Data Acquisition and Storage v2

Adam West

March 29, 2017

1 Generation 1

1.1 Data Amount

In this section I summarise the procedure used in Generation 1 and the resulting data rates. Table 1 shows all of the relevant specifications in our DAQ procedure. I also include values for Generation 2 for comparison; motivations for some of these numbers come in later sections. Using these numbers we can calculate the rate at which data is

Description	Gen. 1 Value	Gen. 2 Value
Bits of DAQ precision	20	8
Sample rate	5 MS/s	15 MS/s
Record length	9 ms	9 ms
Repetition rate	50 Hz	50 Hz
Number of traces averaged together	25	25
Number of PMTs	2	8
Bits of precision written to disk	64	16
Number of days ran for	200	200
Hours run per day	12	12

Table 1: Summary of relevant quantities determining data acquisition rates for both Gen. 1 and Gen. 2.

acquired from the PMTs:

$$\text{Data acquisition rate} = 20 \text{ bits} \times 5 \text{ MHz} \times 9 \text{ ms} \times 50 \text{ Hz} \times 2 \text{ PMTs} \times 8 \text{ bits/byte} = 11.25 \text{ MB/s}. \quad (1)$$

The amount of data saved is reduced by a factor of 25 due to averaging together of molecular pulses and increased by (64/20) due to us writing double-precision (64 bit) data to disk:

$$\text{Data write rate} = 11.25 \text{ MB/s}/25 \times (64/20) = 1.4 \text{ MB/s}. \quad (2)$$

Using this value we can also estimate the amount of data which is saved over the course of an entire year. Assuming we run for an average of 8 hours every day for 200 days out of the year this amounts to:

$$\text{Yearly data storage} = 1.4 \text{ MB/s} \times 60 \times 60 \times 12 \times 200 \approx 8 \text{ TB/year} \quad (3)$$

If I examine some of the data taken during Generation 1 I can get an independent estimate of the amount of space required. Over the period June 4th–11th we wrote a total of around 267 GB of data, which gives an average of around 40 GB of data per day. If we were to write this amount of data 200 days out of the year this gives a total of around 8 TB of data. In practice these estimates are likely rather conservative.

By examining the Labview VIs used in Gen. 1 I found that double precision floats are being written to binary. Reading in some of the raw data I see that the values therein have a fractional precision specified by 16 significant digits. Since 64 bit floats have 52 bits in the mantissa and $\log 10(2^{52}) \approx 16$ this clearly corresponds to 64 bit data.

1.2 Procedure and Hardware

An outline of the hardware we used in Generation 1 is shown in Fig. 1. A PXIe chassis was used which connected to the DAQ computer via PCIe, however the actual digitizers used are PXI, rather than PXIe. The chassis has 8

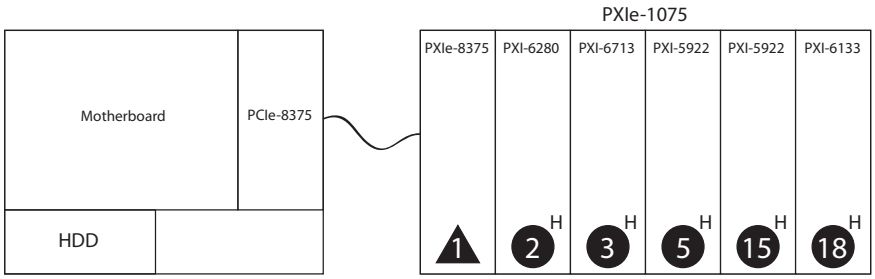

Figure 1: Schematic of the hardware used in Generation 1.

PXIe slots, 8 hybrid slots (meaning they can accept PXI, PXIe or 'PXI hybrid compatible' modules) and 1 PXIe system timing slot. Slots are grouped in sets of four with a PCI switch and bus for each group. There is a per bus, per direction maximum bandwidth of 1 GBps. This means that two modules on the same bus must share this bandwidth, which would be the case if we used all 4 PXI-5922 cards. The various modules are shown in Fig. 1. In slot 1 we have the system controller (PXIe-8375) — communication is made through this device. The digitizers used with the PMTs (PXI-5922) are in slots 5 and 15. In slot 2 we have a multifunction DAQ (PXI-6280) with 16 analogue inputs and 24 digital in/outs. In slot 3 we have a DAQ with 8 analogue outputs and 8 digital in/outs (PXI-6713). In slot 18 we have another multifunction DAQ with 8 anlogue inputs and 8 digital in/outs (PXI-6133). Since all the modules (except for the controller) are PXI (not PXIe) they are mounted in hybrid slots.

The data flow in Generation 1 was as follows:

- PMT data are collected by the PXI-5922 modules at a 5 MS/s acquisition rate for 20 bit resolution. 9 ms records are acquired at a rep rate of 50 Hz. This is a data rate of 45 Mb/s = 5.625 MB/s per channel. It is worth noting that there is likely significant overhead to this data rate; data is usually sent in 32 bit words, and requires additional data to actually perform the transfer. Increasing the sample rate to 15 MS/s reduces the precision to 16 bits (note that we cannot independently change the precision). This corresponds to a data rate of 108 Mb/s = 13.5 MB/s per channel. Each of these modules can simultaneously read from two channels. We are currently using 2 modules and 1 channel on each to read data to optimise available bandwidth — this way each channel has a separate associated PCIe bus and switch (see e.g. here).

- Data from the DAQ modules passes to the backplane of the PXIe chassis. The associated buses and switches have a bandwidth of 1 GB/s per direction. If the modules are on separate buses/switches then this is the spec'd per module bandwidth. The backplane then has a spec'd bandwidth of 6 GB/s (see here for some more details on PXIe).

- From the backplane the data passes to the controller module (PXIe-8375) which transfers the data to the PCIe card. The spec'd sustained data throughput for the controller module is 838 MBps.

- The data transfer rate to the motherboard is not spec'd, so I instead consider the PCIe bus itself. A PCIe (and PCI, PXI, PXIe) bus (also referred to as a 'link') is divided into lanes. More lanes gives a higher bandwidth. The lanes are usually specified by e.g. 'x4' for four lanes. The version of PCIe specifies the per lane bandwidth. Our PCIe module seems to operate on the PCIe 1.0a standard (a little unclear), while the computer has PCIe 2.0. I will assume the former, which has a per-lane transfer rate of 250 MBps (reference). The PCIe module operates as a 'x4' device which means that the total throughput (per direction) is 1 GB/s. This reference suggests that the 'real world' speed is typically around 75% of this value, giving 750 MB/s.

- The data from the PMTs is averaged over 25 traces in Labview to give an effective rep rate of 2 Hz. Assuming this is done in software the data must be transferred to RAM and then processed. This transfer is very fast; our RAM has a clock speed of 666 MHz which corresponds to a spec of around 10.7 GB/s.

- The data is then written to a hard drive. According to CPU-Z software the motherboard is model number 09KPNV which has a SATA 2.0 connection to hard drives, giving it a speed of 3 Gbps=375 MBps. Upgrading to a motherboard with SATA 3.0 would double this rate. We are currently using Seagate ST2000DL003-9VT166 drives which have a 2 TB capacity, 5,900 rpm spin speed, and a benchmarked sequential (random) read/write speed of 140/139 MBps (55/46 MBps) (see here).

- Data is transferred from the DAQ computer to an analysis computer. Data is read from the drive and transferred over ethernet via the Harvard network. We currently use a Broadcom NetXtreme 57xx card which connects via PCIe x1, which has a data rate of 250 MBps (assuming PCIe 1.0a), on its output it has GbE connectivity, giving a maximum of 1 Gbps=125 MBps transfer (details available e.g. here). I do not know what the ethernet cabling is spec'd to be, but will assume it is GbE. Brendon estimated that it took around 15 minutes to transfer 10 GB of data, equivalent to around 90 MBps.

