experimenter to be certain that the data that they are taking looks like it will turn out well. As an added bonus, the photo-electron count of every run is frequency binned and fit to a double gaussian function, which allows the computer to determine what the threshold should have been set at for the previous run.

IV.2. NI-PCI 6221

Next it was necessary to establish communication between the Andor computer and the experiment computer. The experiment computer was already set up to receive digital inputs and return digital outputs, so no new hardware was necessary for the experiment computer. The Andor computer, however, was not set up to give and receive digital signals so research was done as to which National Instruments PCI card should be purchased to accomplish that goal. It was determined that, although precision timing and reliability are highly important, that

simply because cooling mechanisms occur in both loading and cooling stages and because much more goes on in the "cooling" cycle than just Raman cooling, for multiple atoms the whole point of needing to trap all of the atoms is so that experiments can be performed on entanglement and tunneling, so referring to this next procedure as the cooling cycle fails to capture the proper idea. However, we will continue to use this notation.

we would be able to obtain the functionality that we needed from NI's lower lever cards, which run on software time. The NI-PCI 6221 card gave us the maximum flexibility to accomplish what we needed while also allowing for the potential to have analog inputs and outputs as well.

As with almost everything in coding, there is more than one way to do something, but different methods may have different unforeseen consequences or behave differently in certain circumstances. Once the many ways of controlling the voltage levels of the DO levels on the card and reading the voltage levels on the DI lines were determined it was a matter of simulating experimental conditions as closely as possible with function generators and reading the DO values using oscilloscopes. The different methods were assessed for their reliability and speed at which they completed their intended action.

It was determined that the lag between telling a DO line to change it's value and when it actually occurs in hardware was $\sim$5 μs, by calling two change value commands immediately in a row in software and watching the length of the corresponding pulse on an oscilloscope. This was deemed sufficient to be used to measure the length of longer processes. By telling a DO line to change its value immediately before a certain process started and then telling the same line to change its value after the process was completed and measuring the length of the corresponding pulse on an oscilloscope, a rough estimate of the time that that process took was able to be obtained.

Determining the most reliable way to read incoming pulses was a significantly more challenging process. Due to the nature of the communication scheme outlined in Section **??**, if at any point the Andor computer misses one of the triggers from the experiment computer then the entire code gets derailed and fails. The logical scheme would involve a while loop that tells the NI card to constantly read a certain line until its voltage goes high, and then proceed with the next step, but due to the inconsistency in the amount of time that it takes to read a line (5 μs $\sim$ 300μs), this scheme didn't work, because pulses were capable of slipping by unread. This appeared to occur even for pulses longer than 300μs.

The scheme that eventually was used for reading incoming digital trigger pulses employed the edge counting functionality of the NI-PCI 6221 card. Essentially, there are two "counter input" channels on the card that can be configured to count the number of rising edges in an incoming signal. This is typically used for more precise timing mechanisms, utilizing the edge counters as a means of counting the exact number of high-frequency clock cycles before starting a process. Instead, in this application, the counters were told to wait until they read one edge, and then as soon as their internal count is $\geq$1, proceed with the next step in the code. The counters, designed for reading pulse trains with MHz frequencies, were able to easily handle the length and frequency of the incoming trigger pulses and count them reliably.

V. RESULTS

The results were that the code successfully complied with all demands placed on it: after the loading cycle is complete, a picture of an atom is taken, analyzed for the presence of an atom, and then communicated to the experiment computer successfully.

There are two relevant timing measurements that characterize the efficiency of the feedback system, namely the length of time between once a picture of an atom has been taken and once the NI card outputs its 2+ DO lines and the amount of time after those DO lines are written that it takes the experiment computer to respond to the feedback. Using the same timing mechanism discussed in the previous section, the first time interval was measured to be <3 ms—completely compatible with the timescales of the rest of the experiment. Unfortunately, time ran out before I was able to make a measurement of the latter time interval, but future work on the project should easily be able to measure this.

VI. ACKNOWLEDGEMENTS

I would like to thank my PI, Cindy Regal, for providing the opportunity to work in her lab this summer; Brian Lester and Adam Kaufman for providing help and guidance throughout the process; Deborah Jin, Dan Dessau, and Leigh Dodd for their work in organizing the REU program.

[1] Kerman, Andrew James. *Raman sideband cooling and cold atomic collisions in optical lattices.* Diss. Stanford University, 2002. Dissertations & Theses: Full Text, ProQuest.

[2] Kaufman, Adam M., B. J. Lester, and C. A. Regal. "Raman sideband cooling of single neutral atom in a microscale dipole trap." Presentation. http://jila.colorado.edu/regal/sites/default/files/uploads/poster.pdf.