Here is a summary of the data transfer speeds involved: We can see that in terms of collecting experimental data

Link	**Maximum Data Rate**	**Type of Spec**
PMT to Digitizer module	Analogue	Observed
Digitizer module to backplane	1 GBps per module	Spec'd maximum
Backplane to controller module	6 GBps	Spec'd maximum
Controller to PCIe card	838 MBps	Spec'd real world
PCIe card to motherboard	750 MBps	Spec'd maximum
Motherboard to RAM	10.7 GBps	Spec'd maximum
Motherboard to hard drive	375 MBps	Spec'd maximum
Hard drive write speed	140 MBps (50 MBps non-sequential)	Spec'd real world
Hard drive read speed	140 MBps (50 MBps non-sequential)	Spec'd real world
Hard drive to motherboard	375 MBps	Spec'd maximum
Motherboard to ethernet card	250 MBps	Spec'd maximum
Ethernet card to network	125 MBps	Spec'd maximum

Table 2: Summary of data transfer rates during acquisition and transfer to analysis computers in Generation 1.

the slowest process is actually writing the data to disk, however one must bear in mind that although we acquired about 25 MBps of data, we averaged 25 traces together at a time, meaning we would only write 1 MBps. If we were to increase the sampling rate to 15 MS/s and use 8 such cards we would be collecting 240 MBps of data, and write around 10 MBps.

In terms of transferring the data for backup/analysis, the slowest points are the hard drive read speed and the rate of transfer over ethernet.

It is worth noting that there are in fact 8 channels available to us with the current DAQ modules, however according to Paul it was rather difficult to get all 8 channels working at once: 'I could never get more than 2 channels to work well at the full bandwidth our data requires. I think a real-time or streaming acquisition loop might be necessary to capture all 8 signals.'. My preliminary tests with our hardware seem to agree with this statement.

2 Generation 2

2.1 Data Precision

2.1.1 Assumptions

In this section I will list for easy reference all of the assumptions I make in doing these calculations:

- We detect 1,500 photoelectrons per molecular pulse in Gen. 1 and hence 150,000 in Gen. 2, with appropriate Poissonian noise. In Gen. 1 there were 2 PMTs and 2 DAQ channels, so we detect 750 photoelectrons per channel. In Gen. 2 there will be 8 PMTs and 8 channels, resulting in 18,750 photoelectrons per channel.
- The molecular pulse temporal profile is described by a Gaussian, $\sigma = 0.5$ ms, extent $t_{\text{pulse}} = 2$ ms.
- The PMT response to each photon is contained in a single time bin (PMT photon response ~ 2 ns).
- The PMT has a gain, G, of 3×10^6 and is terminated by $R = 500\ \Omega$.
- The PMT response has a Gaussian distribution with a standard deviation which is 60% of the spec'd (central) value. See here for more information.
- We use four stages of $5\times$ amplification with a saturation voltage of ± 1 V.
- The signal is digitized in the time domain with a step size of 200 ns for Gen. 1 and 67 ns for Gen. 2.
- The DAQ dynamic range is assumed to be $\beta = 24$ times the maximum voltage of the average fluorescence signal, in accordance with Gen. 1. I.e. the voltage range of the DAQ is $-\beta V_{\max}/2$ to $+\beta V_{\max}/2$.
- The average signal size across the molecular pulse is half the peak voltage i.e. $R = 2$.
- We use a signal cut of half the average signal size, i.e. $C = 2$.
- Additional additive white Gaussian noise (AWGN) is injected by the equipment used. This is dominated by the first preamplification stage. I assume this to have a standard deviation of 0.1 mV, as per the spec.
- I assume that the DAQ has a Gaussianly distributed differential non-linearity with a mean absolute size of 0.1 LSB. This is based on a typical ADC spec, e.g. here.

2.1.2 Back of the envelope

A good question we should consider is how many bits of precision we require our data analysis to have. Digitization of our signals will unavoidably introduce errors through quantization. The question is how precise our DAQ should be to ensure such errors do not jeopardise our measurements.

In our experiment the DAQ digitizes time into steps δt. If we have on average N photons in each δt, the associated shot noise is given by $\sqrt{N}$. The signal associated with that time step is read in with a precision given by the number of bits that the DAQ has (often called 'vertical resolution'). Thus the signal size is also inherently digitized. This digitization has inherent 'quantization error'; if we have N_{b} bits the number of signal levels is given by $2^{N_{\text{b}}}$. We write the signal voltage as V, its maximum value as $V_{\max}$ and the dynamic range as $\pm V_{\text{range}}$. We then write the digitized signal as V_{d}. For $2^{N_{\text{b}}}$ levels the signal is divided into $2^{N_{\text{b}}} - 1$ steps. The relationship between these two signals can be written as

$$V + \frac{1}{2}\frac{2V_{\text{range}}}{2^{N_{\text{b}}} - 1} \geq V_{\text{d}} \geq V - \frac{1}{2}\frac{2V_{\text{range}}}{2^{N_{\text{b}}} - 1} \tag{4}$$

$$V + \frac{V_{\text{range}}}{2^{N_{\text{b}}} - 1} \geq V_{\text{d}} \geq V - \frac{V_{\text{range}}}{2^{N_{\text{b}}} - 1} \tag{5}$$

$$\tag{6}$$

The average absolute size of the quantization error, δV_{quant} is uniformly distributed across this range so is on average

$$\delta \bar{V}_{\text{quant}} = \frac{1}{2}\frac{V_{\text{range}}}{2^{N_{\text{b}}} - 1} = \frac{V_{\text{range}}}{2^{N_{\text{b}}+1} - 2} \tag{7}$$

A sensible criterion is that quantization error should not significantly contribute to the overall error on our data. In Generation 1 we were close to the shot noise limit, and anticipate being again in Generation 2, making shot noise

the dominant contribution. To compare the shot noise and the quantization error we must formulate the former in terms of photon number per bin. For a total of $N_{\rm tot}$ photoelectrons detected per molecule pulse there are on average

$$\bar{N}_{\rm bin} = N_{\rm tot} t_{\rm bin}/t_{\rm pulse} \tag{8}$$

photoelectrons per bin where $t_{\rm pulse}$ is the molecular pulse duration and $t_{\rm bin}$ is the time bin size. We can convert this to a signal voltage as follows:

$$V_{\rm avg} = \frac{\bar{N}_{\rm bin} GRAe}{t_{\rm bin}} = \alpha \bar{N}_{\rm bin} \tag{9}$$

where $\alpha = GRAe/t_{\rm bin}$, G is the PMT gain, R is the PMT termination, A is the amplification and e is the electron charge. Shot noise manifests as fluctuations in the number of photons in a bin so we can then write

$$\delta\bar{V}_{\rm shot} = \alpha \sqrt{\frac{N_{\rm tot} t_{\rm bin}}{t_{\rm pulse}}}. \tag{10}$$

The ratio between the shot noise for the average photon number and the quantization error, which I label η, is given by

$$\eta \equiv \delta\bar{V}_{\rm shot}/\delta\bar{V}_{\rm quant} = \alpha \sqrt{\frac{N_{\rm tot} t_{\rm bin}}{t_{\rm pulse}}} \frac{(2^{Nb+1}-2)}{V_{\rm range}}. \tag{11}$$

and we can then specify how large we wish η to be. We can rewrite the value of $V_{\rm range}$ as some multiple $\beta \equiv V_{\rm range}/V_{\rm avg}$ of the average voltage, giving (using Eqs. 9 and 8)

$$\eta = \alpha \sqrt{\frac{N_{\rm tot} t_{\rm bin}}{t_{\rm pulse}}} \frac{(2^{Nb+1}-2)}{\alpha\beta\bar{N}_{\rm bin}} \tag{12}$$

$$= \sqrt{\frac{N_{\rm tot} t_{\rm bin}}{t_{\rm pulse}}} \frac{(2^{Nb+1}-2) t_{\rm pulse}}{\beta N_{\rm tot} t_{\rm bin}} \tag{13}$$

$$= \sqrt{\frac{t_{\rm pulse}}{N_{\rm tot} t_{\rm bin}}} \frac{(2^{Nb+1}-2)}{\beta}. \tag{14}$$

Within a given molecule pulse there is variation in the shot noise. The quantization error will be fractionally most important for the smallest signals. In Gen. 1 we set a threshold on the signal size which I shall define as some factor C smaller than the average signal size. Eq. 10 is then modified to

$$\delta\bar{V}_{\rm shot} = \alpha \sqrt{\frac{N_{\rm tot} t_{\rm bin}}{C t_{\rm pulse}}} \tag{15}$$

and Eq. 14 becomes

$$\eta = \sqrt{\frac{t_{\rm pulse}}{C N_{\rm tot} t_{\rm bin}}} \frac{(2^{Nb+1}-2)}{\beta}. \tag{16}$$

From this we can calculate the number of bits needed to achieve a given η:

$$N_{\rm b} = \left\lceil \log_2 \left(\eta\beta \sqrt{\frac{C N_{\rm tot} t_{\rm bin}}{t_{\rm pulse}}} + 2 \right) - 1 \right\rceil. \tag{17}$$

We can then substitute the values for the case of Generation 1, for the specification of $\eta = 10$. As per Section 2.1.1 we have $\beta = 24$, $C = 2$, $N_{\rm tot} = 750$, $t_{\rm bin} = 200$ ns, $t_{\rm pulse} = 2$ ms. This gives a value of $N_{\rm b} = \lceil 5.5 \rceil = 6$.

In Generation 2 we are aiming for a factor of 100 increase in the signal, giving $N_{\rm tot} = 18,750$ and a threefold increase in sampling rate, giving $t_{\rm bin} = 67$ ns. Using these values we then get that $N_{\rm b} = 7$. Moving to Gen. 2 we have around 8 times as many photoelectrons per time bin. The shot noise then scales as the square root of this factor, whereas the quantization error scales linearly. Thus the bit increase is approximately $\log_2 \sqrt{8} \approx 1.5$, as shown above.

2.1.3 Simulation

To look at the problem of precision in more detail I wrote a simulation that produced some fake data and manipulated it according to the process of data acquisition. I describe these simulations here. It turns out that the inclusion of the randomness of Poissonian statistics makes a significant difference.

To begin with I produce some photons which are randomly distributed across a molecule pulse, weighted by a function which describes the mean number of photons, averaged over many pulses, to describe the molecular pulse shape. If for a particular position in the molecular pulse the mean photon number is $\bar{N}$ I randomly choose a photon number, N, with probability

$$P(N) = e^{-\bar{N}}\bar{N}^N/N!. \tag{18}$$

For Generation 1 there are 750 photons distributed across a Gaussian pulse with standard deviation 0.5 ms. Within such a pulse there are 10,000 time bins, which are thus sparsely populated with photons, as shown in Fig. 2.

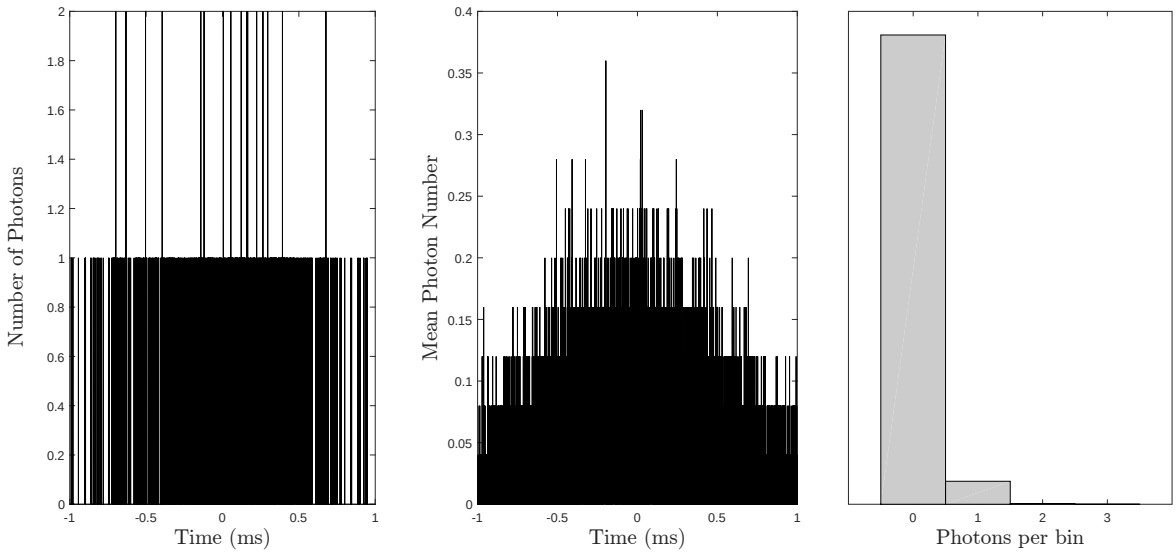

Figure 2: Simulated data. 750 photons are generated amongst 200 ns bins across a 2 ms pulse with a Gaussian weighting and Poissonian noise. Left: photons within a single pulse. Centre: photon number averaged over 25 pulses. Right: histogram of the photon number per bin, note that the majority of time bins are empty.

The photon number is then converted to a voltage according to the properties of the PMT and subsequent termination and amplification. I consider the temporal extent of the signal from individual photons to be negligibly small compared to the time digitization step; the FWHM is spec'd to be 1.5–3.0 ns. I also include variation on the PMT response as random Gaussian fluctuation of the voltage. Both of these spec's are in accordance with the datasheet for our PMT and this reference from Hamamatsu.

I also include additional white Gaussian noise to represent technical noise. The dominant source of this is from the first stage of preamplification; the pre-amp we use has a spec'd input noise of 6.4 nV/$\sqrt{\text{Hz}}$ and a bandwidth of 350 MHz (ref), giving Gaussian noise with standard deviation of 0.1 mV. By comparison the dark current on the PMT anode is spec'd to be 60 nA. The shot noise on this is 0.1 pA, which when terminated into 500 Ω gives just 50 pV of noise. Examples of the voltage signals produced are shown in Fig. 3. Note that there is a small fraction (around 1.5%) of time bins which saturate one of the preamplifiers — this saturation is also something that we observe during running of the experiment. I am told that we were aware of this but did not worry about it as it should be switch independent and happens only a small fraction of the time. For this analysis I ignore such saturation for Gen. 2 simulations, as the voltages are much larger than the 1 V saturation value.

I also considered the effect of including a low-pass filter, as per the experiment. This has a cut-off frequency of 2 MHz. To include its effect requires much finer discretization in the time domain, making the simulation much more computationally intensive. I examined the signal and its noise spectrum and observed that there was a negligible effect when including this low-pass filter. Thus I do not include it in the simulations in general, and it was not used in the results presented here.

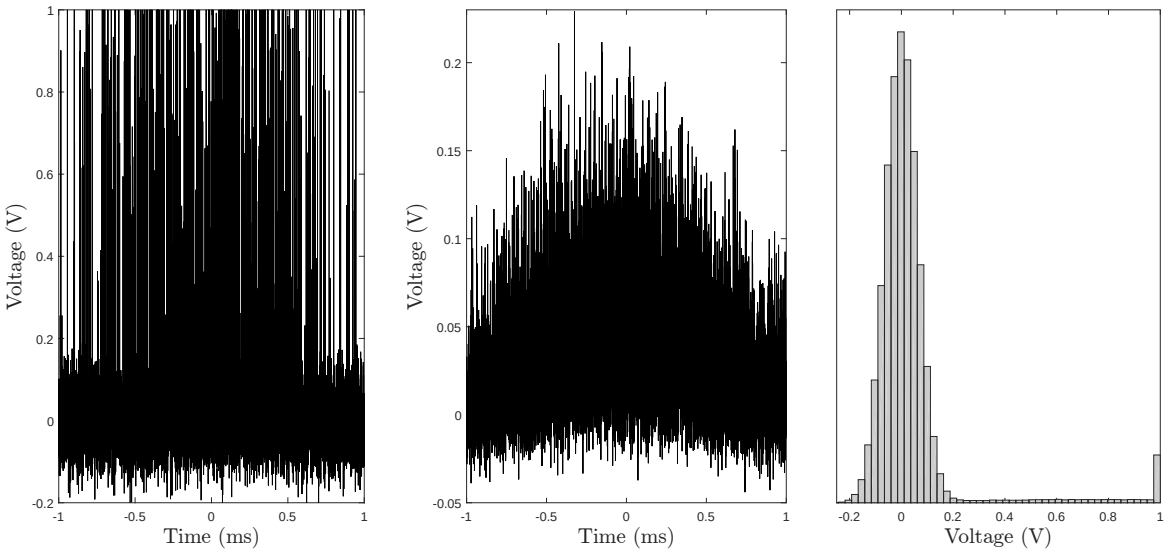

Figure 3: Examples of generated voltage signals. Left: signal from a single pulse of molecules. Centre: voltage signal averaged over 25 molecule pulses. Right: histogram of voltage levels. These 3 plots correspond directly to those in Fig. 2.

After producing the data, each trace is then digitized on a pulse-by-pulse basis. The dynamic range is set based on what was used in Gen. 1, i.e. the dynamic range is ± 8 times the maximum signal voltage for a signal averaged over 25 molecular pulses. In Gen. 1 this factor was deemed more than sufficient to accommodate additional fluctuations such as those due to changes in the overall beam flux. The digitization process is illustrated in Fig. 4.

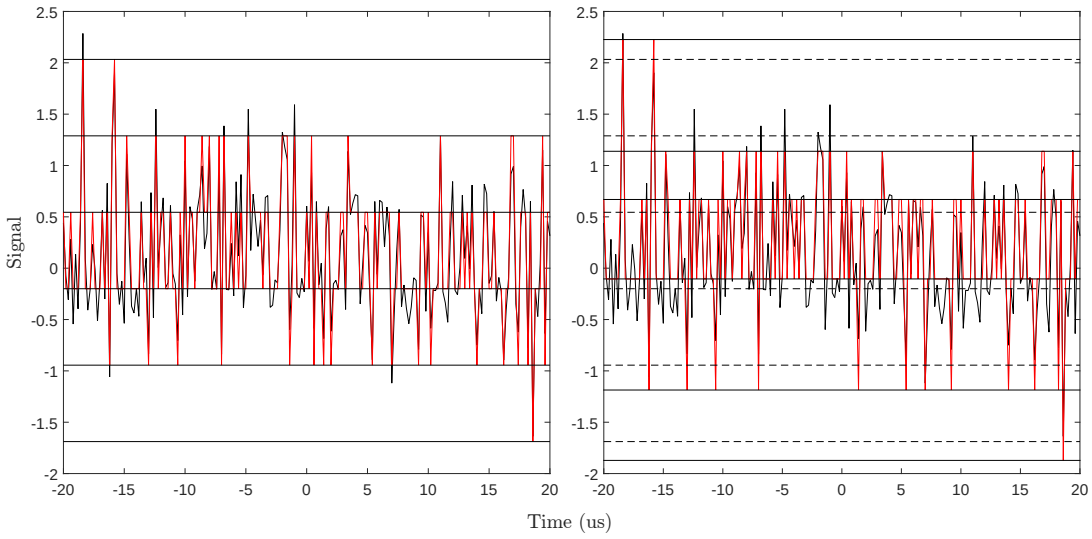

Figure 4: Example of the digitization process. In both plots the 'analogue' data is in black, and the digitized data in red. The left-hand plot has no differential non-linearity (DNL), meaning the digital levels are equally spaced. The right-hand plot has DNL included; dashed lines show the levels without DNL. These plots show digitization with 4 bits of precision. The dynamic range and size of the DNL in this plot were chosen to be slightly different to those actually used during the simulation in order to aid clarity of explanation.

In addition I optionally include the effect of differential non-linearity (DNL), which describes inaccuracy in the analogue to digital conversion characterised by variation in the digital levels. Each of the digital levels is displaced

by some amount δ, sampled from a Gaussian distribution centred on zero. NI struggled to give me an answer but quote the DNL as 0.25 LSB. They also guarantee monotonicity, suggesting the mean value is a little lower than this specified value. As a guideline for the DNL distribution one can consider comparable ADCs, such as the one here. The distribution of DNL I use is shown in Fig. 5 and can be seen to match well the spec sheet. An example of the

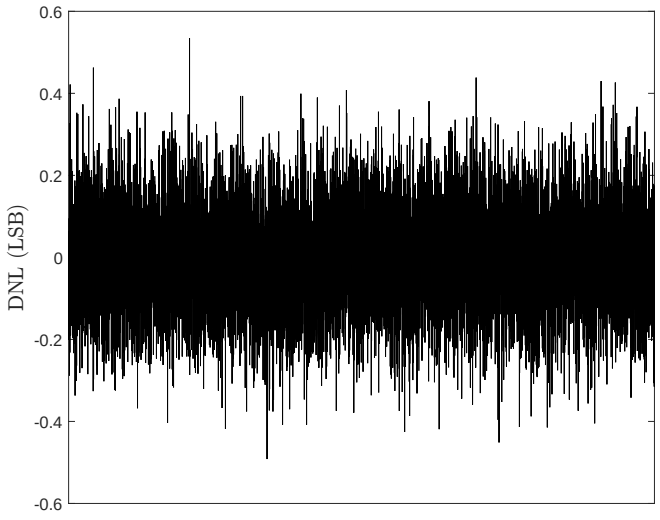

Figure 5: Distribution of the differential non-linearity used in the simulation in units of the least significant bit. Cf. here.

digitization process with DNL included is also shown in Fig. 4.

After performing this procedure we must quantify the error contributions, which come from two sources: the random fluctuations of shot noise and Gaussian noise, and the ADC quantization error. The former can be quantified by considering the deviation of the analogue signal from the 'real' signal. The quantization error can then be computed as an additional error, i.e. it is the difference between the analogue and quantized signals, i.e. the red and black traces in Fig. 4. Fig. 6 shows some example data illustrating the noise levels calculated.

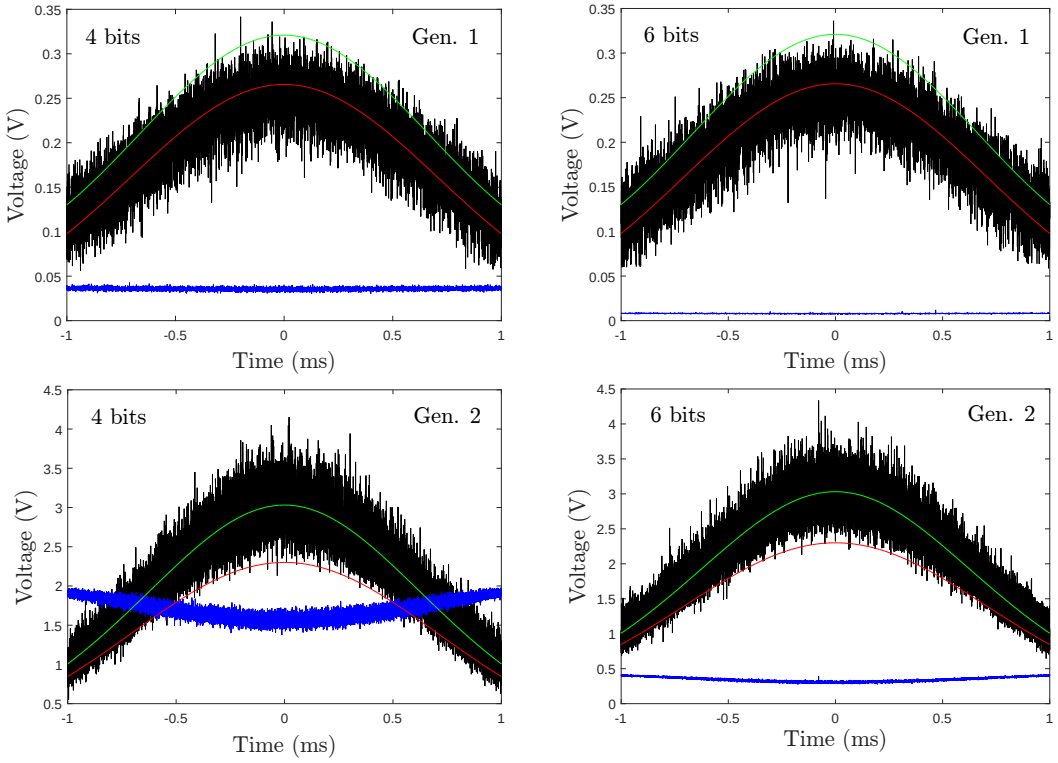

Figure 6: Example data illustrating the noise contributions from shot noise and from quantization noise. In all plots the red line shows the shot noise of the 'ideal' signal given by the square root of the expected photon number, scaled to a voltage. The green line corresponds to the total random noise, also including the PMT response distribution and additional technical noise. The black trace shows the standard deviation of the data (note it follows the green line in Gen. 2). The blue trace is the absolute size of the additional error introduced due to quantization. The left-hand plots correspond to 4 bits of precision. The right-hand plots correspond to 6 bits of precision. The top row is Gen. 1, bottom row is Gen. 2. The dynamic range is defined as described earlier in this section. DNL is included as described in the main text.

The data were simulated for a total of 250 molecular pulses and then the errors described in the caption of Fig. 6 are calculated across these pulses. We see that the noise produced due to random (Poissonian and Gaussian) fluctuations follows the expected trend well for Gen. 2, but less well for Gen. 1. The reason for this is the saturation of the preamplifier; by clipping extreme signal values the overall error is slightly reduced. As expected, the quantization error is more important for Gen. 2 than Gen. 1, where the shot noise is fractionally smaller.

I now consider in more detail the variation of the noise with bit number, shown in Fig. 7. Each data point represents a mean taken over the 2 ms pulse duration and over 250 independent pulses. The y-axis now shows the ratio between the mean quantization error (over all pulses) and the standard deviation of the signal (the random noise). Where DNL is included I assume a mean value of 0.1 LSB and a Gaussian distribution. There are two important conclusions from these data: DNL has a negligible effect, and we need about 5 (7) bits of precision to make quantization error an order of magnitude smaller than random noise for Gen. 1 (Gen. 2). This calculation is in good agreement with the analytic treatment presented in Section 2.1.2.

To study in more detail the sensitivity to DNL I examined the quantization error introduced as a function of the DNL size. This was done assuming 8 bit precision. The results are shown in Fig. 8. We can see that as the size of the DNL increases, the fractional contribution of quantization error also increases, but only by a small amount. We note that the behaviour is very similar between Generation 1 and Generation 2. As expected from Fig. 7 the quantisation error is slightly more significant for Gen. 2 compared to Gen. 1, for a given number of bits of precision. All ADCs that I have looked at specify that the DNL is always less than 1 LSB (i.e. there is monotonicity).

I will now explicitly check the resolution is suitable and that DNL doesn't impact our sensitivity to an EDM-like phase. Ideally this would be tested with an actual EDM-style analysis, but is impractical here. Instead I examine in a more general manner the precision with which we can measure the fluorescence signal, and hence the phase.

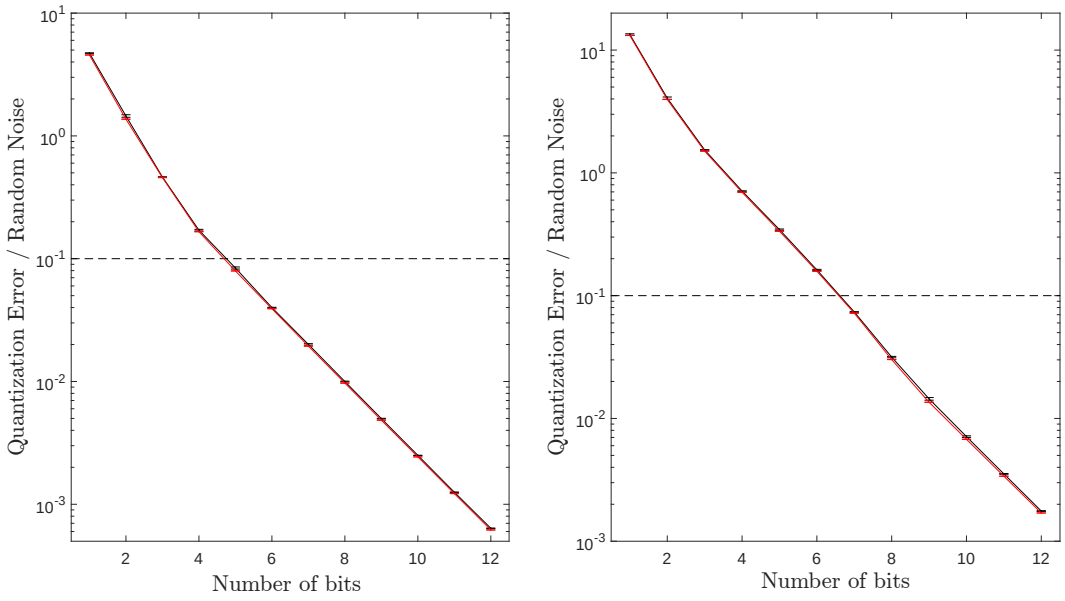

Figure 7: Error contributions vs number of bits of precision. The left-hand plot shows data for Generation 1 (750 photons per pulse, 200 ns time step). The right-hand plot shows data for Generation 2 (18,750 photons per pulse, 67 ns time step). In both plots the black line is the result without DNL included, the red line is with DNL included. The horizontal dashed line shows the threshold where quantization error is an order of magnitude smaller.

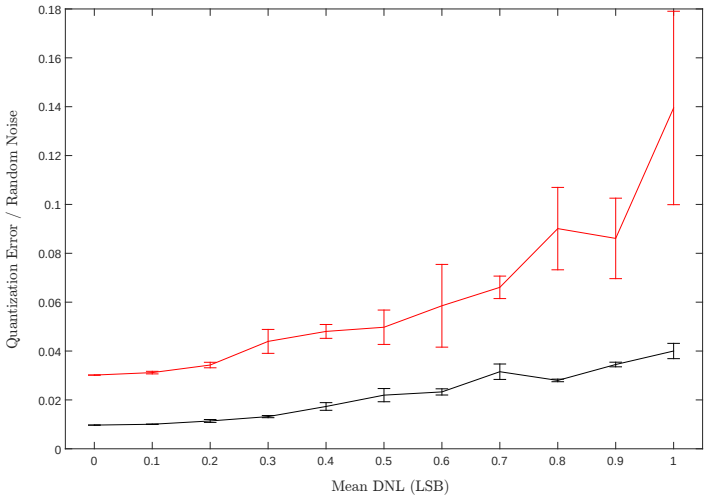

Figure 8: The quantization error as a fraction of the error due to random process (Poissonian and Gaussian noise) as a function of the size of the differential non-linearity. The DNL is described by a Gaussianly distributed displacment of the digital levels with standard deviation σ, specified in units of the least significant bit (LSB). 8 bit precision is used throughout.

The precision of our measurement can be written as

$$\delta d_e = \delta\phi_e \frac{\hbar}{\mathcal{E}_{\text{eff}}\tau}, \tag{19}$$

where $\delta\phi$ is the uncertainty on the EDM phase, $\mathcal{E}$ is the effective electric field and τ is the precession (coherence) time. Putting in numbers we find

$$\delta d_e \approx 1 \times 10^{-23} \delta\phi_e \ e.\text{cm}. \tag{20}$$

So if we want an overall precision of 10^{-28} e.cm then we require $\delta\phi_e = 10^{-5}$. We can relate the phase ϕ_e to the asymmetry $\mathcal{A}$ and to the fluorescence signal size F by

$$\mathcal{A} \equiv \frac{F_X - F_Y}{F_X + F_Y} \propto \cos(\phi + \phi_e) \tag{21}$$

where $F_{X,Y}$ are the fluorescence signals from x and y polarised detection light, and ϕ is the phase accumulated from all other terms in the interaction Hamiltonian (mostly the Zeeman interaction). We choose $\phi \approx \pi/4$ and we know that $\phi_e \ll \phi$. Using this we see that

$$\mathcal{A} \propto F_{X,Y} \propto \phi_e. \tag{22}$$

Thus if we require a fractional uncertainty on ϕ_e equal to $\delta\phi_e = 10^{-5}$ we must measure the fluorescence signal with the same fractional uncertainty.

As a test I assumed that the molecular pulse shape was completely flat, i.e. constant fluorescence signal size, I maintain the same number of photoelectrons per pulse as previously defined (1,000 for Gen. 1 and 100,000 for Gen. 2) and created data accordingly. Then I calculated the mean signal and standard error over a large number of pulses. For the case of Gen. 1 with 50,000 pulses I found a fractional uncertainty of around 2×10^{-4}, around 20 times larger than the desired uncertainty. The data amount corresponds to around 0.28 hours, so to get the desired uncertainty we require 20^2 times as much data, giving $0.28 \times 400 = 112$ hours, which is close to the actual amount of data we took (estimated around 200 hours). Similarly, for Gen. 2, 5,000 pulses (≈ 0.028 hours) gives a fractional uncertainty of around 1×10^{-4}. This is 100 times larger than the desired uncertainty, so the total amount of data we require is $0.028 \times 100^2 = 280$ hours, again close to the anticipated required data amount. The uncertainties quoted changed negligibly upon digitization; the attainable precision is not mitigated by digitization.

The next question is whether accuracy is affected by digitization. To do this I examined the relative change of the measured fluorescence signal as the specified signal size was varied. This variation was performed by changing the envelope of the generated PMT data, in this case a constant value. Example data from this procedure are shown in Fig. 9, for the case of Gen. 1.

Perhaps the first thing to notice is the discrepancy between x and y axes; there is an offset of around 7.5 mV between the specified signal size and the measured signal size. This is caused by the saturation of the pre-amplifiers, and is not a cause for concern as there is a much larger offset in the PMT output, observed to be around 100 mV in Gen. 1 and varying over time.

In each of the traces the black (red) line shows the mean and error of the analogue (digitized) signal. The first (top-left) plot assumed 8-bit resolution and a mean DNL of 0.1 LSB. We see straight away that the digitization process introduces inaccuracy to our measurement at an unacceptable level (much greater than the statistical uncertainty). The top right signal is then the same data but without any DNL, showing a vast improvement. There is an offset between the digitized and analogue data due to amplifier saturation again, but it is negligibly small. To mitigate the effect of DNL one can simply use more bits of precision. The bottom-left plot shows data with 10 bits of precision and 0.1 LSB of DNL, showing the uncertainty introduced by digitization to be less than the statistical uncertainty. The bottom-centre plot shows the case with 12 bits of precision, 0.1 LSB DNL, and bottom-right is 12 bits of precision, 0.3 LSB of DNL. Note that for a mean DNL of 0.3 LSB and a Gaussian distribution, there are some digital levels which break the monotonicity condition, so this is a significantly worse situation than we will be faced with. If we use 12 bits of precision the additional uncertainty introduced by digitization, systematic or statistical, will be negligibly small.

Carrying out this procedure for the case of Gen. 2 yields very similar results. Data assuming 12 bits of precision and a mean DNL of 0.3 LSB are shown in Fig. 10. We see again a small offset produced by digitization, larger than for Gen. 1, but the relative change is preserved very well. This is illustrated in the right-hand plot where the data are normalised to the first point in the series. As can be seen, the data series overlap extremely well indicating negligible impact of digitization.

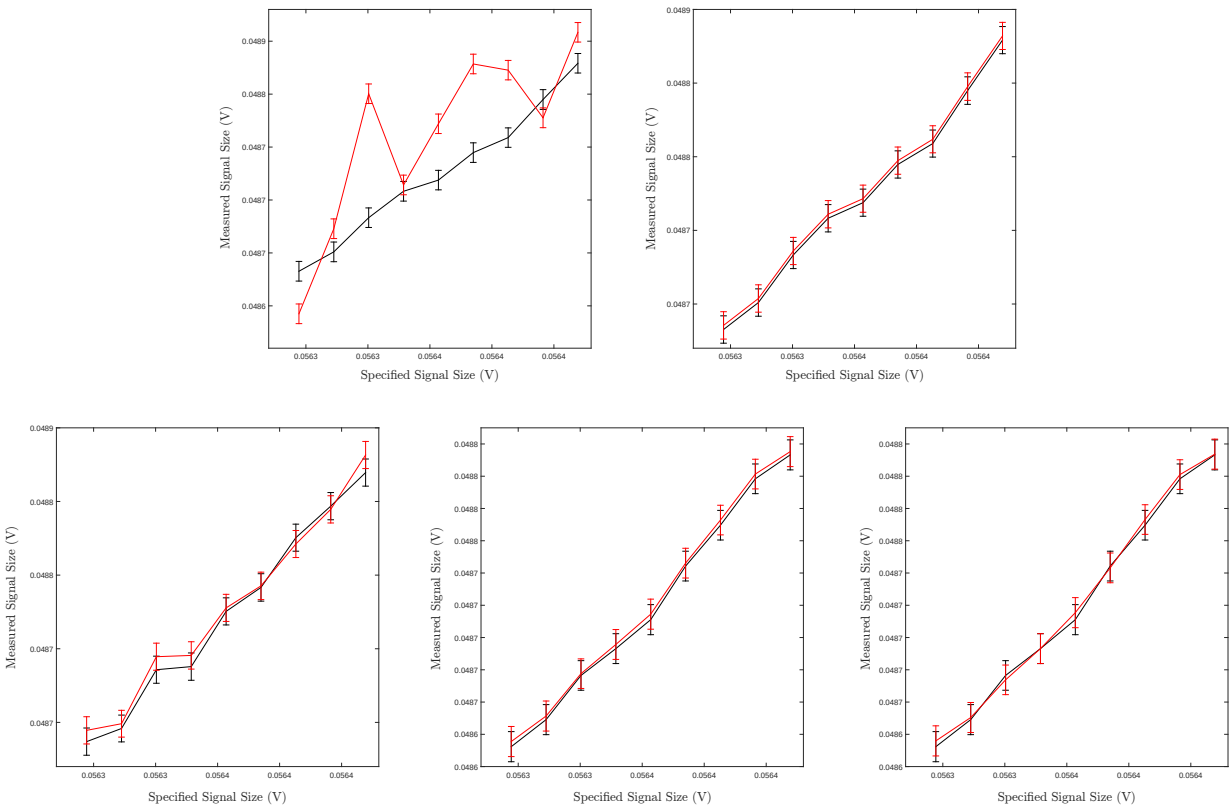

Figure 9: Testing the effect of digitization and DNL. Fluorescence data was produced in pulses of constant amplitude and the mean value and associated error computed (black data). This was then compared to the mean and error after performing digitization (red data) to analyse any mitigation of the measurement accuracy. The top left plot here assumes 8 bits of precision during digitization and a mean DNL of 0.1 LSB. Top-right corresponds to 8 bits and 0 LSB, bottom-left to 10 bits, 0.1 LSB, bottom-centre to 12 bits, 0.1 LSB, and bottom-right to 12 bits, 0.3 LSB.

2.2 Hardware Testing

As described in Section 1.2 we currently have four PXI-5922 modules, each with two channels, however when Paul tried to use 8 channels at once he could not achieve the data acquisition rate that we required. We should try and verify that that is indeed the case, and if it is, where the backlog is in the system in order to inform our choice of new data acquisition hardware.

To do this I created my own VI, based on that of Paul, to test the data acquisition rate. This VI has the minimal functionality which we require in the experiment, that is acquisition of the data at a rate defined by an external trigger, followed by averaging together of traces and saving to disk. The VI is called 'data_rate_test_new'.

For reference I will outline the modules in use here. Slots 4, 5, 15 and 16 contain PXI-5922 modules. In Labview they have the VISA resource names 'Ch56HiRes', 'Ch12HiRes', 'Ch34HiRes' and 'Ch78HiRes' respectively (the order is like this because slots 5 and 15 were initially populated before adding in the two extra modules). Each of these modules has 2 channels, labelled '0' and '1'. From hereon I will label the channels 1–8 according to the module order above, e.g. Ch34HiRes1 is now labelled 6.

I now present some data on the achievable acquisition rate. I use a sample rate of 15 MS/s and resolution of 16 bits throughout. For a first test I use immediate triggering and measure the achievable repetition rate whilst averaging and recording data to disk. The results are summarised in Table **??**. The results are qualitatively as expected, i.e. the repetition rate decreases with the number of channels. It does not, however seem to be a linear function.

The next test I performed was to take a single channel and look at the repetition rate as a function of the record length. The results are shown in Fig. 11. We see that the repetition rate decreases non-linearly with an increase in record length. This may suggest there are two different processes limiting the repetition rate in the limits of small and large record length. I also examined how the repetition rate was altered by not saving, or by not averaging

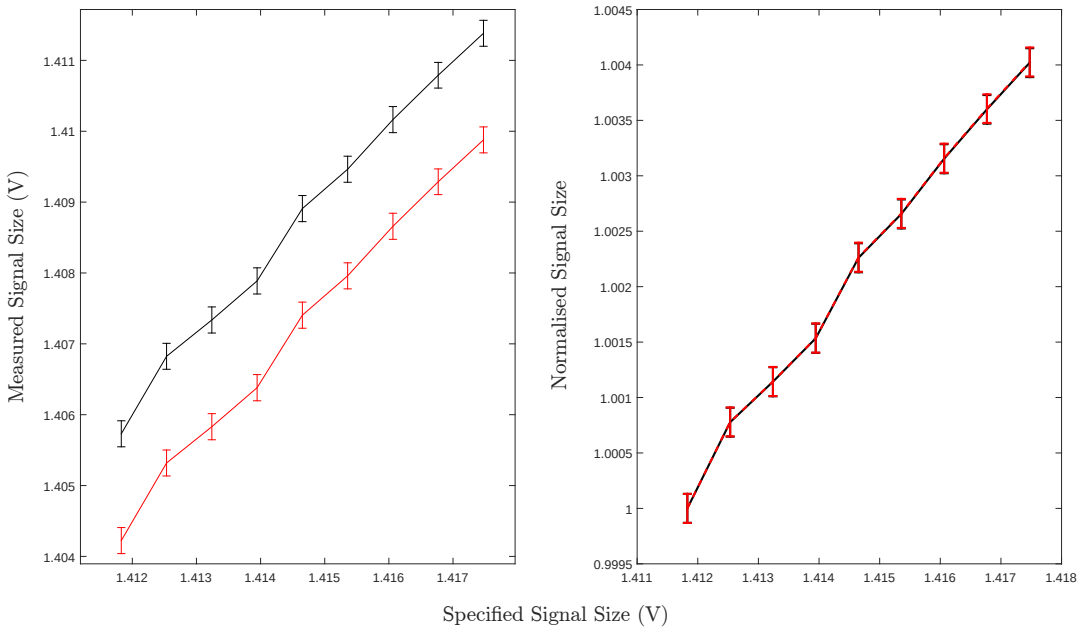

Figure 10: The same treatment as for Fig. 9, but for the case of Gen. 2. Black (red) data correspond to the analogue (digitized) signal. The left-hand plot shows data for the case of 12 bits of precision and a mean DNL of 0.3 LSB. The right-hand plot shows the same data, but the y-axis is now normalised to the first point in the series so that the relative change in signal can be seen.

and hence saving every single trace. In both cases there was no discernible change in the achievable repetition rate, showing that writing data to disk is certainly not the bottleneck.

To try and find out what the bottleneck in fact is, I tried a limiting case of a 1 µs record length, with 1 channel. This gave a repetition rate that alternated between 333 Hz and 500 Hz. If I bypass the fetching of the data altogether (i.e. the device is triggered, the hardware performs the acquisition, but the computer doesn't get the data from it), the repetition rate is unchanged. If I forego the data fetch for a 10 ms record and 1 channel, the repetition rate is increased to around 80 Hz from 46 Hz. For 8 channels it is increased to 67 Hz from 11 Hz. It seems quite clear that the transferring of data from the PXI modules to the computer is at least contributing to the bottleneck.

As an additional test I was advised by NI to try using their 'niScope EX Multi Record Fetch More Than Available Memory' VI. I adapted this to also measure the repetition rate, and to acquire from two channels from a module, rather than just one. This program does not allow immediate triggering, so I used a 50 Hz external trigger during testing. Acquiring from just one channel the VI functions correctly and the acquisition rate is 50 Hz; there doesn't seem to be any evidence of missed triggers. Acquiring from two channels leads to the VI reporting an error:

Record Length (ms)	Channels Used	Min. Rep. Rate (Hz)	Avg. Rep. Rate (Hz)
10	1/2/3/4/5/6/7/8	40	46
10	1+2/3+4/5+6/7+8	27	32
10	1+3/1+4/2+3/2+4	27	32
10	1+5/1+6/2+5/2+6	27	32
10	1+7/1+8/2+7/2+8	27	32
10	1+2+3/1+2+4/1+2+5/1+2+6/1+2+7/1+2+8	20	25
10	1+2+3+4	16	20
10	1+2+3+4+5	13	17
10	1+2+3+4+5+6	11	14
10	1+2+3+4+5+6+7	9	12
10	1+2+3+4+5+6+7+8	9	11

Table 3: Caption

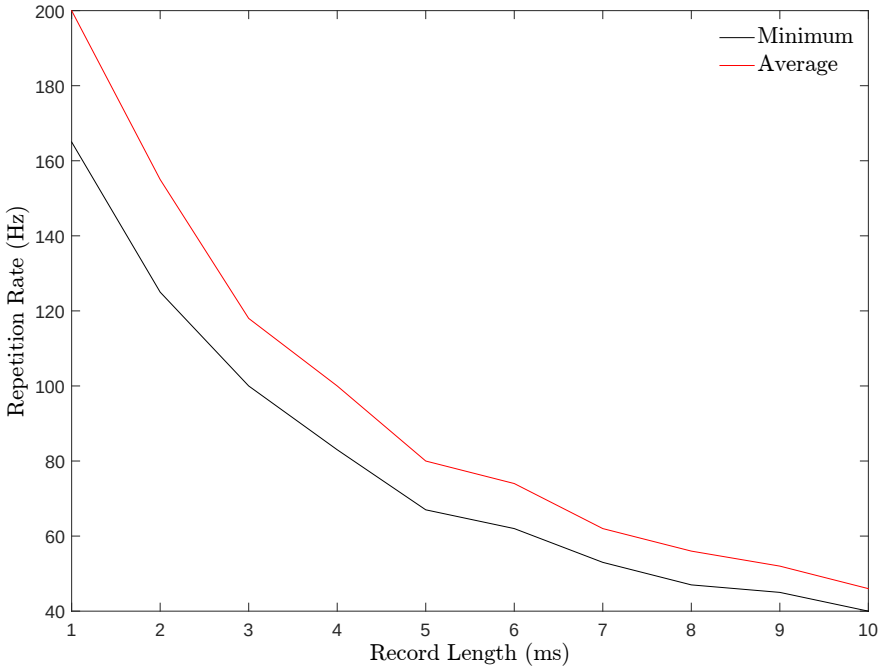

Figure 11: Minimum and average repetition rate as a function of the specified record length when acquiring from one channel at 15 MS/s with 16 bit resolution.

```
niScope Multi Fetch Cluster.vi:1<ERR>Acquisition has been stopped to prevent an
    input buffer overwrite. Your application was unable to read samples from the
    buffer fast enough to prevent new samples from overwriting unread data.

To avoid this error, you can do any of the following:
1. Increase the size of the buffer.
2. Increase the number of samples you read each time you invoke a read operation.
3. Read samples more often.
4. Reduce the sample rate.
5. Reduce the number of applications your computer is running concurrently.

In addition, if you do not need to read every sample that is acquired, use the
    Relative To and Offset properties to read the desired samples.

Status Code: -200613
```

An additional interesting piece of information is that when acquiring from just one channel, the same error message can be caused by moving the Labview window around. This, together with the above error message suggests that the computer is in some way having trouble keeping up with the data acquisition, however it is unclear why this should be the case, as the data transfer rate is relatively low (15 MS/s×10 ms×4 byte word×2 channels = 1.2 MB/s). Examining the system resource monitor shows no significant increase when the VI is running. If I turn off the plotting option for the VI I am able to acquire from two channels without the aforementioned error message. The following article from NI also states that moving windows around can severely impact performance: http://www.ni.com/product-documentation/5897/en/ (section 'Performance issues').

Based on this observation I tried disabling the plotting for the 'data_rate_test_new'. The repetition rate changed by only a small amount, around 2 Hz, when turning the plotting of data on and off, so this does not seem to be a particular problem for that VI.

2.3 Data Amount

In the next generation we wish to re-analyse how we are performing data acquisition and analysis as we anticipate having more data. The change in amount of data comes from the following factors:

- We move from using 2 PMTs to 8 PMTs — factor 4 increase in data.
- Write half or single-precision words to file instead of double precision (64 bit) — factor 2 or 4 reduction in data.
- Increase the digitization frequency from 5 MS/s to 15 MS/s — factor of 3 increase in data. We would like to do this because the decay rate of the I state is approximately three times that of the C state, so to see dynamics within our fluorescence signal with the same resolution requires the same increase in sample rate.

Assuming single-precision data I estimate we will write around 6 times as much data to disk, i.e. around 40 TB, in a year, assuming 200 days of 'buckled in' running. The option of not averaging and recording raw data was also mentioned. This would result in a factor of 25 increase in the amount of data we save, giving 1 PB of data. This would be significantly more difficult — in a simplest solution we would be swapping out hard drives much more frequently (probably around every week).

2.4 Data Acquisition Rate

There are a couple of choices as to how we can proceed in Generation 2. Since we are moving from 2 PMTs to 8 PMTs we need to acquire more data. This could be done by using 8 channels of the DAQ cards we already have, however Paul reports that he was not able to get more than 2 channels working well at the same time. If somebody wanted to investigate this further, that might be useful. An alternative is to use a new DAQ card which has multiple channels built in. This seems like an easier and more reliable solution; if a card is spec'd to read in data from 8 channels simultaneously at the rate and precision we desire, that should work. An example of such a DAQ module that would work with our current chassis and controller is here.

For the sake of this analysis I will assume we use the above module giving a sample rate of 15 MS/s from 8 channels with 12 bit precision. If we assume the same duty cycle/rep rate as in Generation 1 (5 ms long molecular pulses, 50 Hz rep rate) then we have:

$$\text{Data acquisition rate} = 12 \text{ bits} \times (67 \text{ ns})^{-1} \times 5 \text{ ms} \times 50 \text{ Hz} = 5.625 \text{ MBps/PMT}. \tag{23}$$

For 8 PMTs this is a total rate of 45 MBps, which is about 7 times the rate in Generation 1. Looking at Table 2 it seems that this should not pose a problem; the data rate from backplane to controller to PCIe card to motherboard to RAM is much greater than this.

The data acquired from the PMTs would then be written to disk in the same way, but using single-precision floats converted to binary rather than double precision. If we assume that we average over 25 molecule pulses as before and write single-precision data then the total data rate to disk is 45 MBps / 12 bits × 32 bits / 25 = 19.2 MBps. This is significantly below the estimated write speed of the current hard drives, and we can gain even more headroom there by moving to an SSD, and transferring to normal hard drives on a slower time scale. If we are not concerned about the data write speed then it seems it may be just as easy to write directly to a disk over the network. If we do not average the data we would have to write 480 MBps to disk. SSDs have specified (sequential) write speeds of 520 MBps, making it questionable whether this would in fact be possible.

2.5 Data Analysis

After the data has been written to hard drives we must get it off again to analyse the data. Again the rate-limiting step is the disk itself, i.e. reading the data. Read speeds are comparable to write speeds and significant speed up is only really available by moving to SSDs. From here we must transfer the data to whatever computer is doing the analysis, which will almost certainly be via the network. From experience we estimate that this transfer rate is around 90 MBps — approximately consistent with a hard drive read rate. If we are transferring from/to SSDs this rate will probably increase, something that would be pretty easy to test.

2.6 Backup

Obviously we wish to back up the data. In Generation 1 this was done by copying the data to extra hard drives, and this is certainly an option that we could replicate in Generation 2. To streamline this process we could have the data stored centrally on a RAID. This is an array of hard drives which the data is spread over in some fault tolerant manner. If we were to use a mirroring RAID (e.g. RAID1) then we would double the amount of drive space required. If we go with a parity based RAID (e.g. RAID5) we would only need a relatively small amount of extra space. Table 4 shows the relative merits of the options available for backup:

Method	Pros	Cons
Simple backup	Resiliant against many kinds of failure, e.g. hardware error, user error, power failure, theft, acts of god...	A slow process that will have to take place intermittently. Requires doubling of the hard drive space.
RAID1	Creates a complete copy as per the simple backup. Happens automatically as data is written.	Not necessarily resiliant to all kinds of failure e.g. if the power goes out. Requires doubling of the hard drive space.
RAID5	Happens automatically as data is written. Only requires fractional increase of hard drive space.	Not necessarily resiliant to all kinds of failure e.g. if the power goes out.
Cloud backup	Potentially lots of space. Resiliant to lab power loss etc.	Slow transfer rate. Shifts responsibility.

Table 4: Data backup options.

2.7 Proposal

In this section I will outline a proposed scheme for Generation 2 along with the necessary hardware to put it in place. Figure 12 shows an outline. The experiment runs with 8 PMTs, whose signals are read in by a single

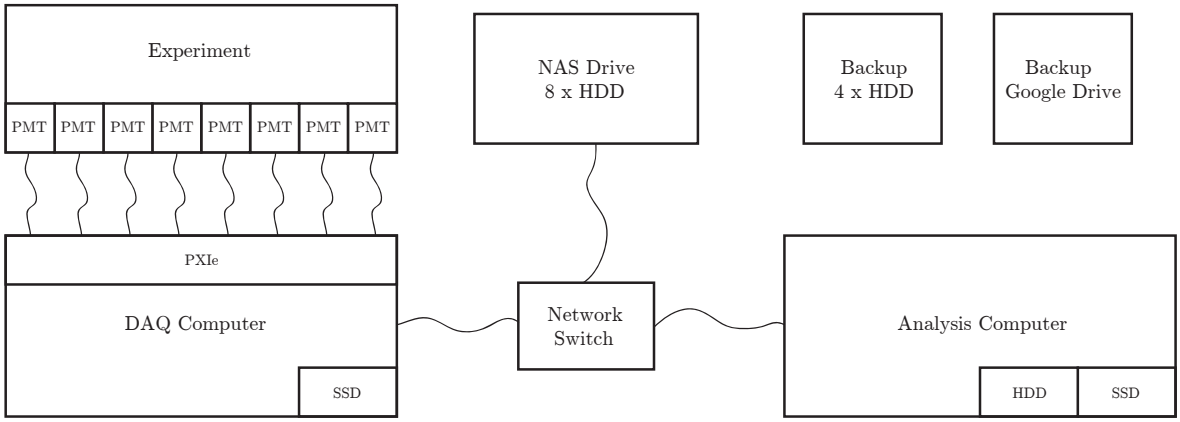

Figure 12: An outline of the proposal for data acquisition and storage in Generation 2.

PXIe DAQ module, reading 8 channels simultaneously. The data read in by these cards is then saved with 16- or 32-bit resolution either directly to a NAS drive, or, if we wish for greater writing speed, to a SSD from which the data is backed up to the NAS drive. The NAS drive runs a RAID configuration, with a maximum capacity of 48 TB (reduced to 24 TB effective capacity for RAID1, or to 36 TB for RAID5). For analysis, the required data is transferred from the NAS drive to the analysis computer, which has an SSD to speed up the transfer and processing of data. There is a separate disconnected backup on hard drives which is performed intermittently, e.g. every week/month or when a new drive can be filled. A third backup is provided by a Google Drive account. A list of equipment is given in Table 5.

Description	Model	Quantity	URL	Lead Time	Item Price	Line Price
DAQ: 8 channel, 12 bit digitizer	NI PXIe-5105	1	Link	12-20 days	$4,499–$6,749	$4,499–$6,749
NAS: Network Attached Storage controller	Synology DS1815+	1	Link	1 day	$967	$967
NAS HDD: Hard drives for the NAS	WD60EFRX	16	Link	1 day	$277	$4,432
Backup: Hard drives for data backup	WD60EFRX	8	Link	1 day	$277	$2,216
SSD: Solid state drive for DAQ and analysis computers	Samsung 850 EVO	2	Link	1 day	$380	$760
					Total:	$12,874–$15,124

Table 5: List of equipment required to upgrade the data acquisition for Generation 2.

3 Other considerations

While I have outlined a general scheme for the data acquisition, there are a few additional detailed things that we might want to consider:

- I am continuing to investigate whether we can perform the averaging of our signals in hardware. If possible this could reduce the required data transfer rate to the PC by a factor of 25.
- We could consider summing the PMTs in a separate channel and only working on this channel for most of the analysis. This would add fractionally to the overall data amount, but significantly reduce the actual data we work with regularly.
- Analysis computation may benefit from either GPU computing or cluster computing. This seems to be a matter of taste; the former has the benefit of being able to be done on a local machine, but has the possibility of not working well, the latter has the benefit of definitely working, but the disadvantage of requiring additional transfer of data to the cluster.
- Data transfer is known to be significantly slower for many small files compared to a few large files with the same overall amount of data. We may wish to investigate modifying how we manage/group data to take advantage of this.
- If desired we could increase the backup redundancy so that we have the data in triplicate.